Analytics

Salesforce, Spring '21

'21

CONTENTS

ANALYZE YOUR DATA

Salesforce offers a powerful suite of reporting and analytics tools that work together to help you understand and act on your data, as well as distribute insights to business users.

Reports and Dashboards

Salesforce offers a powerful suite of reporting tools that work together to help you understand and act on your data.

Explore Data and Take Action with Tableau CRM

Salesforce Tableau CRM, formerly known as Einstein Analytics and Wave, is a cloud-based platform for connecting data from multiple sources, creating interactive views of that data, and sharing those views in apps. Tableau CRM is a better way to distribute insight to business users so they can understand and act on changing information.

Explain, Predict, and Take Action with Einstein Discovery

Einstein Discovery augments your business intelligence with statistical modeling and supervised machine learning to identify, surface, and visualize insights into your business data. It uses predictive and prescriptive analysis to predict future outcomes, as well as suggests ways in which you can improve predicted outcomes. Einstein Discovery requires either the Tableau CRM Plus license or the Einstein Predictions license, both of which are available for an extra cost.

Modify and Enrich Salesforce Data with Data Pipelines

Salesforce Data Pipelines is a high-performance data platform that can clean, transform, and enrich large volumes of data at scale. Use Salesforce Data Pipelines to enrich and modify Salesforce data without needing third-party tools or taking data outside your trusted Salesforce environment. Unlike external ETL tools and data warehousing solutions, Data Pipelines is built natively into your Salesforce CRM, ensuring that updates are fast and secure. No more round-tripping data through expensive, fragile, lower-performing external systems.

Reports and Dashboards

Salesforce offers a powerful suite of reporting tools that work together to help you understand and act on your data.

Get Started with Reports and Dashboards	
Learn Reporting Concepts and Build Your First Report and Dashboard	**Trailhead** • Reports and Dashboards (Lightning Experience) • Reports and Dashboards (Salesforce Classic) **Help** • Reports Basics • Dashboards Basics **Video** • Series: Get Started with Reports and Dashboards in Lightning Experience • Series: Get Started with Reports and Dashboards in Salesforce Classic

Get Started with Reports and Dashboards	
Review Report and Dashboard Limitations	**Help** • Report Limits, Limitations, and Allocations • Lightning Reports and Dashboards Technical Requirements

Report on Your Data	
Build Reports	**Help** • Build a Report **Video** • Build a Report (Lightning Experience) • Getting Started with Buckets (Salesforce Classic) • Building Matrix Reports (Salesforce Classic) • Introducing Joined Reports in Salesforce (Salesforce Classic)
Filter Reports	**Help** • Filter Reports by Values • Filter Across Objects with Cross Filters • Filter Reports Via URL Parameters in Lightning Experience **Video** • How to Use Cross Filters (Lightning Experience) • Dynamically Filter Reports Using URL Parameters (Lightning Experience) • Using Cross Filters in Reports (Salesforce Classic)
Share Reports and Report Data	**Help** • Report and Dashboard Folders **Video** • Manage Report and Dashboard Folders (Lightning Experience)
Subscribe to and Schedule Reports	**Help** • Schedule and Subscribe to Reports **Video** • Subscribe to Reports and Dashboards (Lightning Experience)

Report on Your Data	
	• ▶ Report Notifications (Salesforce Classic) • ▶ Tips for Scheduling Reports (Salesforce Classic)
Work with Report Data in Other Tools	**📖 Help** • Salesforce Reports in Quip • Report on Salesforce Data with Excel • Export a Report
Analyze Report Data Using AI	**📖 Help** • Analyze Reports with Einstein Data Insights
Troubleshoot Reports	**▶ Video** • ▶ Making Your Reports Run Faster (Salesforce Classic)

Share Insights with Dashboards	
Build Dashboards	**📖 Help** • Build a Dashboard • Edit and Customize Lightning Experience Dashboard Components • Work with Salesforce Classic Dashboards • Dynamic Dashboards: Choose Who People View a Dashboard As • Sample CRM Dashboards (PDF) **▶ Video** • ▶ An Overview of Dashboards (Salesforce Classic)
Filter Dashboards	**📖 Help** • Filter a Dashboard
Subscribe to and Schedule Dashboards	**📖 Help** • Subscribe to or Schedule Dashboards **▶ Video** • ▶ Subscribe to Reports and Dashboards (Lightning Experience)

Share Insights with Dashboards	

Share Dashboards	📖 **Help**
	
	   • Report and Dashboard Folders
	
	▶️ **Video**
	
	   • ⏺ Manage Report and Dashboard Folders (Lightning Experience)

Troubleshoot Dashboards	📖 **Help** • Why Doesn't My Dashboard Display the Data I Expect?

Reports

Reports give you access to your Salesforce data. You can examine your Salesforce data in almost infinite combinations, display it in easy-to-understand formats, and share the resulting insights with others. Before building, reading, and sharing reports, review these reporting basics.

As you prepare to report on your Salesforce data, keep these tips in mind:

- Well-designed reports run faster.
- Before building your report, consider writing down each of the questions your report must answer. This way, your report is sure to return all the data you need.
- Reports are shared via folders. Whomever has permission to the folder your report is saved in also has access to your report. Ensure that you save your report in an appropriate folder.

Before building your first report, familiarize yourself with these features and concepts.

Report Builder

The *report builder* is a visual, drag-and-drop tool which you use to create reports and edit existing ones. The report builder is where you choose a report type, report format, and the fields that make up your report.

To launch the report builder, click **New Report**.

Fields

One or more *fields* describe each report result. If you imagine that your report as a table of information, then each row is a result and each column is a field.

For example, a human resources manager creates a report about employees. Each result is an employee, and each field is a different piece of information about the employee: first name, last name, job title, start date, and so forth.

When you create or edit a report, you choose which fields you want to include in your report. To ensure your reports run quickly, it's a good idea to include only the fields that you need.

Filters

Limit the data that your report returns by using *filters*. Filters are useful for many reasons, such as focusing your report on specific data, or ensuring that your report runs quickly.

For example, say your report returns all the Cases in your company, but you only want to see Cases which are open and assigned to you. Filter the report on the `Owner` field and `Status` field.

Add filters in the Report Builder. In Lightning Experience, you can add, edit, or remove filters while reading a report, too.

EDITIONS

Available in: Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

Report Types

The *report type* governs which fields are available in your report. For example, File and Content reports have fields like `File ID`, `File Name`, and `Total Downloads`. Accounts reports have fields like `Account ID`, `Account Name`, and `Phone`.

The first thing you do when creating a report is choose a report type.

Report Format

The *report format* specifies how your report results are laid out. Available formats are tabular (no grouping), summary (grouped by rows), matrix (grouped by rows and columns), or joined (with report blocks that provide different views of your data). In the Salesforce Classic report builder, you must choose a report format before grouping data. In the Lightning report builder, the report format automatically updates as you group report data.

1. [Build a Report](#)

   When you have questions about your Salesforce data, like "How much revenue did we earn in the South East last quarter?", "Which lead source is generating the most closed opportunities?", or "What is the average age of all open cases?", build a report to get the answers.

2. [Customize Report Views in the Run Page](#)

   Use the power of Lightning Experience when you review and analyze your report records in the run page.

3. [Filter Report Data](#)

   What if your report gives you more data than you need? Use filters to pare down your report until it only shows the data that you want.

4. [Schedule and Subscribe to Reports](#)

   Schedule and Subscribe to Reports and receive notifications that keep you informed about metrics you care most about without having to manually run reports. In Salesforce Classic, you can specify criteria that trigger report notifications.

5. [Export and Connect Reports to Other Tools](#)

   Export or connect a report to another tool, such as Quip, to work with report data outside of Salesforce.

6. [Drill Down into Your Reports to Learn Even More](#)

   Drill-down helps you take a closer look at records in a report. For example, as a sales manager, drill-down can help you track the progress of just a few of your reps or review the breakdown of current opportunities based on type.

7. [Organize Reports](#)

   Keep your reports at your fingertips by sorting them into folders and deleting unused reports. If you have a lot of reports, you can use the search field to find the one you need.

8. [Analyze Reports with Einstein Data Insights](#)

   Einstein Data Insights scans your report data - quickly and thoroughly - using artificial intelligence and comprehensive statistical analysis powered by Einstein Discovery. Einstein Data Insights goes deep into the report data, explores underlying patterns, identifies insights, and surfaces those insights with charts and explanations that are easy to understand. Einstein Data Insights works with Tabular and Summary reports.

9. [Troubleshoot Reports](#)

   Use these tips to help solve problems that arise when you're working with reports.

10. [Improve Report Performance: Best Practices](#)

    When you run a report, the report looks for and then returns data. If a report is running slowly, it's because parts of the report that take a long time to find and return data. By optimizing the slow parts of a report, the report can be made to run much faster. Follow the tips in this guide to speed up sluggish reports.

Build a Report

When you have questions about your Salesforce data, like "How much revenue did we earn in the South East last quarter?", "Which lead source is generating the most closed opportunities?", or "What is the average age of all open cases?", build a report to get the answers.

1. Build a Report in Lightning Experience

   Lightning report builder is a powerful and intuitive tool for analyzing your Salesforce data. Group, filter, and summarize records to answer business questions like "How much revenue did we generate from new business in California last quarter?" For a visual overview of your data, add a report chart. When finished, run your report to see full results.

2. Build a Report in Salesforce Classic

   Report Builder is a drag-and-drop tool for accessing your data quickly and comprehensively. Use it to set up new reports and edit existing ones.

3. Categorize Data with Bucket Columns

   Quickly categorize report records without creating a formula or a custom field by bucketing them. When you create a bucket column, you define multiple categories (buckets) used to group report values. Like any other column in your report, you can sort, filter, and group by bucket columns.

4. Evaluate Report Data with Formulas

   Reports feature built-in basic math functions (sum, average, min, and max) that you can apply to any numerical column in a report. When you're ready to perform advanced logical or mathematical operations, write a formula.

5. Count Unique Values in Report Results

   See how many distinct values a column in your report returns with a unique count.

6. Combine Different Types of Information in a Joined Report

   The joined report format lets you view different types of information in a single report. A joined report can contain data from multiple standard or custom report types. You can turn any existing report into a joined report using the report builder.

7. Report on Historical Changes

   On top of the standard up-to-the-minute reporting on the current state of your business, you can analyze day-to-day and week-to-week changes in opportunities, cases, forecasts, and custom objects.

8. Report Type Reference

   The report type you choose determines which records and fields appear in your report. For example, the Opportunities report type gives you access to Opportunity records and fields like Amoung, Stage and Opportunity Owner.

9. Turn Automatic Updates to the Report Preview On or Off

   Edit reports faster by turning off automatic preview updates. When off, you can make multiple edits without waiting for the preview to refresh after each edit. When you're ready to preview data, manually refresh the report preview. Or, see sample records returned after each edit by keeping automatic previews on.

Build a Report in Lightning Experience

Lightning report builder is a powerful and intuitive tool for analyzing your Salesforce data. Group, filter, and summarize records to answer business questions like "How much revenue did we generate from new business in California last quarter?" For a visual overview of your data, add a report chart. When finished, run your report to see full results.

📝 **Note:** These instructions are for building reports in the Lightning report builder. To learn how to build a report using the Classic report builder, see Build a Report in Salesforce Classic in Salesforce Help.

▶ Watch a video: ◉ Build a Report (Lightning Experience)

1. From the reports tab, click **New Report**.

2. Choose a report type, then click **Continue**.

The report type you choose determines which records are returned and which fields are available in your report.

3. The report opens in edit mode, and shows a preview.

In edit mode, add and remove fields to your report as columns, group by rows and columns, filter report data, or add a chart. Customize your report until it shows exactly the data that you need.

4. To add a column to your report,

   a. Choose a field from the **Add column...** picklist.

   b. Alternatively, expand the **Fields** pane, then drag a field onto the Columns list or directly onto the report preview. To select multiple fields, press Ctrl (Windows), Cmd (Mac), or Shift when you click.

   To remove a column from your report,

   a. From the Columns list, find the column you want to remove. Then click ✕ .

   b. Alternatively, from the preview pane, find the column you want to remove. Click ▼ > **Remove Column**.

   c. To remove all columns from your report, from the Columns list, click 🗑 .

   Removing a column from your report doesn't delete the field. If you remove a column, but want it back, add it again.

5. To summarize a column in your report,

   a. From the preview pane, find the column you want to summarize. Click ▼ > **Summarize**.

   b. Choose how you want to summarize the column: **Sum**, **Average**, **Max**, **Min**.

6. To group records in your report,

   a. Choose a column from the **Add group...** picklist under GROUP ROWS.

   After grouping a row, you can group a column by choosing a column from the **Add group...** picklist under GROUP COLUMNS. Group up to 2 rows and 2 columns.

   After grouping records by a date field, you can also customize date granularity. First select the date field you want to group your report by. Then, for **Group Date By...**, apply a calendar or fiscal period.

   b. Alternatively, drag a column from the Columns list or from the preview pane onto the GROUP ROWS or GROUP COLUMNS list.

   c. Alternatively, from the preview pane, find the column you want to group. Click ▼ > **Group Rows by This Column** (or **Group Columns by this Column)**.

   After adding a group, you can show or hide detail rows, subtotals, and a grand total by clicking the switches at the bottom of the preview pane.

   📝 Note: For a report that is grouped by rows and columns (matrix report), the preview sometimes shows different results than the actual report.

   To ungroup records in your report,

   a. From the Groups list, find the group you'd like to ungroup and then click ✕ .

   b. Alternatively, drag the group onto the preview pane.

   c. To ungroup all groups in your report, from the Groups list, click 🗑 .

Unless you drag the group onto the preview pane, removing a group also removes the column from your report. If you still want to show the column, add it back.

7. To filter records from your report, click ▼ **FILTERS**.

Depending on which report type you chose, your report has between two and four standard filters that are applied by default. Most templates include a Show Me filter and a Date filter. The Show Me filter scopes report results around common groups, like "my opportunities" or "all opportunities". The Date filter scopes results around a date field, like "created date" or "closed date".

a. To add a field filter, choose a field from the **Add filter...** picklist.

b. To edit a filter, including standard filters, click the filter.

c. To remove a filter, click the ✕ icon on the filter.

For more information about filtering reports, see Filter Report Data in Salesforce Help.

8. To add a chart, first add at least 1 group, then click **Add Chart**.

A chart appears. To customize the chart, click ⚙ . Change the chart type, color palette, and more.

To show or hide the chart, click 🄴 .

To remove the chart, click ⚙ > **Remove Chart**.

9. Click **Save**. If you're creating a brand new report, give it a name. Optionally, give it a description. With access and sharing in mind, save the report in an appropriate folder.

10. To view complete report results, click **Run**.

Now you know how to harness your Salesforce data to answer your business questions. So, "How much revenue did we generate from new business in California last quarter?", anyways?

To find out, first create a report based on the Opportunities report type. Then, add these filters:

- Close Date range is Previous FQ.
- Billing State is California.
- Type is New Business.

Summarize the Amount column.

Finally, run the report and take note of the total.

📝 Note: As you get ready to build reports with the Lightning Experience report builder, take note of how it differs from the Salesforce Classic report builder. Remember, with the appropriate user permissions, both builders are available in Lightning Experience. If a feature or tool isn't available in one builder, save your report and edit it in the other.

Differences Between the Lightning and Classic Report Builders

Feature	Difference
Report Formats	In the Salesforce Classic report builder, you must choose a report format before grouping data.
	In the Lightning report builder, the report format automatically updates as you group report data. You don't need to select the tabular, summary, or matrix format.
Charts	The Lightning Experience report builder features the same charts as the report-view-page in Lightning Experience.

Features Not Available in the Lightning Report Builder Beta

These features aren't available in the beta version of the Lightning report builder, but we're working hard to implement them in a future release.

Reports built in the Lightning report builder are editable in the Classic builder, and vice versa. To use unavailable features, edit a report in the Classic report builder.

- Row Limit Filters
- Dashboard Settings Menu

Build a Report in Salesforce Classic

Report Builder is a drag-and-drop tool for accessing your data quickly and comprehensively. Use it to set up new reports and edit existing ones.

Watch a Demo: ▶ Getting Started with Report Builder (Salesforce Classic)

📝 Note: These instructions are for building reports in the Classic report builder. To learn how to build a report using the Lightning report builder, see Build a Report in Lightning Experience (Beta) on page 7 in Salesforce Help.

- To customize an existing report using report builder, click the name of a report and click **Customize**.
- To optimize screen real estate, report builder uses a compressed page header.
- To view your application tabs, close the builder or click the Salesforce logo.

1. Choose a Report Type

   The report type you choose determines which records and fields appear in your report. For example, the Opportunities report type gives you access to Opportunity records and fields like Amount, Stage, and Type.

2. Choose a Report Format

   A report can use the tabular, summary, matrix, or joined format. Choose a format that's complex enough to capture the data you want to show, but simple enough to communicate it effectively.

3. Group Your Report Data

   Group data in columns or rows in summary, matrix, and joined reports to display meaningful information. For example, group opportunities by Close Date to see closed opportunities or group cases by product to see the number of cases for each product. You can have groupings inside groupings.

4. Keep Working While Your Report Preview Loads

   For most actions, you can continue working on your report while the preview loads. For example, when editing a report you can drag multiple fields into the report, then create a grouping while those columns load.

5. Customizing Reports

   Report builder is a visual editor for reports. The report builder screen lets you work with report fields and filters, and shows you a preview of your report with just some of the data.

6. Report Fields

   The Fields pane displays fields from the selected report type, organized by folder. It also lists custom summary formulas, which you can create, edit, and delete.

EDITIONS

Available in: Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

USER PERMISSIONS

To create, edit, and delete reports:

- **Legacy Folder Sharing**
  Create and Customize Reports

  AND

  Report Builder

  Enhanced Folder Sharing
  Create and Customize Reports

  AND

  Report Builder

7. Subtotal Report Results

    Subtotaling your reports gives you a tool to analyze trends in the data. You can group sets of information, sort the groupings, and compare subtotals for each set against the overall total. In summary and joined reports, you can also subtotal by multiple fields to give you cascading sets of information.

8. Smart Totaling in Reports

    "Smart" totaling means that duplicate data is counted only once in any subtotal or total. Salesforce uses "smart" totaling when you run reports that include duplicate data in any of the columns chosen for summing or averaging.

9. Save Your Report

    Click **Save** to update an existing report with recent changes, or **Save As** to clone the original report without changing it. In Lightning Experience, click **Clone** to clone the report.

10. Show Report Data Graphically

    To help readers understand your data quickly and easily, show the data in chart form. Charts appear just above the report table. They can help users get a feel for the data before they delve into the details. Use line charts to track changes over time, or a bar or pie chart to compare values at a point in time. Charts can also appear in dashboard components.

11. Show Report Data in Tables

    To help readers scan for data easily, try hiding details and ranges, limiting the number of results shown, and highlighting with color. You can also show your table in a dashboard component.

SEE ALSO:

Keep Working While Your Report Preview Loads

Create a Report

Create a Report

Report Fields

Choose a Report Type

Customizing Reports

Choose a Report Format

Creating a Custom Report

Combine Different Types of Information in a Joined Report

Choose a Report Type

The report type you choose determines which records and fields appear in your report. For example, the Opportunities report type gives you access to Opportunity records and fields like Amount, Stage, and Type.

There are two types of report types: standard report types and custom report types.

Standard report types give you access to most Salesforce data. For example, the Opportunities report type gives you access to Opportunity records and fields in your report. If you're going to report on Opportunity Amounts or Probability or Type, then Opportunities is the report type is for you.

Custom report types give you access to custom objects in Salesforce, or custom views of standard objects (like Opportunities), which your administrator configures. For example, your administrator can create a custom report type which gives access to Opportunities, plus related fields from Products. With that custom report type, you can easily report on Opportunities for a given product.

1. From the Reports tab, click **New Report**.

2. Select the report type, and then click **Create**.

   📝 Note: You can't change the report type after the report is created.

The report builder opens, granting access to records and fields based on your selected report type.

SEE ALSO:

Choose a Report Format

A report can use the tabular, summary, matrix, or joined format. Choose a format that's complex enough to capture the data you want to show, but simple enough to communicate it effectively.

Choose one of the following report formats using the **Format** menu of the report builder. Tabular format is the default.

Format	Description
Tabular	Tabular reports are the simplest and fastest way to look at data. Similar to a spreadsheet, they consist simply of an ordered set of fields in columns, with each matching record listed in a row. Tabular reports are best for creating lists of records or a list with a single grand total. They can't be used to create groups of data or charts, and can't be used in dashboards unless rows are limited. Examples include contact mailing lists and activity reports.
Summary	Summary reports are similar to tabular reports, but also allow users to group rows of data, view subtotals, and create charts. They can be used as the source report for dashboard components. Use this type for a report to show subtotals based on the value of a particular field or when you want to create a hierarchical list, such as all opportunities for your team, subtotaled by `Stage` and `Owner`. Summary reports with no groupings show as tabular reports on the report run page.
Matrix	Matrix reports are similar to summary reports but allow you to group and summarize data by both rows and columns. They can be used as the source report for dashboard components. Use this type for comparing related totals, especially if you have large amounts of data to summarize and you need to compare values in several different fields, or you want to look at data by date *and* by product, person, or geography. Matrix reports without at least one row and one column grouping show as summary reports on the report run page. ⊙ Building Matrix Reports (Salesforce Classic)
Joined	Joined reports let you create multiple report blocks that provide different views of your data. Each block acts like a "sub-report," with its own fields, columns, sorting, and filtering. A joined report can even contain data from different report types. Joined reports are available only in **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions. ⊙ Introducing Joined Reports in Salesforce (Salesforce Classic)

Changing the Report Format

Changing the format affects filters and groupings, as follows:

When you change...	What Happens?
Tabular to Summary or Matrix	The `Rows to Display` filter is removed.

When you change...	What Happens?
Summary, Matrix, or Joined to Tabular	All groupings, charts, and custom summary formulas are removed from the report. Grouping fields are not converted to columns in the tabular report. If the joined report contained multiple blocks, the columns from only the first block are included in the tabular report.
Summary to Matrix	The first summary grouping becomes the first row grouping. The second becomes the first column grouping. The third becomes the second row grouping. If you're using the report wizard, the third summary grouping is removed.
Matrix to Summary	The first row grouping becomes the first summary grouping. The second row grouping becomes the *third* summary grouping. The first column grouping becomes the *second* summary grouping. The second column grouping is removed. If you're using the report wizard, *both* the second row grouping and second column grouping are removed.
Tabular, Summary, or Matrix to Joined	The existing report becomes the first block in the joined report, and the report type becomes the principle report type for the joined report. Joined report blocks are formatted as summary reports, so if you switch from a summary to a joined report, your groupings stay the same. If you switch from a matrix to a joined report, groupings are converted the same way as when you switch from a matrix to a summary report. The following items aren't supported in joined reports, and aren't converted: • Bucket fields • Cross filters • The `Rows to Display` filter

SEE ALSO:

Build a Report in Salesforce Classic
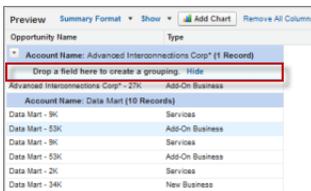
Group Your Report Data

Group data in columns or rows in summary, matrix, and joined reports to display meaningful information. For example, group opportunities by Close Date to see closed opportunities or group cases by product to see the number of cases for each product. You can have groupings inside groupings.

Data for Grouping

The Fields pane displays fields from the selected report type, organized by folder. Before you group data in a summary report, drag and drop at least a few fields into the preview pane.

Add a Grouping

Add a group by dropping a field onto a drop zone.

Click **Show** > **Drop Zones** to make them visible. You can also click a column menu for a field in the report and choose **Group by this Field**.

> 💡 Tip: If you group your report by a date field, you can click the group menu, select **Group Dates By**, and specify the grouping time frame: day, week, month, quarter, year, etc.

Grouping Data in Different Report Formats

Summary and joined reports can have up to three grouping levels. Matrix reports can have two row and two column groupings. For matrix reports, you can't use the same field for both row and column groupings. For joined reports, you can use the fields from the Common Fields category in the Fields pane to group across all report blocks.

Remove a Grouping

To remove a group, click the group menu and choose **Remove Group**. You can also grab the group and:

- Drag it to the column bar to remove the group, but keep the field as a column in the report.
- Drag it back to the Fields pane to remove the group and the field from the report.

Change Order of a Grouping

Drag groups to change their order, or click the group menu and choose **Move Group Up** or **Move Group Down** for column groupings, or **Move Group Left** or **Move Group Right** for row groupings.

SEE ALSO:

Report Fields

Subtotal Report Results

Keep Working While Your Report Preview Loads

For most actions, you can continue working on your report while the preview loads. For example, when editing a report you can drag multiple fields into the report, then create a grouping while those columns load.

The data you see in the Preview panel is real data, but it's only a subset of the data in the report, designed to show you what the report will look like when you run it. Don't worry if rows are missing or not sorted as expected in the Preview panel.

> **Note:** For each external object in the report, your org calls out to the external system each time the preview loads. If the URL of a report callout approaches or exceeds 2 KB, your org splits the request into multiple HTTP calls, with each URL being less than 2 KB.

You can continue working in the report preview while the following actions occur:

- Add, remove, or reorder fields
- Add or remove summary fields
- Add, remove, or reorder groupings
- Remove formulas

> **Note:**
> - If you remove a summary field or formula used in a chart, the chart reloads, but the report preview loads asynchronously.
> - Asynchronous loading isn't available for matrix and joined reports.

You can't work in the report preview while the following actions occur:

- Add or edit formulas
- Remove a report's only grouping
- Remove a column used to limit the row count for a tabular report
- Sort the report by group or column
- Update standard or custom filters
- Change report format
- Show or hide report details
- Add or remove a chart
- Add or remove conditional highlighting
- Change **Group Dates By**
- Change a converted currency field
- Save the report

SEE ALSO:

Build a Report in Salesforce Classic

Report Fields

Customizing Reports

Report builder is a visual editor for reports. The report builder screen lets you work with report fields and filters, and shows you a preview of your report with just some of the data.

To optimize screen real estate, report builder uses a compressed page header. To view your application tabs, simply close the builder or click the Salesforce logo.

Fields Pane (1)

The Fields pane displays fields from the selected report type, organized by folder. Find the fields you want using the Quick Find search box and field type filters, then drag them into the Preview pane to add them to the report.

Create, view, edit, and delete custom summary formulas and bucket fields in the Fields pane as well.

In the joined report format, the Fields pane displays fields from all report types added to the report, organized by report type.

Filters Pane (2)

Set the view, time frame, and custom filters to limit the data shown in the report.

Preview Pane (3)

The dynamic preview makes it easy for you to customize your report. Add, reorder, and remove columns, summary fields, formulas, groupings, and blocks. Change the report format and display options, or add a chart.

The preview shows only a limited number of records. Run the report to see all your results.

Watch a Demo: ▶ Getting Started with Report Builder (Salesforce Classic)

SEE ALSO:

Create a Report

Build a Report in Salesforce Classic

Report Fields

The Fields pane displays fields from the selected report type, organized by folder. It also lists custom summary formulas, which you can create, edit, and delete.

Adding Field Filters

With tabular, summary, and matrix reports, you can drag a field from the Fields pane to the Filters pane to add a report filter.

Finding Fields

Find a field by typing its name into the Quick Find search box. You can also filter the list by type:

- Click ▣ All to see all field types, as well as custom summary formulas.
- Click ▣ to see just text fields.
- Click ▣ # to see just number fields (numeric, percentage, or currency).
- Click ▣ to see just date fields.

Adding and Removing Fields

To add a field to a tabular, summary, or matrix report, double-click it or drag it into the Preview pane. To add a field to a joined report, drag it to the Preview pane. Press **CTRL** to select multiple fields. Drag an entire folder to add all its fields. If a tabular, summary, or matrix report already contains a field, you can't add it again. You can add the same field multiple times to a joined report as long as you add it to different blocks. In the preview pane, click **Show** > **Details** to see your report fields. While **Show** > **Details** is disabled, you can only add summary fields.

To remove a field, grab its column header and drag it back to the Fields pane. With tabular, summary, and matrix reports, you can click the column menu and choose **Remove Column**, or click **Remove All Columns**.

Working with More than One Field

You can select multiple fields to add, remove, or reorder. For example, you can add `Created By, Type` and `Opportunity Name` to your report at the same time.

To select multiple fields or columns, press CTRL (Windows) or Command (Mac).

📝 Note:

- When you add multiple fields, they appear in the report in the order selected.
- For summary and tabular formats, disable **Show** > **Details** when adding multiple summarizable fields to a report. The Summarize dialog automatically appears, letting you select summaries for all the fields at once.
- Enable **Show** > **Details** when adding non-summary fields, such as text fields to a report.
- When working with a joined report, you can select multiple fields from the Common Fields category and one report type. You can't select multiple fields from different report types.

Ordering and Sorting Fields

Reorder report columns by grabbing a column header and dragging it to a new location. Press **CTRL** to select multiple columns. To sort your report by a column, click its column header. You can also click the column menu and choose **Sort Ascending** or **Sort Descending** from the drop-down list. Sort is disabled when **Show** > **Details** isn't selected.

Changing the Currency Displayed

If your organization has enabled multiple currencies, you can change the currency shown for all currency fields. Click **Show** > **Display Currencies Using**, then select an active currency to display.

SEE ALSO:

Evaluate Groups and Totals with Summary Formulas

Group Your Report Data

Summarize Report Data

Highlight Data Ranges with Conditional Formatting

Subtotal Report Results

Subtotaling your reports gives you a tool to analyze trends in the data. You can group sets of information, sort the groupings, and compare subtotals for each set against the overall total. In summary and joined reports, you can also subtotal by multiple fields to give you cascading sets of information.

For example, if you subtotal a summary report by Opportunity Owner, the report groups the accounts by Opportunity Owner, lists the number of opportunities owned by each user, and shows subtotals by Opportunity Owner for all the columns in the report. You could further subtotal each user's opportunities by product.

1. Click **Customize** or **Edit** from any report.

2. In the report builder, add a summary field to the report.
   Summaries show up at grouping levels as well as for individual rows.

SEE ALSO:

Add a Summary Formula Column to a Report

Build a Report in Salesforce Classic

Subtotal Report Results

Smart Totaling in Reports

"Smart" totaling means that duplicate data is counted only once in any subtotal or total. Salesforce uses "smart" totaling when you run reports that include duplicate data in any of the columns chosen for summing or averaging.

For example, suppose an opportunity has two products, and you run the Opportunity Product Report with the total opportunity amount selected as a column to sum by. The amount appears twice in the details of the report, once for each product on the opportunity.

In this case, "smart" totaling correctly calculates any subtotals, grand totals, and averages, adding that opportunity amount only once.

EDITIONS

Available in: Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

USER PERMISSIONS

To subtotal report results:
- **Legacy Folder Sharing**
  Create and Customize Reports

  AND

  Report Builder
- **Enhanced Folder Sharing**
  Create and Customize Reports

  AND

  Report Builder

> 📝 **Note:** For dashboard components, "smart" totaling isn't used. For example, in a dashboard table, the total displayed is simply the sum of the values listed in the table.

SEE ALSO:

Subtotal Report Results

Subtotal Report Results

Save Your Report

Click **Save** to update an existing report with recent changes, or **Save As** to clone the original report without changing it. In Lightning Experience, click **Clone** to clone the report.

1. Verify the name, description, and folder, then choose where to go next:
   - Click **Save** to save the report and go to the Reports home page
   - Click **Save & Return to Report** to save it and go back to the report run page.

   > 📝 **Note:** To save your report at a specific role hierarchy drill-down level on sales, forecast, opportunity, and activity reports, select `Save Hierarchy Level`.

> 💡 **Tip:** If you add a colon to your report name, it displays in two separate lines when you view the report. Use this to categorize reports by name, or better display long names. For example, if you enter `First Line: Second Line` for `Report Name`, you'll see this on the run page:

Show Report Data Graphically

To help readers understand your data quickly and easily, show the data in chart form. Charts appear just above the report table. They can help users get a feel for the data before they delve into the details. Use line charts to track changes over time, or a bar or pie chart to compare values at a point in time. Charts can also appear in dashboard components.

1. Add a Chart to a Report

   Add a chart to give users a visual way to understand the data in your report.

2. Chart Properties

   You can add a chart to any standard or custom summary or matrix report. The chart properties specify the data that appears in the chart, its labels and colors, and any conditional highlighting you apply.

3. Present Data Effectively with Charts

   When you add a chart to a report, things like negative values, very large or small numbers, custom summary formulas, and field-level security can affect the charts' appearance. Switching the report format and changing groupings and blocks also affects charts.

4. Show Multiple Sets of Data in One Chart

   A *combination chart* plots multiple sets of data on a single chart. Each set of data is based on a different field, so values are easy to compare. You can also combine certain chart types to present data in different ways in a single chart.

5. Combination Chart Examples

   Use a combination chart to show multiple values against a single axis range, show two chart types together, or compare two continuous summary values.

6. Chart Formatting Options

SEE ALSO:

Present Data Effectively with Charts

Add a Chart to a Report

Chart Properties

Combination Chart Examples

Show Multiple Sets of Data in One Chart

Add a Chart to a Report

Add a chart to give users a visual way to understand the data in your report.

> 📝 **Note:** Your report must have at least one grouping before you can add a chart.

In Lightning Experience, add or edit a chart from the Report Run Page.

1. To show or hide the chart, click ⟳ .

2. To edit the chart, click ✱ .

   From the chart editor, change chart type, give the chart a title, change axes, show or hide a reference line, and show or hide chart values.

In Salesforce Classic, add or edit a chart from the Report Builder.

1. Click **Add Chart** in report builder. For existing charts, click **Edit Chart**.

2. Select a chart type.

3. Enter the appropriate settings on the Chart Data tab for the chart type you selected.

4. Enter the appropriate settings on the Formatting tab.

5. Click **OK**.

SEE ALSO:

   Chart Types

   Present Data Effectively with Charts

   Build a Report in Salesforce Classic

EDITIONS

Available in: Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

USER PERMISSIONS

To create, edit, and delete reports:

- **Legacy Folder Sharing**
  Create and Customize Reports

  AND

  Report Builder

  Enhanced Folder Sharing
  Create and Customize Reports

  AND

  Report Builder

Chart Properties

You can add a chart to any standard or custom summary or matrix report. The chart properties specify the data that appears in the chart, its labels and colors, and any conditional highlighting you apply.

📝 **Note:** This topic only applies if you're not using report builder. *Report builder* is a visual editor for reports.

To customize chart properties, click **Add Chart** or **Edit Chart** in any matrix or summary report, and use the fields on the Chart Data and Formatting tabs.

See Present Data Effectively with Charts on page 26 for limits, considerations, and tips.

Chart Data Settings

Control the data that appears in your chart with these options.

Field	Description
Chart Type	Select the type of chart to use when representing the data in your report. The chart type that you choose determines which chart properties are available to set.
X-Axis and Y-Axis	Choose what values to display on the axes of your chart. Depending on the chart type, axis values can be record count, summary fields, or groupings defined in the report.
	📝 **Note:** If the Y-axis corresponds to a custom summary formula that has the `Where Will this Formula Be Displayed?` option set to a grouping level other than `All summary levels`, then the X-axis and Groupings selection must correspond to that custom summary formula's grouping level.
Combination Chart	Select this option to plot additional values on this chart. The chart type you chose must allow combination charts.
Groupings	Choose how to group information on your chart. You can only pick from groupings defined in the report. For bar and column charts, click an icon to select the grouping display: side-by-side, stacked, or stacked to 100%. For either single or grouped line charts, you can select `Cumulative`.
Values	Choose what to display as values for your pie, donut, or funnel chart.
Wedges	Choose what to display as wedges for your pie or donut chart.
Segments	Choose what to display as segments for your funnel chart.

Chart Presentation

Control the appearance and behavior of your chart using these options.

Field	Description
Chart Title	Enter a name for the chart.
Title Color	Select the color for the text of your chart title.
Title Size	Select the font size for the text of your chart title. The maximum size is 18. Larger values are shown at 18 points.
Text Color	Select a color for all the text and labels in your chart.
Text Size	Select a font size for all the text and labels in your chart. The maximum size is 18. Larger values are shown at 18 points.
Background Fade	Choose a direction for a gradient color background. Also select a `Start Color` and `End Color` for the gradient. Use white for both if you do not want a background design.
Legend Position	Choose a place to display the legend in relation to your chart.
Combine Small Groups into "Others"	Combine all groups less than or equal to 3% of the total into a single "Others" wedge or segment. Deselect to show all values individually on the chart. This only applies to pie, donut, and funnel charts. This option is on by default for pie and donut charts, and off for funnel.
X- or Y-Axis Range	Choose a manual or automatic axis range for bar, line, or column charts. If you choose manual, enter numbers for the minimum and maximum axis values to be displayed. If there are data points outside the range that you set, the axis automatically extends to include those values when you generate the chart.
Show Axis Labels	Display labels for each axis of your chart. This only applies to bar and line charts.
Show Labels	Display labels for your pie, donut, or funnel chart.
Show Group %	Display the percentage value for each group in the chart.
Show X- or Y-Axis Values	Display the values of individual records or groups on the chart axis. This only applies to certain horizontal bar and vertical column charts.
Show Values	Display the values of individual records or groups on the chart. This only applies to certain chart types.
Show Wedge %	Display the percentage value for each wedge of pie and donut charts.
Show Total	Display the total value for the donut chart.
Show Segment %	Display the percentage value for each segment of funnel charts.
Show Details on Hover	Display values, labels, and percentages when hovering over charts. Hover details depend on chart type. Percentages apply to pie, donut, and funnel charts only. Hover is disabled when viewing charts that have more than 200 data points.
Chart Size	Select a size for the chart, from Tiny to Extra Large.
Chart Position	Place the chart above or below your report.

Conditional Highlighting

Highlight field values reports based on ranges and colors you specify. You can apply conditional formatting to reports that are grouped by rows (summary reports) or grouped by rows and columns (matrix reports).

Field	Description
Summary	Choose a summary field whose number ranges you want represented by colors.
Low Color	Select a color to represent data that falls below the Low Breakpoint value.
Low Breakpoint	The number that acts as the threshold between the Low Color and the Mid Color. Values that are the same as the Low Breakpoint value are shown as the Mid Color.
Mid Color	Select a color to represent data that falls between the Low Breakpoint and High Breakpoint values.
High Breakpoint	The number that acts as the threshold between the Mid Color and the High Color. Values that are the same as the High Breakpoint value are shown as the High Color.
High Color	Select a color to represent data that falls above the High Breakpoint value.

SEE ALSO:

Chart Types

Customizing Reports

Present Data Effectively with Charts

When you add a chart to a report, things like negative values, very large or small numbers, custom summary formulas, and field-level security can affect the charts' appearance. Switching the report format and changing groupings and blocks also affects charts.

- You can't have more than 250 groups or 4,000 values in a chart. If you see an error message saying that your chart has too many groups or values to plot, adjust the report filters to reduce the number. In combination charts, all groups and values count against the total.

- If you lose access to a field used in a chart, another field may be used in its place. If no other fields are available, record count is used.

- Decimal-place precision on charts is not customizable. Numeric and currency values round to two decimal places. Percentage values round to one decimal place.

- If numeric values are too large or too small, they are shown in scientific notation. For example, the number 5,750,000,000 is displayed as 5.75E9; -0.0000000061 is displayed as -6.1E-9.

- Negative values are displayed on all line charts and bar and column charts.

  In Salesforce Classic, negative values don't display on pie, donut, and funnel charts. Groupings containing negative values display in the legend, and negative values are reflected in the calculation of all summary values, including the total for donut charts.

  In Lightning Experience, negative values display on donut and funnel charts.

- When creating charts, don't group by a field on a child object then sum by a field on the parent object. It's not good practice. If you do this with a donut chart, the total shown may not match the sum of the wedges.

- In Salesforce Classic, dashboard and report charts that display values from summary formulas display decimal places using your default currency setting instead of what you specified for the formula. For example, if the summary formula specifies zero decimal places, no decimal places appear in columns, but chart values show the number of decimal places specified for your default currency (usually two decimal places). This limit applies to currencies, numbers, and percentages.

- With joined reports, summary field names contain both the field name and the block name. For example, if you've summarized the Amount field in Block 1, it appears as `Block 1 — Sum of Amount` in the Chart Editor. A cross-block or standard custom summary formula contains the block name when the formula is included in multiple blocks.

- When a report already has a chart, changing the report format or removing blocks, groupings, or summary fields has impacts described in this table.

When you make this change ...	The effect is ...
Switch format from summary, matrix, or joined to tabular	All charts are removed from the report.
Switch format from summary to matrix	The first summary grouping becomes the first row grouping. The second becomes the first column grouping. The third becomes the second row grouping. The chart is unchanged.
Switch format from matrix to summary	The first row grouping becomes the first summary grouping. The second row grouping becomes the *third* summary grouping. The first column grouping becomes the *second* summary grouping. The second column grouping is removed. If the chart used the second column grouping, that grouping is replaced by the first available summary field that's not already used in the chart.
Switch format from summary or matrix to joined	The existing report becomes the first block and the chart remains in the report.
	If the matrix report used the second column grouping, that grouping is replaced by the first available summary field that's not already used in the chart.
	If the summary or matrix report included a grouping from a bucket field, the grouping is replaced with the next available grouping not already used in the chart.
Switch format from joined to summary or matrix	The first block becomes the report, and groupings and the chart are preserved. If the chart in the joined version contained summary fields from the first block, they remain in the chart in the summary report. Summaries from other blocks are removed from the chart.
Remove a block containing a summary field from a joined report	The summary field is replaced with the next available summary field. If no additional summary fields are available, the original field is replaced with the record count from the first remaining block. Removing all blocks from a report removes the chart completely.
Remove a grouping used in the chart	The grouping is replaced with the next available grouping. If all groupings are removed, the chart is removed as well. Note that groupings can be removed automatically when you add a report type to a joined report or if a field becomes unavailable, for example as a result of field-level security.
Remove a summary field used in the chart (including custom and cross-block summaries)	The summary field is replaced with the next available summary field. If no additional summary fields are available, the field is replaced with the record count for either the

When you make this change ...	The effect is ...
	report or the first block. Note that summary fields can be removed automatically, for example as a result of field-level security or when the report format changes.

SEE ALSO:

Add a Chart to a Report

Chart Formatting Options

Show Multiple Sets of Data in One Chart

Show Multiple Sets of Data in One Chart

A *combination chart* plots multiple sets of data on a single chart. Each set of data is based on a different field, so values are easy to compare. You can also combine certain chart types to present data in different ways in a single chart.

With combination charts, you can:

- Add a line to an existing line, vertical column, grouped vertical column, or stacked vertical column chart.
- Add a cumulative line to an existing line cumulative chart.
- Add up to three columns to a vertical column chart.
- Add up to three bars to a horizontal bar chart.

For example, as a sales manager, you can display "Pipeline amount" as a line and "Number of open deals" as vertical bars on the same chart.

In Lightning Experience, to display multiple measures on the same chart, follow these steps:

1. Edit the chart for any report grouped by rows (summary report) or grouped by rows and columns (matrix report).

2. Verify that the fields are summarized and of the same data type.

3. Click [icon] to add a chart to the report.

4. To edit the chart, click [icon].

5. Choose the Bar, Column, or Line chart.

6. The X axis for the chart is the field for the row grouping. If the report is grouped by multiple rows, select the one to use as the X axis.

7. For the Y axis, select a measure field.

8. To display a reference line, click **Show Reference Line** and enter the reference value.

9. If your chart is a line chart and you want to display cumulative values, select **Cumulative**.

10. To add another measure of the same type, click **+ Measure** and select the measure field to add. For bar chart and column charts, you can add up to 3 additional measures. If your chart is a column chart, you can display the second measure as a line chart. Select **Plot as Line Chart**. To display a separate axis for the line chart on the right, select **Plot on Second Axis**.

11. After adding measures and modifying any other settings, click **Run Report** or **Save**.

In Salesforce Classic, to display multiple measures on the same chart, follow these steps:

1. Edit the chart for any summary or matrix report, or edit a dashboard component that displays a summary or matrix report.

2. Choose a chart type that allows combination charts:

   - Bar chart
   - Column chart
   - Grouped column chart
   - Stacked column chart
   - Line chart
   - Cumulative line chart

3. Select the **Plot additional values** checkbox. The chart preview updates as you configure your combination chart.

4. To plot on the chart, select a `Value`.

5. Choose a `Display` option. Available options differ based on your chart type and whether you're editing a chart or a dashboard component.

   - For columns or bars, click **Add Bar** or **Add Column** links to add up to three sets.
   - When adding a line to a vertical column chart, select **Use second axis** to show a separate axis for the added line on the right side of the column chart. A separate axis can be useful when the two values have different ranges or units.

     Note: Selecting **Use second axis** makes more values available in the `Value` dropdown list. Without this option, you can only pick from values of the same type as the primary Y axis—for example, number, currency, or percentage. This option is only available for certain combination charts.

6. When you've finished setting up your report, click **Run Report** or **Save**.

   Note:
   - If you lose access to a field used in a chart, another field can be used in its place. If no other fields are available, the record count is used.
   - Don't set both the X axis and Y axis to `Auto`. If they're both set to `Auto`, you can't plot the chart.
   - Filtered drilldown doesn't work for combination charts in dashboards.

SEE ALSO:
   Chart Types
   Add a Chart to a Report
   Combination Chart Examples

Combination Chart Examples

Use a combination chart to show multiple values against a single axis range, show two chart types together, or compare two continuous summary values.

A *combination chart* plots multiple sets of data on a single chart.

Column-on-Column
   Add columns to a column chart to show multiple values against a single axis range.

   To create the chart in this example, choose the Vertical Column chart type, set the opportunity sum of amount as the `Y-Axis`, stage as the `X-Axis`, and use the **Plot additional values** option to add the sum of expected revenue as a column.

You can quickly compare the actual values against the expected values for each stage.
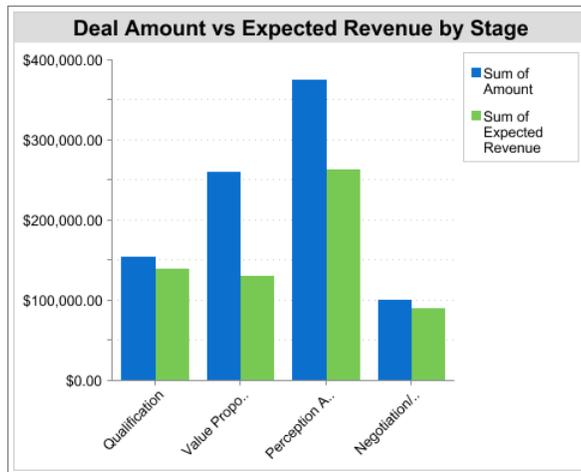

Line-on-Column

Add a line to a column chart to show two chart types together. Using a second axis allows you to add different types of values to the chart.

To create the chart in this example, choose the Vertical Column chart type, set the opportunity sum of amount as the `Y-Axis`, fiscal year as the `X-Axis`, and use the **Plot additional values** option to add the number of deals as a line. Summary values of different types won't be available in the `Values` drop-down list unless you select **Use second axis**.

You can see both the total amount and number of deals for each year on a single chart.
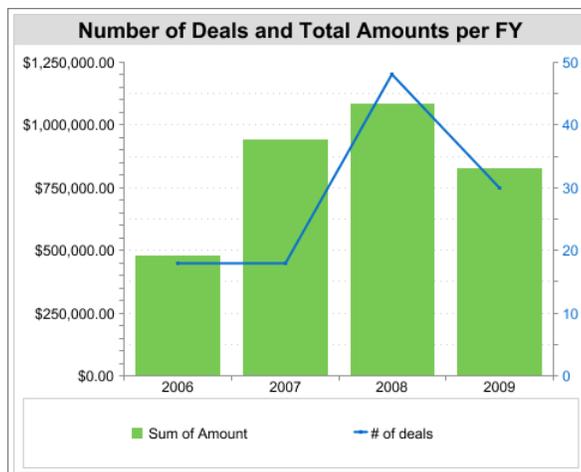
Line-on-Line

Add a line to a line chart to compare two continuous summary values.

To create the chart in this example, set up a custom summary formula to calculate a three-week moving average of opportunity amounts, then choose the Line chart type, set the opportunity sum of amount as the `Y-Axis`, date as the `X-Axis`, and use the **Plot additional values** option to add the calculated three-week moving average as a line.

You can compare sales against the moving average over time.
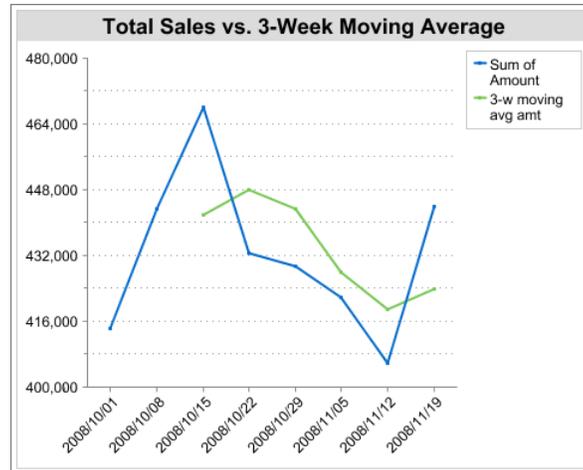
The custom summary formula used in this example is shown here:

```
(OppProductTrends__c.Amount__c:SUM+ PREVGROUPVAL(OppProductTrends__c.Amount__c:SUM,
OppProductTrends__c.as_of_date__c) + PREVGROUPVAL(OppProductTrends__c.Amount__c:SUM,
OppProductTrends__c.as_of_date__c,2))/3
```

SEE ALSO:

Show Multiple Sets of Data in One Chart

Chart Formatting Options

Control the appearance and behavior of your chart using these options.

Field	Description
Chart Title	Enter a name for the chart.
Title Color	Select the color for the text of your chart title.
Title Size	Select the font size for the text of your chart title. The maximum size is 18. Larger values are shown at 18 points.
Text Color	Select a color for all the text and labels in your chart.
Text Size	Select a font size for all the text and labels in your chart. The maximum size is 18. Larger values are shown at 18 points.
Background Fade	Choose a direction for a gradient color background. Also select a `Start Color` and `End Color` for the gradient. Use white for both if you do not want a background design.
Legend Position	Choose a place to display the legend in relation to your chart.
Combine Small Groups into "Others"	Combine all groups less than or equal to 3% of the total into a single "Others" wedge or segment. Deselect to show all values individually on the chart. This only applies to pie, donut, and funnel charts. This option is on by default for pie and donut charts, and off for funnel.
X- or Y-Axis Range	Choose a manual or automatic axis range for bar, line, or column charts. If you choose manual, enter numbers for the minimum and maximum axis values to be displayed. If there are data points outside

Field	Description
	the range that you set, the axis automatically extends to include those values when you generate the chart.
Show Axis Labels	Display labels for each axis of your chart. This only applies to bar and line charts.
Show Labels	Display labels for your pie, donut, or funnel chart.
Show Group %	Display the percentage value for each group in the chart.
Show X- or Y-Axis Values	Display the values of individual records or groups on the chart axis. This only applies to certain horizontal bar and vertical column charts.
Show Values	Display the values of individual records or groups on the chart. This only applies to certain chart types.
Show Wedge %	Display the percentage value for each wedge of pie and donut charts.
Show Total	Display the total value for the donut chart.
Show Segment %	Display the percentage value for each segment of funnel charts.
Show Details on Hover	Display values, labels, and percentages when hovering over charts. Hover details depend on chart type. Percentages apply to pie, donut, and funnel charts only. Hover is disabled when viewing charts that have more than 200 data points.
Chart Size	Select a size for the chart, from Tiny to Extra Large.
Chart Position	Place the chart above or below your report.

SEE ALSO:

> Add a Chart to a Report

Show Report Data in Tables

To help readers scan for data easily, try hiding details and ranges, limiting the number of results shown, and highlighting with color. You can also show your table in a dashboard component.

1. Show and Hide Report Details

    You can show or hide report details from either the run reports page or the report builder. When you hide details, individual records don't display in the report. Groupings, summary formulas, and record counts remain visible.

2. Highlight Data Ranges with Conditional Formatting

    Highlight field values reports based on ranges and colors you specify. You can apply conditional formatting to reports that are grouped by rows (summary reports) or grouped by rows and columns (matrix reports).

3. Use a Tabular Report in a Dashboard

    You can use a tabular report as the source report for a dashboard table or chart component, if you limit the number of rows it returns.

4. Limit Report Results

    Set limits to the scope of your report to avoid processing too many records. The built-in choices for limiting your results vary according to the object you are reporting on.

SEE ALSO:

Highlight Data Ranges with Conditional Formatting

Limit Report Results

Show and Hide Report Details

Use a Tabular Report in a Dashboard

Show and Hide Report Details

You can show or hide report details from either the run reports page or the report builder. When you hide details, individual records don't display in the report. Groupings, summary formulas, and record counts remain visible.

- From the run reports page, click **Hide Details** to hide individual records. Click **Show Details** to show all records.

- From the report builder, click **Show** > **Details**. A check mark beside the **Details** menu item means that details are displayed. Click **Details** to toggle between showing or hiding records.

SEE ALSO:

Show and Hide the Record Count for a Block

Build a Report in Salesforce Classic

Combine Different Types of Information in a Joined Report

Highlight Data Ranges with Conditional Formatting

Highlight field values reports based on ranges and colors you specify. You can apply conditional
formatting to reports that are grouped by rows (summary reports) or grouped by rows and columns
(matrix reports).

In Salesforce Classic, you can set up to three formatting rules. To set conditional highlighting, click
Show > **Conditional Highlighting** in report builder, then set the breakpoint values and their
range colors as follows:

Field	Description
Summary	Choose a summary field whose number ranges you want represented by colors.
Low Color	Select a color to represent data that falls below the `Low Breakpoint` value.
Low Breakpoint	The number that acts as the threshold between the `Low Color` and the `Mid Color`. Values that are the same as the `Low Breakpoint` value are shown as the `Mid Color`.
Mid Color	Select a color to represent data that falls between the `Low Breakpoint` and `High Breakpoint` values.
High Breakpoint	The number that acts as the threshold between the `Mid Color` and the `High Color`. Values that are the same as the `High Breakpoint` value are shown as the `High Color`.
High Color	Select a color to represent data that falls above the `High Breakpoint` value.

In Lightning Experience, you can set up to five rules and define custom colors for each range. To
enable conditional highlighting, your report must be a summary report (grouped by rows) or matrix
report (grouped by rows and columns). It must also contain at least one summary field or custom
summary formula.

Click **Conditional Formatting**. In Add Conditional Formatting Rule, select a summary or custom
summary formula field that serves as a KPI for your business. You can also apply conditional
formatting to grand totals in a matrix report. Set the breakpoint values and their range colors for each bin. When you're ready to apply
the conditional formatting rule, click **Done**.

Notes:

- When you apply more than three conditional formatting rules to a report, conditional formatting can be viewed and edited only in
  Lightning Experience.

- You can subscribe to reports with conditional highlighting, but conditional highlighting doesn't appear in the subscription email.

- The rounding of data values for conditional formatting differs slightly for Lightning Experience versus Salesforce Classic. The following
  table shows some sample differences.

Range	Interpretation in Salesforce Classic	Interpretation in Lightning Experience
<=5	All negative values plus positive values up to but not including 5.0	All negative values plus positive values up to and including 5.99999999

Range	Interpretation in Salesforce Classic	Interpretation in Lightning Experience
Classic: Between 5 and 6 Lightning Experience: > 5 to 6	5.00000001 to 6.0	6.0 to 6.99999999
> 6	7.0 and greater	7.0 and greater

SEE ALSO:

Build a Report in Salesforce Classic

Add a Summary Formula Column to a Report

Use a Tabular Report in a Dashboard

You can use a tabular report as the source report for a dashboard table or chart component, if you limit the number of rows it returns.

1. Click **Add** > **Row Limit**.

2. Set the `Row Limit` to *10*, *25*, or *Custom*. If you choose custom enter a number between one and 99.

3. Set the `Sort By` and sort order options. If you chose **Limit Rows by this Field** for a column, these options are already set.

4. Click **OK**.

5. Click **Dashboard Settings** in the toolbar.

6. Choose a `Name` and `Value` to use in dashboard tables and charts. Tables show both name and value. Charts are grouped by name.

7. Click **OK**. You can now use this tabular report as the source report for a dashboard component.

> 💡 Tip: When you create a dashboard component to display your tabular report, you can use the dashboard component editor to override the settings you chose in **Dashboard Settings**.

Limit Report Results

Set limits to the scope of your report to avoid processing too many records. The built-in choices for limiting your results vary according to the object you are reporting on.

- To see a collapsed view of a report showing only the headings, subtotals, and total in report builder, deselect **Show** > **Details**.

  On the report run page, click **Hide Details** or **Show Details** at the top of the report.

- To filter by a field, click **Add** > **Field Filter**. With tabular, summary, and matrix reports, you can drag a field from the Fields pane to the Filters pane to add a report filter.

- As you create and work with cross filters, take note of these considerations and limits. To add one, click **Add** > **Cross Filter**.

- In Professional, Enterprise, Unlimited, Performance, and Developer edition organizations, the **Hierarchy** links let you browse report results based on the role or territory hierarchies.

- If your organization uses divisions to segment data and you have the "Affected by Divisions" permission, use the Division drop-down list to include records in just one division or all divisions. Select `Current` to show records in your current working division. Reports that are already scoped (such as My Cases or My team's accounts) include records in all divisions, and you can't further limit them to a specific division. If you do not have the "Affected by Divisions" permission, your reports include records in all divisions.

- You can set the maximum number of records to display in a tabular report by clicking **Add** > **Row Limit** in report builder. Set the number of rows, then choose a field to sort by, and the sort order. Limiting rows on a tabular report allows you to use it as a source report for dashboard table and chart components.

  The `Row Limit` option on tabular reports shows only fields from the primary object on reports created from custom report types where object A may or may not have object B. For example, in an accounts with or without contacts report, only fields from accounts are shown. Fields from objects after a may-or-may-not association on custom report types aren't shown. For example, in an accounts with contacts with or without cases report, only fields from accounts and contacts are available to use. If you change the report format, **Row Limit** settings are lost.

> 📝 **Note:** Only the first 255 characters in a custom text field count for filtering purposes. For example, if you add a field filter to find opportunities where the custom text field `Customer notes` includes the word "phone," but "phone" appears after the 255th character in the field, the filter will not find that record. In standard text fields, all characters count, regardless of the length of the field.

SEE ALSO:

Filter Across Objects with Cross Filters

Categorize Data with Bucket Columns

Quickly categorize report records without creating a formula or a custom field by bucketing them. When you create a bucket column, you define multiple categories (buckets) used to group report values. Like any other column in your report, you can sort, filter, and group by bucket columns.

Watch a Demo: ▶ Getting Started with Buckets (Salesforce Classic)

For example, get a view of your accounts based on how many employees they have. Create a bucket column named `Size` based on the `# Employees` field. Then, create buckets that group records into Small, Medium, and Large ranges that you define. Small includes accounts with 5,000 or less employees. Medium includes accounts with 5,000 to 10,000 employees. Large includes accounts with more than 10,000 employees. Now you can sort, filter, or group records based on how they're grouped in buckets. If one account grows or shrinks, it'll automatically switch buckets.

> 📝 **Note:** If your report is of a custom type and object and you create a bucket column with a total of more than 1000 characters across the bucket values, the buckets may not appear when you group by columns in dashboards that use the report.

1. Add a Bucket Column

   Create a bucket column while editing a report.

2. Edit a Bucket Column

   When it's time to add, remove, or change buckets in a bucket column, edit it.

3. Delete a Bucket Column

   When you no longer need a bucket column, delete it.

Add a Bucket Column

Create a bucket column while editing a report.

1. Edit a report.

2. Find the column you want to bucket in the report preview, then click ▼ > **Bucket This Column**.

    Alternatively, from the Columns section, click ▼ > **Add Bucket Column**.

    The Edit Bucket Column menu opens.

3. From **Field**, choose a field from the report type. The bucket column is based on the field you choose. If you started bucketing a column in the report preview, then the field is already set.

4. From **Bucket Name**, enter a name for the bucket column.

5. Add buckets and choose values for each bucket.

    Depending on the data type of the column you're bucketing, you see different bucket options in the Edit Bucket Column menu. You can bucket 3 data types: numeric, picklist, and text.

    Numeric

    A numeric bucket helps you sort data that can be described in terms of numbers. Numeric buckets include columns with field types like `currency` (Amount), `number` (Number of Employees), and `percent` (Probability), but not necessarily mixed alphanumeric field types like `auto number`. Alphanumeric values don't reside on the number line, which means they can't get numerically evaluated, which prevents them from getting bucketed in a numeric bucket. Alphanumeric values can be included in text buckets.

    Each numeric bucket includes a range and a name.

    To gain insight into your deals, use bucketing to group by deal size instead of looking at individual deals. For example, create 3 buckets based on deal size: Small, Medium, and Large. Small holds deals worth $1,000 or less. Medium holds deals worth between $1,000 and $25,000 dollars. Large holds deals worth more than $25,000.

    a. To create a numeric bucket, click **Add**. Each numeric bucket consists of a range and a name.

    b. From **RANGE**, enter the lower-bound and upper-bound breakpoints for ranges. The first number entered buckets values equal to or less than the number you enter. Each subsequent number you enter buckets values greater than the prior number you enter. The last number entered buckets values greater than the number you enter.

    c. From **BUCKET**, enter a name for the bucket.

    d. Optionally, to move all empty or null values to the bucket that contains the value zero, select **Treat empty values in the report as zeroes**. If deselected, unbucketed values appear as a dash (-).

Picklist

A picklist bucket contains items that can be selected from a list.

To get a generalized view of your accounts, bucket them by industry. Create buckets with industry names like Manufacturing, Technology, and Media. Then, add accounts into the bucket that best represents their operation.

    a. To create a picklist bucket, click **Add Bucket**. Then, give the new bucket a name.

    b. To put values in a bucket, select the values you want to move, then click **Move To** and select bucket you want to move them into. To move values into a new bucket, click **Move To** > **New Bucket**, enter a name for the new bucket.

    c. To take values out of a bucket, click **Move To** > **Unbucketed Values**. Unbucketed values appear in the bucket column as they would in a normal report column. You can put them all into a bucket named Other by enabling **Bucket remaining values as Other**.

    d. To rename a bucket, click ✏️ . To delete a bucket, click 🗑️ .

Text

At first glance, text buckets seem to have the same options as picklist buckets. Values of the text column appear as though they were values in a picklist, and you can add them into buckets in much the same way.

Unlike picklist fields, text fields feature the **Enter Values** button. If you know which value you want to bucket, you can use the Enter Values function to quickly bucket it without searching. Entering values is useful if your report has a large number (such as millions) of values and searching for a value is slow. You can also use this method to enter and bucket values that appear in the report later.

6. Click **Apply**.

7. Click **Save**.

The bucket column appears in the report preview and in the Columns section. Like any other column in the report, you can sort, filter, and group by the bucket column.

To edit the bucket column, find the column in the report preview and click 🔽 > **Edit Bucket Column**. To delete the bucket column, click 🔽 > **Delete Bucket Column**.

As you prepare to add bucket columns to your report, take note of these limitations:

- Each report can include up to 5 bucket fields.
- Each bucket field can contain up to 20 buckets.
- Each bucket can contain up to 20 values.
- Bucket columns are available for use only in the report where they're generated. To use a bucket in multiple reports, create the bucket column for each report. Or create a separate formula field for the object that's dependent on the bucket.
- Buckets and bucket fields aren't available for reports that include external objects.
- If a bucket column's source column has a custom index, and you filter by the bucket column, then the custom index performance gains are lost.
- If a bucket column has a filter on it, modifying the values in the buckets removes any filters that refer to the bucket column.
- Bucket columns don't support picklist values that contain a tab character.

Edit a Bucket Column

When it's time to add, remove, or change buckets in a bucket column, edit it.

1. Edit the report.

2. In Lightning Experience, from the report preview, click ▼ > **Edit Bucket Column**.

   In Salesforce Classic, from the Fields pane under Bucket Fields, hover over a bucket field and click 🖉 . Or in the Preview pane, click the bucket field column menu and select **Edit Bucket Field**.

   The Edit Bucket Column menu opens.

3. Add, remove, or change buckets and bucket values.

4. Click **Apply**.

5. Click **Save**.

For more information about the Edit Bucket Column menu, see Add a Bucket Column in Salesforce Help.

Delete a Bucket Column

When you no longer need a bucket column, delete it.

Deleting a bucket column does not affect the source fields. Deleting a bucket column from the report preview also deletes it from the Columns section. If you accidentally delete a bucket column and need to retrieve it, undo the delete action.

1. Edit the report.

2. From the report preview, find the bucket column you want to delete and click ▼ > **Delete Bucket Column**. Alternatively, find the bucket column in the Columns section and click **X**.

3. Click **Save**.

The bucket column is deleted.

Evaluate Report Data with Formulas

Reports feature built-in basic math functions (sum, average, min, and max) that you can apply to any numerical column in a report. When you're ready to perform advanced logical or mathematical operations, write a formula.

There are three types of formulas, *summaries*, *summary formulas*, and *row-level formulas*. The type of formula you write depends in part on where you want that formula to apply.

Summaries

Summaries are mathematical functions that you can apply to numeric columns in a report. Apply summaries to find the sum, average, minimum, or maximum value. Summaries evaluate data in and show subtotals for each report group, plus a grand total for all data in the report. No need to write a formula to find these essential math functions; instead, apply them from a numeric column's More Actions menu.

Apply a summary to answer business questions like these:

- What is the total value of all our open opportunities?
- What is the average age of all our cases?
- Which of our accounts has the most employees, and which has the least?

Row-level Formulas

Row-level formulas apply to every record in a report, and can evaluate data mathematically or logically. For example, use a row-level formula in an opportunity report to subtract `Created Date` from `Close Date` to how long it took to close each opportunity. Write another row-level formula to flag each opportunity with an emoticon based on whether customer satisfaction score (CSAT) is above 9, above 6, or below 6.

Write a row-level formula to answer business questions like these:

- How many days did it take each opportunity to close?
- Which case subjects contain the word "widget"?
- As defined by a customer satisfaction (CSAT) score, which customers are satisfied or upset?

Summary Formulas

Summary formulas evaluate data in either a specified group, all groups, or all groups and the entire report. If an opportunity report is grouped by `Type` and `Stage`, a summary function can apply to any of these places:

- Just to `Type`
- Just to `Stage`
- Just to the grand total of all report data
- To `Type`, `Stage` and the grand total of all report data

Summary formulas are perfect for comparing grouped data to other groups or to report totals. For example, write a summary formula to evaluate cases by source or product.

Write a summary formula to answer business questions like these:

- After deducting taxes and expenses, what are our net earnings?
- What percentage of our opportunities are attributable to each lead source?

PREVGROUPVAL() and PARENTGROUPVAL()

PREVGROUPVAL() and *PARENTGROUPVAL()* are two types of special summary formulas that are particular useful in reporting. Use PREVGROUPVAL() to compare one group to other groups in the report. Use PARENTGROUPVAL() to compare a child group to a parent group or to a report total.

Write summary formulas with PREVGROUPVAL() and PARENTGROUPVAL() to answer business questions like these:

- PREVGROUPVAL(): How much more or less business did we do in February compared to January?
- PARENTGROUPVAL(): In which month did we do the most business last year?

1. Summarize Report Data

   When you want to know sum, average, highest, or lowest value of a numeric column (like Amount, Age, or Probability), summarize the column. Summaries calculate subtotals for every group in a report, as well as a grand total for all report data.

2. Evaluate Each Record in Reports with Row-Level Formulas

   Write row-level formulas to when business questions like "how many days did it take each opportunity to close?" prompt you to logically or mathematically evaluate each record in a report.

3. Evaluate Groups and Totals with Summary Formulas

   Write summary formulas to evaluate a report's group subtotals and grand totals. For example, summary formulas can adjust earnings after tax.

4. Compare Groups with PARENTGROUPVAL() and PREVGROUPVAL()

   Write row level formulas to evaluate each record in a report. For example, see how many days it took each opportunity to close by subtracting the Open Date from the Close Date.

Summarize Report Data

When you want to know sum, average, highest, or lowest value of a numeric column (like Amount, Age, or Probability), summarize the column. Summaries calculate subtotals for every group in a report, as well as a grand total for all report data.

👁 **Example:** After building a basic opportunity report grouped by `Stage`, you may find yourself wondering what the sum and average amounts are for each stage.

No need to add amounts together. No need to write a formula. Instead, summarize the `Amount` column. Click ▼ > **Summarize**, and then choose how you'd like to evaluate the data: **Sum**, **Average**, **Max**, **Min**. For this example, choose **Sum** and **Average**.

Summarize Report Data in Lightning Experience

Summarize report data from the report builder.

1. From the Reports tab, edit a report. Click ▼ > **Edit**.

2. Find the numeric column you'd like to summarize. Click ▼ > **Summarize**, and then choose how you'd like to evaluate the data: **Sum**, **Average**, **Max**, **Min** (1).

    A check mark appears next to already-applied summaries. Click You can apply all four summaries at once.

3. If you don't see the **Summarize** option, it means that the column is not numeric. To add a numeric column, expand the **Fields** pane and either double click a numeric field or drag it into the Columns list.

   Each field's type is denoted by an icon directly to the left of the field name. Numeric fields are identified by a number sign (#).

4. Optionally, there is a second way to summarize a numeric field (2).

   a. From the Columns section of the OUTLINE panel, click the numeric field you want to summarize.

   b. Then, select the summary function you wish to calculate: **Sum**, **Average**, **Max**, **Min**.

   c. Click **Apply**.
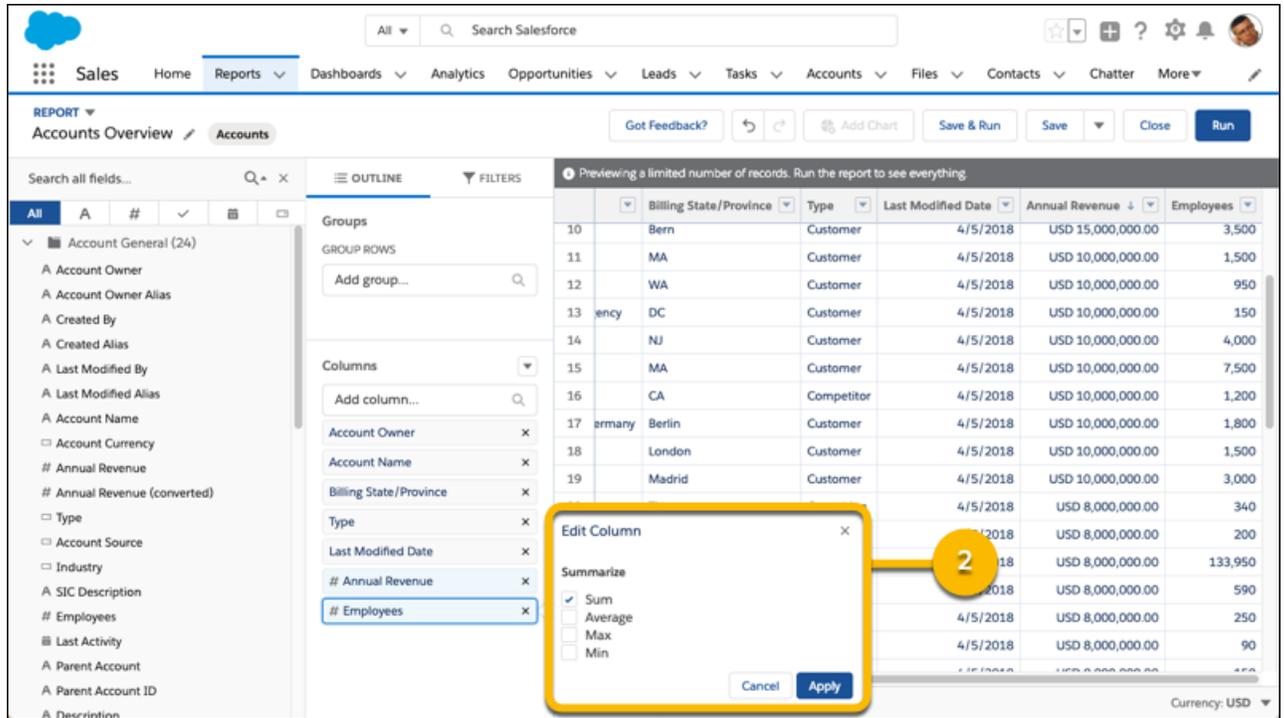
For each group in the report, every summary you apply appears as a subtotal. At the bottom of the report, summaries appear as a total (3). The report builder preview displays data for a limited number of records. Save and run the report to see everything.
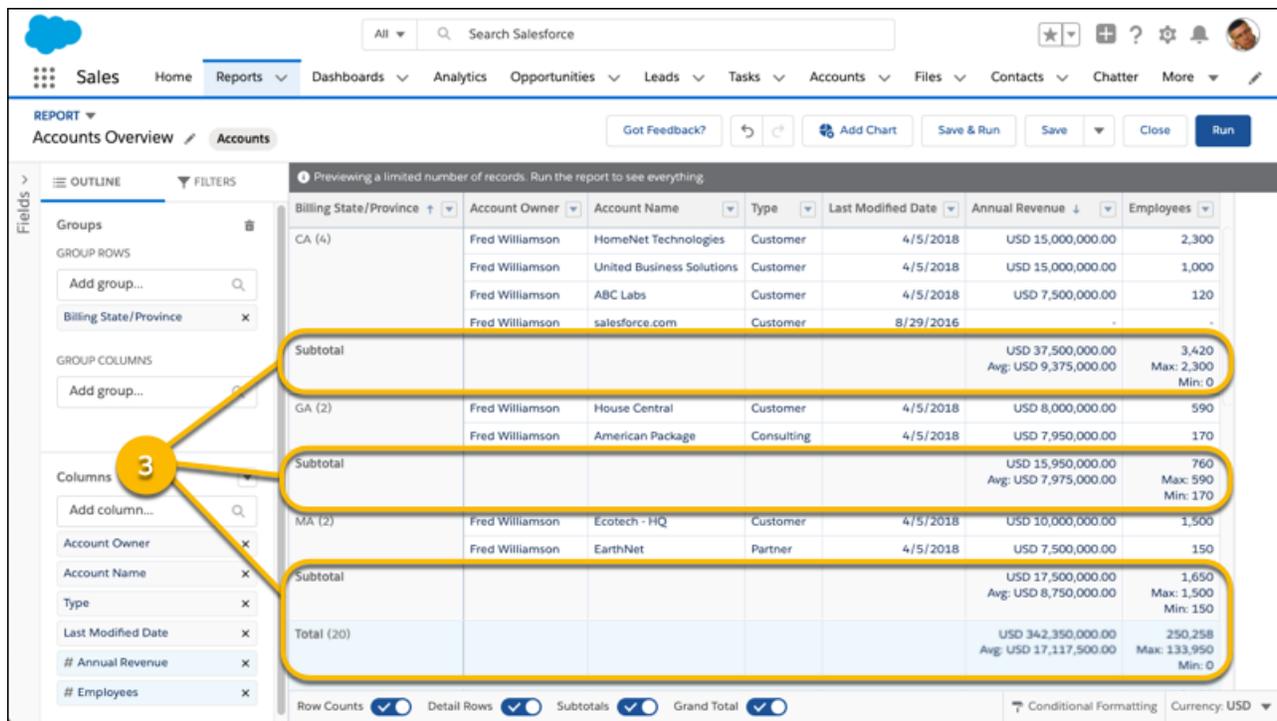
When viewing the report on the run page, summaries also appear at the top of the report (4).

Summarize Report Data in Salesforce Classic

The Fields pane displays fields from the selected report type, organized by folder.

A summary is the `Sum`, `Average`, `Max`, or `Min` for a number field. (Use the `#` filter to find them faster.)

To add a summary field:

- Double-click a number field in the Fields pane.
- Drag a number field into the preview. Press **CTRL** to select multiple fields. For matrix reports, there are drop zones before, between, and after sets of summaries. For example, the sum, average, max, and min of `Annual Revenue` are a set, and you can't drop a new summary field between them.
- Choose **Summarize this Field** in the column menu for a field already in the report.

To change an existing summary field, or add other summaries on that same field, click ▼ next to the summary field and choose **Summarize this Field**.

To remove a summary field:

- Click its menu and choose **Summarize this Field** and deselect all options.
- Click its menu and choose **Remove Summary**.
- Drag the summary set back to the Fields pane. Note that all summaries for that field are removed. Press **CTRL** to select multiple summary fields.

Reorder sets of summary fields in matrix reports by dragging them. The summaries for each field move together when dragged. For example, if your report contains the sum and average of `Annual Revenue`, and the max and min of `Probability`, you can drag

the `Probability` summaries before or after the `Annual Revenue` summaries, but not between. Summaries can't be placed after custom summary formulas or `Record Count`.

SEE ALSO:
   Report Fields
   Subtotal Report Results

Evaluate Each Record in Reports with Row-Level Formulas

Write row-level formulas to when business questions like "how many days did it take each opportunity to close?" prompt you to logically or mathematically evaluate each record in a report.

To answer common questions like these, you need to assess each record in a report:

- How many days did it take each opportunity to close?
- Which cases mention the word "widget"?
- As defined by a customer satisfaction (CSAT) score, which customers are satisfied or upset?

The answer to these questions, and many more, are a row-level formula away.

Write a Row-Level Formula
Writing a row-level formula adds a row-level formula column to your report that makes calculations on every report row. Write row-level formulas directly in the Lightning report builder.

Edit a Row-Level Formula
Edit a row-level formula to expand, refine, or change how it evaluates report data.

Delete a Row-Level Formula
When you no longer need a row-level formula in your report, delete it.

Examples: Evaluate Each Record in Reports with Row-Level Formulas
"How many days did it take each opportunity to close? Which case subjects mention the word `widget`?" Here's how to write row-level formulas that answer these business questions, and others.

Get the Most Out of Row-Level Formulas: Tips, Limits, and Limitations
As you get ready to write row-level formulas, review these tips, limits, and limitations.

Write a Row-Level Formula

Writing a row-level formula adds a row-level formula column to your report that makes calculations on every report row. Write row-level formulas directly in the Lightning report builder.

Here's how to add a row-level formula to a report.

1. Create or edit a report.

2. From the Columns section of the OUTLINE pane, click ▼ > **Add Row-Level Formula**.

3. From the Edit Row-Level Formula Column window, enter a **Column Name** (2), choose a **Formula Output Type** (3), and write a **Formula** (4). Optionally, enter a **Description** and set the number of **Decimal Points**.

To test your formula for errors, click **Validate**. If necessary, resolve any errors. Then, click **Apply** (5).

💡 Tip: For a little row-level formula inspiration, see Examples: Evaluate Each Record in Reports with Row-Level Formulas on page 60 in Salesforce help.

To learn more about writing formulas, see Formula Operators and Functions in Salesforce help.
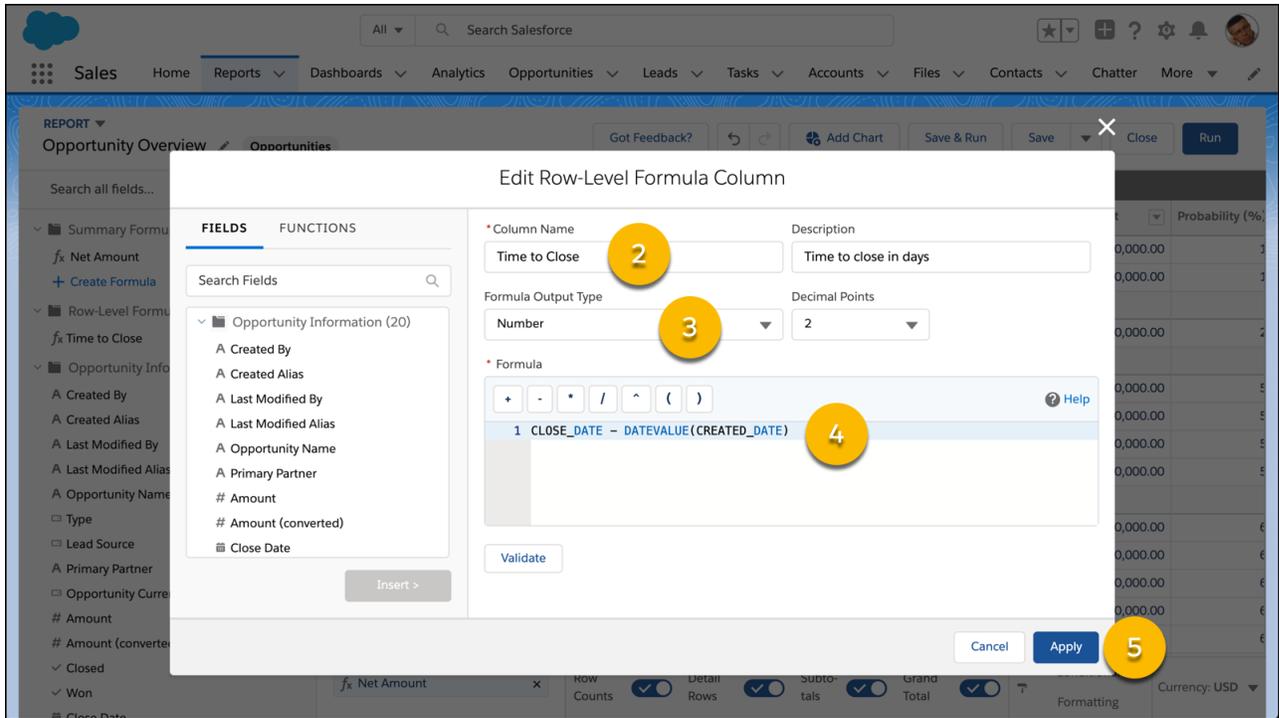
4. The row-level formula appears as a column on the report (6).
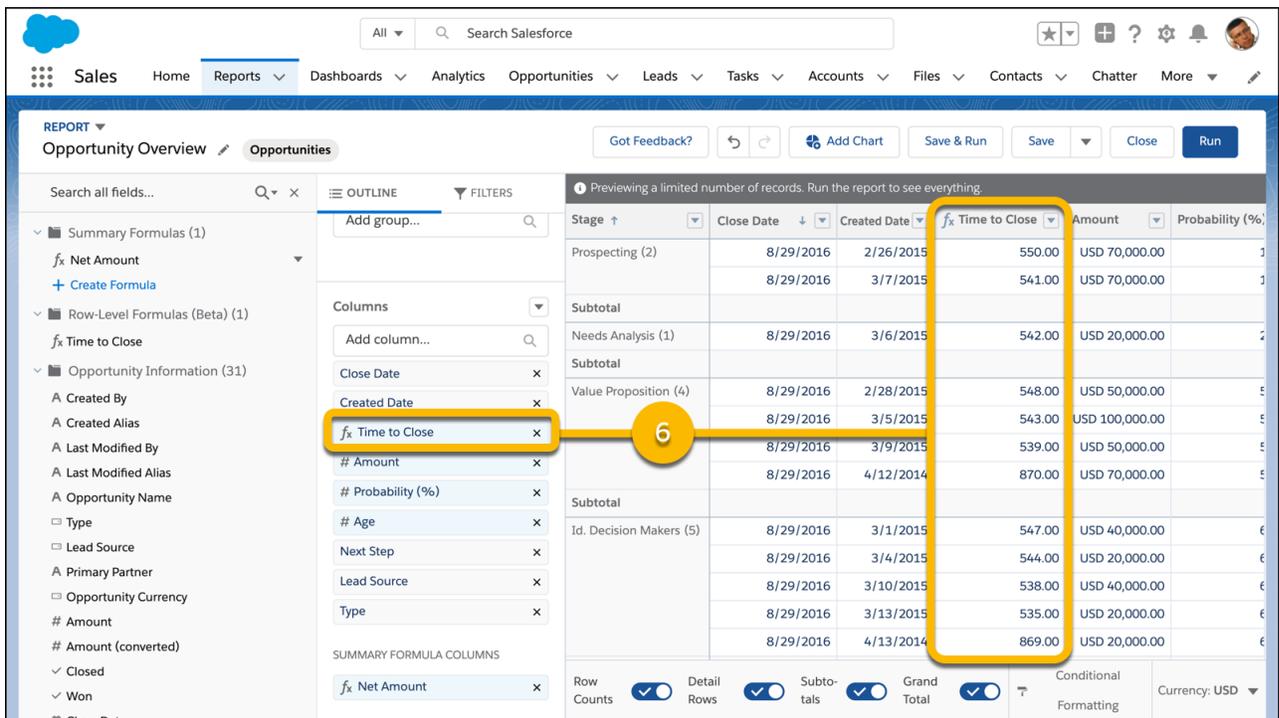
5. To save and run the report, click **Save & Run**.

The report now features a row-level formula.

Edit a Row-Level Formula

Edit a row-level formula to expand, refine, or change how it evaluates report data.

Edit row-level formulas directly in the Lightning report builder. Here's how.

1. Edit a report.

2. From the Columns section of the OUTLINE pane, click the row-level formula column's name.
   For example, if the row-level formula you want to edit is named *Time to Close*, then click
   Time to Close (1).

3. From the Edit Row-Level Formula Column window, edit the **Column Name** (2), choose a **Formula Output Type** (3), and edit the **Formula** (4). If applicable, edit the **Description** and change the number of **Decimal Points**.

   To test your formula for errors, click **Validate**. If necessary, resolve any errors. Then, click **Apply** (5).

   💡 Tip: For a little row-level formula inspiration, see Examples: Evaluate Each Record in Reports with Row-Level Formulas on page 60 in Salesforce help.

   To learn more about writing formulas, see Formula Operators and Functions in Salesforce help.
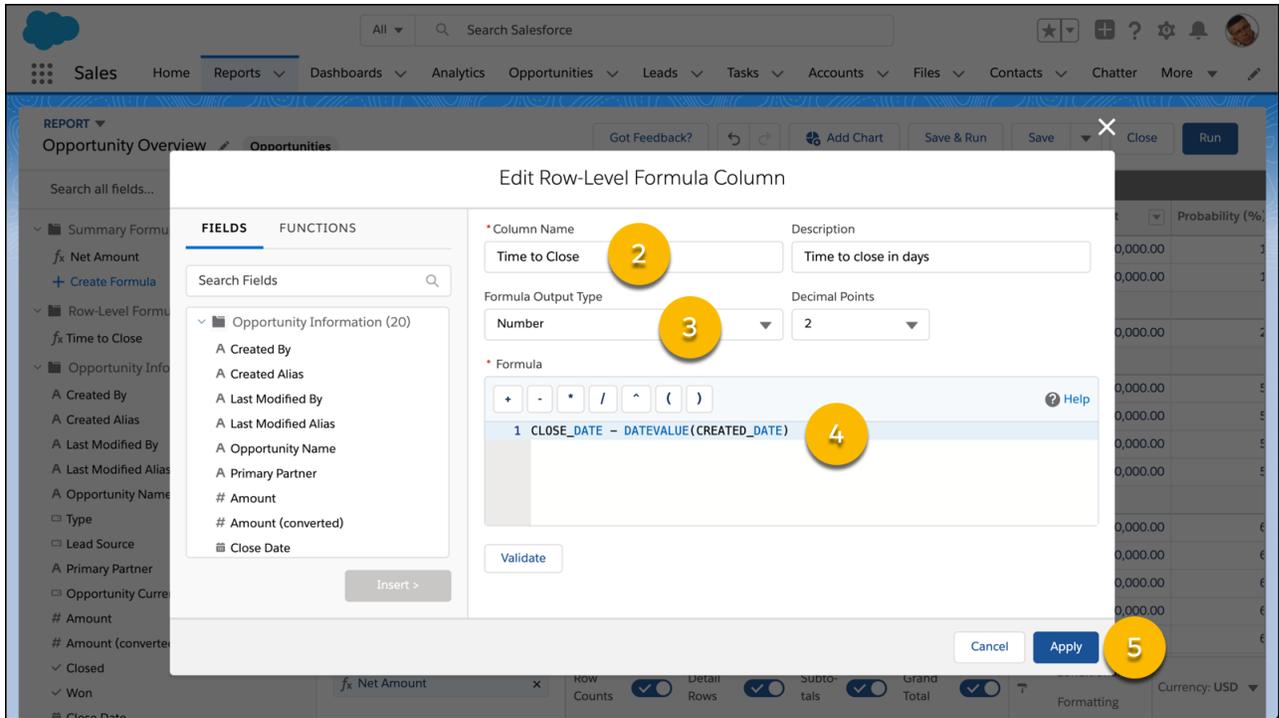
4. The edited row-level formula appears as a column on the report (6).
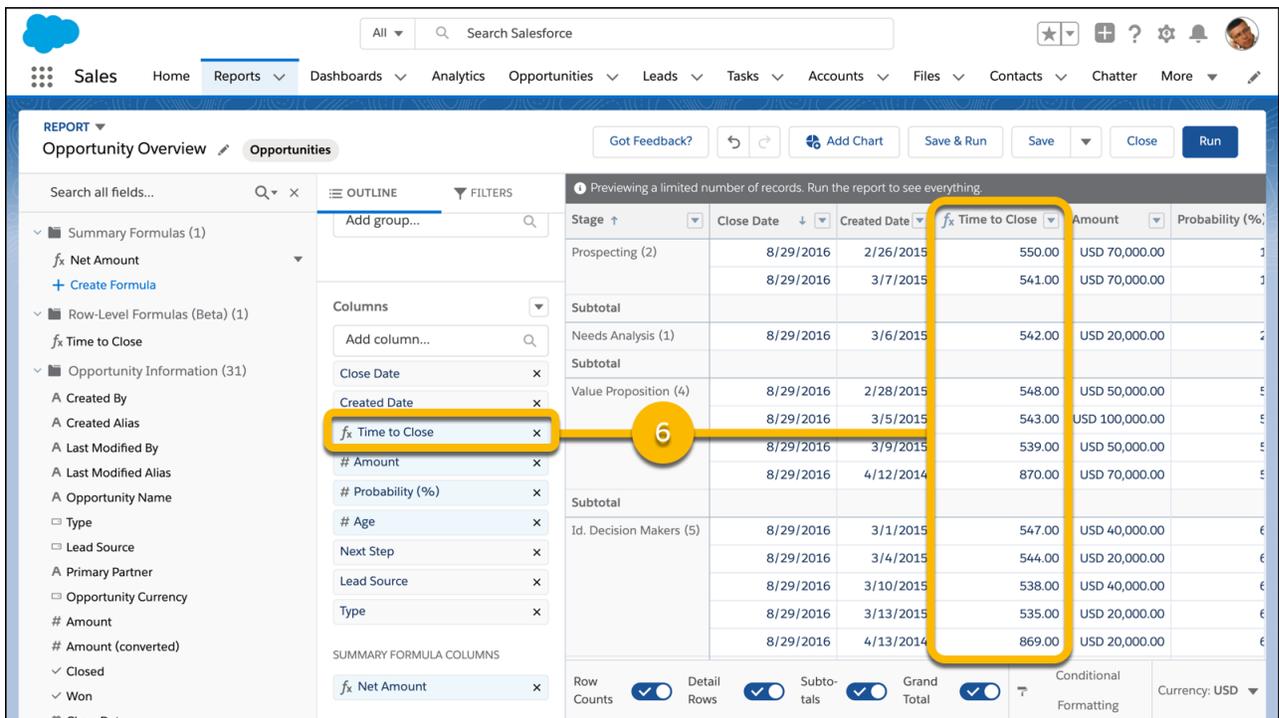
5. To save and run the report, click **Save & Run**.

The row-level formula column on the report is updated to reflect your edits.

Delete a Row-Level Formula

When you no longer need a row-level formula in your report, delete it.

Delete row-level formulas directly from the Lightning report builder. Here's how.

1. Edit a report.
2. From the Columns section of the OUTLINE pane, find the row-level formula you want to delete and click **x**.

3. If you mistakenly remove a row-level formula from the report, you can recover it by clicking the undo button (⟲).

4. To save and run the report, click **Save & Run**.

The row-level formula column is removed from the report and from the row-level formulas folder in the Fields pane.

Examples: Evaluate Each Record in Reports with Row-Level Formulas

"How many days did it take each opportunity to close? Which case subjects mention the word *widget*?" Here's how to write row-level formulas that answer these business questions, and others.

To answer common questions like these, you need to assess each record in a report:

- How many days did it take each opportunity to close?
- Which case subjects mention the word "widget"?

The answer to these questions, and many more, are a row-level formula away. Here's how.

Example: How many days did it take each opportunity to close?

To answer this question, create an opportunity report and write a row-level formula that subtracts the Created Date from the Closed Date.

1. Create or edit an opportunities report.
2. From the Columns section of the OUTLINE pane, click ▼ > **Add Row-Level Formula**.

3. From the Edit Summary-Level Formula Column window, write the formula.

   a. In **Column Name**, enter `Time to Close`.

   b. In **Description**, enter `Time to Close in Days`.

   c. From **Formula Output Type**, select **Number**.

      Even though this formula works with date values, number is the output type needed because the formula performs subtraction and returns a number.

   d. From **Decimal Points** to **0**.

   e. In **Formula**, write:

   ```
   CLOSE_DATE - DATEVALUE(CREATED_DATE)
   ```

   📝 Note:

   - `CLOSE_DATE` and `CREATED_DATE` are the API names of the Close Date and Created Date fields. Formulas work with the API names of fields, not the display names. The easiest way to enter the API name of a field is to search for it from the FIELDS menu, select it, and click **Insert >**.

   - Because `CLOSE_DATE`'s data type is `date` and `CREATED_DATE`'s data type is `datetime`, we convert the value of `CREATED_DATE` to the `date` data type with the `DATEVALUE()` function. The two values must have a matching data type so that we can subtract one from the other. (`Datetime` and `date` are different data types because `datetime` includes information about units of time smaller than a day - hours, minutes, seconds, and so on - and `DATE` does not.)

   f. To make sure that the formula is error-free, click **Validate**. If necessary, resolve errors.

   g. Click **Apply** (5).
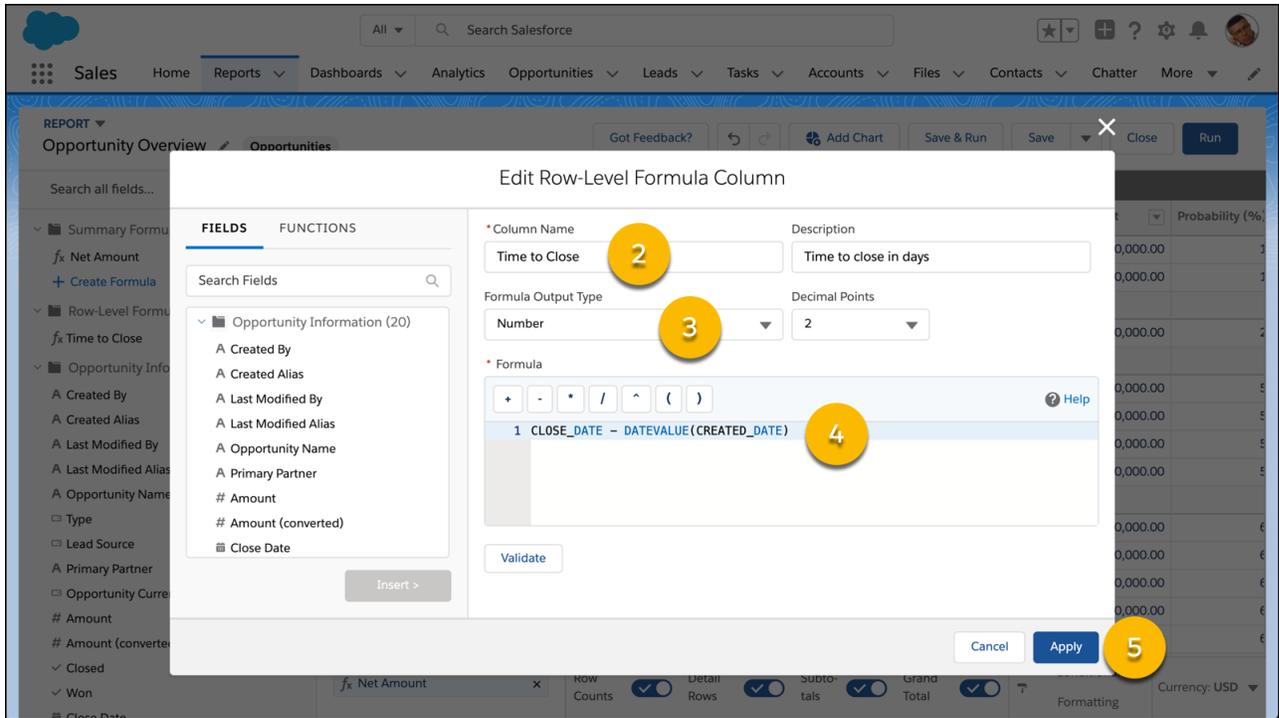
4. To save the report, click **Save**.

The row-level formula appears as a report column named `Time to Close` (6). The Time to Close column calculates how many days each opportunity was open by subtracting Created Date from Closed Date.
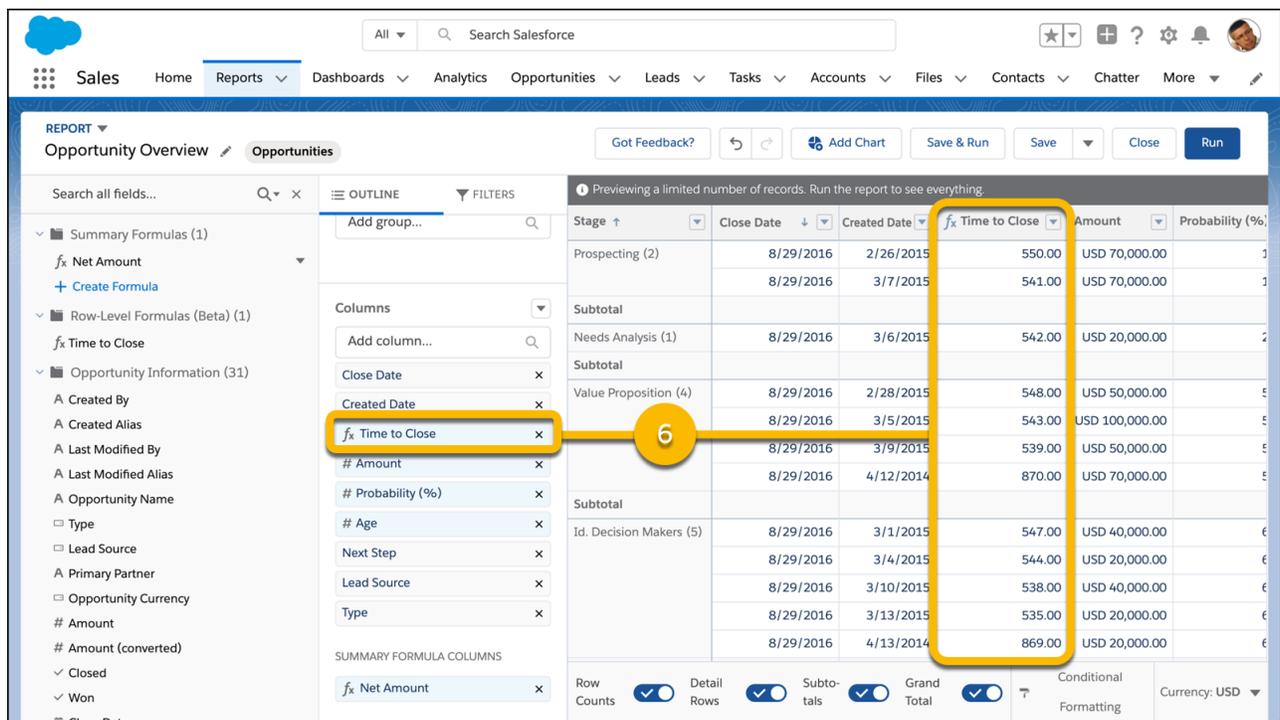
Example: Which case subjects mention the word "widget"?

To answer this question, create a cases report and write a row-level formula that evaluates each case subject to see whether or not it includes the term "widget".

1. From the Columns section of the OUTLINE pane, click ▼ > **Add Row-Level Formula**.

2. From the Edit Summary-Level Formula Column window, write the formula.

   a. In **Column Name**, enter *Mentions Widgets?*.

   b. In **Description**, enter *Determines whether or a case subject includes the term "widget"*.

   c. From **Formula Output Type**, select **Text**.

   d. In **Formula**, write:

   ```
   IF(CONTAINS(SUBJECT, "widget"), "Yes", "No")
   ```

   📝 Note:
   - SUBJECT is the API name of the Subject field. Formulas work with the API names of fields, not the display names. The easiest way to enter the API name of a field is to search for it from the FIELDS menu, select it, and click **Insert >**.

   e. To make sure that the formula is error-free, click **Validate**. If necessary, resolve errors.

3. Click **Apply**.

4. Click **Save**.

The row-level formula appears as a report column named *Mentions Widgets?*. The *Mentions Widgets?* column evaluates each case's subject field to determine whether or not it includes the text *widget*. If it does, the row-level formula returns *Yes*. If it doesn't, the row-level formula returns *No*.

Get the Most Out of Row-Level Formulas: Tips, Limits, and Limitations

As you get ready to write row-level formulas, review these tips, limits, and limitations.

- Each report supports 1 row-level formula.
- Each row-level formula can reference up to 5 unique fields. If the formula references the same field 5 times, that counts as referencing 1 unique field.
- You can't use row-level formulas for:
  - Cross filters
  - Buckets
- Row-level formulas can't reference:
  - Bucket fields
  - Summary formulas
  - Other row-level formulas
- Row-level formulas don't support these field types:
  - Boolean
  - Timeonly
  - Email
  - Multiselect picklist
- Some date fields aren't supported, including:
  - Due Date
  - Birthdate
- Some text fields aren't supported, including:
  - Billing Address
- Row-level formulas always use an org's default currency. Row-level formulas don't respect multi-currency settings on reports.

  For example, if USD is your org's default currency and a report is set to show GBP, the row-level formula displays USD.
- "With or Without" custom report types (like Accounts with or without Contacts) don't support row-level formulas. The **Add Row-Level Formula** button doesn't appear.
- You can create row-level formulas on historical trend reports, but you can't reference historical fields in row-level formulas.
- Reporting snapshots don't support row-level formula fields.
- Row-level formulas aren't available on joined reports. To convert a report with row-level formulas into a joined report, first remove the row-level formulas.
- The embedded Salesforce Classic report builder in Lightning Experience doesn't support row-level formulas. If a report has a row-level formula, then the **Edit (Salesforce Classic)** button doesn't appear.
- You can't create, edit, or delete row-level formulas from the Fields panel. Create, edit, or delete them from the OUTLINE panel.
- Row-level formulas aren't available in Salesforce Classic. You can't run or edit reports with row-level formulas in Salesforce Classic.
- Row-level formulas aren't available in the Apex API.

Evaluate Groups and Totals with Summary Formulas

Write summary formulas to evaluate a report's group subtotals and grand totals. For example, summary formulas can adjust earnings after tax.

With the report builder, you can summarize numeric columns with four built-in functions: `Sum`, `Average`, `Min`, and `Max`. If you want to know the total value of closed opportunities, you can calculate it by finding the Amount column in the report preview and clicking 🔽 > **Summarize** > **Sum**.

But what if you want to know the total value of closed opportunities after tax, and what if earnings are taxed at a progressive rate? (Say, 15% on sales below $1,000,000, and 20% on sales above $1,000,000). Add a summary formula column named "Amount After Tax" to your report and use a formula like this one to evaluate the Amount column:

```
IF(AMOUNT:SUM < 1000000, AMOUNT:SUM * (1 – 0.15), AMOUNT:SUM * (1 – 0.20))
```

Now you can easily incorporate after-tax earnings into fiscal quarter planning!

1. Add a Summary Formula Column to a Report
   Before adding a summary formula to your report, be sure to group report data. Summary formula columns require at least 1 group.

2. Edit a Summary Formula Column

    To update a summary formula after adding it to a report, edit it.

3. Delete a Summary Formula Column

    If you no longer need a summary formula column, delete it.

4. Get the Most Out of Summary Formulas: Tips, Limits, and Limitations

    Summary formulas are a powerful reporting technique, but they can be tricky. Here are some tips, limits, and limitations to keep in mind when working with them.

SEE ALSO:

Add a Summary Formula Column to a Report

Report Fields

Add a Summary Formula Column to a Report

Before adding a summary formula to your report, be sure to group report data. Summary formula columns require at least 1 group.

Summary formula columns are available in both Lightning Experience and Salesforce Classic. They use the same formula language, so the summary formulas you add in Lightning Experience work in Salesforce Classic, and vice versa.

The steps for adding a summary formula column are a little different depending on whether you're using Lightning Experience or Salesforce Classic. Follow the instructions based on whether you're using the Lightning Experience or the Salesforce Classic report builder.

> 📝 **Note:** Summary formulas are powerful and complex. For help building your formula, refer to Formula Operators and Functions in Salesforce Help.

Add a Summary Formula Column in Lightning Experience

1. Edit or create a report.

2. If necessary, group report data. From the Groups section, select a field from the **Add group...** lookup menu under GROUP ROWS.

3. From the Columns section, click ▼ > **Add Summary Formula Column**.

   Alternatively, from the Fields pane, under Summary Formulas, click **+ Create Formula**.

   If a summary formula is available, but not yet added as a summary formula column, you can add it from the Fields pane. Under Summary Formulas, click ▼ > **Add**

4. Enter a name for the summary formula column.

5. Choose the Formula Output Type.

6. Enter a summary formula. For example:

```
IF(AMOUNT:SUM < 1000000, AMOUNT:SUM * (1 - 0.15), AMOUNT:SUM * (1 - 0.20))
```

7. To see if your formula contains errors, click **Check Syntax**. Resolve any errors.

8. Optionally, enter a description for the formula.

9. If applicable, choose the number of decimal points.

10. Optionally, click **FORMAT** to choose how to format results, how many decimal points to show, and to change where the formula gets applied.

11. Click **Apply**.

12. Click **Save**.

Add a Summary Formula Column in Salesforce Classic

1. Edit or create a report.

2. If necessary, group report data. Change the format to **Summary**, **Matrix**, or **Joined**, find the field you want to group by, and drop it into the grouping field.

3. From the Fields pane, in the Formulas folder, click **Add Formula**.

4. Enter a name for your formula column. The label must be unique. Optionally, enter a description.

5. From the `Format` dropdown list, select the appropriate data type for your formula based on the output of your calculation.

6. From the `Decimal Places` dropdown, select the number of decimal places to display for currency, number, or percent data types. This setting is ignored for currency fields in multicurrency organizations. Instead, the `Decimal Places` for your currency setting apply.

7. Set the `Where will this formula be displayed?` option. The formula calculation displays in the report at the level you select. To display the formula calculation at every level, including the Grand Total, select `All summary levels`.

8. Build your formula.

    a. Select one of the fields listed in the **Summary Fields** dropdown list. This field's value is used in your formula.

    b. Select a summary type for use in your formula. This option is not available for `Record Count`.

Summary Type	Description
Sum	The summary value of data in a field or grouping of fields.
Largest Value	The largest value of data in a field or grouping of fields.
Smallest Value	The smallest value of data in a field or grouping of fields.
Average	The average of data in a field or grouping of fields.

    c. To add operators (like > or <) to your formula, click **Operators**.

    d. Select the function category (All, Logical, Math, or Summary), choose the function you want to use in your formula, and click **Insert**.

    e. Repeat these steps as necessary.

9. To see if your formula contains errors, click **Check Syntax**. If there are errors, the cursor highlights them.

10. Click **OK**.

11. Click **Save**.

The summary formula column appears in your report.

SEE ALSO:
Build a Report in Salesforce Classic
Evaluate Groups and Totals with Summary Formulas
Get the Most Out of Summary Formulas: Tips, Limits, and Limitations
Add a Summary Formula to a Joined Report

Edit a Summary Formula Column

To update a summary formula after adding it to a report, edit it.

The steps for editing a summary formula column are a little different depending on whether you're using Lightning Experience or Salesforce Classic. Follow the instructions based on whether you're using the Lightning Experience or the Salesforce Classic report builder.

> **Note:** Summary formulas are powerful and complex. For help building your formula, refer to Formula Operators and Functions in Salesforce Help.

Edit a Summary Formula Column in Lightning Experience

1. Edit a report.

2. From the Formula Columns section, click the formula column you want to edit. (Don't click **X**. It deletes the formula column. If you accidentally delete the formula column, undo the deletion by clicking ⟲ .)

   Alternatively, from the Fields pane, under Summary Formulas, find the formula column you want to edit and click ▼ > **Edit**.

3. Edit the summary formula.

4. To see if your formula contains errors, click **Check Syntax**. Resolve any errors.

5. Optionally, click **FORMAT** to choose how to format results, how many decimal points to show, and to change where the formula gets applied.

6. Click **Apply**.

7. Click **Save**.

Edit a Summary Formula Column in Salesforce Classic

1. Edit a report.

2. From the Fields pane, find the summary formula you want to edit and click 🖊 .

   The Custom Summary Formula menu opens.

3. Enter a name for your formula column. The label must be unique. Optionally, enter a description.

4. From the `Format` dropdown list, select the appropriate data type for your formula based on the output of your calculation.

5. From the `Decimal Places` dropdown, select the number of decimal places to display for currency, number, or percent data types. This setting is ignored for currency fields in multicurrency organizations. Instead, the `Decimal Places` for your currency setting apply.

6. Set the `Where will this formula be displayed?` option. The formula calculation displays in the report at the level you select. To display the formula calculation at every level, including the Grand Total, select `All summary levels`.

7. Build your formula.

   a. Select one of the fields listed in the **Summary Fields** dropdown list. This field's value is used in your formula.

   b. Select a summary type for use in your formula. This option is not available for `Record Count`.

Summary Type	Description
Sum	The summary value of data in a field or grouping of fields.
Largest Value	The largest value of data in a field or grouping of fields.
Smallest Value	The smallest value of data in a field or grouping of fields.
Average	The average of data in a field or grouping of fields.

   c. To add operators (like > or <) to your formula, click **Operators**.

   d. Select the function category (All, Logical, Math, or Summary), choose the function you want to use in your formula, and click **Insert**.

   e. Repeat these steps as necessary.

8. To see if your formula contains errors, click **Check Syntax**. If there are errors, the cursor highlights them.

9. Click **OK**.

10. Click **Save**.

The summary formula column updates to reflect your edits.

Delete a Summary Formula Column

If you no longer need a summary formula column, delete it.

The steps for deleting a summary formula column are a little different depending on whether you're using Lightning Experience or Salesforce Classic. Follow the instructions based on whether you're using the Lightning Experience or the Salesforce Classic report builder.

Delete a Summary Formula Column in Lightning Experience

1. Edit or create a report.

2. From the Formula Columns section, find the summary formula column you want to remove and click **X**. When you remove a summary formula this way, it's still available in the Fields pane under Summary Formulas. You can add it back to the report as a summary formula column from the Fields pane later.

    Alternatively, from the Fields pane, under Summary Formulas, find the formula column you want to edit and click ▼ > **Delete**. When you delete a summary formula this way, it is removed for good. If you want to add it back later, you'll have to rewrite it.

3. Click **Save**.

Delete a Summary Formula Column in Salesforce Classic

1. Edit or create a report.

2. From the Fields pane, in the Formulas folder, find the summary formula column you want to delete and click 🗑

3. Click **Save**.

The summary formula column is removed from your report.

In Lightning Experience, If you accidentally delete a formula column in Lightning Experience, undo the deletion by clicking ↺ .

In Salesforce Classic, you can't undo a deletion. If you can't undo the deletion, deleting a summary formula column is permanent. To get it back, you'll have to recreate it from scratch.

Get the Most Out of Summary Formulas: Tips, Limits, and Limitations

Summary formulas are a powerful reporting technique, but they can be tricky. Here are some tips, limits, and limitations to keep in mind when working with them.

Functions for use with summary formulas in reports are available under the function category. Select a function, then click **Help on this function** for information.

- Summary level formulas don't support date or date-time functions or fields.

- A summary formula can't reference another summary formula. Nor can a summary formula reference a row-level formula.

- You can't group report data by summary formula columns.

- You can't filter report data by summary formula columns.

- You can't reorder summary formula columns.

- Before adding a summary formula column, group report data at least one time.

- Summary formulas can contain 3900 or fewer characters.

- Regardless of the summary formula data type, your summary formula can contain fields of different data types, including: number, currency, percent, and checkbox (true/false) fields. For example, a summary formula in an Opportunities with Partners report can reference opportunity `Amount` or `Stage Duration`, as well as account `Annual Revenue`.

- In Salesforce Classic, dashboard and report charts that display values from summary formulas display decimal places using your default currency setting instead of what you specified for the formula. For example, if the summary formula specifies zero decimal places, no decimal places appear in columns, but chart values show the number of decimal places specified for your default currency (usually two decimal places). This limit applies to currencies, numbers, and percentages.

- When a field is deleted or is unavailable (for example, because of field-level security), all summary formulas that contain the field are removed from the report.

- The summary types `Sum`, `Largest Value`, `Smallest Value`, and `Average` aren't available for use with the `Record Count` field.

- The `Smallest Value` summary type includes blank (null) or zero values in the summary formula calculation if these values are present in your report data.

- The `Largest Value` summary type includes the largest blank (non-null) value present in your report data.

- Percents are represented as decimals in summary formulas. 20% is represented as 0.20.

- Operators can be used to give fields in summary formulas a negative value. For example: `{!EMPLOYEES:SUM} + - {!SALES:SUM}`.

- For summary formulas on reports that are grouped by rows and columns (matrix reports), Salesforce calculates results for all formulas where the `Where will this formula be displayed?` option isn't set to `All summary levels`. The formula determines the value returned. If null is returned, the cell is empty.

- Summary fields can display up to 21 digits. If a summary field has more than 21 digits, an accurate result might not be displayed. If "`#Too Big!`" appears in a report cell, check your formula for calculations that could result in more than 21 digits. Avoid multiplying large numbers, raising a large number to a power, or dividing by a very small number.

- Formulas treat blank (null) report cells as zero values.

- "`#Error!`" displays on report cells whenever an error occurs while calculating a formula's value. "`#Error!`" also displays when formulas divide by zero. To resolve the error, check your formula and provide an alternative value.

- Cross-block summary formulas are available for joined reports.

SEE ALSO:

Compare Groups with `PARENTGROUPVAL()` and `PREVGROUPVAL()`

Write row level formulas to evaluate each record in a report. For example, see how many days it took each opportunity to close by subtracting the Open Date from the Close Date.

PARENTGROUPVAL and PREVGROUPVAL

Use `PARENTGROUPVAL` to calculate values relative to a parent grouping. Use `PREVGROUPVAL` to calculate values relative to a peer grouping.

PARENTGROUPVAL and PREVGROUPVAL

Use `PARENTGROUPVAL` to calculate values relative to a parent grouping. Use `PREVGROUPVAL` to calculate values relative to a peer grouping.

PARENTGROUPVAL

Use this function to calculate values relative to a parent grouping.

Description:	This function returns the value of a specified parent grouping. A "parent" grouping is any level above the one containing the formula. You can use this function only in custom summary formulas and at grouping levels for reports, but not at summary levels.
Use:	**For reports that are grouped by rows (summary) and for joined reports:** `PARENTGROUPVAL(summary_field, grouping_level)` **For reports that are grouped by rows and columns (matrix):** `PARENTGROUPVAL(summary_field, parent_row_grouping, parent_column_grouping)` Where `summary_field` is the summarized field value, `grouping_level` is `GRAND_SUMMARY` or the API name of the parent level group for summary reports, and `parent_row_level` and `parent_column_level` are the parent levels for matrix reports. In reports with multiple grouping levels, you can set the `grouping_level` to be any group level higher than the formula display level.

Example:

```
TOTAL_PRICE:SUM/PARENTGROUPVAL(TOTAL_PRICE:SUM, GRAND_SUMMARY)
```

This formula calculates, for each product, its relative size compared to the grand total. In this example, the report is a summary of opportunities and their products, grouped by `Product Name`.

PREVGROUPVAL

Use this function to calculate values relative to a peer grouping. If there's no previous grouping, the function returns a null value.

Description:

This function returns the value of a specified previous grouping. A "previous" grouping is one that comes before the current grouping in the report. Choose the grouping level and increment. The increment is the number of columns or rows before the current summary. The default is 1; the maximum is 12. You can use this function only in custom summary formulas and at grouping levels for reports, but not at summary levels.

Use:

```
PREVGROUPVAL(summary_field, grouping_level [, increment])
```

When `summary_field` is the name of the grouped row or column:

- `grouping_level` is the API name of the peer level group whose summary value is used for the previous grouping.
- `increment` is the number of previous groupings.

In reports with multiple grouping levels, you can specify `grouping_level` to be either the same group level as the formula display level or a group level higher than the formula display level.

Example:

```
AMOUNT:SUM - PREVGROUPVAL(AMOUNT:SUM, CLOSE_DATE)
```

This formula calculates, for each month, the difference in amount from the previous month shown in the report. In this example, the report is an opportunity matrix with columns grouped by `Close Date` and rows by `Stage`.

Example: Three Week Moving Average

To calculate a three-week moving average of opportunity amounts:

```
(OppProductTrends__c.Amount__c:SUM+ PREVGROUPVAL(OppProductTrends__c.Amount__c:SUM,
OppProductTrends__c.as_of_date__c) + PREVGROUPVAL(OppProductTrends__c.Amount__c:SUM,
OppProductTrends__c.as_of_date__c,2))/3
```

SEE ALSO:

Add a Summary Formula Column to a Report

Evaluate Groups and Totals with Summary Formulas

Add a Summary Formula to a Joined Report

Count Unique Values in Report Results

See how many distinct values a column in your report returns with a unique count.

When you add a unique count to a column, it shows the number of unique column values in report results. Duplicate values aren't counted.

1. While editing a report in the report builder, find the column for which you want to count unique values.

2. Click ▼ > **Show Unique Count**.

3. Click **Save**.

A count of unique values appears as a grand total for the whole report and as a subtotal for each group.

👁 **Example:** Opportunity reports often list multiple opportunities with the same account. Add a unique count to the Account Name column to see how many individual accounts values appear in the report.

To remove the unique count, click ▼ > **Hide Unique Count**.

As you get ready to count unique values in report results, take note of these limitations.

- Each report support up to 3 unique counts.

- Lightning table dashboard components don't support unique counts.

- Fields of these data types don't support unique counts:

  - Blobs

  - Clobs

  - Encrypted fields

- – Multi-select picklist
- – Text area, including long text area and rich text area

Combine Different Types of Information in a Joined Report

The joined report format lets you view different types of information in a single report. A joined report can contain data from multiple standard or custom report types. You can turn any existing report into a joined report using the report builder.

Watch a Demo: ▶ Introducing Joined Reports in Salesforce (Salesforce Classic)

Joined reports are made up of up to 5 *blocks*. Each block shows data from one standard or custom report type, so a joined report can include data from up to 5 different report types. Report types can only be joined if they share some common fields with each other. For example, Opportunities and Accounts can be joined because they share the common fields Account Name and Account ID. Each block returns up to 2,000 records, which means that a joined report with 5 blocks can return up to 10,000 records. When you group data, you group data across blocks, which means you can only group data by *common fields*.

When you filter joined reports, you add filters to each block independent of one another.

You can create the ultimate sales report by building a joined report. Add three blocks to the joined report for a complete view of your operation: Accounts, Opportunities, and Cases.

Or, build a joined report to get the birds-eye-view of your customer service operation. Add three blocks with the Cases report type. Filter block 1 by `Status equals Open`, block 2 by `Status equals In Progress`, and block 3 by `Status equals Closed`.

Joined reports are based on SQL *outer join*. Each row in the source blocks has a corresponding row in the joined report, regardless of whether all the blocks that make up the joined report have matching data.

For example, here's a simple joined report that shows Accounts and Opportunities grouped by Account Name. Each row represents an account. The first 2 accounts (Abbott) also have opportunity data. The last 2 accounts (ABC and ABM) don't have any associated opportunities, so those entries are null in the Opportunities block.

EDITIONS

Available in: both Lightning Experience and Salesforce Classic

Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

USER PERMISSIONS

To create, edit, and delete joined reports:

- **Legacy Folder Sharing**
  Create and Customize Reports

  AND

  Report Builder

  Enhanced Folder Sharing
  Create and Customize Reports

  AND

  Report Builder

> 💡 **Tip:** You can show a joined report that includes a chart on a dashboard. Edit the joined report dashboard component and select **Use chart as defined in the source report**.

1. **How Joined Reports Work**

   A joined report can contain data from multiple standard or custom report types. You can add report types to a joined report if they have relationships with the same object or objects. For example, if you have a joined report that contains the Opportunities report type, you can add the Cases report type as well because both have a relationship with the Accounts object.

2. **Create a Joined Report**

   With up to five blocks, joined reports let you view different types of information in a single report. Each block shows data from one standard or custom report type.

3. **Add a Report Type to a Joined Report**

   Adding a report type lets you expand the set of data available for analysis in a joined report.

4. **Change the Principal Report Type**

   The principal report type controls how common fields are named. A joined report must have a principal report type. You can change the principal report type at any time.

5. **Summary Formulas with Joined Reports**

   Two types of custom summary formulas are available with joined reports: standard and cross-block.

6. **Work with Blocks**

   Blocks let you create different views of the information contained in a joined report.

7. **Joined Report Examples**

   Creating a sales rep performance scorecard, reviewing support cases by status, and predicting your opportunity pipeline are some examples of what you can do with joined reports. Here's how.

8. Joined Report Limits, Limitations, and Allocations

Most of the things you can do with reports you can also do with joined reports, such as find, add, and remove fields; summarize fields; and run and save reports. However, there are some things you can't do. When working with Lightning Experience joined reports, there are some Lightning Experience-specific limits and limitations to consider.

SEE ALSO:

Build a Report in Salesforce Classic

How Joined Reports Work

How Joined Reports Work

A joined report can contain data from multiple standard or custom report types. You can add report types to a joined report if they have relationships with the same object or objects. For example, if you have a joined report that contains the Opportunities report type, you can add the Cases report type as well because both have a relationship with the Accounts object.

A joined report consists of up to five report *blocks*, which you add to the report to create multiple views of your data. For each block, you can add regular and summary fields, create standard and cross-block custom summary formulas, apply filters, and sort columns. You apply groupings across all blocks in the report, and can add up to three groupings to the blocks, the same as for the summary format. You can also add a chart to a joined report.

Note: When you add a new block to a joined report and the block has multiple entities in common with the report, only the first entity (in alphabetical order) is shown. Only the fields from the first entity are shown in the common fields area.

Each joined report has a *principal* report type. By default, the principal type is the first one added to the report. For example, if you create the joined report by selecting the Opportunities report type, and then add the Cases type, the Opportunities type is the principal report type.

The principal report type controls how common fields are named. Some common fields have different names or appear in different sections in different report types.

When a joined report contains multiple report types, some fields are identified as *common* fields. A field is a common field if it's shared by all report types or if all report types share a lookup relationship to the field. Common fields can be used to group report blocks. In Lightning Experience, common fields are identified by the ⤙ icon. In Salesforce Classic, common fields appear in the Common Fields area in the Fields pane.

Joined reports are based on SQL *outer join*. Each row in the source blocks has a corresponding row in the joined report, regardless of whether all the blocks that make up the joined report have matching data.

SEE ALSO:

Add a Report Type to a Joined Report

Add a Summary Formula to a Joined Report

Work with Blocks

Create a Joined Report

With up to five blocks, joined reports let you view different types of information in a single report. Each block shows data from one standard or custom report type.

You can turn any existing report into a joined report, or start fresh with a new one.

Create a Joined Report in Lightning Experience

1. From the Reports tab, click **New Report**.

2. Choose a report type and click **Continue**.

    The report type you choose becomes the joined report's principal report type. The principal report type determines how common fields shared by different report types in a report are named. For more information about principal report types and common fields in joined reports, see How Joined Reports Work on page 79.

3. The report opens in the report builder. To turn it into a joined report, click **Report** ▼ > **Joined Report** > **Apply**.

4. To add another report type to the joined report, click ⬚ **Add Block**.

    a. Choose a report type.

    b. If you'd like to include standard columns for this report type, leave **Include default columns** checked. If you'd like to add an empty block, uncheck **Include default columns**. In either case, you can customize the block after adding it by adding or removing columns.

    > 📝 Note: If you add an empty block, it won't appear in the preview until you add columns to it.

    c. Click **Add Block**.

    d. To learn more about working with blocks, see Work with Blocks on page 85.

5. Customize the joined report with columns, groups, filters, and formulas.

6. Click **Save** and name the report.

7. Optionally, describe the report and click **Select Folder** to choose a report folder.

8. Click **Save**.

9. To run the report, click **Run**.

Create a Joined Report in Salesforce Classic

1. From the Reports tab, click **New Report...**

2. Choose a report type, and click **Create**.

    The report type you choose becomes the joined report's principal report type. The principal report type determines how common fields shared by different report types in a report are named. For more information about principal report types and common fields in joined reports, see How Joined Reports Work on page 79.

3. The report opens in the report builder. To turn it into a joined report, click **Tabular Format** ▼ > **Joined**.

4. To add another report type, click **Add Report Type**.

    a. Choose a report type.

    b. Click **OK**.

    c. The additional report type appears in a new block. To learn more about working with blocks, see Work with Blocks on page 85.

5. Customize the joined report with columns, groups, filters, and formulas.

6. Click **Save** and name the report.

7. Optionally, describe the report and click **Select Folder** to choose a report folder.

8. Click **Save**.

9. To run the report, click **Run**.

Add a Report Type to a Joined Report

Adding a report type lets you expand the set of data available for analysis in a joined report.

Add a Report Type in Lightning Experience

1. To add another report type to the joined report, click 🔲 **Add Block**.

2. Choose a report type.

3. If you'd like to include standard columns for this report type, leave **Include default columns** checked. If you'd like to add an empty block, uncheck **Include default columns**. In either case, you can customize the block after adding it by adding or removing columns.

4. Click **Add Block**.

Add a Report Type in Salesforce Classic

1. To add another report type, click **Add Report Type**.

2. Choose a report type.

3. Click **OK**.

The additional report type is added. Notice that:

- A new block appears in the report.
- In Lightning Experience, common fields are marked with the common field icon (⤵) in the Fields pane. In Salesforce Classic, The Fields pane updates with a new area that contains fields unique to the report type. Fields common to all report types are in the Common Fields area.

To learn more about working with blocks, see Work with Blocks on page 85.

SEE ALSO:

    Add a Summary Formula to a Joined Report

    How Joined Reports Work

    Combine Different Types of Information in a Joined Report

EDITIONS

Available in: both Lightning Experience and Salesforce Classic

Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

USER PERMISSIONS

To create, edit, and delete joined reports:

- **Legacy Folder Sharing**
  Create and Customize Reports

  Enhanced Folder Sharing
  Create and Customize Reports

Change the Principal Report Type

The principal report type controls how common fields are named. A joined report must have a principal report type. You can change the principal report type at any time.

By default, the first report type you add to the report is the principal.

The principal report type doesn't affect what data is available for reporting.

Change the principal report type by removing its blocks. If there are multiple blocks based on the principal report type, you must remove them all. Alternatively, in Lightning Experience, change principal report type by reordering blocks. In Lightning Experience, the first block's report type is always the principal report type (the first block is the top-most in the OUTLINE).

When you remove the principal report type, the way the new report type is selected depends on how many report types the report contains.

- If the report contains only two report types, the other report type automatically becomes the principal.

- In Lightning Experience, if the report contains more than two report types, the report type of the first block is the principal report type. The first block is the top-most block in the columns pane.

  In Salesforce Classic, if the report contains more than two report types, a dialog displays that prompts you to pick one of them as the new principal report type.

SEE ALSO:

> Add a Block
>
> Add a Report Type to a Joined Report
>
> Joined Report Limits, Limitations, and Allocations
>
> Combine Different Types of Information in a Joined Report

Summary Formulas with Joined Reports

Two types of custom summary formulas are available with joined reports: standard and cross-block.

Standard Custom Summary Formulas

Standard custom summary formulas apply to one report type, and can be added to blocks that are based on that report type only. For example, a summary formula created for the Cases report type can only be applied to Cases blocks. The configuration options for custom summary formulas are the same for joined reports, reports grouped by rows (summary), and reports grouped by rows and columns (matrix).

Keep the following in mind when working with custom summary formulas in joined reports.

- In Lightning Experience, formulas are automatically added to the block from which they're created. In Salesforce Classic, the formulas aren't automatically added to the report when you create them. To add a formula to a block in Salesforce Classic, drag it to a block with the same report type.

- When you add custom summary formulas to a block, they appear to the right of the standard fields in the order in which you added them to the block. If you also add cross-block custom summary formulas, they appear to the right of the standard ones.

- The results of custom summary formulas are affected by the filter options applied to the blocks they're included in. As a result, the same formula can yield different results in different blocks.

- You can add up to 10 custom summary formulas to each block in a joined report. A joined report can have a total of 50 custom summary formulas.

- Each custom summary formula must have a unique name. However, standard and cross-block custom summary formulas can have the same name.

- Custom summary formula names can't include brackets ("[" or "]").

Cross-Block Custom Summary Formulas

Cross-block custom summary formulas let you calculate values across multiple blocks in a joined report. For example, you can use a cross-block formula to calculate the ratio of open to closed opportunities for an account or the ratio of closed pipeline deals to sales targets.

Building a cross-block formula is similar to creating a standard one. The same data formats, formula options, functions, and calculation display locations are available. The formula syntax is also similar, except that block information is also included. For example, when calculating the ratio of opportunities to cases for each account, the formula also includes the block title: `[Opportunities block 1]RowCount / [Cases block 2]RowCount`. If you omit block title, you see an error message when you check formula syntax or save the formula.

Keep the following in mind when working with cross-block custom summary formulas.

- You can add a cross-block formula to any block in the report.

- In Lightning Experience, cross-block formulas are automatically added to the report when you create them. In Salesforce Classic, cross-block formulas aren't automatically added to the report when you create them. To add a cross-block formula in Salesforce Classic, drag it to a block.

- When you add cross-block formulas to a block, they appear to the right of standard ones in the order in which you add them to the block.

- The results of cross-block formulas are affected by the filter options applied to the blocks in the report. As a result, a cross-block formula can yield different results when you change filter options.

- Each joined report can have up to 10 cross-block custom summary formulas.

- Deleting a block that's used in a cross-block formula also deletes the formula from both the Fields pane and any remaining blocks containing it.

- Each cross block formula must have a unique name. However, standard and cross-block custom summary formulas can have the same name.

- Cross-block formula names can't include brackets ("[" or "]").

Add a Summary Formula to a Joined Report

Summary formulas in joined reports are just like summary formulas in standard reports, except they can apply to multiple blocks or just one. When you add a summary formula to a block, you specify whether it should apply across blocks or only to that block.

SEE ALSO:

Add a Summary Formula to a Joined Report

Summary formulas in joined reports are just like summary formulas in standard reports, except they can apply to multiple blocks or just one. When you add a summary formula to a block, you specify whether it should apply across blocks or only to that block.

Add a summary formula to a joined report in Lightning Experience

1. Create or edit a joined report.

2. Ensure that there is at least one grouping on the joined report. If necessary, group data by clicking **Add group...** and selecting a field to group by.

3. From the Columns section of the OUTLINE panel, click the More Actions icon (▼) adjacent to a block name.

4. To add a summary formula that only evaluates data in one block, click **Add Summary Formula**. To add a cross-block summary formula that evaluates all the data in the joined report, click **Add Cross-Block Summary Formula**.

5. Write the formula. For help writing the formula, see Evaluate Groups and Totals with Summary Formulas on page 65.

6. Click **Apply**.

Add a custom summary formula to a joined report in Salesforce Classic

1. Create or edit a joined report.

2. Ensure that there is at least one grouping on the joined report. If necessary, group data by dragging and dropping a field onto the area that reads **Drop a field here to group across report blocks**.

3. To add a standard custom summary formula, from the Fields pane, go to the report type category and double-click **Add Formula**. To add a cross-block custom summary formula, from the Fields pane, go to the Common category and double click **Add Cross Block Formula**.

4. Write the formula. For help writing the formula, see Evaluate Groups and Totals with Summary Formulas on page 65.

5. Click **OK**.

The summary formula column appears in your report.

SEE ALSO:

> Evaluate Groups and Totals with Summary Formulas
>
> Summary Formulas with Joined Reports
>
> Get the Most Out of Summary Formulas: Tips, Limits, and Limitations
>
> Joined Report Examples

Work with Blocks

Blocks let you create different views of the information contained in a joined report.

1. Add a Block

   Adding blocks to joined reports lets you create multiple views of the data included in a single report.

2. Reorder Blocks

   You can reorder blocks in joined reports. Reordering blocks affects the report's appearance but doesn't affect the data in the blocks.

3. Rename a Block

   You can rename blocks to provide more user-friendly descriptions of the information they contain.

4. Show and Hide the Record Count for a Block

   You can choose to show or hide the number of records, or *record count*, for each block in a joined report. By default, record count is displayed for each block in the report builder and on the run reports page.

5. Delete a Block

   When you no longer need a block, delete it.

EDITIONS

Available in: both Lightning Experience and Salesforce Classic

Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

USER PERMISSIONS

To create, edit, and delete joined reports:

- **Legacy Folder Sharing**
  Create and Customize Reports

  AND

  Report Builder

  Enhanced Folder Sharing
  Create and Customize Reports

  AND

  Report Builder

Add a Block

Adding blocks to joined reports lets you create multiple views of the data included in a single report.

Add a block to a joined report in Lightning Experience

1. Create or edit a joined report.

2. Click 🔲 **Add Block**.

3. Choose a report type.

4. If you'd like to include standard columns for this report type, leave **Include default columns** checked. If you'd like to add an empty block, uncheck **Include default columns**. In either case, you can customize the block after adding it by adding or removing columns.

5. Click **Add Block**.

6. Click **Save**.

Add a block to a joined report in Salesforce Classic

1. Click and hold a field from the Fields pane, then drag it to the empty area of the Preview pane.
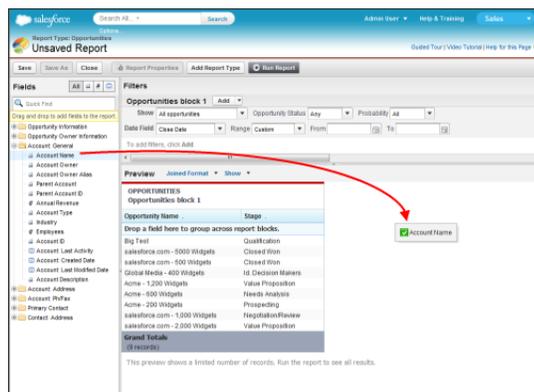
2. Drop the field to create the block.
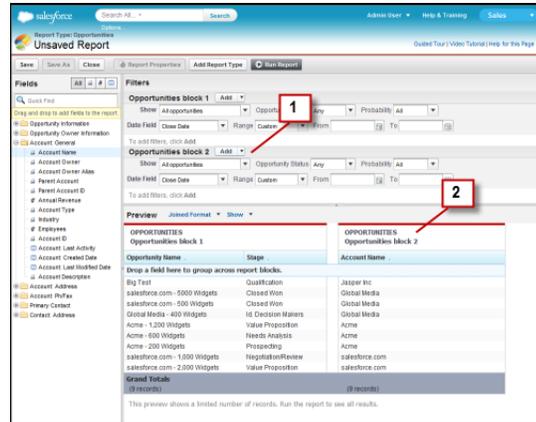
When you've added the block, notice that standard and field filters for the additional block are added to the Filters pane (1). Also, the new block appears in the Preview pane (2).

3. Click **Save**.

To delete a block, click ▼ in the block header, then click **Remove Block**.

SEE ALSO:

Add a Report Type to a Joined Report

Delete a Block

Add a Summary Formula to a Joined Report

Work with Blocks

Combine Different Types of Information in a Joined Report

Reorder Blocks

You can reorder blocks in joined reports. Reordering blocks affects the report's appearance but doesn't affect the data in the blocks.

When you reorder blocks, block numbers don't change. For example, if you have a report containing three blocks, and you move block 1 to a new position to the right of block 3, the blocks now display as block 2, block 3, and block 1. Rename the blocks to avoid confusion.

When reordering a block, you move it to either the left or right of an existing block.

Reorder a block in Lightning Experience

1. From the Columns section of the OUTLINE panel, click the More Actions icon (▼) adjacent to the name of the block you wish to reorder.

2. To move the block left, click **Move Block Up**. To move the block right, click **Move Block Down**.

3. Click **Save**.

Reorder a block in Salesforce Classic

1. In the Preview pane, drag the block to either the left or right side of an existing block. A blue bar beside the block indicates an acceptable drop location.

2. Drop the block in the desired location.

3. Click **Save**.

SEE ALSO:

Rename a Block

Work with Blocks

Rename a Block

You can rename blocks to provide more user-friendly descriptions of the information they contain.

When you add a block to a joined report, it's named automatically based on the report type and the number of blocks in the report. For example, if your report contains two blocks and you add a new block that's based on the Opportunities report type, it's named Opportunities block 3.

Rename a block in Lightning Experience

1. Create or edit a joined report.

2. From the preview panel, find the block you wish to rename and click the pencil icon (✏) next to the current block name.

3. Enter a new block name.

4. To keep the name, on your keyboard, press **Enter**.

5. Click **Save**.

Rename a block in Salesforce Classic

1. Create or edit a joined report.

2. Click the block's name.
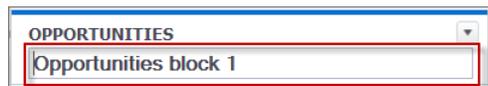
3. Enter the new name.

4. To apply the name, press **Enter** on your keyboard or click outside the block.

5. Click **Save**.

SEE ALSO:

Work with Blocks

Combine Different Types of Information in a Joined Report

Show and Hide the Record Count for a Block

You can choose to show or hide the number of records, or *record count*, for each block in a joined report. By default, record count is displayed for each block in the report builder and on the run reports page.

Show or hide record count for a block in Lightning Experience

For joined reports created in Lightning Experience, record count shows by default. You can't hide record count until you summarize another field in the report, like Amount. After adding a summary, click the **Row Count** toggle to hide record count. Click it again to show it.

If you hide record count and then remove all summaries from the report, then record count is automatically shown.

Show or hide record count for a block in Salesforce Classic

You can hide or show record count two ways.

The first: Click the arrow in the block header (▼) to display the block menu. The check mark beside the **Record Count** menu item shows that record count is enabled for the block. Click **Record Count** to toggle between showing and hiding the count.

The second: When you've hidden report details, position the cursor over the Record Count column to display an arrow (▼). Click the arrow, and select **Remove Column**. To display record count again, click the arrow in the block header and select **Record Count**.

If you haven't summarized any rows in your report blocks and have also hidden both details for the report and row counts for all blocks, your blocks will be hidden on the run reports page. To display the blocks, choose **Show Details** (in Salesforce Classic) or toggle the **Row Counts**, **Detail Rows**, **Subtotals**, or **Grand Total** switches (in Lightning Experience) from the report run page or the report builder.

SEE ALSO:

Show and Hide Report Details

Use a Summary Function in a Custom Summary Formula

Delete a Block

When you no longer need a block, delete it.

Delete a block from a joined report in Lightning Experience

1. From the Columns section of the OUTLINE panel, click the More Actions icon (▼) adjacent to the name of the block you wish to delete.

2. Click **Remove Block**.

3. Click **Save**.

Delete a block from a joined report in Salesforce Classic

1. Click the arrow in the block header (▼) to display the block menu, then click **Remove Block**. Or, click and hold in the block header, and drag the block to the Fields pane.

2. Click **Save**.

SEE ALSO:

Work with Blocks

Combine Different Types of Information in a Joined Report

Joined Report Examples

Creating a sales rep performance scorecard, reviewing support cases by status, and predicting your opportunity pipeline are some examples of what you can do with joined reports. Here's how.

Accounts with Open Cases (Lightning Experience)

A joined report is an ideal way to see Accounts that have open Cases.

1. From the Reports tab, click **New Report** .

2. Select the *Accounts* report type. Then, click **Create**.

3. If necessary, customize standard filters. For example, set *Show* to **All accounts** and *Range* to **All Time**.

4. The report opens in the report builder. To turn it into a joined report, click **Report** ▼ > **Joined Report** > **Apply**.

5. To add another report type to the joined report, click 🔲 **Add Block**.

   a. Choose the **Cases** report.

   b. Click **Add Block**.

6. If necessary, continue to customize the report. For example, we want to see only open cases, which we can achieve by adding a field filter to block 2.

   a. From FILTERS, under the Cases block, click **Add filter...**.

   b. Select **Closed**.

   c. Select **False** and click **Apply**.

7. If necessary, group across blocks using common fields. Common fields are shared between all objects in the joined report. For example, we can group this report by *Account Owner*.

8. Click **Save and Run Report**.

Admire your handiwork and start gaining actionable insights!

Creating a Sales Rep Performance Scorecard (Salesforce Classic)

A sales rep scorecard lets your sales management team understand the performance and actions of your organization's sales reps. To create it, you need to have three separate custom report types, each of which creates a relationship between User (as the primary object) and one of the following three objects: Opportunity Owner, Opportunity Creator, and Activity Owner. Note that, in this example, we've named the custom report types `User and Opportunity Owner Custom Report`, `User and Opportunity Creator Custom Report`, and `User and Activity Owner Custom Report`.

Start by creating a new custom report based on the User and Opportunity Owner custom report type, and then add the User and Opportunity Creator and User and Activity owner custom report types as two additional blocks. Then, group by Sales Rep (opportunity owner) and set the filters as described in the procedure.

To create the report:

1. Create a new report, selecting `User and Opportunity Owner Custom Report` as the report type.

2. Select `Joined` from the **Format** drop-down.

3. Click **Add Report Type**.

4. Select `User and Opportunity Creator Custom Report`.

5. Click **Add Report Type** again, and choose `User and Activity Owner Custom Report`.

6. Group the blocks by `Full Name`.

7. Add additional fields and filters to the report as needed. For example, you might want to change the date filters to focus on rep performance during a particular time frame. Or, to make sure that only sales people are included as opportunity owners, create a filter on the `Role: Name` filter limit your results to users with "Sales" in their roles.

8. Optionally, provide names for the blocks.

9. Click **Save** or **Run Report**.

Reviewing Support Cases by Status (Salesforce Classic)

You can also create a report comparing the number of support cases that are new, closed, or in-progress by priority. The report contains a single standard report type: Cases. First, create the report, add three blocks to the report, filter each block by the appropriate status, and then use the `Priority` field for grouping.

To create the report:

1. Create a new report, selecting `Cases` as the report type.

2. Select `Joined` from the **Format** drop-down.

3. Remove unwanted fields by dragging them to the Fields pane.

4. Create three blocks, each containing the `Case Number` and `Status` and, optionally, `Case Owner` fields.

5. For each block, filter on all cases. Then, filter each of the blocks by `Status` as follows:

   - Block 1: *Status equals Closed*
   - Block 2: *Status equals New*
   - Block 3: *Status not equal to Closed, New, Closed in Portal, Closed — First Call*

6. Group the blocks by `Priority`.

7. Optionally, rename the blocks.

8. Click **Save** or **Run Report**.

Predicting Opportunity Pipeline (Salesforce Classic)

Using cross-block custom summary formulas, you can create a report that predicts future opportunity revenue based on your sales reps'
past performance. Create a report based on the Opportunities standard report type, add three blocks to the report, filter each block by
the appropriate status, and then create a formula that uses fields from different blocks.

To create the report:

1. Create a new joined report, selecting `Opportunities` as the report type.

2. Remove unwanted fields by dragging them to the Fields pane.

3. Create three blocks, each containing the `Opportunity Name`, `Account Name` and `Amount` fields.

4. For each block, show `All Opportunities`. Then, filter each block.

Block	Filters
Block 1	*Opportunity Status equals Closed*
	Date Field equals Close Date
	Range equals Current and Previous FY
	Stage equals Closed Lost
Block 2	*Opportunity Status equals Closed Won*
	Date Field equals Close Date
	Range equals Current and Previous FY

Block	Filters
Block 3	Date Field equals Close Date
	Range equals Next Month
	Opportunity Status equals Open

5. Group the blocks by Opportunity Owner.

6. Rename the blocks. For example, "Closed — Won", "Close — Lost", and "Closing Next Month".

7. Create a cross-block custom summary formula that predicts upcoming revenue based on past sales rep performance:

```
[Closing Next Month]AMOUNT:SUM*([Closed - Won]RowCount/([Closed -
                                Lost]RowCount+[Closed - Won]RowCount))
```

8. Add the formula to one or more of the blocks.

9. Optionally, add a cross-block custom summary formula that calculates the win ratio of each sales rep:

```
[Closed - Won]RowCount/([Closed - Lost]RowCount+[Closed - Won]RowCount)
```

10. Click **Save** or **Run Report**.
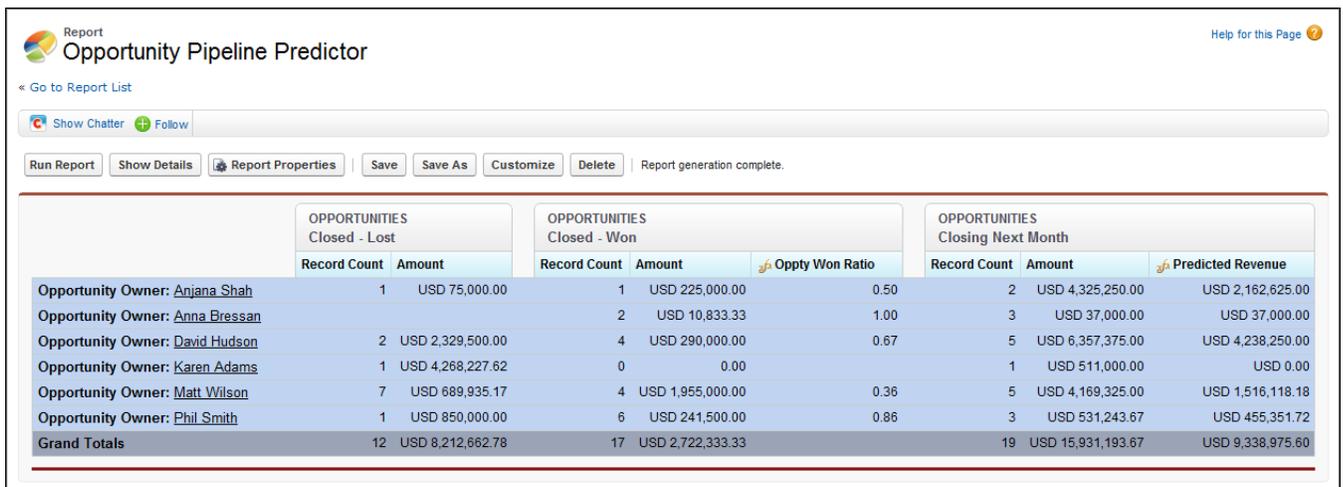
Adding a Chart to the Opportunity Pipeline Predictor Report (Salesforce Classic)

You can include a chart with a joined report to provide a visual representation of the data. For example, you can add a chart to the Opportunity Pipeline Predictor report that shows actual versus predicted revenue.

To add the chart:

1. Create the Predicting Opportunity Pipeline (Salesforce Classic) report.

2. Click **Add Chart**.

3. For the Y-axis, select Closed — Won — Sum of Amount.

4. Select **Plot additional values**.

5. Select Line for the Display option, and Predicted Revenue as the Value.

6. Click the Formatting tab, and enter `Actual versus Predicted Revenue` as the chart title.

7. Click **OK**.

8. Click **Save** or **Run Report**.
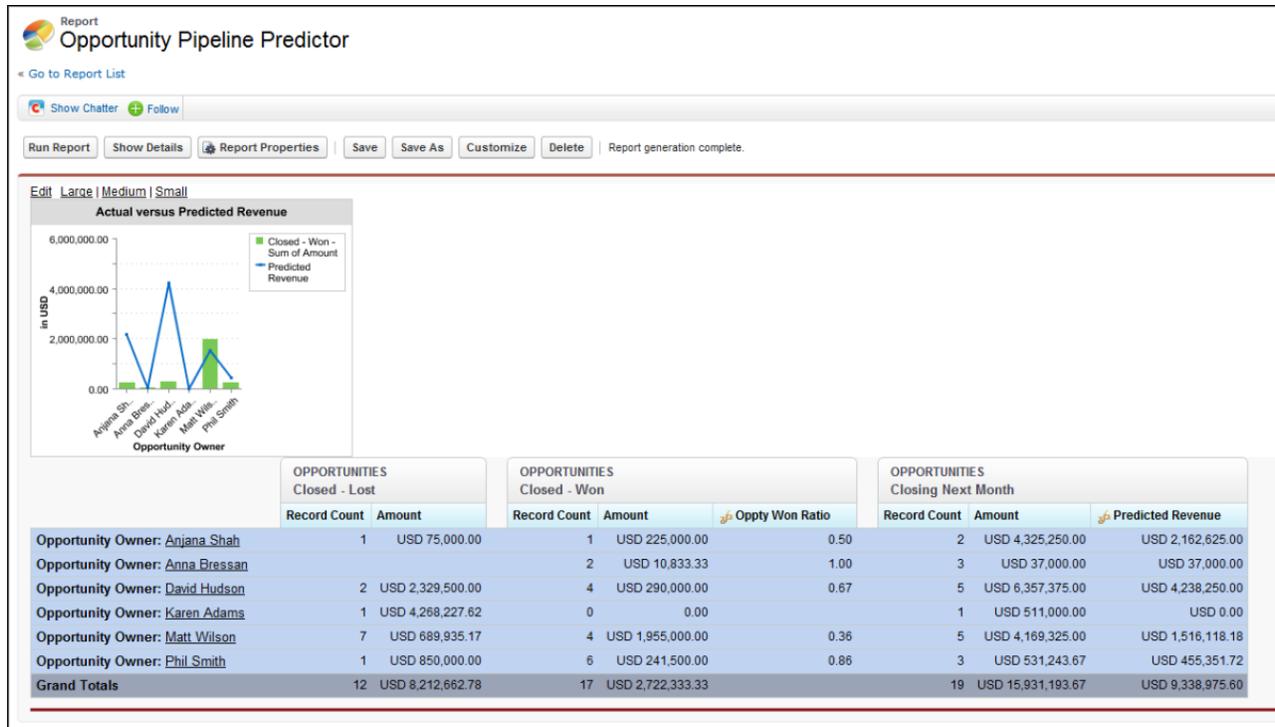
SEE ALSO:

Add a Chart to a Report

Summary Formulas with Joined Reports

How Joined Reports Work

Work with Blocks

Combine Different Types of Information in a Joined Report

Joined Report Limits, Limitations, and Allocations

Most of the things you can do with reports you can also do with joined reports, such as find, add, and remove fields; summarize fields; and run and save reports. However, there are some things you can't do. When working with Lightning Experience joined reports, there are some Lightning Experience-specific limits and limitations to consider.

Here are some things you can't do with joined reports.

- Add bucketed fields.

- Add cross filters.

- Drag and drop filters from the Fields pane on to the Filter pane.

- Apply conditional highlighting.

- Change the hierarchy for account, activity, lead, and opportunity reports.

EDITIONS

Available in: both Lightning Experience and Salesforce Classic

Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

- Create reporting snapshots based on joined reports.
- Schedule or subscribe to joined reports in Salesforce Classic.

> **Note:**
>
> - For users to be able to create and edit joined reports, report builder must be enabled for your entire organization. When report builder isn't enabled, users can run joined reports, but can't create them.
> - Joined reports require that the new user interface theme is enabled. Users without the new theme are unable to create, edit, or run joined reports.
> - Internet Explorer 6 is not supported for joined reports.
> - You can't filter data on a joined report in dashboard view or add a filter to a dashboard that only has joined reports.
> - When using the console in run mode, the **Hide Details** and **Show Details** buttons aren't available for joined reports. To use the buttons, click **Customize** to enter edit mode.
> - Joined reports aren't supported in console apps.

Standard Report Types That Can't Be Used in Joined Reports

- Accounts and Contacts
    - Account History
    - Account Owners*
    - Contact History

- Activities
    - My Delegated Approval Requests*

- Administrative
    - All Pending Approval Requests*
    - API Usage Last 7 Days*

- Campaign
    - Campaign Call Down*
    - Campaign Member
    - Campaign Member Analysis*
    - Campaigns with Influenced Opportunities

- Contract
    - Contract History
    - Order History

- Customer Support
    - Case History
    - Self Service Usage
    - Solution History

- File and Content
    - Content Authors
    - Content Publication Time Frame

- File and Content Downloads
- File and Content Engagement
- File and Content Links
- Library Administrators
- Library Content
- Most Content Downloads
- Most Content Subscriptions
- Stale Content

- Lead
  - Lead All
  - Lead History
  - Lead Status*

- Opportunity
  - Opportunities with Contact Roles and Products
  - Opportunities with Opportunity Teams and Products
  - Opportunity Field History

- Price Books, Products and Assets
  - Assets without Products*

> **Note:**
> - You might not have access to all the report types listed here. Certain factors can affect the report types you see, such as which features your organization has enabled and how your administrator has set up report folder visibility. Check with your administrator if you think you should see a report type that you don't.
> - If your organization has renamed standard objects, the names of the standard report types will contain your organization's names instead of the original ones. For example, if your organization has renamed the "Opportunity" object as "Deal," the standard report type "Opportunity Field History" will be renamed "Deal Field History."
> - In this list, report types marked with an asterisk (*) aren't available when you create a new report. Instead, you access them by customizing standard reports, which are in folders on the Reports tab.

Lightning Experience Joined Report Feature Gaps, Differences, and Limitations

As you get ready to work with joined reports in Lightning Experience, take note of these limitations. These features aren't available yet, but we're working hard to implement them in a future release.

- Subscriptions
- Export as details only (Formatted report export is supported)
- Open in Quip
- Conditional formatting
- Trend report data in Einstein Analytics

- Einstein Data Insights

SEE ALSO:

Choose a Report Type

Change the Principal Report Type

Combine Different Types of Information in a Joined Report

Standard Report Types

Report on Historical Changes

On top of the standard up-to-the-minute reporting on the current state of your business, you can analyze day-to-day and week-to-week changes in opportunities, cases, forecasts, and custom objects.

📝 **Note:** Historical trend reports are also called historical tracking reports.

1. Track Changes Over Time with Historical Trend Reporting

   Historical trend reporting uses a special custom report type designed to highlight changes between five snapshot dates, such as five business days or five business weeks. You can visually represent the data changes in charts and on dashboards.

2. Report on Historical Data with Reporting Snapshots

   A reporting snapshot lets you report on historical data. Authorized users can save tabular or summary report results to fields on a custom object, then map those fields to corresponding fields on a target object. They can then schedule when to run the report to load the custom object's fields with the report's data. Reporting snapshots enable you to work with report data similarly to how you work with other records in Salesforce.

Track Changes Over Time with Historical Trend Reporting

Historical trend reporting uses a special custom report type designed to highlight changes between five snapshot dates, such as five business days or five business weeks. You can visually represent the data changes in charts and on dashboards.

For organizations created in Winter '14 and thereafter, historical trend reporting is activated by default. If your organization is older than that, you must activate historical trend reporting in the Setup menu.

Salesforce retains historical data for the previous three months, plus the current month. (The amount of historical data you can work with in practice depends on your organization's data design and use patterns.) You can select up to five date snapshots in that span to compare, using up to four historical filters.

Longer durations than days or weeks are not recommended. They may result in reports timing out and not returning. For small organizations with fewer records, month-to-month trend reporting may work, but this is not what historical trend reporting is designed for.

📝 **Note:** Historical trend reports are also called historical tracking reports.

💡 **Tip:** You can also create and customize historical trend reporting reports via the Metadata API and the Reports and Dashboards REST API.

Track Changes in Your Sales Pipeline

A historical report can help you monitor your company's sales pipeline to make sure it contains enough activity to meet current and future sales goals. You may want to focus on deals whose value grew or shrank in the last three months, or deals that moved into or out of a given target period.

Let's build a historical report, in matrix format, of the amounts your team members have assigned to each forecast category. Changes in those amounts can reveal how accurately your team is forecasting.

Monitor activity across cases and identify your case status changes in the last three months of a case's life cycle.

Historical trending in Salesforce is subject to certain limits on the time during which data is tracked, the number of rows of data, and the fields and objects you can track. These limits are designed to restrict the data set so that reports return data quickly.

Track Changes in Your Sales Pipeline

A historical report can help you monitor your company's sales pipeline to make sure it contains enough activity to meet current and future sales goals. You may want to focus on deals whose value grew or shrank in the last three months, or deals that moved into or out of a given target period.

You'll probably want to compare historical and current values of key attributes of opportunities, such as dates, amounts, and status, to see how your pipeline has changed over time. For example, this Lightning Experience report points out opportunity amounts and close dates that have changed since 3 months ago.

- Snapshot up to five dates to track day-to-day or week-to-week trends.
- Current and historical values are shown side by side for easy comparison.
- Changed dates and amounts are highlighted in red or green.

1.

   Let's say you want to know how the total value of your pipeline today compares with its value yesterday. Create a simple report to compare the two dates.

2.

   Suppose you want to focus only on deals in your pipeline worth more than $5,000, but you're not concerned about today's fluctuations. You'll need a historical trending report that filters out any deal whose value was below $5,000 yesterday.

3.

   To focus on deals that are taking more time to close than expected, create a historical trending report that finds deals in your pipeline that have had their close dates moved to a later date.

4. Identify Shrinking Deals

Historical trending analysis can help you prioritize by quickly identifying deals that may be at risk. For example, target the deals in your pipeline that have decreased in value since yesterday. You'll need a historical trending report that gives you the deals whose value yesterday was greater than their value today.

Watch Your Pipeline Change Over Time

Let's say you want to know how the total value of your pipeline today compares with its value yesterday. Create a simple report to compare the two dates.

To see Opportunities with Historical Trending, enable Historical Trend Reporting for Opportunities in Setup. For more information, see *Set Up Historical Trend Reporting* in Salesforce Help.

1. Create an opportunity history report.

   a. On the Reports tab, click **New Report**.

   b. Select **Select Report Type** > **Opportunities** > **Opportunities with Historical Trending**.

   To see Opportunities with Historical Trending, enable **Historical Trend Reporting for Opportunities** in Setup. For more information, see *Set Up Historical Trend Reporting* in Salesforce Help.

   c. Click **Continue**.

2. Extend the report to all opportunities and a date range of interest.

   💡 Tip: To select all opportunities click **All Opportunities** when the report first opens, or click 🔻 **FILTERS** and then click **Show Me** to make your selection. To change the time range for the close date comparison, click **Close Date** under 🔻 **FILTERS**.

3. Click **Outline**.

   Note that the default value in the `Snapshot Dates` is *Yesterday*, and the `Amount (Historical)` field shows yesterday's date.

   💡 Tip: "Yesterday" is a *rolling* date value, meaning that it points to a date that is relative to today's date. If you run this same report tomorrow, `Amount (Historical)` will show today's date.

4. To add a different start date for the snapshot comparison, click **+** and select the date. To remove a snapshot date, click **X**.

5. In the report table, click the down arrow in the headers of the `Amount (Historical)` and the `Amount` columns. Click *Summarize*, then select *Sum*.

6. In the report table, click the down arrow in the header of the `Amount` column. Click *Show Change*, then select *Value*.

7. Click **Run Report**.

8. In the `Change` column, observe the difference between the total value of the `Amount (Historical)` column and that of the `Amount` column.

   💡 Tip: You can see changes at a glance by looking for values that are colored green or red.

You've created a simple historical trending report that can help you analyze your pipeline's behavior.

SEE ALSO:

Opportunities with Historical Trending Report

Identify Historical Deals Over a Given Value

Suppose you want to focus only on deals in your pipeline worth more than $5,000, but you're not concerned about today's fluctuations. You'll need a historical trending report that filters out any deal whose value was below $5,000 yesterday.

1. Create an opportunity history report.

   a. On the Reports tab, click **New Report**.

   b. Select **Select Report Type** > **Opportunities** > **Opportunities with Historical Trending**.

      To see Opportunities with Historical Trending, enable **Historical Trend Reporting for Opportunities** in Setup. For more information, see *Set Up Historical Trend Reporting* in Salesforce Help.

   c. Click **Create**.

2. For Show, select **All Opportunities**.

3. Filter for historical values over $5,000.

   a. Click **Add** and select *Historical Field Filter*.

   b. Set *Amount (Historical)* to *Yesterday*.

   c. Select *greater than* for the operator.

   d. Enter *5,000* in the last field.

   e. Click **OK**.

4. Click **Run Report**.

5. In the finished report, observe that all the deals in the `Amount — Historical` column are worth more than $5,000.

> 💡 Tip: Values in the Amount, Close Date and Stage fields are shown in green or red to indicate the direction of change. You can
> reverse the colors by clicking the down arrow in the column header.

You've created a simple report that pulls out all the deals that have were worth more than a given amount as of a given historical snapshot
date.

SEE ALSO:

[Opportunities with Historical Trending Report](#)

Find Deals that Have Been Pushed Out

To focus on deals that are taking more time to close than expected, create a historical trending
report that finds deals in your pipeline that have had their close dates moved to a later date.

1. Create an opportunity history report.

    a. On the Reports tab, click **New Report**.

    b. Select **Select Report Type** > **Opportunities** > **Opportunities with Historical Trending**.

    To see Opportunities with Historical Trending, enable **Historical Trend Reporting for
    Opportunities** in Setup. For more information, see *Set Up Historical Trend Reporting* in
    Salesforce Help.

    c. Click **Create**.

2. For Show, select **All Opportunities**.

3. Filter for close dates that are later now than they were in the past.

    a. Click **Add** and select `Historical Field Filter`.

    b. Set `Close Date (Historical)` to Feb. 1, 2013.

    > 💡 Tip: Use the calendar under **Fixed Days** to select the date.

    c. Click **OK**.

4. Click the down arrow in the header of the `Close Date` column and select `Show Changes`.

5. Click **Run Report**.

6. For each deal in the report, compare the date in the `Close Date — Historical`
   column with the date in the `Close Date — Today` column.

   > 💡 Tip: You can see changes at a glance by looking for values that are colored green or red.

SEE ALSO:

Opportunities with Historical Trending Report

Identify Shrinking Deals

Historical trending analysis can help you prioritize by quickly identifying deals that may be at risk. For example, target the deals in your pipeline that have decreased in value since yesterday. You'll need a historical trending report that gives you the deals whose value yesterday was greater than their value today.

1. Create an opportunity history report.

   a. On the Reports tab, click **New Report**.

   b. Select **Select Report Type** > **Opportunities** > **Opportunities with Historical Trending**.

      To see Opportunities with Historical Trending, enable **Historical Trend Reporting for Opportunities** in Setup. For more information, see *Set Up Historical Trend Reporting* in Salesforce Help.

   c. Click **Create**.

2. For Show, select **All Opportunities**.

3. Filter out any deal that wasn't worth more yesterday than today.

   a. Click **Add** and select *Historical Field Filter*.

   b. Set `Amount (Historical)` to `greater than`.

   c. Set the last field to `Field`.

      Here we're choosing to compare the historical amount with whatever amount is in the `Amount — Today` column, and not with a specific amount.

   d. Click **OK**.
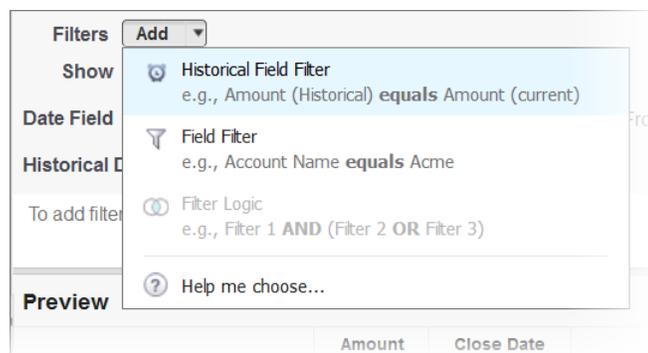
4. Click **Run Report**.

You've created a simple report that flags deals that have shrunk since yesterday. Note that all the results in the `Amount — Today` column are shown in red, to indicate decreases in value.

💡 **Tip:** To change the color-coding for amount changes, click the down arrow above the Change column and click **Reverse Colors**.

SEE ALSO:

[Opportunities with Historical Trending Report](#)

See How Forecast Amounts Have Changed

Let's build a historical report, in matrix format, of the amounts your team members have assigned to each forecast category. Changes in those amounts can reveal how accurately your team is forecasting.

For this example, we'll assume you're about a month into the current fiscal quarter.

1. Create a forecast history report.

   a. On the Reports tab, click **New Report**.

   b. Select **Select Report Type** > **Forecasts** > **Forecasting Items with Historical Trending**. Before you can select this report type, your org must have enabled **Historical Trending for Forecasting Items**. If you do not see the report type listed, contact your administrator.

   c. Click **Create**.

2. In the Filters area, choose **Show** > **All forecasting items**, then set up your filters.

   a. For **Date Field**, choose End Date. For **Range**, choose Current FQ. The **From** and **To** date fields automatically get the beginning and end dates for the current quarter.

   b. For **Historical Date**, choose `3 Months Ago`, `2 Months Ago`, then `1 Month Ago` from the dropdown menu. Each date you select is added to the field.

Filter for the historical data you want.
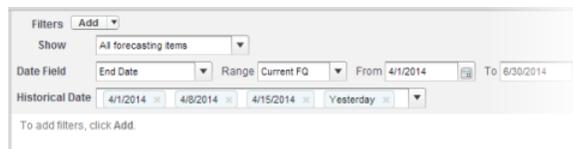
💡 **Tip:** To move `Yesterday` to the right side of the field, delete it, and then add it again from the dropdown menu.

If you're using multiple forecast types, add a filter for Forecasting Type: API Name to prevent duplicate values from appearing on the report.

3. Choose the data you want to monitor for historical changes. In this case, we're interested in the category in which the changes were made and how the forecast amount changed.

   a. In the **Preview** pane, change **Tabular Format** to **Matrix Format**.

   b. Drag the `Owner: Full Name` field from the Field area to the Preview pane to create a row grouping.

   c. Drag the `Forecast Category` field over to the right of `Owner: Full Name` to create another row grouping.

   d. Drag the `Forecast Amount (Historical)` field to the matrix area of the Preview pane, below the yellow bar. In the **Summarize** dialog, select `Sum`.

Select the fields that contain the data you want to track.
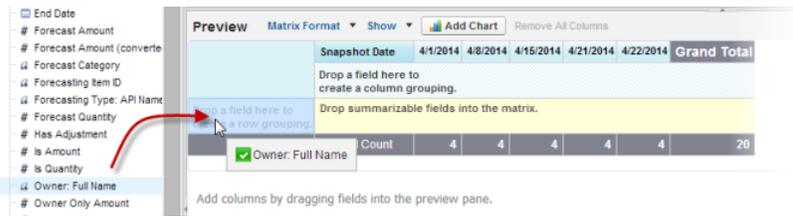
Columns appear for each of the historical dates you chose in the filters area that contain records.

4. Click **Run Report** to see what data you've gathered so far. Then click **Customize** to keep adjusting your filters and fields if necessary.

5. Once you've defined the data you want to track, create a chart to show it graphically.

    a. In the Preview area, click **Add Chart** and select the line chart icon.

    b. On the Chart Data tab, choose `Sum of Forecast Amount (historical)` for the **Y-Axis** and `Snapshot Date` for the **X-Axis**.

    c. For **Group By**, choose `Forecast Category`, and select **Cumulative**.

    d. Click **OK**, then click **Run Report** again.

6. Click **Save As**. Choose a name and folder for the report and click **Save**.

Track History for Cases

Monitor activity across cases and identify your case status changes in the last three months of a case's life cycle.

1. One way to optimize contact center operations is to observe the `Status` field over time, watching for cases that move backward to a previous status. This can help reveal ways to resolve cases more effectively.

2. For another example, try analyzing historical values of the `Priority` field to identify cases that may have been incorrectly classified when they were opened. Watching for frequent changes in priority may also lead to ways to improve the handling of complex cases.

Limitations on Historical Trend Reporting

Historical trending in Salesforce is subject to certain limits on the time during which data is tracked, the number of rows of data, and the fields and objects you can track. These limits are designed to restrict the data set so that reports return data quickly.

- Salesforce retains historical data for the previous three months, plus the current month.

- Up to 5 million rows of historical trending data can be stored for each object. Historical data capture stops when the limit is exceeded. Email is sent to the admin when any object reaches 70 percent of the limit, and again if the limit is exceeded. Alerts continue until the percentage is reduced below 70.

- Each historical trend report can contain up to 100 fields. In Opportunities reports, the fields include the standard preselected fields, which can't be disabled.

- Formula fields aren't supported.

- Row limit filters aren't supported.

- The summary report format isn't supported.

- You can specify up to five historical snapshot dates in each historical trend report.

- You can use up to four historical filters on each historical trend report.

- These field types are supported: Number, Currency, Date, Picklist, Lookup.

- Dynamic exchange rates aren't supported. When you run a historical trend report, it uses a static exchange rate, which could be outdated.
- Internet Explorer 6 isn't supported.
- You can't subscribe to historical trend reports in Lightning Experience or Salesforce Classic. For matrix reports in Salesforce Classic, the UI provides settings to subscribe, but the subscription emails aren't sent.
- The Report Wizard isn't supported. Historical trend reports can only be created with the Report Builder.
- Historical trend reporting with charts is supported in Lightning Experience, but tabular views of historical trend reports aren't available.
- When you enable historical trending, data collection starts from when the record was last modified.

> ⓘ **Important:** If a picklist field is already being used in a historical trending data filter, think hard before changing any of its values. You could make that field less useful for historical reporting in the future.

Report on Historical Data with Reporting Snapshots

A reporting snapshot lets you report on historical data. Authorized users can save tabular or summary report results to fields on a custom object, then map those fields to corresponding fields on a target object. They can then schedule when to run the report to load the custom object's fields with the report's data. Reporting snapshots enable you to work with report data similarly to how you work with other records in Salesforce.

After you set up a reporting snapshot, users can:

- Create and run custom reports from the target object.
- Create dashboards from the source report.
- Define list views on the target object, if it's included on a custom object tab.

For example, a customer support manager could set up a reporting snapshot that reports on the open cases assigned to his or her team everyday at 5:00 PM, and store that data in a custom object to build a history on open cases from which he or she could spot trends via reports. Then the customer support manager could report on point-in-time or trend data stored in the custom object and use the report as a source for a dashboard component. For the total number of reporting snapshots you can create, see *Salesforce Limits*.

> 📝 **Note:** Reporting snapshots don't support row-level formula fields.

1. Prepare Reporting Snapshots

   To set up a reporting snapshot, you need a source report and a target object with fields to contain the data in the source report.

2. Define a Reporting Snapshot

   After you create a source report, target object, and target object fields, you can define your reporting snapshot. You define a reporting snapshot by naming it and choosing the source report that will load report results into the target object you specify when the reporting snapshot runs.

3. Map Reporting Snapshot Fields

   After you create a source report, target object, target object fields, and define your reporting snapshot, you can map the fields on your source report to the fields on your target object. You map source report fields to target object fields so that when the report runs, it automatically loads specific target object fields with data from specific source report fields.

4. Schedule and Run a Reporting Snapshot

   After you create a source report, target object, target object fields, define your reporting snapshot, and map its fields, you can schedule when it runs. You can schedule a reporting snapshot to run daily, weekly, or monthly so that data from the source report is loaded into the target object when you need it.

5. Manage Reporting Snapshots

    After you set up a reporting snapshot, you can view details about it and edit and delete it. From Setup, enter `Reporting Snapshots` in the `Quick Find` box, then select **Reporting Snapshots** to display the Reporting Snapshots page, which shows the list of reporting snapshots defined for your organization.

6. Troubleshoot Reporting Snapshots

    The Run History section of a reporting snapshot detail page displays if a reporting snapshot ran successfully or not. When a reporting snapshot fails during a scheduled run, the failure is noted in the `Result` column. To view the details of a run, click the date and time of the run in the `Run Start Time` column.

SEE ALSO:

    Build a Salesforce Classic Dashboard

    Create a Custom Report in Accessibility Mode

Prepare Reporting Snapshots

To set up a reporting snapshot, you need a source report and a target object with fields to contain the data in the source report.

To set up a reporting snapshot:

1. Create a new custom report that includes the fields you want to load as records into a target object.

2. Create a new custom object in which to store the records loaded from the source report.

3. Create fields on the target object that will receive the source report's results when the reporting snapshot runs.

Tips on Source Reports for Reporting Snapshots

- If you save a tabular source report with its details hidden, the report will not be available to include in a reporting snapshot. Furthermore, if you hide the details of a tabular source report included in a reporting snapshot, the reporting snapshot will fail when it runs. To verify that the details of the tabular source report are not hidden, view the report, click **Show Details**, and save the report. The **Show Details** button only displays if the report's details are hidden.

- When creating the source report for your reporting snapshot, note the names of the fields you added to the report, as those field names may be useful to you when you create fields on the target object in which to store the report results.

- You can choose any custom tabular or summary report as the source report, except legacy forecast reports, Quota vs Actual reports, and Leads by Source reports. The `Source Report` drop-down list does not display standard reports..

- You can include up to 100 fields in your source report.

- You can delete the schedule of when a reporting snapshot runs. You can't stop or pause a reporting snapshot when it is running, nor can you delete its source report. To delete the source report, you must first remove the report from the reporting snapshot by changing the report in the `Source Report` drop-down list..

- If you select Load No Data in the Fields from Source Report column, no data will load into the corresponding field in the Fields in Target Object column when the reporting snapshot runs. .

- The (No fields with compatible data type) field displays in the Fields from Source Report column when a field on the target object does not match the data type of a field on the source report..

Tips on Target Objects for Reporting Snapshots

Consider the following when setting up target objects for reporting snapshots:

- In Professional, Enterprise, Unlimited, Performance, and Developer Editions, use field-level security to make the target object's fields visible to the appropriate users.
- You can't delete a custom object if it's a target object in a reporting snapshot.
- The fields on the target object determine field mapping availability. For example, your source report may include ten fields, but if your target object includes one field, then you can only map one field in your reporting snapshot.
- You can add up to 100 fields to the target object.
- Target objects cannot contain validation rules or be included in a workflow.
- Reporting snapshots cannot contain target objects that trigger Apex code to run when new records are created..
- When a reporting snapshot runs, it can add up to 2,000 new records to the target object. If the source report generates more than 2,000 records, an error message is displayed for the additional records in the Row Failures related list. You can access the Row Failures related list via the Run History section of a reporting snapshot detail page.

Tips on Reporting Snapshots

- Be aware of the type of license your Running User has. For example, if the Running User of a reporting snapshot has a Salesforce license, users who have Lightning Platform or Salesforce Platform One licenses will not be able to view it. Alternatively, if the Running User has a Lightning Platform or Salesforce Platform One license, users who have Salesforce licenses will be able to see the reporting snapshot. If you have users with Lightning Platform or Salesforce Platform One licenses, we recommend creating a separate reporting snapshot for them with a Running User that has a Lightning Platform or Salesforce Platform One user license.
- You can only map fields with compatible data types. For example, you can map a currency field to a number field.
- If you change the source report or target object on a reporting snapshot with existing field mappings, the field mappings are deleted when you save the reporting snapshot. You can also view Summary Fields in Source Report and Fields in Target Object to see the number of summary or target fields, respectively..
- You must map at least one field from the source report to one field on the target object or data will not load from the source report to the target object when the reporting snapshot runs.
- When a reporting snapshot is defined, deleted, or its source report or target object is changed, it is tracked in your organization's setup audit trail history.
- The Run History section on a reporting snapshot detail page displays details on when the reporting snapshot ran. Details include:
    - The date and time at which the reporting snapshot ran
    - The name of the source report, target object, and running user
    - The time it took for the reporting snapshot to run
    - The total number of detail or summary rows in the source report, depending on the report type
    - The number of records created in the target object

– Whether or not the reporting snapshot ran successfully

Define a Reporting Snapshot

After you create a source report, target object, and target object fields, you can define your reporting snapshot. You define a reporting snapshot by naming it and choosing the source report that will load report results into the target object you specify when the reporting snapshot runs.

1. From Setup, enter "Reporting Snapshots" in the `Quick Find` box, then select **Reporting Snapshots**.

2. Click **New Reporting Snapshot**.

3. Enter a name, unique name, and description for your reporting snapshot.

4. Choose a user in the `Running User` field by clicking the lookup icon.

   The user in the `Running User` field determines the source report's level of access to data. This bypasses all security settings, giving all users who can view the results of the source report in the target object access to data they might not be able to see otherwise.

   Only users with the "Modify All Data" permission can choose running users other than themselves.

5. Select a report from the `Source Report` drop-down list.

   The report you choose determines the report results that will load as records into the target object when the reporting snapshot runs.

   You can choose any custom tabular or summary report as the source report, except legacy forecast reports, Quota vs Actual reports, and Leads by Source reports. The `Source Report` drop-down list does not display standard reports.

6. Select a custom object from the `Target Object` drop-down list.

   The custom object you choose will receive the source report's results as records when the reporting snapshot runs.

   If a record used for an reporting snapshot has no record type associated with it, the record type of the running user is associated with the reporting snapshot by default.

7. Click **Save** to save the definition of your reporting snapshot, or click **Save & Edit Field Mappings** to save your reporting snapshot and map its fields.

8. Map the fields on the source report to the fields on the target object.

EDITIONS

Available in: both Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

USER PERMISSIONS

To create, save, and schedule a reporting snapshot:
- Manage Reporting Snapshots

To run a reporting snapshot as a running user and add the results to a custom object, the running user must have:
- Run Reports

  AND

  Create on the target object

Map Reporting Snapshot Fields

After you create a source report, target object, target object fields, and define your reporting snapshot, you can map the fields on your source report to the fields on your target object. You map source report fields to target object fields so that when the report runs, it automatically loads specific target object fields with data from specific source report fields.

1. From Setup, enter `Reporting Snapshots` in the `Quick Find` box, then select **Reporting Snapshots**.

2. Select the name of the reporting snapshot whose fields you want to map.

3. Click **Edit** on the Field Mappings section.

4. For summary reports, select the `Grouping Level` at which summary data is extracted from the source report. Data loaded into the target object is taken from summary fields at the grouping level you specify. The `Grand Summary` summarizes on the total for all grouping levels.

5. In the Fields from Source Report column, click a `Load No Data` drop-down list and select a field from the source report to map to a custom object field in the Fields in Target Object column. Only summary fields can be mapped for reporting snapshots based on summary reports. Note that the fields for summary reports may vary depending on the grouping level selected.

6. Click **Quick Save** to save field mappings and continue mapping fields, or click **Save** to save field mappings and return to the reporting snapshot's detail page.

7. Next, schedule the reporting snapshot to run.

Considerations for Mapping Reporting Snapshot Fields

- You must map at least one field from the source report to one field on the target object or data will not load from the source report to the target object when the reporting snapshot runs.

- You can only map fields with compatible data types. For example, you can map a currency field to a number field.

- A custom summary formula can be mapped only if the grouping level in the reporting snapshot and the grouping level in the custom summary formula match.

- If you select Load No Data in the Fields from Source Report column, no data will load into the corresponding field in the Fields in Target Object column when the reporting snapshot runs..

- The `(No fields with compatible data type)` field displays in the Fields from Source Report column when a field on the target object does not match the data type of a field on the source report.

- The fields on the target object determine field mapping availability. For example, your source report may include ten fields, but if your target object includes one field, then you can only map one field in your reporting snapshot.

- You cannot map fields from the source report to the following fields on the target object: `Created By, Last Modified By, Created Date,` and `Last Modified Date`.

- When you map fields from the source report to the target object, some data may lose its context when loaded to the target object. For example, if you map a date and time field from the source report to a text field on the target object, the date and time load to the target object without the time zone.

- When executing a reporting snapshot, if the running user does not have "read" or "write" access to a mapped field in the target object, that field is dropped from the mapping, but does not cause the execution to fail. If a required field in the target object is not mapped, the execution fails. To ensure that fields are always mapped, make them required or set default values for them.

- To map a field in the source report to a lookup field on the target object, you must map to the ID of the object associated with the lookup. For example, to map to an opportunity lookup field, you must map to the `Opportunity ID`. To get the `Opportunity ID` in the source report, you may need to use a custom report type to include ID and other related fields.

SEE ALSO:

Schedule and Run a Reporting Snapshot

After you create a source report, target object, target object fields, define your reporting snapshot, and map its fields, you can schedule when it runs. You can schedule a reporting snapshot to run daily, weekly, or monthly so that data from the source report is loaded into the target object when you need it.

The number of reporting snapshots you can schedule to run is determined by your Edition. After a reporting snapshot has run, you can send an email notification to yourself and other users that includes details about the reporting snapshot run, such as the date and time it ran, whether it ran successfully, and how many records were loaded into the target object from the source report. Also, the notification includes a link to the reporting snapshot detail page in Salesforce.

EDITIONS

Available in: both Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

1. From Setup, enter "Reporting Snapshots" in the `Quick Find` box, then select **Reporting Snapshots**.

2. Select the name of the reporting snapshot that you want to schedule to run.

   Reporting snapshots run as scheduled only if the user in the `Running User` field has access to the folder in which the source report is stored.

   📝 Note: If the running user becomes inactive, the report doesn't run. Salesforce sends an email notification to either activate the user, delete the report schedule, or change the running user to an active one. Salesforce sends the notification to users with the "Manage Users," "Modify All Data," and "Manage Billing" permissions. If no user has all these user permissions, Salesforce sends the notification to users with the "Manage Users" and "Modify All Data" user permissions.

USER PERMISSIONS

To create, save, and schedule a reporting snapshot:
- Manage Reporting Snapshots

To run a reporting snapshot as a running user and add the results to a custom object, the running user must have:
- Run Reports

   AND

   Create on the target object

3. Click **Edit** on the Schedule Reporting Snapshot section.

   The Schedule Reporting Snapshot section on a reporting snapshot detail page displays details on when the reporting snapshot is scheduled to run.

4. Select a notification setting to send an email when the reporting snapshot finishes running:

   - Click `Me` to send an email to the email address specified on your Salesforce user record.
   - Click `Others...` to send an email to additional users. You can only send reporting snapshot notifications to email addresses included on Salesforce user records. You can only select Users and Public Groups in the `Search` drop-down list. You can only share with internal users, not external users (such as Experience Cloud site users).

5. Schedule the reporting snapshot to run:

   📝 Note: If you schedule multiple report snapshots for the same time slot (for example, from multiple browsers), the report snapshot jobs may fail.

   - In the `Frequency` field, select the frequency at which the reporting snapshot runs. When you click the `Daily`, `Weekly`, or `Monthly` fields, more options display that allow you to refine frequency criteria.

If you schedule a reporting snapshot to run on a specific day of every month, the reporting snapshot will only run on months that have that specific day. For example, if you schedule a reporting snapshot to run on the 31st day of every month, then the reporting snapshot will only run on months that have 31 days. If you want a reporting snapshot to run on the last day of every month, choose last from the `On day of every month` drop-down list.

- In the `Start` and `End` fields, specify the dates during which you wish to schedule the reporting snapshot to run.

- In the `Preferred Start Time` drop-down list, click the **Find available options...** link to choose a preferred start time for the reporting snapshot to run.

  - Your preferred start time may not be available if other users have already selected that time to run a reporting snapshot or your organization has reached its reporting snapshot limit.

  - Reporting snapshots run in the time zone of the user who schedules the run. For example, if the `Time Zone` field on your user record is set to Pacific Standard Time, and you schedule a reporting snapshot to run every day at 2:00 PM, then the reporting snapshot will run every day at 2:00 PM Pacific Standard Time.

  - If you view and save a schedule in a time zone different from the one in which it was previously scheduled, the time slot could potentially change.

  - The reporting snapshot runs within an hour of the time you select in the `Preferred Start Time` drop-down list. For example, if you select 2:00 PM as your preferred start time, the reporting snapshot may run any time in between 2:00 PM or 2:59 PM, depending on how many other reporting snapshots are scheduled to refresh at that time.

6. Click **Save** to schedule the reporting snapshot to run.

When the reporting snapshot runs, it adds new records to the target object.

Optionally, once you have scheduled a reporting snapshot to run, you can perform the following actions after you click **Edit** on the Schedule Reporting Snapshot section of a reporting snapshot detail page:

- Click **Edit** to update the notification and frequency settings of the reporting snapshot.

- Click **Delete** to permanently delete the existing schedule of when the reporting snapshot runs.

A reporting snapshot will fail during a scheduled run if:

- The source report includes more than 100 fields.

- The source report was changed from summary to tabular.

- The selected grouping level for a summary source report is no longer valid.

- The running user does not have access to the source report.

- The running user does not have the "Run Reports" permission.

- The target object has more than 100 custom fields.

- The target object contains validation rules.

- The target object is included in a workflow.

- The target object is a detail object in a master-detail relationship.

- The target object runs an Apex trigger when new records are created on it.

- The running user does not have the "Create" permission on the target object. If the target object's status is `In Development`, the running user must have the "Customize Applications" permission.

- The reporting snapshot is scheduled with frequency set as Monthly and the run date is changed after the snapshot has run for the current month. The snapshot doesn't run on the new day for the current month unless the snapshot frequency is changed to Daily.

To unschedule a reporting snapshot from Setup, enter "Reporting Snapshots" in the **Quick Find** box, then select **Reporting Snapshots**. Click **Edit** for the snapshot and then click **Unscheduled Snapshot.** After unscheduling, you can set up another schedule as needed.

Manage Reporting Snapshots

After you set up a reporting snapshot, you can view details about it and edit and delete it. From Setup, enter `Reporting Snapshots` in the `Quick Find` box, then select **Reporting Snapshots** to display the Reporting Snapshots page, which shows the list of reporting snapshots defined for your organization.

From the Reporting Snapshots page, you can:

- Select a list view from the `View` drop-down list to go directly to that list page, or click **Create New View** to define your own custom view.

- Define a new reporting snapshot by clicking **New Reporting Snapshot**.

- Update the reporting snapshot name, description, running user, source report, and target object by clicking **Edit** next to its name. Only users with the "Modify All Data" permission can choose running users other than themselves. If you have the "Customize Application" permission, enter a unique name to be used by the API and managed packages.

  If you change the source report or target object on a reporting snapshot with existing field mappings, the field mappings are deleted when you save the reporting snapshot. You can also view `Summary Fields in Source Report` and `Fields in Target Object` to see the number of summary or target fields, respectively.

- Delete a reporting snapshot by clicking **Del** next to its name. After the reporting snapshot is deleted, it cannot be restored from the Recycle Bin.

  🛑 **Important:** When you delete a reporting snapshot, the source report and target object aren't deleted; however, when the source report runs, it won't load the target object with data.

  You can delete the schedule of when a reporting snapshot runs. You can't stop or pause a reporting snapshot when it is running, nor can you delete its source report. To delete the source report, you must first remove the report from the reporting snapshot by changing the report in the `Source Report` drop-down list.

- Display detailed information about a reporting snapshot and customize it further by clicking its name. Then you can:

  - Click links in the Identification section that redirect you to the reporting snapshot running user, source report, and target object. In addition, you can view the preferred date and time at which the reporting snapshot will approximately run next in the `Next Run` field, and view the date and time at which it last ran in the `Last Run` field.

  - Click **Edit** in the Field Mappings section to further customize the fields mapped from the source report to the target object.

    The Reporting Snapshot Field Mappings section displays which source report fields are mapped to the target object's fields. You can view the number of fields in the source report available for mapping to the target object in the `Columns in Source Report` field. Also, you can view the number of fields available for mapping in the target object in the `Fields in Target Object` field.

  - Click **Edit** in the Schedule Reporting Snapshot section to schedule when to run the reporting snapshot.

    The Schedule Reporting Snapshot section displays details about when the reporting snapshot is currently scheduled to run.

  - The Run History section displays details about when the reporting snapshot ran. Details include:

    - The date and time at which the reporting snapshot ran
    - The name of the source report, target object, and running user
    - The time it took for the reporting snapshot to run

- The total number of detail or summary rows in the source report, depending on the report type
- The number of records created in the target object
- Whether or not the reporting snapshot ran successfully

Up to 200 records are stored in the Run History section. After 200 records are stored, the oldest record is automatically deleted and cannot be retrieved from the Recycle Bin.

SEE ALSO:

Report on Historical Data with Reporting Snapshots

Troubleshoot Reporting Snapshots

The Run History section of a reporting snapshot detail page displays if a reporting snapshot ran successfully or not. When a reporting snapshot fails during a scheduled run, the failure is noted in the `Result` column. To view the details of a run, click the date and time of the run in the `Run Start Time` column.

💡 **Tip:**

- If the `Total Row Number` is blank, the run failed before the report was completed (for example, the report was invalid or the running user is inactive).
- When a reporting snapshot runs, it can add up to 2000 new records to the target object. If there are more than 2000 new records, the additional records are not recorded, and the notification indicates that some rows failed.
- The details of a failed run are available on the Row Failures related list for 14 days before they are automatically deleted. You can't retrieve details about row failures from the Recycle Bin.
- If you have a unique field in the target object, and records in the report have more than one of the same value in the column mapped to that unique field, duplicate records are not added. The run history indicates when records are not added to the reporting snapshot.
- If field mappings failed, the snapshot still runs, but the run history shows that there was a partial error.

A reporting snapshot could fail during a scheduled run for a number of reasons. This table lists the errors a failed run may display and how the errors can help you troubleshoot the reporting snapshot so that it will run successfully.

Running user does not have permission to run reports.
The user in the `Running User` field does not have the "Run Reports" or "Create and Customize Reports" permission. Choose a user with the appropriate permissions or enable the appropriate permissions for the running user.

Cannot run reporting snapshot because source report has been deleted.
The report in the `Source Report` field was deleted and no longer available to run. Choose another source report for your reporting snapshot or restore the deleted report from the Recycle Bin.

Running user does not have permission to access source report.
The user in the `Running User` field does not have access to the folder in which the source report is stored. Choose a user with access to the source report or provide the existing running user with access to the folder in which the source report is stored.

Source report definition is obsolete.
The report in the `Source Report` field references a custom or external object that is no longer available for reports or the relationships between the objects in the report have changed.

Source report definition is invalid.
The report in the `Source Report` field can't run because it contains invalid formulas or filter criteria. Update the report so that it can run without errors.

Running user does not have permission to access report type.
The user in the `Running User` field does not have permission to access a report type associated with the report in the `Source Report` field. Choose a running user that has the correct permissions or provide the existing running user with the appropriate permissions.

Source report must be tabular.
The report in the `Source Report` field is no longer in tabular format. Choose a new source report or update the existing source report's format to tabular.

Source report last saved with details hidden on report results.
The report in the `Source Report` field was saved with its details hidden. To display the details of the source report, view the report, click the **Show Details** button, and save the report.

Target object has been deleted or is inaccessible to running user.
The custom object in the `Target Object` field has been deleted or the user in the `Running User` field does not have permission to access the target object. Restore the deleted custom object, choose a new target object, or provide the existing running user with "Create" permissions to the custom object in the `Target Object` field. Note that if the custom object's status is `In Development`, the running user must have the "Customize Application" permission to access the target object.

Running user does not have permission to edit target object.
The user in the `Running User` field does not have "Create" permissions on the custom object in the `Target Object` field. Choose a running user that has such permissions or provide the existing running user with "Create" permissions to the custom object in the `Target Object` field.

Target object must not be a detail object in a master-detail relationship.
The custom object in the `Target Object` field is a detail object in a master-detail relationship, meaning that a master object controls certain behaviors on the target object's records. Choose a target object that is not included in a master-detail relationship.

Target object must not be included in a workflow.
The custom object in the `Target Object` field is included in a workflow. Choose a target object that is not included in a workflow.

Target object must not include an insert trigger.
An Apex trigger runs when new records are created for the custom object in the `Target Object` field. Remove the Apex trigger or choose a target object for which an Apex trigger does not run when new records are created.

Target object must not include validation rules.
The custom object in the `Target Object` field contains validation rules. Choose a target object that does not contain validation rules or delete validation rules from the existing target object.

Running user is inactive.
The user in the `Running User` field is no longer active. Choose an active user.

One or more required fields on the target object are not mapped.
One or more required fields on the target object are not mapped. Map all required fields on the target object to fields on the source report.

`Source report contains too many fields, has a formula field with too many functions, or contains too many criteria.`

The report in the `Source Report` field contains too many fields, criteria, or functions within a formula field. Remove any unnecessary fields, criteria and functions within formula fields from the source report.

`Source report cannot have more than 100 selected columns.`

The report in the `Source Report` field contains more than 100 fields. Remove any unnecessary fields from the source report.

`Target object cannot have more than 100 custom fields.`

The custom object in the `Target Object` field contains more than 100 custom fields. Remove any unnecessary fields from the target object.

`Your report exceeded the time limit for processing.`

The report in the `Source Report` field may contain too much data to process. Reduce the amount of data the report processes when running by limiting the report's date range and remove any unnecessary fields from the source report.

`This Reporting Snapshot Unique Name already exists. Please choose a unique name.`

The new snapshot you are trying to create has the same unique name as that of a existing snapshot.

`A summary field did not return a valid number.`

A summary field in the results has not returned a valid number. For example, the field may have attempted to divide by 0. Check your formulas and test for 0 and "null" in calculations if they appear in your data.

`There is a problem with this reporting snapshot. The source report format was changed from tabular to summary. The field mappings in the reporting snapshot are no longer correct. You can change the report format back to tabular or update the field mappings in the snapshot definition.`

The source report format was changed from tabular to summary, which made the field mappings in the reporting snapshot incorrect. To fix this error, either change the report format back to tabular or update the field mappings in the snapshot definition. This error only applies to reporting snapshots with summary reports.

`There is a problem with this reporting snapshot. Source report must be tabular or summary.`

The source report format must be either tabular or summary. Reports grouped by rows and columns (matrix) can't be used with reporting snapshots.

`The grouping level you specified in the reporting snapshot is no longer valid. The running user may no longer have access to that field, the grouping level may have been removed from the source report, or the grouping level was never set.`

This error may result when the running user no longer has access to the field specified in the grouping level, or the grouping level was removed from the source report or was never set. This error only applies to reporting snapshots with summary reports. Make sure the running user has access to all necessary fields.

SEE ALSO:

[Creating a Custom Report](#)

Report Type Reference

The report type you choose determines which records and fields appear in your report. For example, the Opportunities report type gives you access to Opportunity records and fields like Amoung, Stage and Opportunity Owner.

There are two kinds of report types: *standard report types* and *custom report types*.

Standard report types give you access to most Salesforce data. For example, the Opportunities report type gives you access to Opportunity records and fields. If you're going to report on Opportunity Amounts or Probability, then Opportunities is the report type for you.

Custom report types give you access to custom objects in Salesforce, or custom views of standard objects (like Opportunities), which your administrator configures. For example, your administrator can create a custom report type which gives access to Opportunities, plus related fields from Products. That way, you can report on Opportunities for a given product.

Standard Report Types

Salesforce provides a rich collection of standard report types that you can tailor to your unique requirements. You rarely need to create a brand-new report.

Pre-Designed Custom Report Types

Some Salesforce features come with custom report types that are designed for you in advance, so you don't have to create a new report.

Standard Report Types

Salesforce provides a rich collection of standard report types that you can tailor to your unique requirements. You rarely need to create a brand-new report.

> 💡 **Tip:** You may not see some of these folders if your administrator has customized the visibility of the Report tab folders.

If you can't find a report to customize for your own needs, you can also create a custom report to access exactly the right information.

1. ### Account and Contact Reports

   Use account and contact reports to learn about active, neglected, or new accounts, as well as accounts by account owner or partner. The two standard contact reports let you create a mailing list of contacts or track opportunities by contact role.

2. ### Activity Reports

   Activity reports are useful for gathering information about open activities, completed activities, multi-person events, or pending approval requests for which you are a delegated approver.

3. ### Administrative Reports

   Administrative reports help you analyze your Salesforce users, documents, and pending approval requests. You can report on the active Salesforce users and see who has been logging in.

4. ### Campaign Reports

   Use campaign reports to analyze your marketing efforts. You can report on the ROI of your campaigns, track who you targeted with your campaigns and who has responded, or analyze which opportunities resulted from your campaigns.

5. ### File and Content Reports

   Run File and Content reports to analyze how users are engaging with files and Salesforce CRM Content.

6. ### High Velocity Sales Reports

   High Velocity Sales (HVS) reports give you information about your inside sales efforts. As your reps work through cadences, you can evaluate how your prospects become qualified leads and which cadence step is most effective. You can also examine phone call statistics for your reps, which reps are converting the most prospects, and more.

7. ### Lead Reports

   Use lead reports to show information about the source and status of leads, how long it takes to respond to leads, neglected leads, and the history of lead fields.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** Editions except **Database.com** (The edition determines which reports you see.)

USER PERMISSIONS

To run reports:
- Run Reports

To schedule reports:
- Schedule Reports

To create, edit, and delete reports:
- Create and Customize Reports

  AND

  Report Builder

---

8. Opportunity Reports

   Opportunity reports provide information about your opportunities, including owners, accounts, stages, amounts, and more. The default settings show you the most commonly used information from each object, but you can customize a report to view other information, such as primary campaign source, forecast category, and synced quote.

9. Product and Asset Reports

   Use product and asset reports to view information about the products your users currently have installed. Find out what assets your customers have, list the cases filed for a particular asset, or identify assets that aren't associated with a product.

10. Self-Service Reports

   Self-Service reports help you analyze the effectiveness of your Self-Service portal. Find out how many cases are being viewed, how many customers are logging in, or what customers think of the solutions you're offering.

11. Reporting on Support Activity

   Use support reports to track the number of cases created, case comments, case emails, case owners, case contact roles, cases with solutions, the length of time since the case last changed status or owner, and the history of cases.

SEE ALSO:

Search for Reports and Dashboards in Lightning Experience

Search for Reports and Dashboards from the Reports Tab in Salesforce Classic

Creating a Custom Report

Account and Contact Reports

Use account and contact reports to learn about active, neglected, or new accounts, as well as accounts by account owner or partner. The two standard contact reports let you create a mailing list of contacts or track opportunities by contact role.

Standard Report: Field History

If your organization tracks field history on accounts or contacts, you can report on that information using the account history or contact history report.

Standard Report: Person Accounts

If your organization uses person accounts, fields specific to person accounts are available and prefixed with `Person Account:` in account reports. In addition, you can include the `Is Person Account` field in both account and contact reports. Your administrator may have given a different label to `Person Account`.

Mass Mail Merge

You can also create a report of your contact information, export that data to Microsoft® Excel®, and then do a mass mail merge using Microsoft® Word®.

`View` Filter for Account Reports

The standard `View` filter for account reports allows you to limit your account data according to the following options. These options vary depending on your organization's edition and setup.

- My accounts: Shows accounts that you own.

- My account team accounts: Shows accounts where you are on the account team.

- My account team and my accounts: Shows accounts you own and those where you are on the account team.

- My team's accounts: Shows your accounts and accounts owned by all of your subordinates in the role hierarchy.

- My territories: For organizations that use territory management, this option shows accounts that belong to the territories to which you are assigned.

- My territory team's accounts: For organizations that use territory management, this option shows accounts that belong to your territories and your territories' descendants.

- My team's account team and their own accounts: For users who report to you in the role hierarchy, shows accounts they own or for which they are on the account team.
- All visible accounts: Shows all accounts that you can view, as determined by your sharing model.
- Territories: For organizations that use previous versions of territory management (not Enterprise Territory Management), the additional `Territories` filter can be set to All, Multiple Territories, or Missing Territory. In custom report types, when using the `Territories` filter that includes territories, Multiple Territories or Missing Territories are not shown in the report results.
- Customer Portal: If your organization uses a Salesforce Customer Portal, add the `Customer Portal Account` field to your account reports to view which accounts have contacts enabled to use the portal.

SEE ALSO:

Limit Report Results

Activity Reports

Activity reports are useful for gathering information about open activities, completed activities, multi-person events, or pending approval requests for which you are a delegated approver.

Standard activity reports allow you to select the date range and status of the activities you want included. The standard activity reports list your tasks and appointments for a selected date range or events with all invitees.

You can also create custom reports for activities by clicking the Reports tab, **New Report**, and choosing **Activities** as the type of data on which to report.

> Note: Click **Show Hierarchy** to see your org's role hierarchy above the report results. You can use the role hierarchy to share report data with people at different levels in the hierarchy. For example, if you see **CEO > VP of Global Sales > Sales Operations Director**, you're viewing data for the Sales Operations Director role. Click any role name to see and share the data that's visible to people in that role.

Special Features of Activity Reports

Consider the following when running activity reports:

Standard Reports

- Choose the HTML Email Status report if you have the HTML email tracking enabled. This report covers anything in the HTML Email Status related list of your leads and contacts.
- Choose the Events with Invitees report to include only multi-person events in your report. The standard filters for events with invitees are:
  - Assigned to...—Shows only multi-person events that you own.
  - Assigned to the team of...—Shows multi-person events that anyone in your team owns.
  - Invitee is...—Shows only multi-person events that list you as an invitee.
  - Invitee is in the team of...—Shows the multi-person events that show anyone on your team as an invitee.
- The My Delegated Approval Requests report lists all the approval requests for which you are the approval proxy.

  > Note: The All Pending Approval Requests report is listed in the Activity Reports folder.

- In Professional, Enterprise, Unlimited, Performance, and Developer Edition organizations, to show the activities for users who report to you, use the **Hierarchy** links in the Tasks and Appointments report.

  > Note: You can view only your own activities and activities owned by users below you in the role hierarchy.

- If your organization uses Shared Activities, reports (including custom report types and Tasks and Events reports) display different results depending on your permissions. Say you're reporting on events, and your report results include an event that is related to two or more contacts and also has invitees. If you're an administrator, your report results show an event for the primary contact plus a separate event for each invitee. If you're not an administrator, your report results show just one event, for the primary contact.

  A user can access an activity's child event when one of the following guidelines is met:

  – The user is the owner of the activity or higher up in the hierarchy than the owner.

  – The user can access the who (parent record or contact) and the who count is one.

Report Types

- Using report builder, you can create activity reports that show activities related to another type of record. For example, a custom Activities with Leads report shows activities associated with leads.

- Activities with Leads reports do not display data for the `Address Line 1`, `Address Line 2`, and `Address Line 3` fields.

Tips for Activity Reports

- Set a search criteria of "Event Invitation equals 0" to filter out events that are meeting invites sent to users for a multi-person event.

- Archived activities aren't included in reports. Events and closed tasks older than a year are archived. However, open tasks aren't archived. You can still see archived activities for a record by selecting **View All** in the `Activity History` section of a record's detail page.

- Activities for private contacts are displayed only in reports for the contact owner.

- The standard filters for activity reports allow you to limit your report results using the following options. Some of these options aren't visible, depending on your edition.

  – My Activities—Shows activities that you own.

  – My Delegated Activities—Shows activities that you created but that someone else owns. That owner is in the same role as you, or below, in the role hierarchy.

  – My Team's Activities—Shows activities owned by users who report to you in the role hierarchy.

  – All Activities—Shows all activities that you can view, as determined by your sharing model.

- If you receive an error message that your activity report has too many results, customize the report to include a filter on a picklist, text, or date field. Alternatively, rerun the report using a different activity data type such as "Activities with Accounts" or "Activities with Opportunities."

SEE ALSO:

Limit Report Results

Administrative Reports

Administrative reports help you analyze your Salesforce users, documents, and pending approval requests. You can report on the active Salesforce users and see who has been logging in.

📝 **Note:** You can see the Administrative Reports folder on the Reports tab only if you have the "View Setup and Configuration" permission. You don't need this permission to view the Administrative Reports report type—all users can view it and manage any document reports associated with that type. To view other types of reports associated with that type, such as user and approval request reports, you must have the permission.

Special Features of Administrative Reports

Consider the following when running administrative reports:

Standard Reports

- The All Active Users report lists the active users in your organization and when they last logged in.
- The Users Logged in This Week report lists all of the users who have logged in to Salesforce in the past seven days. If Communities is enabled, you can add a `Community` column so that you can see which communities users have logged in to as well.
- The Documents report lists the documents within each document folder.
- The All Pending Approval Requests report lists the approval requests awaiting approval for each approval process. Note that the My Delegated Approval Requests report is listed in the Activity Reports folder.
- If your organization uses territories, the User Territory Report in the Territory Reports folder summarizes the territories to which users have been assigned.

Report Types

- Create a custom report that lists your organization's reports and the last time each report was used. Choose Administrative Reports and select Reports as the report type.
- If your organization uses Communities, you can report on login history for external members of your communities.
  1. Create a custom report.
  2. Choose Administrative Reports.
  3. Select Users as the report type.
  4. Add `Community` to your report columns.
  5. Add a filter that includes profiles of community users.
- If your organization uses a Salesforce Customer Portal, you can report on Customer Portal users:
  1. Create a custom report.
  2. Choose Administrative Reports.
  3. Select Users as the report type.
  4. Add `License Type`, `Profile`, and `Role` to your report columns. You can't report on roles for high-volume portal users because they don't have roles.

     📝 **Note:** Permission sets aren't supported.

  You can also add the `Customer Portal Account` field to your account reports to view which accounts have contacts enabled to use the portal.

- You can create custom report types from which users can report on your organization's reports and dashboards. When defining a custom report type, select Reports or Dashboards from the `Primary Object` drop-down list on the New Custom Report Type page.

Tips for Administrative Reports

- Users with the "Manage Internal Users" permission can create a custom user report that lists the details of users' login attempts. The relevant fields—such as `Login Date/Time`, `Source IP Address`, and `Login Status`—are grouped in the Login History section of the Select Columns step. Note that the `Client Type` field shows whether the user logged in via a Web browser or an alternate interface such as Connect for Lotus Notes or a partner portal. You can also see which users have never logged in by setting report criteria: choose the `Login Date/Time` field and the "equals" operator and leave the third value blank.

SEE ALSO:

[Limit Report Results](#)

Campaign Reports

Use campaign reports to analyze your marketing efforts. You can report on the ROI of your campaigns, track who you targeted with your campaigns and who has responded, or analyze which opportunities resulted from your campaigns.

Special Features of Campaign Reports

Consider the following when running Campaign Reports:

Standard Reports

- On the Campaign Detail custom report and the Campaign ROI Analysis report, you can include campaign-hierarchy statistics that provide aggregate values for a parent campaign and all the campaigns below it in the campaign hierarchy. If your campaigns include a custom picklist that indicates hierarchy level (for example, "tactic," "program," and "initiative"), you can run a report that summarizes data at any hierarchy level across all campaigns.

- Use the Campaign Leads or Campaign Contacts reports to list the leads or contacts associated with your campaigns.

- Use the Campaign Member Report for a list of campaign members by campaign.

- Use the Campaign Member Analysis report to summarize information about who has responded to campaigns.

- Use the Campaign Revenue Report to analyze which opportunities have resulted from your campaigns. In Enterprise, Unlimited, Performance, and Developer Editions, you can also analyze products, quantity schedules, and revenue schedules in this report.

- The Campaign ROI Analysis Report calculates the return on investment (ROI) and average costs for your campaigns. The ROI is calculated as the net gain (`Total Value Won Opps - Actual Cost`) divided by the `Actual Cost`. The ROI result is expressed as a percentage.

- Use the Campaigns with Influenced Opportunities report to view opportunities that have been influenced by multiple campaigns.

  📝 Note: The Campaigns with Influenced Opportunities report respects sharing rules on accounts, contacts, and campaigns. Objects with sharing rules set to private will not display in the report.

Report Types

- Use the Campaigns with Campaign Members custom report type to create a report that contains information about the leads and contacts on multiple campaigns. Use the Campaign Call Down report to see contacts and leads for a specific campaign. These reports are only available to users that have the "Read" permission on both contacts and leads.

- Use the Campaigns with Leads and Converted Lead Information report to view lead lifetime information sorted by a campaign or campaigns.

- Use the Campaigns with Influenced Opportunities report to view opportunities that have been influenced by multiple campaigns.

Tips for Campaign Reports

- Some reports allow you to limit the data to one campaign by using the lookup icon to select a campaign. If the user running a report no longer has access to view the selected campaign, the report does not show any results. This report behavior is similar to what happens when a campaign is deleted.

- `Member Status` is the status of a lead or contact in reference to the campaign. The campaign owner can create up to 50 member status values. Sample Member Status values include, "Planned," "Sent," or "Attended." Additionally, you can now add the `Member First Associated`, `Responded`, and `Member First Responded` fields to campaign reports. These fields allow you to see the date the member was added to the campaign, whether the member responded to the campaign, and the date the member initially responded to the campaign.

- The `Last Activity` of a campaign is the most recent due date of an activity on the record. The following past or future activities set this date:

  - Any event
  - Closed tasks

SEE ALSO:

Limit Report Results

File and Content Reports

Run File and Content reports to analyze how users are engaging with files and Salesforce CRM Content.

Standard File and Content Reports

Run these prebuilt reports about files and content.

Content Authors

View how many files, content packs, Google Docs, and Web links each author has published in Salesforce CRM Content.

Content Publication Time Frame

Determine how many files, content packs, Google Docs, and Web links were published in a library during a given time frame.

File and Content Downloads

See which users downloaded what files, and when they downloaded the files.

File and Content Engagement

See the number of times a file has been downloaded, shared, liked, and commented on.

File and Content Links

See which files users are sharing using content deliveries or **Share via link**. See when each link was created, when it expires, when it was last accessed, whether it is password protected, and the total number of times the link has been accessed. Each of the links in this report can be accessed by users outside of your organization.

Library Administrators

View the number of files, content packs, Google Docs, and Web links in each library as well as the total amount of file storage used by each library.

Library Content

View the number of files, content packs, Google Docs, and Web links in each library as well as the total amount of file storage used by each library.

Most Content Downloads

Determine which files and content packs are downloaded most frequently and which Web links and Google Docs are opened most frequently.

Most Content Subscriptions

Determine which files, content packs, Google Docs, and Web links have the most subscribers.

Stale Content

Determine which files have not been downloaded or revised recently.

File and Content Report Types

Build standard reports about files and content.

Content Report

Generate a report about Salesforce CRM Content.

Field	Description
`Archived`	Flag that indicates whether a file has been archived. Archiving a file removes it from its library but does not permanently delete the file from Salesforce CRM Content. Archived files can be restored as needed.
`Content Created By`	Contributor who published the file, content pack, Google Doc, or Web link.
`Content ID`	Identifier that enables you to group by file rather than version. The `Content Title` is not guaranteed to be unique because multiple versions of the same file can have different titles.
`Content Published Date`	Date a file, content pack, Google Doc, or Web link was first published.
`Content Title`	Title of a file, content pack, Google Doc, or Web link.
`Record Type`	Title of the record type associated with the file, content pack, Google Doc, or Web link.
`Featured Content`	Flag that indicates whether a piece of content is featured.
`Individual Content Size (MB)`	Size of an individual file, exclusive of other versions of the same file.
`Last Subscribed Date`	Date on which the content was subscribed to most recently.
`Num Downloads`	Number of times a file has been downloaded or the total number of downloads in a library. The number of times a Google Doc or Web link has been opened is also included in this count. **Tip:** To see which users have downloaded a certain file, go to the file's content details page and click the **Downloads** tab.
`Num Negative Ratings`	Number of thumbs-down votes.
`Num Positive Ratings`	Number of thumbs-up votes.
`Num Subscriptions`	Number of users who are subscribed to a file, content pack, Google Doc, or Web link, or the total number of subscriptions in a library.
`Num Versions`	Number of times a new version of a file has been published.

Field	Description
Overall Rating	The number of positive votes minus the number of negative votes. For example, if a file has two positive votes and no negative votes, its `Overall Rating` is 2.
Tag	Salesforce CRM Content tag assigned to a file, content pack, Google Doc, or Web link.
Total Content Size (MB)	Size of a file, inclusive of all the file's versions.
Version Revised Date	Date and time a new version of the file was published.
Version Revised By	Contributor who published the file version.
Library Name	Name of the library.
Library Created Date	Date and time the library was created.

Library and User Report

Generate a list of users who are assigned to libraries.

Field	Description
Administrator	Flag that indicates whether or not the user is a library administrator.
Group	Flag that indicates whether or not the library member is a single user or a public group.
Library Created	Date and time the library was created.
Library Name	Name of the library.
Member Name	First and last name of the library member.

File and Content Report

Generate a report about files uploaded to your organization.

Field	Description
Archived	Flag that indicates whether or not a file has been archived. Archiving a file removes it from its library but does not permanently delete the file from Salesforce CRM Content. Archived files can be restored as needed.
Title	Title of a file, content pack, Google doc, or Web link.
Created By	User who uploaded the file, content pack, Google Doc, or Web link.
Download Date	Date of the first file download.
Downloaded By	User who downloaded the file.
Download User Type	Profile type of the user who download the file.
File Type	MIME type of the uploaded file.
ID	Identifier that enables you to group by file rather than version. The `Title` is not guaranteed to be unique because multiple versions of the same file can have different titles.

Field	Description
`Last Revised By`	Last user to edit or upload a new version of the file.
`Last Revised Date`	Date of last file revision.
`Likes`	Number of times file posts have been liked. This number doesn't count likes applied to comments on the file post.
`Links`	Number of external file links that have been generated. External links are generated when a user shares a file using **Share via link** or when a user shares content from a library using a content delivery.
`Post Comments`	Number of comments made on file posts in feeds.
`Posts`	Number of times the file has been posted to a feed. This number doesn't count the number of times the file has been attached to comments on posts.
`Published Date`	Date a file, content pack, Google Doc, or Web link was first published or uploaded.
`Shares`	Number of times the file has been posted to a user's feed, shared using **Share with people**, and shared using **Share with groups**.
`Size (MB)`	Size of the most recent file version in megabytes.
`Total Downloads`	Number of times the file has been downloaded.
`Total Size (MB)`	Combined Size of all uploaded file versions.
`Versions`	Number of uploaded file versions.

Tips for Using File and Content Reports

- Salesforce CRM Content users who have the `Manage Content` option checked in their library permission can sort report data by the library they have access to ("My Libraries") or by all the libraries in an organization ("All Libraries"). Users without the `Manage Content` option can only sort data by the libraries they have access to.

- Sort custom reports by `Library Name` to view data for individual files as well as library summaries, such as total storage used and total number of downloads.

- Sort according to the `Content ID` to view data for a particular document.

- If you have Customer Portal or partner portal users with the "Create Libraries" user permission, run the Library Administrators report to determine which new libraries have been created by portal users.

SEE ALSO:

Limit Report Results

High Velocity Sales Reports

High Velocity Sales (HVS) reports give you information about your inside sales efforts. As your reps work through cadences, you can evaluate how your prospects become qualified leads and which cadence step is most effective. You can also examine phone call statistics for your reps, which reps are converting the most prospects, and more.

Once sales reps take some prospects through a sales cadence, six included reports provide managers with insights into the performance of their sales cadences, how their team is doing, and where they can improve.

Sales Cadence Completion Reason

For cadences that had contacts or leads added within the past 30 days, this report shows the percentage of cadence completions, by completion reason. The completion reason field is empty until a cadence is completed.

> 📝 **Note:** If you increase the report's time span and your company contains many leads or contacts in each cadence, the report may take longer to run. To improve the performance, use filtering to limit the result set returned by the report. For instance, reduce the time span to cadences less than the default 30-day setting.

Sales Cadence Engagement

Shows which sales cadence and which outreach step is the most effective for each type of prospect engagement, such as total calls or email delivery rate.

Lead Conversion

Shows how many leads sales reps are converting, the dollar value of the related opportunities, and who the top sales performer is.

Call Activity

Shows which sales reps are making the most phone calls, and what the results are.

Call Script Engagement

Shows how prospects responded to calls based on your call scripts. See exactly how many calls were successful.

Email Template Engagement

Shows how prospects responded to emails based on your email templates. See exactly how many prospects replied to each email.

EDITIONS
Available in: Lightning Experience
Available with High Velocity Sales, which is available for an extra cost in: **Enterprise**, **Performance**, and **Unlimited** Editions

Lead Reports

Use lead reports to show information about the source and status of leads, how long it takes to respond to leads, neglected leads, and the history of lead fields.

Special Features of Lead Reports

Consider the following when running lead reports:

Standard Reports

Choose the Lead History report type to track the history of standard and custom fields on leads where field histories are set to tracked. Use this report to see tracked fields' old and new values.

> 💡 **Tip:** If you have the "Create and Customize Reports" permission, you can use the `View` drop-down on a Lead History Report to view lead history data by My Leads, My Team's Leads, User Owned Leads, Queue Owned Leads, and All Leads.

Tips for Lead Reports

- Limit your report view to "My team's leads" to see leads owned by users who report to you in the role hierarchy.

- Lead reports can show all leads, both converted and unconverted. To limit your report to just unconverted leads, enter filter criteria of "Converted equals 0."
- The `Last Activity` of a lead is the most recent due date of an activity on the record. The following past or future activities set this date:
  - Any event
  - Closed tasks
- You can create a report of your lead information, export that data to Excel, and then do a mass mail merge using Microsoft® Word.

Lead Report Limitations

- You can't use filter conditions to search the results of the `Old Value` and `New Value` fields.
- You can't use filter logic if you are filtering by *Field/Event*.

SEE ALSO:

Limit Report Results

Opportunity Reports

Opportunity reports provide information about your opportunities, including owners, accounts, stages, amounts, and more. The default settings show you the most commonly used information from each object, but you can customize a report to view other information, such as primary campaign source, forecast category, and synced quote.

1. Tips for Working with Opportunity Reports

   Opportunity reports can include all opportunity fields plus some extra columns for more detail.

2. Standard Opportunity Reports

   Standard opportunity reports help you report on your opportunity pipeline and history, opportunity sources, opportunity types, and more.

3. Opportunity Report Types

   Report types provide a report template that you can customize to capture the unique data you're after without creating a report from scratch. Many of the opportunities custom report types include information from associated objects, such as products, partners, and quotes.

SEE ALSO:

Limit Report Results

Tips for Working with Opportunity Reports

Opportunity reports can include all opportunity fields plus some extra columns for more detail.

- You can use the following columns to provide more detail:
  - Age—Age counts the number of days passed between opportunity open date and close date. On a given day, it's possible for opportunity A to have been closed for 0:00 - 23:59 hours (which appears as 0) and opportunity B to have been closed for 24:00 hours or more (which appears as 1).
  - Stage Duration—The number of days the opportunity was in the stage listed in the Stage column. You can run the Opportunity Stage Duration report to see how much time an opportunity spent at different stages.

- Last Activity—The most recent due date of an activity on the opportunity record, including any opportunity event or closed task.

- Not all Opportunity Product fields are available in report filters. For example, the `Product Family` field is not available in Opportunity Product report filters because it is related to that object through the Pricebook Entry object. To make the `Product Family` field available in Opportunity Product report filters—for cross-sell or upsell reporting—create a custom formula field to store its contents and use that formula field in your filter. From the object management settings for opportunities, go to the fields area, then create a custom field of type Formula, making sure to use `Text` for your formula return type and `TEXT(PricebookEntry.Product2.Family)` for your formula.

- In Professional, Enterprise, Unlimited, Performance, and Developer edition organizations, the **Hierarchy** links let you browse report results based on the role or territory hierarchies.

- If your organization uses previous versions of territory management (not Enterprise Territory Management), the Hierarchy filter on opportunity reports lets you view data according to either the role or territory hierarchies. In addition, the Territories filter lets you display either opportunities from all territories or opportunities that lack an associated territory.

- You can include the `Primary Campaign Source` field on all standard opportunity reports. This field is controlled by field-level security.

- Use the View filter to limit your opportunity report results. View options vary depending on your organization's Edition and setup.

  - My opportunities—Shows only your opportunities.

  - My team-selling opportunities—Shows opportunities for which you are on the opportunity team.

  - My team-selling and my opportunities—Shows your opportunities and opportunities for which you are on the opportunity team.

  - My team's opportunities—Shows your opportunities and opportunities owned by all of your subordinates in the role hierarchy.

    If your organization uses previous versions of territory management (not Enterprise Territory Management), the effect of this option depends on the value of the Hierarchy filter above. If you select **Role**, you see your opportunities and opportunities owned by all of your subordinates in the role hierarchy. If you select **Territory**, you see opportunities that you own and any opportunities owned by your territories' descendants.

  - My team's team-selling and their opportunities—For users who report to you in the role hierarchy, this shows opportunities that they own or for which they are on the opportunity team.

  - My territories—For organizations that use original territory management or Enterprise Territory Management, this option shows opportunities that belong to the territories to which you are assigned.

  - My territories' opportunities—For organizations that use Enterprise Territory Management, this option shows opportunities that belong to the territories to which you are assigned.

  - My territory team's opportunities—For organizations that use Enterprise Territory Management, this option shows opportunities that belong to the territories to which you are assigned and their child territories.

  - All opportunities—Shows all opportunities that you can view.

SEE ALSO:

Opportunity Reports

Opportunity Report Types

Standard Opportunity Reports

Standard Opportunity Reports

Standard opportunity reports help you report on your opportunity pipeline and history, opportunity sources, opportunity types, and more.

1. In the `Folders` pane on the Reports tab, select **Opportunity Reports**.
2. Click the report you want.

Report Name	Description
Closed Opportunities	Won opportunities
Opportunities by Type	Types of available opportunities.
Opportunity Field History	Field history on opportunities. Available only if your organization tracks this information.
Opportunity History	Status changes on opportunities. Available only if your organization tracks this information.
Opportunity Pipeline Trend	A historical snapshot of your opportunities; opportunity amounts are grouped by historical stage for specified months. This report is available in Professional, Enterprise, Unlimited, Performance, and Developer Edition organizations only.
Opportunity Pipeline	Opportunities by stage.
Opportunity Pipeline with Splits	Opportunities summarized by split information such as assigned user and percentage.
Opportunity Product	Opportunities by month and product.
Opportunity Product Report with Splits	Split assignments and percentages for each product. You can also summarize information by these fields: • `Split Total Price`–Split percentage multiplied by Total Price. • `Split Expected Product Amount`–Split Total Price multiplied by Probability.
Opportunity Schedule Report with Splits	Opportunities by month, including split percentages for the opportunity team. To access this report, product scheduling must be enabled for your organization. You can also summarize information by these fields: • `Split Total Price`–Split percentage multiplied by Total Price. • `Split Schedule Amount`–Split percentage multiplied by Schedule Amount. • `Split Expected Product Amount`–Split Total Price multiplied by Probability. • `Split Expected Schedule Amount`–Split Schedule Amount multiplied by Probability.
Opportunity Sources	Sources of your opportunities.
Opportunity Stage Duration	Duration of an opportunity at each stage.

Report Name	Description
Opportunity Teams	Information about opportunities and opportunity teams to which you belong.
Partner Opportunities	All partners associated with an opportunity or the primary partners only. To limit your results to primary partners, customize the report and enter `Primary equals True` on the criteria page of the report wizard.
Stuck Opportunities	Open opportunities grouped by stage and then sorted by age.

3. Run the report.

SEE ALSO:

Opportunity Report Types

Report types provide a report template that you can customize to capture the unique data you're after without creating a report from scratch. Many of the opportunities custom report types include information from associated objects, such as products, partners, and quotes.

Opportunities Reports

View standard information about your opportunities, including owners, accounts, stages, amounts, and more. The default settings show you the most commonly used information from each object, but you can customize the report to view other information, such as primary campaign source, forecast category, and synced quote.

Opportunities with Contact Roles Report

View information about the contacts associated with your opportunities, including name, title, and role.

Opportunities with Contact Roles and Products Report

View information about the contacts and opportunities associated with a selected product. You must select a product to filter results by when you run the report. This report isn't supported in Lightning Experience. In Lightning Experience, select a different report type.

Opportunities with Competitors Report

View information about your company's competitors for opportunities, including their strengths and weaknesses.

Opportunities with Historical Trending Report

The Opportunities with Historical Trending report is a custom report type designed to help you analyze historical trends in your sales pipeline.

Opportunity Field History Report

View information about the change history of key opportunity fields, including old and new values and the dates edits were made.

Opportunity History Report

View information about the history of your opportunities, including stages and close date.

Opportunities with Opportunity Teams Report

View information about the members of your opportunity teams and their roles, organized by opportunity.

Opportunities with Opportunity Teams and Products Report

View information about your opportunity team members and their products, organized by opportunity. You must specify either a product or an opportunity team member to filter results by when you run the report.

Opportunities with Partners Report

View information about the partners you team with on your opportunities, including opportunity name, amount, and partner role.

Opportunities with Products Report

View information about the products associated with your opportunities, including product name and opportunity stage.

Opportunity Trends Report

View information about trends shared by the opportunities in your pipeline.

Opportunities with Quotes and Quote PDFs Report

View details about the quote PDFs created for each quote associated to an opportunity. The default settings show you the most commonly used information from each object, but you can customize the report to view other information, such as who created or last modified each listed quote PDF.

Opportunities with Quotes and Quote Line Items Report

View details about the quotes associated with opportunities, and the line items for each quote. The default settings provide the most commonly used information from each object, but you can customize the report to see any opportunity, quote, or quote line item field.

SEE ALSO:

Choose a Report Type

Opportunity Reports

Tips for Working with Opportunity Reports

Opportunities Reports

View standard information about your opportunities, including owners, accounts, stages, amounts, and more. The default settings show you the most commonly used information from each object, but you can customize the report to view other information, such as primary campaign source, forecast category, and synced quote.

> **Note:** Click **Show Hierarchy** to see your org's role hierarchy above the report results. You can use the role hierarchy to share report data with people at different levels in the hierarchy. For example, if you see **CEO > VP of Global Sales > Sales Operations Director**, you're viewing data for the Sales Operations Director role. Click any role name to see and share the data that's visible to people in that role.

The default settings for this report are:

Format

Tabular

Selected Columns

Object Information Type	Columns
Opportunity Information	- Opportunity Name - Amount - Close Date

Object Information Type	Columns
	• Stage
	• Age
	• Type
	• Probability (%)
	• Lead Source
	• Fiscal Period
	• Next Step
	• Created Date
Opportunity Owner Information	• Opportunity Owner
	• Owner Role
Account Information	• Account Name

SEE ALSO:

Opportunities with Contact Roles Report

View information about the contacts associated with your opportunities, including name, title, and role.

The default settings for this report are:

Format

Tabular

Selected Columns

Object Information Type	Columns
Opportunity Information	• Opportunity Name
Opportunity Owner Information	• Opportunity Owner
Account: General Information	• Account Name
Contact Role: General Information	• Title
	• First Name
	• Last Name
Contact Role: Phone/Fax/Email	• Phone
	• Email

Object Information Type	Columns
Contact Role: Address	• Mailing Street • Mailing City • Mailing State/Province • Mailing ZIP/Postal Code • Mailing Country

SEE ALSO:

Limit Report Results

Opportunity Reports

Opportunities with Contact Roles and Products Report

View information about the contacts and opportunities associated with a selected product. You must select a product to filter results by when you run the report. This report isn't supported in Lightning Experience. In Lightning Experience, select a different report type.

The default settings for this report are as follows.

Format

Tabular

Selected Columns

Object Information Type	Columns
Opportunity: Information	• Opportunity Name
Product Information	• Product Name
Opportunity Owner Information	• Opportunity Owner
Account: General Information	• Account Name
Contact Role: General	• Title • First Name • Last Name
Contact Role: Phone/Fax/Email	• Phone • Email
Contact Role: Address	• Mailing Street • Mailing City • Mailing State/Province • Mailing ZIP/Postal Code

Object Information Type	Columns
	• Mailing Country

Opportunities with Competitors Report

View information about your company's competitors for opportunities, including their strengths and weaknesses.

The default settings for this report are:

Format

    Summary

Selected Columns

Object Information Type	Columns
Opportunity Information	• Opportunity Name • Close Date • Amount
Competitor Information	• Competitor Name • Strengths • Weaknesses
Account: General Information	• Account Name

Opportunities with Historical Trending Report

The Opportunities with Historical Trending report is a custom report type designed to help you analyze historical trends in your sales pipeline.

We've selected some of the most useful Opportunity fields for you in the default report:

Amount

    Estimated total sale amount. For organizations using multiple currencies, the amount is shown in your personal currency by default. Change the `Opportunity Currency` picklist to track the amount in another currency.

Close Date

    Date when you plan to close the opportunity. You can enter a date, or choose a date from the calendar that displays when you put your cursor in the field.

Stage

    Current stage of opportunity based on selections you make from a predefined list, for example, Prospect or Proposal.

Probability
> Percentage of estimated confidence in closing the opportunity.

Forecast Category
> Forecast category name that is displayed in reports, opportunity detail and edit pages, opportunity searches, and opportunity list views. The setting for an opportunity is tied to its `Stage`.

> 📝 **Note:** If you edit this report type, it is no longer automatically updated. If you remove this report type, it will not be regenerated.

Opportunity Field History Report
View information about the change history of key opportunity fields, including old and new values and the dates edits were made.

> 📝 **Note:** You must enable and set up field history tracking and select fields in order to use the Field History Tracking report.

The default settings for this report are:

Format
> Tabular

Selected Columns

Object Information Type	Columns
History Data	• `Edit Date` • `New Value` • `Edited By` • `Field/Event` • `Old Value`
Opportunity Fields	• `Opportunity Name`
Opportunity Owner Information	• `Opportunity Owner`

SEE ALSO:
> Limit Report Results
> Opportunity Reports

Opportunity History Report
View information about the history of your opportunities, including stages and close date.

The default settings for this report are:

Format
> Summary

Selected Columns

Object Information Type	Columns
Opportunity Information	• `Opportunity Name`

Object Information Type	Columns
Opportunity Owner Information	• Owner
Opportunity History Information	• From Stage • Amount • Last Modified • To Stage • Probability (%) • Last Modified By • Close Date

SEE ALSO:

[Limit Report Results](#)

[Opportunity Reports](#)

Opportunities with Opportunity Teams Report

View information about the members of your opportunity teams and their roles, organized by opportunity.

To be able to use this report, enable team selling.

The default settings for this report are:

Format

Tabular

Selected Columns

Object Information Type	Columns
Opportunity information	• Opportunity Name
Team member information	• Team Member Name • Team Role

If your admin has created custom opportunity team fields, you can include them in this report.

SEE ALSO:

[Limit Report Results](#)

[Opportunity Reports](#)

Opportunities with Opportunity Teams and Products Report

View information about your opportunity team members and their products, organized by opportunity. You must specify either a product or an opportunity team member to filter results by when you run the report.

The default settings for this report are:

Format
Tabular

Selected Columns

Object Information Type	Columns
Opportunity Information	• Opportunity Name
Product Information	• Product Name
Team Member Information	• Team Member Name • Team Role

SEE ALSO:

Limit Report Results

Opportunity Reports

Opportunities with Partners Report
View information about the partners you team with on your opportunities, including opportunity name, amount, and partner role.

The default settings for this report are:

Format
Summary

Selected Columns

Object Information Type	Columns
Opportunity Information	• Opportunity Name • Close Date • Amount
Opportunity Owner Information	• Opportunity Owner
Partner Information	• Partner Owner • Partner Role • Partner
Account Information	• Account Owner • Account Name

SEE ALSO:

Limit Report Results

Opportunity Reports

Opportunities with Products Report

View information about the products associated with your opportunities, including product name and opportunity stage.

The default settings for this report are:

Format

Matrix

Summary Fields

`Total Price` (sum)

Selected Columns

Object Information Type	Columns
Opportunity Information	`Opportunity Name` `Amount` `Close Date` `Stage` `Age` `Type` `Probability` `Created Date`
Product Information	`Product Name` `Product Code` `Quantity` `Active Product` `Sales Price` `Product Date` `Product Description` `Total Price` `Product: Month` `List Price`
Opportunity Owner Information	`Opportunity Owner` `Owner Role`
Account: General Information	`Account Name`

SEE ALSO:

Limit Report Results

Opportunity Reports

Opportunity Trends Report

View information about trends shared by the opportunities in your pipeline.

The default settings for this report are:

Format

Matrix

Groupings

The default report shows rows grouped by `Historical Stage` and columns grouped by `As of Date`.

Summary Fields

`Historical Amount` (sum)

Selected Columns

Object Information Type	Columns
Opportunity Information	• Opportunity Name
Opportunity Trend Information	• Historical Amount • As of Date • Historical Stage
Opportunity Owner Information	• Opportunity Owner Alias

SEE ALSO:

Opportunities with Quotes and Quote PDFs Report

View details about the quote PDFs created for each quote associated to an opportunity. The default settings show you the most commonly used information from each object, but you can customize the report to view other information, such as who created or last modified each listed quote PDF.

The default settings for this report are:

Format

Summary

Grouping

The default report shows quotes and quote PDFs grouped by `Opportunity Name`.

Selected Columns

Object Information Type	Columns
Opportunity Information	• Opportunity Name
Quote Information	• Quote Name • Syncing
Quote PDF Information	• Quote PDF: Created Date • Quote PDF: Name

Object Information Type	Columns
	• `Quote PDF: Discount`
	• `Quote PDF: Grand Total`

SEE ALSO:

[Limit Report Results](#)

[Opportunity Reports](#)

Opportunities with Quotes and Quote Line Items Report

View details about the quotes associated with opportunities, and the line items for each quote. The default settings provide the most commonly used information from each object, but you can customize the report to see any opportunity, quote, or quote line item field.

If your organization uses multicurrency or advanced currency management, you have additional options for customizing this report. When you select report columns, you can select the "converted" version of an amount or total column to show its value converted to a different currency. Select the currency you want to convert to under Advanced Settings when you select your report criteria.

The default settings for this report are:

Format

Summary

Summary Fields

`Amount` (sum)

`Quote Discount` (sum)

Groupings

The default report shows you results grouped first by `Opportunity Name` and then by `Quote Name`. Each quote line item is listed beneath its associated quote.

Selected Columns

Object Information Type	Columns
Opportunity Information	• `Opportunity Name` • `Amount`
Quote Information	• `Quote Name` • `Discount` • `Syncing` • `Status`
Quote Line Item Information	• `Quote Line Item: Discount` • `Product: Product Name` • `Line Item Number` • `Sales Price` • `List Price` • `Quote Line Item: Subtotal`

Object Information Type	Columns
	• Quote Line Item: Total Price

SEE ALSO:

Limit Report Results

Opportunity Reports

Product and Asset Reports

Use product and asset reports to view information about the products your users currently have installed. Find out what assets your customers have, list the cases filed for a particular asset, or identify assets that aren't associated with a product.

Special Features of Product and Asset Reports

Consider the following when running product and asset reports:

Standard Reports

Use the Assets without Products report, available from the "Products and Asset Reports" folder to list the assets that are not associated with a product. Depending on how you use assets, these assets can represent your competitor's products.

Report Types

- To create a custom report showing what assets your customers have, click **New Report** from the Reports tab and choose the Accounts with Assets or Contacts with Assets report type from the Accounts & Contacts report type category.

- To view a list of the cases filed for a particular asset, click **New Report** from the Reports tab and choose the Assets with Cases report type from the Price Books, Products and Assets option.

SEE ALSO:

Limit Report Results

Self-Service Reports

Self-Service reports help you analyze the effectiveness of your Self-Service portal. Find out how many cases are being viewed, how many customers are logging in, or what customers think of the solutions you're offering.

📝 **Note:** Starting with Spring '12, the Self-Service portal isn't available for new Salesforce orgs. Existing orgs continue to have access to the Self-Service portal.

Standard Reports

- The Self-Service Usage Report gives you information on how many cases are viewed and logged, the number of comments that have been added, and the number of searches Self-Service users have performed.

- The Self-Service User Report provides information about the customers who are logging into your Self-Service portal, including the associated account and last login date. You can also include Self-Service fields in any custom contact report.

- The Helpful Solutions report displays statistics from Self-Service portals that display solutions. On each Self-Service solution page, customers can indicate whether the solution is helpful. You can use the results of this survey when choosing the top five solutions for your Self-Service Home page and to verify that customers are able to find the solutions they need.

SEE ALSO:

[Limit Report Results](#)

Reporting on Support Activity

Use support reports to track the number of cases created, case comments, case emails, case owners, case contact roles, cases with solutions, the length of time since the case last changed status or owner, and the history of cases.

You can also report on the solutions for your organization, including solution history, the languages in which solutions have been written, and whether translated solutions are out of date. If you have enabled the Self-Service portal, you can run reports to track usage of your Self-Service portal.

1. [Fields Available for Case Reports](#)

   You can report on a number of key case fields in addition to the ones that are included in the standard and custom report types.

2. [Fields Available for Solution Reports](#)

   You can report on a number of key solution fields in addition to the standard and custom fields.

SEE ALSO:

[Limit Report Results](#)

Fields Available for Case Reports

You can report on a number of key case fields in addition to the ones that are included in the standard and custom report types.

Cases	
Field	**Description**
`Last Activity`	This field for the Cases report type is populated from the `Last Activity` field in the Contacts object. The Cases report type doesn't include a `Last Activity` field from the Cases object.
`# Cases Submitted`	The number of cases submitted by Self-Service users via the Self-Service portal. (Self-Service Usage Report)
`# Cases Viewed`	The number of cases viewed by Self-Service users via the Self-Service portal. (Self-Service Usage Report)
`# Comments Added`	The number of comments added to all cases by Self-Service users via the Self-Service portal. (Self-Service Usage Report)
`Entitlement Process Start Time`	The time the case entered an entitlement process. This field displays if an entitlement process applies to the case.

Cases

Field	Description
`Entitlement Process End Time`	The time the case exited an entitlement process. This field displays if an entitlement process applies to the case.
`Super User`	A contact enabled to view case information, add comments, and upload attachments for all cases anyone in the company submitted via the Self-Service portal. (Self-Service User Report)
Self-Service Commented	A checkbox showing a comment was added to a case via the Self-Service portal.
New Self-Service Comment	A checkbox showing someone added a comment to a case via the Self-Service that the case owner didn't review yet.
`Is Incoming`	A checkbox showing a case was received by email via the Email-to-Case or On-Demand Email-to-Case feature. (Cases with Emails Report)
`Age`	The age of an open case is the elapsed time from creation to the present. The age of a closed case is the elapsed time from creation to the closing time of the case. Age can be expressed in days, hours, or minutes. Note that the age of a case does *not* take into account any holidays that are associated with the case's business hours. Holidays suspend business hours during specified dates and times.
`Closed`	A checkbox showing a case has a closed status.
`Escalated`	A checkbox showing a case was escalated by an escalation rule.
`Public Case Commented`	A checkbox showing a case has comments that may be displayed via the Self-Service portal.
`Has Attachment`	A checkbox showing a case with emails has attachments. (Cases with Emails Report)
`Old Value`	The value in a tracked case or solution field before it was changed. (Case History Report and Solution History Report)
`New Value`	The value in a tracked case or solution field after it was changed. (Case History Report and Solution History Report)
`Business Hours`	The number of business hours that have elapsed since a case was last updated. Note that holidays are *not* taken into account for this field. Holidays suspend business hours during specified dates and times. (Case History Report)

Cases

Field	Description
Business Hours Since Similar Change	The number of hours that elapsed since the same field on a case was last updated. Note that holidays are *not* taken into account for this field. Holidays suspend business hours during specified dates and times. (Case History Report)
History ID	The unique identifier for each change tracked on a specified case or solution field. (Case History Report and Solution History Report)
Contact Account Name	The account associated with the contact on the case. View together with the `Account Name` field to see if the account on the case is different from the account on the contact.
Parent Case ID	The ID of a parent case, which can be used to access a parent case via the API.

Fields Available for Solution Reports

You can report on a number of key solution fields in addition to the standard and custom fields.

Field	Description
# Solution Searches	The number of solution searches performed by Self-Service users via the Self-Service portal. (Self-Service Usage Report)
Self-Service Access Count	The number of times a solution was viewed in the Self-Service portal.
Self-Service Answer Count	The number of times the survey question "Does this Solution help you answer your question?" is answered, either positively or negatively, on a solution in the Self-Service portal.
Self-Service Positive Count	The number of times the survey question "Does this Solution help you answer your question?" is answered positively on a solution in the Self-Service portal.
Author	The name of the user who originally created the solution.
Num Related Cases	The number of cases associated with the solution.
Reviewed	Checkbox that indicates whether the solution has a reviewed status.
Old Value	The value in a tracked case or solution field before it was changed. (Case History Report and Solution History Report)
New Value	The value in a tracked case or solution field after it was changed. (Case History Report and Solution History Report)
History ID	The unique identifier for each change tracked on a specified case or solution field. (Case History Report and Solution History Report)

Field	Description
Language	The language in which the master solution is written.
Translation Language	The language in which the translated solution is written. (Translated Solutions report)
Solution ID	The unique identifier for each solution. (Translated Solutions report)
Out of Date	Checkbox that indicates that the translated solution may need translating to match its master solution. (Translated Solutions report)
Master Solution Title	The title of the master solution. Displays up to 250 characters.
Translated Solution Title	The title of the translated solution. Displays up to 250 characters.
Master Solution Details	The solution details of the master solution. Displays up to 1000 characters.
Translated Solution Details	The solution details of the translated solution. Displays up to 1000 characters.

Pre-Designed Custom Report Types

Some Salesforce features come with custom report types that are designed for you in advance, so you don't have to create a new report.

📝 **Note:** Pre-designed custom report types are not the same as standard report types.

1. Territory Reports

   Use territory reports to analyze your sales territories. Identify which users have been assigned to which territories, which users have been assigned more than one territory, or which users have no territories.

2. Salesforce CRM Call Center Reports

   Call Center reports help you analyze the Salesforce CRM Call Center phone calls that were handled by you and your team.

3. Create a Custom Report Type for Approval History

   Before you can run reports on executed and in-progress approval processes and their steps, you need to create a custom report type for approval process instances.

4. Create a Custom Report Type in Collaborative Forecasts

   To make a forecasting report available to users, administrators must create a custom report type. A *report type* defines the set of records and fields available to a report based on the relationships between a primary object and its related objects. Reports display only records that meet the criteria defined in the report type.

5. Custom Report Types in Collaborative Forecasts

   A *report type* defines the set of records and fields available to a report based on the relationships between a primary object and its related objects. Reports display only records that meet the criteria defined in the report type. When you create a custom report type, choose the primary and related objects carefully, because they determine the forecast types you can report on.

6. Idea Reports

   Create custom report types so users can create reports about ideas, idea comments, and votes.

7. Report on Salesforce Knowledge Articles

   Use Salesforce Knowledge custom reports to track how articles are created, maintained, and delivered.

8. Chat Session Reports

   Use Chat Session reports to consolidate data about agents' activities while they chat with customers–for example, how long agents are online or how many chat requests are assigned to them.

9. Report on Partners

   Some opportunities involve partner relationships. Use this report to identify and analyze those relationships.

10. Report on Relationship Groups

    You can report on relationship groups and relationship group members if your administrator has enabled custom report types for those custom objects.

11. Enable the Account Owner Report

    The Account Owner report lists all accounts and who owns them.

12. Report on File Search Activity

Use a report to determine the top 300 search term queries for file searches and to identify which content is missing or needs to be renamed based on users' search terms and the click-rank.

13. Report on Chatter Top 100 Feed Item Views with Interaction Count Report

Get a report on the top 100 feed item views in your organization or community with the primary object Interaction Count Report. Knowing the posts that people are looking at most is a great way to get insight into current Chatter trends. Actions counted as views include liking, commenting, and viewing a feed item detail. View counts are not unique. For example, a user can add three comments to a post, and that is counted as three views. Once you set up the report, it's run weekly.

14. Related Articles and Questions Reports

Create custom report types so users can report on click-through rates on related articles and questions. Salesforce Einstein suggests related articles and questions in communities. Use the click-through rates reports to create content around the most accessed articles and questions, empowering your customers to find the information they need.

15. Community Case Deflection from Discussions and Articles

Get insight into how well the Contact Support Form and Case Deflection components actually deflect cases from being created. Using the Case Deflection Dashboard, get real-time metrics on potential and confirmed case deflections, the most helpful articles and discussions, and the least helpful articles and discussions.

16. Report on External Documents Attached to Cases

Create a report to see how many and which specific external documents are being attached to cases. Use this information to see which external sources have the most impact on closing cases. This report is helpful when you have set up Salesforce Federated Search, which gives users access to external search results when using the Knowledge One Widget. Users can attach external documents to cases only if Chatter is enabled.

Territory Reports

Use territory reports to analyze your sales territories. Identify which users have been assigned to which territories, which users have been assigned more than one territory, or which users have no territories.

> 📝 **Note:** The original territory management feature is scheduled for retirement for all customers as of Summer '21. After the feature is retired, users can't access the original territory management feature and its underlying data. We encourage you to migrate to Enterprise Territory Management. For more information, see The Original Territory Management Module Will Be Retired in the Summer '21 Release. The information in this topic applies to the original Territory Management feature only, and not to Enterprise Territory Management.

EDITIONS

Available in: Salesforce Classic

Territory management is available in: **Performance** and **Developer** Editions and in **Enterprise** and **Unlimited** Editions with the Sales Cloud.

Special Features of Territory Reports

Consider the following when running territory reports:

Standard Reports

- The territory report lists all territories in your organization. Select **No Users** in the Users drop-down list and click **Run Report** to see the territories in your organization that do not have any assigned users.

- The User Territory report, User Multiple Territory report, and User Missing Territory report summarize the users who have been assigned to any territories, more than one territory, or no territories, respectively.

- The Account Territory report, Account Multiple Territory report, and Account Missing Territory report summarize the accounts that have been assigned to any territories, more than one territory, or no territories, respectively.

- The Opportunity Territory report summarizes the opportunities that are in territories. To see opportunities owned by users who are not currently active in the opportunity's territory, customize the Opportunity Territory report with the following advanced

filter: `Active in Territory` equals "False." The Opportunity Missing Territory report summarizes the opportunities that are associated with accounts that do not have a territory.

- Reports run from custom report types that include territories may display results differently than standard reports that include territories. This is because reports run from custom report types only display results *with* territories, such as accounts with territories, whereas standard reports that include territories may display results *without* territories. For example, if you select the Account Territory Report, results display accounts without territories. In custom report types, when using the `Territories` filter that includes territories, Multiple Territories or Missing Territories are not shown in the report results.

Tips for Territory Reports

- Standard and custom territory fields are available in territory reports. They are also available in account reports, activity reports that include accounts, opportunity reports, and user reports.

SEE ALSO:

Salesforce CRM Call Center Reports

Call Center reports help you analyze the Salesforce CRM Call Center phone calls that were handled by you and your team.

Special Features of Call Center Reports

Consider the following when running call center reports:

Standard Reports

- The My Team's Calls This Week gives you information about the calls that were handled by the call center users on your team during the past week, including associated records and the result of each call.

- The My Calls Today report gives you information about the calls that you initiated or received during the past day, including associated records and the result of each call.

- The My Calls This Week report gives you information about the calls that you initiated or received during the past week, including associated records and the result of each call.

EDITIONS

Available in: Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Essentials**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

USER PERMISSIONS

To run reports:
- Run Reports

  AND

  Read on the records included in report

Create a Custom Report Type for Approval History

Before you can run reports on executed and in-progress approval processes and their steps, you need to create a custom report type for approval process instances.

1. From Setup, enter `Report Types` in the Quick Find box, then select **Report Types**.
2. Click **New Custom Report Type.**
3. Fill out the fields.

For this field...	Do this...
Primary Object	Select Process Instance.
	A process instance represents one instance of an approval process. A new process instance is created each time a record is submitted for approval.
Report Type Label	Enter a label. Users see this label when they create reports.
	Example: *Approval Process Instances*
Report Type Name	Enter a unique name for the report type.
Description	Enter a description. Users see this label when they create reports.
Store in Category	Select Administrative Reports.
	If you like, you may select a different category. This determines in which folder your users find the custom report type when they create approval history reports.
Deployment Status	When you're ready to let all users access the report type, select Deployed.

4. Click **Next**.
5. Click the box under the primary object.
6. Select Process Instance Node.

    A process instance node represents an instance of an approval step. A new process instance node is created each time a record enters a step in an approval process. No process instance node is created when the record doesn't meet the step criteria or if the approval process instance is otherwise completed without entering the step.

7. For the A to B relationship, select one of these options.

Option	Description
Each "A" record must have at least one related "B" record.	The report includes only process instances that enter at least one approval step to create a process instance node.
	The report excludes process instances for records that were submitted for approval but that didn't meet any step criteria.
"A" records may or may not have related "B" records.	The report includes all process instances.

8. Click **Save**.

After the report type is deployed, notify the relevant users with the names of the category folder and the custom report type, so that they can start creating and running approval history reports.

SEE ALSO:

    Approval History Reports

    Create a Custom Report Type

    Build a Report in Salesforce Classic

Create a Custom Report Type in Collaborative Forecasts

To make a forecasting report available to users, administrators must create a custom report type. A *report type* defines the set of records and fields available to a report based on the relationships between a primary object and its related objects. Reports display only records that meet the criteria defined in the report type.

1. Start creating a custom report type from Setup by entering `Report Types` in the `Quick Find` box, then selecting **Report Types** and **New Custom Report Type**.

2. For Primary Object, select Forecasting Items or Forecasting Quotas.

3. For Store in Category, select **Forecasts**.

4. Let your reps know the locations and names of the report types.

EDITIONS

Available in: both Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Professional** (no custom field or opportunity splits forecasts), **Performance**, and **Developer** Editions and in **Enterprise** and **Unlimited** Editions with the Sales Cloud

USER PERMISSIONS

To create or update custom report types:
- Manage Custom Report Types

To delete custom report types:
- Modify All Data

Custom Report Types in Collaborative Forecasts

A *report type* defines the set of records and fields available to a report based on the relationships between a primary object and its related objects. Reports display only records that meet the criteria defined in the report type. When you create a custom report type, choose the primary and related objects carefully, because they determine the forecast types you can report on.

This table lists the forecasting custom report types that you can create.

Primary Object	Use to create a report for...
Forecasting Items	Viewing information about forecasting amounts, including adjustment amount information. As a best practice, if you use a forecast type based on revenue, use these default fields in the report type.

- Owner Only Amount—The sum of a person's revenue opportunities, without adjustments. For example, if you own two opportunities, each worth $10,000, the Owner Only Amount is $20,000.

- Amount Without Adjustments—The sum of a person's owned revenue opportunities and the person's subordinates' opportunities, without adjustments. Subordinates include everyone reporting up to a person in the forecast hierarchy. This amount is visible only on reports. For example, if the sum of the amount of all opportunities owned by you is $20,000, and the sum of the amount of your subordinates' opportunities is $55,000, the Amount Without Adjustments is $75,000.

- Amount Without Manager Adjustments—The forecast number as seen by the forecast owner. This amount is the sum of the owner's revenue opportunities and the owner's subordinates' opportunities, including adjustments made by the forecast owner on the owner's or subordinates' forecasts. It doesn't include adjustments made by forecast managers above the owner in the forecast hierarchy. For example, Anne has an Amount Without Adjustments of $75,000, made up of $20,000 of her own opportunities and $55,000 of opportunities owned by Ben, her subordinate. She adjusts Ben's amount to $65,000 for a total of $85,000. If you adjust Anne's number from $85,000 to $100,000, you see $85,000 in Amount Without Manager Adjustments, because Anne sees this amount (and Anne can't see your adjustments because you're her manager). To see the amount that includes your adjustment to $100,000, look at Forecast Amount.

- Forecast Amount—The revenue forecast from the forecast manager's perspective and the sum of the owner's and subordinates' opportunities, including all forecast adjustments. For example, you're a forecast manager and have another forecast manager reporting to you who has an Amount Without Manager Adjustment totaling $85,000. If you adjust the forecast to $100,000, the Forecast Amount is $100,000.

If you use a forecast type based on quantity, use these default fields in the report type.

- Owner Only Quantity, Quantity Without Adjustments, Quantity Without Manager Adjustments, and Forecast Quantity

Primary Object	Use to create a report for...
	Regardless of whether you forecast based on revenue or quantity, add these fields.
	• Has Adjustment—A checkbox that indicates whether a manager adjustment has been made on a forecast owner's amount.
	• Has Owner Adjustment—A checkbox that indicates whether a forecast user has adjusted the user's own forecast amount.
	If you use cumulative forecast rollups, add this field to your report.
	• ForecastingItemCategory—This field indicates which rollup each forecast amount is for: Open Pipeline, Best Case Forecast, Commit Forecast, Closed Only, Pipeline, Best Case, Commit, or Closed. If you changed the forecast category names, those changes do not appear in the ForecastingItemCategory values.
`Forecasting Items` with `Opportunities` as a related object	Viewing opportunity revenue or opportunity quantity forecasts. View opportunity information for specific forecasting line items. For example, you can create a summary report for each of your subordinates that includes the opportunity names and last activity dates for their forecasting items, with adjustment information and final forecast amounts.
	📝 Note: For opportunities with no opportunity products specified, this report type includes two forecasting items: one for the Opportunity-Revenue forecast type and one for the Product Family forecast type. These product family forecasting items roll up into the Products Not Specified row of the Product Family forecast.
`Forecasting Items` with `Opportunity Splits` as a related object	Viewing opportunity splits or custom field forecasts. View opportunity split or custom field information for specific forecasting line items. For example, you can create a summary report for each of your subordinates that includes the opportunity split amounts and percentages for their forecasting items, with adjustment information and final forecast amounts.
`Forecasting Items` with `Opportunity Product` as a related object	Viewing product family revenue or product family quantity forecasts. View product family information for specific forecasting line items. For example, you can create a summary report for each of your subordinates that includes the product families and total price for their forecasting items, with adjustment information and final forecast amounts.
	📝 Note: This report type shows forecasting items for the Product Family Revenue, Product Family Revenue by Territory, and Product Family Quantity forecast types. It includes opportunities with and without opportunity products specified.
`Forecasting Quotas`	Viewing data about individual or team quotas. As a best practice, include all the default fields in the report type. For example, you can include lookup fields, such as the full name of the owner. When running the report, you can filter by your name to see quotas that you created and their related accounts and owners.
`Forecasting Quotas` with `Forecasting Items` as a related object	Viewing quota attainment. For example, you can use `Forecasting Quotas` and `Forecasting Items` to create the custom report type. Then, when you create the report, include a team's quotas and forecasted revenue for closed forecasts and create a formula field to display the attained quota percentage.

📝 Note: If you delete a forecast type, reports that use that forecast type don't run.

Idea Reports

Create custom report types so users can create reports about ideas, idea comments, and votes.

As an administrator, you can create custom report types so users can analyze what happens to ideas. Custom report types are the only way to make idea reports available for your users—Salesforce does not provide sample idea reports or a standard report folder for ideas.

To create a folder of idea reports for your users:

1. Create a custom report type for ideas.

   If you create a custom report type that uses Ideas as the primary object and Votes as the secondary object, child (merged) ideas will not appear in the report unless you select `"A"` `records may or may not have related "B" records`. Child ideas have no votes because their votes are transferred to the master idea. This means child ideas do not appear in a report if the Votes object is required.

2. Create a new public folder for idea reports. This step requires the "Manage Public Reports" permission.

3. Using your custom report type, create one or more new custom reports for ideas. Assign the reports to the new idea reports folder you created.

After completing these steps, a folder of idea reports will be available to your users on the Reports home page.

Report on Salesforce Knowledge Articles

Use Salesforce Knowledge custom reports to track how articles are created, maintained, and delivered.

As an administrator, you can create custom report types so agents can create reports on Salesforce Knowledge articles. Custom report types are the only way to make reports about articles available for your readers. Salesforce does not provide sample article reports or a standard report folder for articles.

> 💡 **Tip:** The *Knowledge Base Dashboards and Reports* AppExchange package provides over two dozen reports that help you monitor the knowledge base and analyze usage metrics.

1. Create a Folder for Article Reports

   Create a public folder where you can store article reports for your users.

2. Create a Report Type for Article Reports

   Create a custom report type to report on Salesforce Knowledge article data.

3. Create an Article Report

   Run your custom report on your Salesforce Knowledge articles and save them to your article reports folder.

4. Fields Available on Salesforce Knowledge Reports

   The fields you can use in a knowledge report depend on the type of information you are reporting on.

Create a Folder for Article Reports

Create a public folder where you can store article reports for your users.

1. In the Report Folder section of the Reports tab, click **Create New Folder**.

2. Enter `Article Reports` in the `Folder Label` field.

3. Optionally, modify the `Group Unique Name`.

4. Choose a `Public Folder Access` option.

   Select read/write if you want users to be able to add and remove reports.

5. Choose a folder visibility option.

6. Click **Save**.

Reports you store in this folder are available on the Reports tab.

Create a Report Type for Article Reports

Create a custom report type to report on Salesforce Knowledge article data.

1. From Setup, enter `Report Types` in the Quick Find box, then select **Report Types**.

2. Click **New Custom Report Type**.

   See Create a Custom Report Type

3. In the Primary Object dropdown menu, select the article-related object you want to report on.

Primary Object	Related Objects	Description
Knowledge (Lightning Knowledge)	• Knowledge Versions • Feed: Knowledge	Compare information, such as creation dates, audience visibility, and the number of associated cases across record types. To access history and custom fields, add Knowledge Versions as a secondary object. 💡 Tip: If you changed the name of your knowledge base, the custom label shows here.

Knowledge Articles	• Knowledge Article Versions
	• Article View Statistics
	• Article Vote Statistics
	• Case Article
	  – Knowledge. You can then choose a relationship to Knowledge_DataCategorySelection
	  – Articles
	  – There are many additional relationships from Case Article, including Work Orders and Case Comments.
		Report on information about individual published articles, such as their creation date and published channels. When you build this custom report type, you can include article view statistics, article vote statistics, and case associations. In reports using the Knowledge Articles primary object, each article has five records (rows), one for each channel (All Channels, Internal App, Customer, Partner, and Public Knowledge Base).

Knowledge Article Versions	• Article View Statistics
	• Article Vote Statistics
	• Case Article
	  – Knowledge. Create a fourth-level relationship to Knowledge_DataCategorySelection
	  – Articles
		Compare information about individual versions and translations, such as their creation date, published channels, and number of associated cases. If you choose this custom report type, you can also include article view and vote statistics.

Primary Object	Related Objects	Description
	– There are many additional relationships from Case Article, including Work Orders and Case Comments.	
Article types (Knowledge in Salesforce Classic)	• *Article Type* Versions You can then choose a relationship to *Article Type*_DataCategorySelection.	Compare information, such as creation dates, published channels, and number of associated cases, for your custom article type, such as an FAQ. To access version history and custom fields, add the article type's version as a secondary object. If you have multiple article types, each is listed separately, for example FAQs, Issues, and Procedures.

Report types for the search, view, vote, and version history objects don't have secondary relationships.

Table 1: Report Types for Search, Views, Votes, and Version History

Primary Object	Description
Knowledge Search Activity	• Analyze the number of searches per day, month, or year for each channel and language. • For each search, see the date, ID, and title of the article that was clicked. • See which keywords users are looking for in your knowledge base. • For each keyword, see the average number of results, and articles that appear in the search results. • For each article, see the average number of clicks, and unique users who clicked it.
Knowledge Keyword Search	See which keywords users are looking for in your knowledge base. Keyword data is only available for Salesforce Classic.
Article Version History	Compare information about individual article versions, such as their creation dates, published channels, and number of associated cases.
Knowledge Article Views	Analyze the number of views per day, month, or year for each channel and role.
Knowledge Article Votes	Analyze the number of votes per day, month, or year for each channel and role.
Knowledge Article Searches	Analyze the number of searches per day, month, or year for each channel and role. Searches aren't differentiated between internal and external, and all searches are logged as Internal App. Searches in Experience Cloud sites aren't checked.

4. Complete the required fields and click **Save**.

   In the Store in Category dropdown menu, we recommend choosing **Customer Support Reports** or **Other Reports**. This category is where users find the custom report type on the Reports tab.

5. Make your choices on the Define Report Records Set page.

6. Click **Save**.

7. As needed, remove and rearrange fields from your report layout.

   To learn which fields are available on each primary object, see Fields Available on Salesforce Knowledge Reports on page 163.

   > **Note:** An article's score is calculated slightly differently in the API than it is in a custom report. We recommend standardizing on one or the other and not attempting to use both.

Create an Article Report

Run your custom report on your Salesforce Knowledge articles and save them to your article reports folder.

Using your custom report types, create article reports and save them to your article reports folder.

1. On the Reports tab, click **New Report**.

2. Choose the category where your custom report types are stored, for example, **Customer Support Reports** or **Other Reports**.

   The Cases with Articles report is available by default in the Customer Support Reports folder.

3. Find your report type and click **Create**.

4. When viewing your report, click **Save As** and save it in the new Article Reports folder to make the report available to other users.

   > **Note:**
   > - In reports using the Knowledge Article custom report type, there are at least five rows per article (one for each channel, including "All Channels").
   >
   > - Daily values are reported independently for the last 90 days and monthly values for the last 18 months. After those time periods, use the monthly aggregate and yearly aggregate values, respectively.
   >
   > - In reports using the Knowledge Article Votes, Knowledge Article Views, or Knowledge Article Searches custom report types, each row represents a day, channel, and role combination. For example, if a user with the Kingmaker role views articles in the internal app and the next day the same reader views more articles in the internal app, the Article Views report has two rows: one for each unique date.
   >
   > - In reports using Knowledge Article Version (KAV), except Knowledge Article Version History, you can filter by data category. You can add up to four filters and set their logic to AT, ABOVE, BELOW, or ABOVE OR BELOW. The logic between filters is OR. You can use the same category group multiple times, however, you must use the same operator each time.
   >
   > - To report on Approval Processes for Knowledge Articles, use **Process Instance** and **Process Instance Node** when creating a custom report type. Then filter the report on object type, which is the article type.

Fields Available on Salesforce Knowledge Reports

The fields you can use in a knowledge report depend on the type of information you are reporting on.

The following tables list the available fields by primary object for Salesforce Knowledge reports.

Fields Available on Article types (Knowledge in Salesforce Classic) Reports

The Article Type primary object allows you to add a secondary object of the article type version.

Table 2: Article Type Report Primary Object Fields

Field	Description
Archived By	User who archived the article.
Archived Date	Date the article was archived.
Article Number	Unique number automatically assigned to the article.
Article Type ID	The ID associated with the article type.
Case Association Count	Number of cases attached to the article.
Created By	User who created the article.
Created Date	Date the current article version was created. If the article has been published more than once, this is the latest draft date. To create reports that use the original article creation date, first create a Custom Report Type that joins the Knowledge Article (__ka) and Knowledge Article Version (__kav) objects for a given article type to allow for use of the created date on the record in the Knowledge Article object, rather than the one for the Knowledge Article Version record.
Custom fields	Any custom fields created on the article types. Add the article type's version as a secondary object to access any custom fields for the article type.
First Published Date	Date the article was originally published.
Knowledge Article Version	The article's version number.
Last Modified By	User who changed the article most recently.
Last Modified Date	Date the article was last changed. The last modified date of a draft article is the time the draft was saved. The last modified date of a published article is time the article was most recently published.
Last Published Date	Date the article was last published.
Master Language	The original language of the article.

> **Note:** Currently you can't use `Knowledge Article Version` and `Last Modified Date` in the same report.

Table 3: Article Type Report Secondary Object Fields

Field	Description
Archived By	User who archived the article.
Article Type	The article type associated with the article.
Created By	User who created the article.
Created Date	Date the current article version was created. If the article has been published more than once, this is the latest draft date. To create reports that use the original article creation date, first create a Custom Report Type that joins the Knowledge Article (__ka) and Knowledge Article Version (__kav) objects for a given article type to allow for use of the created date on the record in the Knowledge Article object, rather than the one for the Knowledge Article Version record.
Custom fields	Any custom fields created on the article types. Add the article type's version as a secondary object to access any custom fields for the article type. **Note:** File fields aren't supported for reports and report types with Knowledge.
Is Latest Version	Indicates if the article is the most recent version.
Is Master Language	Indicates that the article is not a translation, but the original article.
Knowledge Article Version ID	Unique ID automatically assigned to the article translation.
Language	The article's language.
Last Modified By	User who changed the article most recently.
Last Modified Date	Date the article was last changed. The last modified date of a draft article is the time the draft was saved. The last modified date of a published article is time the article was most recently published.
Out of Date	Indicates that the master article has been updated since this translation was published.
Publication Status	Indicates whether the article or translation is in progress (draft), published, or archived.
Summary	Description of the article provided by the author.
Title	The article's title.
Translation Completed Date	Date the translation was completed.
Translation Exported Date	Date the article was exported for translation.
Translation Imported Date	Date the translation was imported.
URL Name	Text used as hyperlink for the article.
Validation Status	Indicates if the article is valid or not.

Field	Description
Version Number	The version number of the article.
Visible in Customer	Indicates that the article is published in the Customer Portal.
Visible in Internal App	Indicates that the article is published in the internal app (Articles tab).
Visible in Partner	Indicates that the article is published in the partner portal.
Visible in Public Knowledge Base	Indicates that the article is published in the public knowledge base.

Fields Available on Knowledge Articles Reports

Field	Description
Article Number	Unique number automatically assigned to the article.
Article Type	The article type associated with the article.
Case Association Count	Number of cases attached to the article.
Created By	User who created the article.
Created Date	Date the current article version was created. If the article has been published more than once, this is the latest draft date. To create reports that use the original article creation date, first create a Custom Report Type that joins the Knowledge Article (__ka) and Knowledge Article Version (__kav) objects for a given article type to allow for use of the created date on the record in the Knowledge Article object, rather than the one for the Knowledge Article Version record.
First Published Date	Date the article was originally published.
Is Latest Version	Indicates if the article is the most recent version.
Knowledge Article Version ID	Unique ID automatically assigned to the article translation.
Last Modified By	User who changed the article most recently.
Last Modified Date	Date the article was last changed. The last modified date of a draft article is the time the draft was saved. The last modified date of a published article is time the article was most recently published.
Last Published Date	Date the article was last published.
Published Version Owner	The user or queue that owns the published version of an article.
Summary	Description of the article provided by the author.
Title	The article's title.
URL Name	Text used as hyperlink for the article.
Validation Status	Indicates if the article is valid or not.
Version Number	The version number of the article.

Field	Description
Visible in Customer	Indicates that the article is published in the Customer Portal.
Visible in Internal App	Indicates that the article is published in the internal app (Articles tab).
Visible in Partner	Indicates that the article is published in the partner portal.
Visible in Public Knowledge Base	Indicates that the article is published in the public knowledge base.

Fields Available on Knowledge Article Searches Reports

Field	Description
Channel	The channel that's applicable to the article. Possible values are `All Channels`, `Internal App`, `Customer`, `Partner`, and `Public Knowledge Base`.
Count	The number of article searches applicable to the duration shown (day, month, or year).
Cumulative Count	The total number of article searches for the history of the record.
Date	Last date on which an article search took place for the record. All rows represent a date, channel, and role combination.
Duration	The time period the search count is applied to. Possible values are `Daily`, `Monthly`, and `Yearly`. For example, a record where the Count is 70 and the Duration is Monthly indicates that 70 searches took place over the past month. Totals are aggregated daily for the current month, monthly from the past full month through the past full year, and yearly beyond that.
Related Role	Name of the role that applies to the record. Each row in the report represents searches per channel per role.

Fields Available on Knowledge Article Versions Reports

Field	Description
Archived By	User who archived the article.
Archived Date	Date the article was archived.
Article Number	Unique number automatically assigned to the article.
Article Type	The article type associated with the article.
Case Association Count	Number of cases attached to the article.
Created By	User who created the article.
Created Date	Date the current article version was created. If the article has been published more than once, this is the latest draft date. To create reports that use the original article creation date, first create a Custom Report Type that joins the Knowledge Article (__ka) and Knowledge Article Version (__kav) objects for a given article type to allow for use of the created date on the record in the Knowledge Article object, rather than the one for the Knowledge Article Version record.

Field	Description
First Published Date	Date the article was originally published.
Is Latest Version	Indicates if the article is the most recent version.
Is Master Language	Indicates that the article is not a translation, but the original article.
Knowledge Article Version ID	Unique ID automatically assigned to the article translation.
Language	The article's language.
Last Modified By	User who changed the article most recently.
Last Modified Date	Date the article was last changed. The last modified date of a draft article is the time the draft was saved. The last modified date of a published article is time the article was most recently published.
Date the article was last published.	Date the article was last published.
Master Language	The original language of the article.
Out of Date	Indicates that the master article has been updated since this translation was published.
Owner	The user or queue that owns a published, draft, or archived version of an article.
Publication Status	Indicates whether the article or translation is in progress (draft), published, or archived.
Summary	Description of the article provided by the author.
Title	The article's title.
Translation Completed Date	Date the translation was completed.
Translation Exported Date	Date the article was exported for translation.
Translation Imported Date	Date the translation was imported.
URL Name	Text used as hyperlink for the article.
Validation Status	Indicates if the article is valid or not.
Version Number	The version number of the article.
Visible in Customer	Indicates that the article is published in the Customer Portal.
Visible in Internal App	Indicates that the article is published in the internal app (Articles tab).
Visible in Partner	Indicates that the article is published in the partner portal.
Visible in Public Knowledge Base	Indicates that the article is published in the public knowledge base.

Fields Available on Knowledge Article Views Reports

You can add up to 6of the 8fields below.

Field	Description
Channel	The channel that's applicable to the article. Possible values are `All Channels`, `Internal App`, `Customer`, `Partner`, and `Public Knowledge Base`.
Count	The number of article views applicable to the duration shown (day, month, year).
Cumulative Count	The total number of article views for the history of the record.
Date	Last date on which an article view took place for the record. All rows represent a date, channel, and role combination.
Duration	The time period the search count is applied to. Possible values are `Daily`, `Monthly`, and `Yearly`. For example, a record where the Count is 70 and the Duration is Monthly indicates that 70 searches took place over the past month. Totals are aggregated daily for the current month, monthly from the past full month through the past full year, and yearly beyond that.
Related Role	Name of the role that applies to the record.
Score	Article's average view rating. Scoresake into account a half-life calculation. Every 15 days, if an article has not been viewed its average rating moves up or down. This calculation ensures that over time, older or outdated articles don't maintain artificially high or low ratings compared to newer, more frequently viewed articles.
Total Views	Number of times a published article has been viewed.

Fields Available on Knowledge Article Votes Reports

Field	Description
Channel	The channel that's applicable to the article. Possible values are `All Channels`, `Internal App`, `Customer`, `Partner`, and `Public Knowledge Base`.
Count	The number of article votes applicable to the duration shown (day, month, year).
Cumulative Count	The total number of article votes for the history of the record.
Date	Last date on which an article vote took place for the record. All rows represent a date, channel, and role combination.
Duration	The time period the search count is applied to. Possible values are `Daily`, `Monthly`, and `Yearly`. For example, a record where the Count is 70 and the Duration is Monthly indicates that 70 searches took place over the past month. Totals are aggregated daily for the current month, monthly from the past full month through the past full year, and yearly beyond that.
Related Role	Name of the role that applies to the record.

Fields Available on Knowledge Keyword Search Reports

Knowledge Keyword Search reports are designed for use with the Salesforce Knowledge in Salesforce Classic data model. These reports include searches from the Knowledge One widget.

Field	Description
Channel	The channel that's applicable to the article. Possible values are `All Channels`, `Internal App`, `Customer`, `Partner`, and `Public Knowledge Base`.
Count	The number of keyword searches applicable to the duration shown (day, month, year).
Date	Last date on which a keyword search took place for the record. All rows represent a date, channel, and role combination.
Duration	The time period the search count is applied to. Possible values are `Daily`, `Monthly`, and `Yearly`. For example, a record where the Count is 70 and the Duration is Monthly indicates that 70 searches took place over the past month. Totals are aggregated daily for the current month, monthly from the past full month through the past full year, and yearly beyond that.
Found	Indicates whether the keyword shown was found during a search of the knowledge base.
Keyword	Search term used to search published articles in the knowledge base.

Fields Available on Knowledge Search Activity Reports

Field	Description
Average Click Rank	The order in which the article appeared in search results when results are sorted by relevance and when readers clicked it from the list of results.
Channel	The channel that's applicable to the article. Possible values are `All Channels`, `Internal App`, `Customer`, `Partner`, and `Public Knowledge Base`.
Clicked Article Title	The title of the clicked article taken when the search results are sorted by relevance by the reader.
Duration	The time period the search count is applied to. Possible values are `Daily`, `Monthly`, and `Yearly`. For example, a record where the Count is 70 and the Duration is Monthly indicates that 70 searches took place over the past month. Totals are aggregated daily for the current month, monthly from the past full month through the past full year, and yearly beyond that. **Note:** Activity totals are collected nightly and aren't in real time.
Language	The language filter that's applied to the reader's search.
Number of Results	The number of search results that were returned for the search term. If Duration is also included, this value is aggregated based on the time period specified.
Number of Searches	The number of searches for the duration that's shown (day, month, or year).
Number of Users	The number of individual users who clicked the article.
Search Date	The date of the search.
Search Term	The first 100 characters of the search term that was used to search published articles in the knowledge base.

Chat Session Reports

Use Chat Session reports to consolidate data about agents' activities while they chat with customers–for example, how long agents are online or how many chat requests are assigned to them.

A *report type* defines the set of records and fields available to a report based on the relationships between a primary object and its related objects. Reports display only records that meet the criteria defined in the report type. You can create a custom report type for Chat sessions to aggregate data about your agents' activity while they support customers. These reports include data for all of the chats that occurred during a specific Chat session.

Customize Chat session reports to include columns of information about any of the following categories, as well as any custom fields related to Chat sessions:

Column Name	Description
Agent: Full Name	Name of the agent associated with the session.
Assistance Flags Lowered (Agent)	Number of times an agent lowered an assistance flag during a Chat Session.
Assistance Flags Lowered (Supervisor)	Number of times a supervisor lowered an assistance flag during an agent's Chat Session.
Assistance Flags Raised	Number of times an agent raised an assistance flag during a Chat Session.
Chat Requests Assigned	Number of chat requests assigned to an agent.
Chat Requests Declined (Manually)	Number of chat requests declined manually by an agent.
Chat Requests Declined (Push Timeout)	Number of chat requests that timed out while assigned to an agent.
Chat Requests Engaged	Number of chats in which an agent was engaged during the session.
Created By: Full Name	Full name of the creator of the session record.
Created Date	Date the session record was created.
Last Modified By: Full Name	Full name of the person who last modified the session record.
Last Modified Date	Date the session record was last modified.
Chat Session ID	ID of the Chat session record.
Chat Session Name	Automatically generated ID of the Chat session.
Login Time	Time and date the agent logged in to the session.
Logout Time	Time and date the agent logged out of the session.
Time Idle	Total amount of time in seconds an agent was not engaged in chats during a session. The following formula indicates how an

	agent's idle time is calculated: (Time Spent Online + Time Spent Away) - Time Spent in Chats = Time Idle.
Time Spent at Capacity	Total amount of time in seconds in which an agent's queue was full.
Time Spent Away	Total amount of time in seconds an agent spent in "Away" status.
Time Spent in Chats	Total amount of time in seconds an agent spent engaged in chats.
Time Spent Online	Total amount of time in seconds an agent spent online.

Report on Partners

Some opportunities involve partner relationships. Use this report to identify and analyze those relationships.

To report on all partnerships or primary partnerships within opportunities:

1. From the Reports tab, choose the Partner Opportunities report.

2. Choose **Customize** to change the report and view only primary partner relationships. Add a field filter where `Primary equals 1`.

In any other opportunity report, when you customize the report to display the Partner column, only the primary partner displays.

You can also run the Partner Accounts report to analyze the partnerships of your accounts.

EDITIONS

Available in: Salesforce Classic

Available in: **Essentials**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

USER PERMISSIONS

To report on partners:
- Read on accounts or opportunities

  AND

  Run Reports

Report on Relationship Groups

You can report on relationship groups and relationship group members if your administrator has enabled custom report types for those custom objects.

Custom report types are the only way to make relationship group reports available for your users—Salesforce does not provide sample relationship group reports or a standard report folder for relationship groups.

Administrators can create a public folder of relationship group reports as follows:

1. Create a custom report type for relationship group objects.

    💡 **Tip:** To create a report type about the members of relationship groups, select `Relationship Groups` as the primary report type object and add `Relationship Group Members` as an object relationship. Alternatively, to create a report type about the accounts that are primary on a relationship group, select `Accounts` as the primary report type object and add `Relationship Groups (Primary Account)` as an object relationship.

2. Create a new public folder for relationship group reports. This step requires the "Manage Public Reports" permission.

3. Using your custom report type, create one or more new custom reports for relationship groups. Assign the reports to the new relationship groups reports folder you created.

After completing these steps, a folder of relationship group reports will be available to users on the Reports home page.

Enable the Account Owner Report

The Account Owner report lists all accounts and who owns them.

Organizations that have their organization-wide sharing access level set to Private for accounts may want to restrict users from running the Account Owner report.

To show or hide this report:

1. From Setup, enter `Account Owner Report` in the `Quick Find` box, then select **Account Owner Report**. This option is available only in organizations that have a private account sharing model.

2. Select the checkbox to allow all users to run this report. If you leave the box unchecked, only administrators and users with the "View All Data" permission can run this report.

3. Click **Save**.

Report on File Search Activity

Use a report to determine the top 300 search term queries for file searches and to identify which content is missing or needs to be renamed based on users' search terms and the click-rank.

User Permissions Needed	
To create or update custom report types:	"Manage Custom Report Types"
To create a public reports folder:	"Manage Public Reports"
To run the File Search Activity report:	"Run Reports" AND "View All Data" OR "Content Administrator". Users with "Manage Library" for the library don't need the "View All Data" or "Content Administrator" permissions to run the report.

Custom report types are the only way to make reports about file search activity available for your users—Salesforce does not provide sample file search activity reports or a standard report folder.

Note: Activity totals are collected nightly and aren't in real time.

Primary Object for File Search Activity Report

When you create a custom report type, select **File Search Activity** for the **Primary Object**.

Available File Search Activity Report Fields

The following table lists the available fields for File Search Activity reports.

Field	Description
Average Click Rank	The order in which the file appeared in search results when users clicked it from the list of results.
Average Number of Results	The number of search results that were returned for the search term. If Duration is also included, this value is aggregated based on the time period specified.
Duration	The time period the search count is applied to. Possible values are Daily, Monthly, and Yearly. For example, a record where the Count is 70 and the Duration is Monthly indicates that 70 searches took place over the past month. Totals are aggregated daily for the current month, monthly from the past full month through the past full year, and yearly beyond that.
Language	The language filter that's applied to the user's search.
Number of Searches	The number of searches for the duration that's shown (day, month, or year).
Number of Users	The number of individual users who clicked the file.

Field	Description
Search Date	The date of the search.
Search Terms	The first 100 characters of the search term that was used to search published files.

SEE ALSO:

Access to Report Folders

Creating and Editing Folders

Create a Custom Report Type

Create a Report

Report on Chatter Top 100 Feed Item Views with Interaction Count Report

Get a report on the top 100 feed item views in your organization or community with the primary object Interaction Count Report. Knowing the posts that people are looking at most is a great way to get insight into current Chatter trends. Actions counted as views include liking, commenting, and viewing a feed item detail. View counts are not unique. For example, a user can add three comments to a post, and that is counted as three views. Once you set up the report, it's run weekly.

User Permissions Needed	
To create or update custom report types:	"Manage Custom Report Types"
To create a public reports folder:	"Manage Public Reports"

Custom report types are the only way to make reports about the 100 top feed item views. Salesforce doesn't provide sample top view activity reports or a standard report folder.
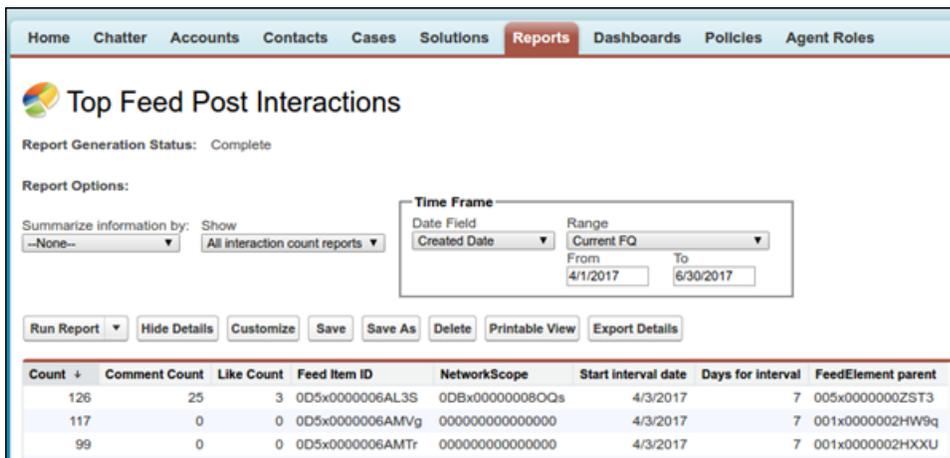
Note: Reports on feed activities don't include information about system-generated posts, such as feed tracked changes.

Primary Object for Interaction Count Report

When you create a custom report type, select **Interaction Count Reports** for the **Primary Object**.

Available Interaction Count Report Fields

The following table lists the available fields for Interaction Count reports.

Field	Description
Comment Count	The number of comments on a post
Count	The number of views on a post
Days for interval	How often the report is run—currently 7, for every seven days (this is a fixed value)
FeedElement parent	The ID of the group, profile, record, or other entity that is the parent to the feed where the content is posted
Feed Item ID	The ID of the post
Like Count	The number of likes on the post
NetworkScope	The scope of the network where this feed item is available.
	This field always has a value. For the default network, it's 00000000000000. If it's posted in a community, it has the ID of the community (also known as the NetworkId). If the value contains an "I," the post is parented by a record, like an account, an opportunity, or some other record type.
Start interval Date	The date that marks the start of the seven-day run interval

SEE ALSO:

Characteristics of Chatter Counts

Related Articles and Questions Reports

Create custom report types so users can report on click-through rates on related articles and questions. Salesforce Einstein suggests related articles and questions in communities. Use the click-through rates reports to create content around the most accessed articles and questions, empowering your customers to find the information they need.

As an admin, you can create custom report types so users can analyze which related articles and questions are most clicked in a community.

To create a folder of reports for your users:

1. Create a custom report type using Related Content Metrics as the primary object.

2. Create a public folder for related content metric reports. This step requires the Manage Public Reports permission.

3. Using your custom report type, create one or more new custom reports. For example, you can make specific reports on related questions click-through rates and related articles click-through rates. Assign the reports to the new reports folder you created.

After completing these steps, a folder of related content metrics reports is available to your users on the Reports home page.

Community Case Deflection from Discussions and Articles

Get insight into how well the Contact Support Form and Case Deflection components actually deflect cases from being created. Using the Case Deflection Dashboard, get real-time metrics on potential and confirmed case deflections, the most helpful articles and discussions, and the least helpful articles and discussions.

As an admin, you can create custom report types so users can analyze which articles and discussions are most useful in deflecting cases in a community.

To create a folder of reports for your users:

1. Create a custom report type using Community Case Deflection Metrics as the primary object.

2. Create a public folder for the reports. This step requires the Manage Public Reports permission.

3. Using your custom report type, create one or more new custom reports. For example, you can make specific reports on which articles and discussions helped deflect cases, and which ones were unsuccessful at deflecting cases. Assign the reports to the new reports folder you created.

After completing these steps, a folder of community case deflection metrics reports is available to your users on the Reports home page.

Report on External Documents Attached to Cases

Create a report to see how many and which specific external documents are being attached to cases. Use this information to see which external sources have the most impact on closing cases. This report is helpful when you have set up Salesforce Federated Search, which gives users access to external search results when using the Knowledge One Widget. Users can attach external documents to cases only if Chatter is enabled.

1. From Setup, enter `Report Types` in the Quick Find box, then select **Report Types**.

2. Click **New Custom Report Type**.

3. Select **Cases** as the Primary Object for your custom report type.

4. Enter the **Report Type Label**. For example, `Cases with External Documents`. The Report Type Name automatically fills.

5. Enter a description for your custom report type. For example, `Reports on the external documents attached to cases`.

6. To indicate where to store the custom report type, select the **Other Reports** category.

7. Select the deployment option of your choice. Click **Next**.

8. On the next page, **Define Report Records Set**, relate another object and create an A to B Relationship. Select **External Documents** as the child object for your custom report type. Leave **Each "A" record must have at least one related "B" record** selected. Click **Save**.

9. Leaving Setup, click the **Reports** tab.

10. Click **New Report**. From the Other Reports folder, select **Cases with External Documents**, or the name of the report you created.

11. Click **Create**.

12. If they're not already included, from the Cases object in the left pane, drag and drop the **Case ID** and **Case Number** fields onto the report. From the External Object: Object Name object, drag and drop the **Display URL** and **Title** fields. Drag and drop other fields that you want to include in the report.

13. Click **Run Report**. Save the report.

EDITIONS

Available in: Salesforce Classic (not available in all orgs)

Salesforce Knowledge is available in **Essentials** and the **Unlimited** Edition with Service Cloud.

Salesforce Knowledge is available for an additional cost in: **Professional**, **Enterprise**, **Performance**, and **Developer** Editions. For more information, contact your Salesforce representative.

USER PERMISSIONS

To create or update custom report types
- "Manage Custom Report Types"

To delete custom report types
- "Modify All Data"

To run reports
- "Run Reports"

To create, edit, and delete reports
- "Create and Customize Reports"

Turn Automatic Updates to the Report Preview On or Off

Edit reports faster by turning off automatic preview updates. When off, you can make multiple edits without waiting for the preview to refresh after each edit. When you're ready to preview data, manually refresh the report preview. Or, see sample records returned after each edit by keeping automatic previews on.

To turn automatic preview updates on or off flip the **Update Preview Automatically** switch on or off.

1. Edit or create a report.

2. Click **Update Preview Automatically**. The switch switches on or off.

   When on, the report preview updates automatically after each edit you make to the report.

   When off, a message tells you whether the preview shows your most recent edits and provides the option to refresh the preview. When hiding recent edits, the preview turns gray.

3. To refresh out-of-date data, click **Refresh** (1).

The preview reloads to show your edits.
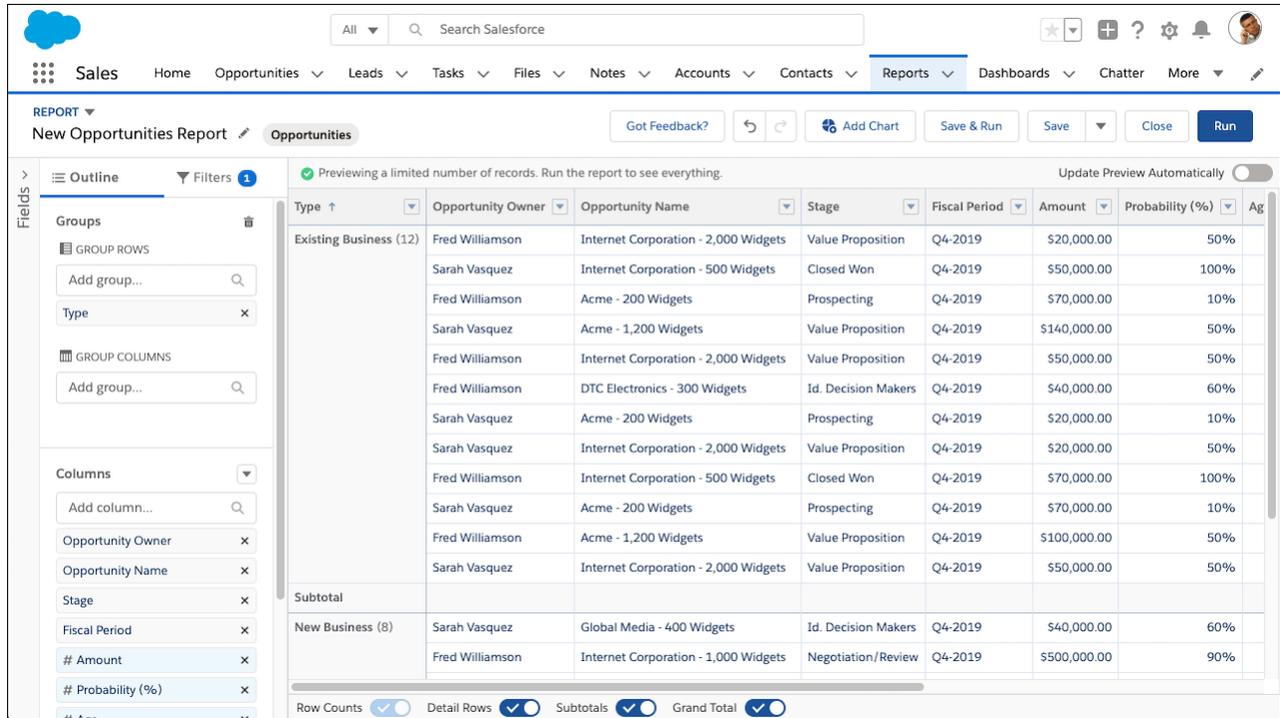
The **Update Preview Automatically** switch does not affect the report run page.

Customize Report Views in the Run Page

Use the power of Lightning Experience when you review and analyze your report records in the run page.

Available in: Lightning Experience
Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
Available in: Enhanced Folder Sharing

Review key report metrics in the header of the report run page in Lightning Experience. The Lightning Experience report header displays up to 8 metrics, in the order that they appear in the report, from left to right. These metrics include summaries such as average amounts, grand totals, subtotals, record counts, and formula column results.

To quickly view or hide a report's count of rows, detail rows, subtotals, or grand total record, use the toggles in the report footer. For reports grouped by rows and columns (matrix), the footer includes the option to switch between stacked and unstacked summaries view. Stacked summaries appear together, whereas unstacked summaries appear in distinct columns.

Get even more done with the tools in the report header. Using the tools, you can:

- trend a report (available with Einstein Analytics license)
- show or hide a chart
- add, remove, or modify report filters
- collaborate with others on the report feed
- refresh the report

Viewing jumbo-sized reports in the run page? Column headers are sticky, so, they don't go out of sight as you page through report records.

You can further customize how the records in the report display. From a column's action menu, you can sort and group records, and even remove the column. You can also sort by various metrics from a grouping column.

Want to take a closer look at records from a report that's grouped by rows (summary) or grouped by rows and columns (matrix)? On the report run page, select the box next to a primary grouping value you want to drill down. Click **Drill Down**. Specify the field you want to group the results by, and click **Apply**.

You can also view the record detail rows behind a report that's grouped by rows and columns (matrix). To do so, click the appropriate record in the summary table on the report run page. You can even click in the Total row.

Let's say you're looking at the Accounts grouped by Industry report in the run page. If you want to review the records for just a specific industry, click the count for that industry in the summary table.

Want a more convenient view of summarized metrics in reports that are grouped by rows and columns (matrix)? For a viewer-friendly version of the report with less scrolling required, enable **Stacked Summaries** in the report footer.

Total Records	Total Amount	Average amount	Amount after Discount
50	USD 7,740,000.00	154,800.00	7,353,000.00

Stage →		Prospecting				Needs Analysis				Value Proposition		
Opportunity Owner ↑ ▼		Sum Amount	Average amount	Amount after Discount	Count	Sum Amount	Average amount	Amount after Discount	Count	Sum Amount	Average amount	Amount after Discou...
Brian Alison		USD 20,000.00	20,000.00	19,000.00	1	USD 70,000.00	70,000.00	66,500.00	1	USD 160,000.00	80,000.00	152,000.0
Fred Williamson		USD 20,000.00	20,000.00	19,000.00	1	USD 0.00	-	-	0	USD 0.00	-	
Hank Chen		USD 40,000.00	20,000.00	38,000.00	2	USD 70,000.00	70,000.00	66,500.00	1	USD 160,000.00	80,000.00	152,000.0
Nadia Smith		USD 0.00	-	-	0	USD 70,000.00	70,000.00	66,500.00	1	USD 160,000.00	80,000.00	152,000.0
Sarah Vasquez		USD 20,000.00	20,000.00	19,000.00	1	USD 140,000.00	70,000.00	133,000.00	2	USD 340,000.00	68,000.00	323,000.0
Total		USD 100,000.00	20,000.00	95,000.00	5	USD 350,000.00	70,000.00	332,500.00	5	USD 820,000.00	74,545.45	779,000.0

Row Counts ☑○   Detail Rows ☑○   Grand Total ☑○   Stacked Summaries ○

Total Records	Total Amount	Average amount	Amount after Discount
50	USD 7,740,000.00	154,800.00	7,353,000.00

Opportunity Owner ↑ ▼	Stage →	Prospecting	Needs Analysis	Value Proposition	Id. Decision Makers	Negotiation/Review	Closed Won	Total
Brian Alison	Sum of Amount Average amount Amount after Discount Record Count	USD 20,000.00 20,000.00 19,000.00 1	USD 70,000.00 70,000.00 66,500.00 1	USD 160,000.00 80,000.00 152,000.00 2	USD 110,000.00 55,000.00 104,500.00 2	USD 100,000.00 100,000.00 95,000.00 1	USD 550,000.00 275,000.00 522,500.00 2	USD 1,010,000.00 112,222.22 959,500.00 9
Fred Williamson	Sum of Amount Average amount Amount after Discount Record Count	USD 20,000.00 20,000.00 19,000.00 1	USD 0.00 - - 0	USD 0.00 - - 0	USD 110,000.00 55,000.00 104,500.00 2	USD 2,000,000.00 2,000,000.00 1,900,000.00 1	USD 0.00 - - 0	USD 2,130,000.00 532,500.00 2,023,500.00 4
Hank Chen	Sum of Amount Average amount Amount after Discount Record Count	USD 40,000.00 20,000.00 38,000.00 2	USD 70,000.00 70,000.00 66,500.00 1	USD 160,000.00 80,000.00 152,000.00 2	USD 290,000.00 58,000.00 275,500.00 5	USD 100,000.00 100,000.00 95,000.00 1	USD 1,050,000.00 350,000.00 997,500.00 3	USD 1,710,000.00 122,142.86 1,624,500.00 14
Nadia Smith	Sum of Amount Average amount Amount after Discount Record Count	USD 0.00 - - 0	USD 70,000.00 70,000.00 66,500.00 1	USD 160,000.00 80,000.00 152,000.00 2	USD 0.00 - - 0	USD 100,000.00 100,000.00 95,000.00 1	USD 550,000.00 275,000.00 522,500.00 2	USD 880,000.00 146,666.67 836,000.00 6
Sarah Vasquez	Sum of Amount Average amount Amount after Discount Record Count	USD 20,000.00 20,000.00 19,000.00 1	USD 140,000.00 70,000.00 133,000.00 2	USD 340,000.00 68,000.00 323,000.00 5	USD 110,000.00 55,000.00 104,500.00 2	USD 300,000.00 100,000.00 285,000.00 3	USD 1,100,000.00 275,000.00 1,045,000.00 4	USD 2,010,000.00 118,235.29 1,909,500.00 17
Total	Sum of Amount Average amount Amount after Discount Record Count	USD 100,000.00 20,000.00 95,000.00 5	USD 350,000.00 70,000.00 332,500.00 5	USD 820,000.00 74,545.45 779,000.00 11	USD 620,000.00 56,363.64 589,000.00 11	USD 2,600,000.00 371,428.57 2,470,000.00 7	USD 3,250,000.00 295,454.55 3,087,500.00 11	USD 7,740,000.00 154,800.00 7,353,000.00 50

Row Counts ☑○   Detail Rows ☑○   Grand Total ☑○   Stacked Summaries ☑○

To resize a column width, hover over the column header's right edge. When the resize cursor appears, drag the edge until the data fits right for you. When you're done, click the cursor to set the new width.

Total Records	Total Amount	Average amount	Amount after Discount
1,962	USD 275,038,105.00	140,182.52	261,286,199.75

Account Na... ▼	Opportunity Name ▼	Opportun... ▼	Created Date ▼	Amount ▼	fx Average amount ▼
ZiffCorp (2)	ZiffCorp - Laptops	Fred Williamson	4/5/2018	USD 20,000....	-
	ZiffCorp- Printers	Fred Williamson	4/5/2018	USD 1,500,0...	-
Subtotal				USD 1,520,0...	760,000.00
West Airlines (2)	West Airlines - 10 AW 5000 Se...	Fred Williamson	4/5/2018	USD 20,000....	-
	West Airlines - 12 AW 5000 Se...	Fred Williamson	4/5/2018	USD 1,403,3...	-
Subtotal				USD 1,423,3...	711,662.50

Filter Report Data

What if your report gives you more data than you need? Use filters to pare down your report until it only shows the data that you want.

Filter Reports by Values

Filter a report by value when you want to define the filter criteria yourself. For example, filter for opportunities worth more than $50,000.00, cases that mention word "widgets" in the subject, or Accounts located in California.

Filter Reports Using Field Comparisons with Field-To-Field Filters

Field-to-field filters let you filter a report by comparing the values of two different report fields. For example, see cases modified after closing date by filtering on cases with a last modified date after the closed date.

Add Filter Logic

Filter logic governs how and when filters apply to your report.

Filter Across Objects with Cross Filters

Use a cross filter to fine-tune your results by including or excluding records from related objects and their fields, without having to write formulas or code. You can apply cross filters by themselves, or in combination with field filters.

Filter Report Data by Role Hierarchy

Want to see records based on org structure or job function? Get records owned by everyone in a job role (like sales manager) and their subordinate roles (like sales person) by filtering your report on a role.

Filter Reports Via URL Parameters in Lightning Experience

No need to futz with filters! Pass URL parameters to set filter values in Lightning Experience reports. When linking to reports or when bookmarking a report, add filter value parameters to the URL to customize how the report filters when opened. For example, bookmark your opportunities report and add a filter value parameter to specify whether you see New Business or Existing Business.

Filter Field History Reports by Old and New Values

Field history reports let you track changes in fields by adding an "Old Value" and a "New Value" column. With a field history report, each report row represents a change to a record. By filtering a field history report, you can answer questions like "Which opportunities have we closed so far this financial quarter?"

Filter Knowledge Reports by Category

Return information about entire categories and subcategories of knowledge articles with category filters.

Filters Type Reference

Several different types of filters help you scope your report data: standard filters, field filters, cross filters, and row limit filters. Each filter type filters your report in different ways. This list of filter types helps you choose the right filter types for your report.

Filter Operators Reference

The operator in a filter is like the verb in a sentence. Operators specify how filter criteria relate to one another. Refer to this list of filter operators when setting filters on list views, reports, dashboards, and some custom fields.

Relative Date Filter Reference

Relative date filters let you filter on date fields using easy-to-understand, human-speech-inspired syntax.

Notes about Filtering on Types of Fields and Values

Keep these tips in mind when filtering on text fields, date fields, numeric values, picklist values, and blank or null values.

Tips for Filtering on Multiple Currencies

Tips for filtering on currency fields when your organization uses multiple currencies.

Filter Reports by Values

Filter a report by value when you want to define the filter criteria yourself. For example, filter for opportunities worth more than $50,000.00, cases that mention word "widgets" in the subject, or Accounts located in California.

In Salesforce Classic, filter your report from the report builder.

In Lightning Experience, there are two ways to filter reports: either from the Report Builder or while viewing a report. To add or edit report filters, use the Report Builder. To edit existing, unlocked report filters while you're reading a report, run the report and then edit filters directly from the filters pane (▼). You can edit existing filters from the filters pane, but you can't add new ones.

Each report supports up to 20 field filters.

> 📝 **Note:** In Lightning Experience, these filters are available in the Report Builder, but are not shown in the filter panel when viewing a report. Even though the filters are not shown, they still filter the report.
>
> - Row limit filters
> - Historical field filters
>
> These filters appear in the filter panel when viewing a report but are not editable.
>
> - Cross filters
> - Chart filters

1. From the Lightning Experience report builder, click ▼ **Filters**. Then, choose a field from the **Add filter...** picklist.

   From the Salesforce Classic report builder, open the **Add** dropdown menu and select a filter type:

   - **Field Filter** to filter on fields. For example, use a field filter to filter by `Account Name equals Acme`.

   - **Filter Logic** to customize how existing filters apply to your report. Each filter is assigned a number. To get your report to return records that meet the criteria of Filter 1 and either Filter 2 or Filter 3, use this filter logic: `Filter 1 AND (Filter 2 OR Filter 3)`. Filter logic requires at least one field filter.

     To add filter logic in the Lightning Experience report builder, click ▼ > **Add Filter Logic**.

   - **Cross Filter** to filter on one object's relationship to another object. Cross filter on `Accounts with Opportunities` so that your report only returns Accounts that have Opportunities. Add a subfilter to a cross filter to further filter by the second object. For example, the Opportunity subfilter `Amount greater than 50000` causes your report to return Accounts that have Opportunities worth more than $50,000.00.

     Cross filters aren't available in the Lightning Experience report builder.

   - **Row Limit** to limit the number of report results in tabular reports. To see which five Accounts have the largest annual revenue, set a row limit of `Top 5 Accounts by Annual Revenue`.

     Row limit filters aren't available in the Lightning Experience report builder.

   Standard filters, such as date filters, are applied by default to most objects. Look for them underneath the Add dropdown menu and customize them as necessary. Different objects have different standard filters.

2. Enter filter criteria.

For help with entering filter criteria, see Filter Operators and Filter Logic.

3. Optionally, to prevent people from editing a field filter while reading your report in Lightning Experience, check **Locked**.

4. Click **Save**.

5. To read your filtered report, click **Run Report**.

👁 Example: Say that you want your team to call new leads at companies with more than 100 employees located in California, Arizona, or Nevada. You have a leads report with fields like `Lead Status`, `Number of Employees`, and `State`. Your report gives a complete overview of your entire company's leads. But you only want to see *new* leads that *have more than 100 employees* and are *located in California*. Apply these filters to your report:

1. Lead Status equals New

2. Number of Employees greater than 100

3. State includes California, Arizona, Nevada

Now your leads report returns only the leads you need.

SEE ALSO:

Create or Clone a List View in Lightning Experience

Filter Reports Using Field Comparisons with Field-To-Field Filters

Field-to-field filters let you filter a report by comparing the values of two different report fields. For example, see cases modified after closing date by filtering on cases with a last modified date after the closed date.

Add a field-to-field filter while editing a report.

1. From the Lightning Experience report builder, click ▼ **Filters**. Then, choose a field from the **Add filter...** picklist.

2. If the field you chose to filter on supports field-to-field filters, then set the Type dropdown to **Field**. If the field does not support field-to-field filtering, then the Type dropdown does not appear.

3. Enter filter criteria.

   For help with entering filter criteria, see Filter Operators and Filter Logic.

4. Optionally, to prevent people from editing a field filter while reading your report in Lightning Experience, check **Locked**.

5. Click **Save**.

6. To read your filtered report, click **Run Report**.

Report data filters based on the criteria you set.

> 👁 **Example:** Say that you want opportunities worth less than projected. You have an opportunity report with fields like `Amount` and `Projected Amount`, and `State`. Apply a field-to-field filter to your report that compares values between Amount and Projected Amount.
>
> 1. Click **Add filter...**.
>
> 2. Select **Amount**.
>
> 3. Set Operator to **less than**, Type to **Field**, and Value to **Projected Amount**. If necessary, clear the default Value by clicking **X**.
>
> 4. Click **Apply**.
>
> 5. Click **Save**.
>
> Now your report has a field-to-field filter (1) that returns opportunities worth less than projected (2).

Get the Most Out of Field-To-Field Filters Tips, Limits, and Limitations

As you get ready to add field-to-field filters, review these tips, limits, and limitations.

- Each report supports up to 4 field-to-field filters.
- Field-to-field filters only support these field types:
    - Numeric
    - Date
    - Datetime

    If your org has enabled multiple currencies, then despite being a numeric field type, currency fields (like Amount) aren't supported.

- Field-to-field filters are unavailable in report types that:
    - Reference an external object
    - Have a "with or without" (outer join) relationship between objects
- Field-to-field filters only compare fields of the same data type. For example, a datetime field can only be compared with another datetime.
- Joined reports don't support field-to-field filters. To convert a report with field-to-field filters into a joined report, first remove the field-to-field filter.
- Historical tracking reports support field-to-field filters, but the filter must be based on a history field.
- Field-to-field filters must compare two different fields.
- When a field-to-field filter compares a null value with a non-null value, it returns `false` or 0. If two null values are compared, then the filter returns `true` or 1.
- Field-to-field filters aren't available in Salesforce Classic. You can't run or edit reports with field-to-field filters in Salesforce Classic.

Add Filter Logic

Filter logic governs how and when filters apply to your report.

After adding a filter to your report, the filter is numbered. Your first filter becomes Filter 1 and your second filter becomes Filter 2. You apply filter logic based on these numbered filters.

For example, let's say you have a report of Accounts with fields like `State`, `Annual Revenue`, and `Industry`. Your report has these filters:

1. State includes California, Arizona, Nevada

2. Industry equals Banking

3. Annual Revenue greater than 1000000

These three filters cause your report to return Accounts located in California, Arizona, or Nevada in the Banking industry with annual revenue greater than $1,000,000. But what if you only want Accounts located in California, Arizona, or Nevada AND in the Banking industry, OR with annual revenue greater than $1,000,000.00? Add filter logic to your report.

To add filter logic,

1. From the Lightning Experience report builder, click ▼ **FILTERS** > ▼ > **Add Filter Logic**.
   From the Salesforce Classic report builder, click **Add** > **Filter Logic**.

2. Enter each filter line number, separated by a filter logic operator.
   For example, **(1 AND 2) OR 3** finds records that match both Filter 1 and Filter 2, or Filter 3.
   Here's a complete table of filter logic operators:

Operator	Definition
AND	Finds records that match both values. **1 AND 2**
OR	Finds records that match either value. **1 OR 2**
NOT	Finds records that exclude values. For example, Filter 1 is **Industry equals "Biotechnology"**. You set filter logic as **Not 1**. Your report returns records which aren't equal to Biotechnology.

3. Click **Save**.

Now your report shows Banking industry Accounts in California, Arizona, and Nevada with any amount of revenue, and any Account with annual revenue above $1,000,000.00.

📝 **Note:**

- Filter logic isn't available for all filters. For example, you can't use them for roll-up summary fields.

- On reports where object A may or may not have object B, you can't use the OR condition to filter across multiple objects. For example, if you enter filter criteria *Account Name starts with M OR Contact First Name starts with M*, an error message displays informing you that your filter criteria is incorrect.
- Except filter logic on lookup fields, you can't use filter logic if your field filters use any of the following fields:

  - `Topics`
  - `Description`
  - any `Address Line 1`, `Address Line 2`, `Address Line 3` fields
  - `Forecast Category`
  - `Campaign: Member Type`
  - `User: Profile Name`
  - `Login Status`
  - custom long-text area fields

SEE ALSO:

[Notes about Filtering on Types of Fields and Values](#)

Filter Across Objects with Cross Filters

Use a cross filter to fine-tune your results by including or excluding records from related objects and their fields, without having to write formulas or code. You can apply cross filters by themselves, or in combination with field filters.

▶ Watch a video:

- ▶ How to Use Cross Filters (Lightning Experience)
- ▶ Using Cross Filters in Reports (Salesforce Classic)

To see which accounts have escalated cases, create an accounts report and add a cross filter. Set the cross filter to show accounts with cases. Then, filter cases for `status equals escalated`.

1. Create a Cross Filter

   Use cross filters to include or exclude records in your report results based on related objects and their fields.

2. Edit a Cross Filter

   To change how a cross filter filters the primary or secondary object, edit it.

3. Delete a Cross Filter

   When you no longer need a cross filter, or a subfilter in a cross filter, delete it.

4. Example: Use WITH in Cross Filters

   Use cross filters to filter a report by an object's child objects using WITH conditions. For example, filter a report to show just accounts with cases.

5. Example: Use WITHOUT in Cross Filters

   Use cross filters to filter a report by an object's child objects using WITHOUT conditions. For example, filter a report to show just contacts without activities.

6. Example: Multiple Cross Filters and a Subfilter

   Use cross filters to filter a report by an object's child objects using both WITH and WITHOUT conditions. For example, filter a report to show accounts that have cases but don't have activities.

7. Cross Filter Considerations and Limits

   As you create and work with cross filters, take note of these considerations and limits.

Create a Cross Filter

Use cross filters to include or exclude records in your report results based on related objects and their fields.

▶ Watch a video:

- ⊙ How to Use Cross Filters (Lightning Experience)
- ⊙ Build Interactive Analytics Dashboards (English Only)

To see which accounts have escalated cases, create an accounts report and add a cross filter. Set the cross filter to show accounts with cases. Then, filter cases for *status equals escalated*.

Cross filters are available in both Lightning Experience and Salesforce Classic. Cross filters that you create in Lightning Experience work in Salesforce Classic, and vice versa.

The steps for creating a cross filter are a little different depending on whether you're using Lightning Experience or Salesforce Classic. Follow the instructions based on whether you're using the Lightning Experience or the Salesforce Classic report builder.

Create a Cross Filter in the Lightning Experience Report Builder

1. Edit or create a report.

2. From the FILTERS tab, click ▼ > **Add Cross Filter**.

3. Choose whether to show your primary object **with** or **without** a secondary object. Then, choose the secondary object.

4. Click **Apply**.

5. Optionally, to add a filter to the secondary object, click **Add** `Secondary Object` **Filter**, where `Secondary Object` is the name of the secondary object.

   a. Choose a field from the secondary object.

   b. Specify an operator and values for the secondary object filter.

   c. Click **Apply**.

6. Click **Save**.

Create a Cross Filter in the Salesforce Classic Report Builder

1. Edit or create a report.

2. From the Filters pane, click **Add** > **Cross Filter**.

3. Select a parent object from the dropdown list. Your choice determines which related objects you see in the child object list.

   💡 Tip: In report types based on Campaigns, the parent object can be the secondary object in the report type. For example, in a "Campaigns with Leads" report, the parent object can be `Campaigns` or `Leads`.

4. Choose *with* or *without*.

5. Select a child object from the dropdown or search by its name. The dropdown list contains all eligible child objects of your selected parent object.

6. Optionally, to add filters on the secondary object, add subfilters:

   a. Click **Add** *Secondary Object* **Filter**, where *Secondary Object* is the name of the secondary object.

   b. Select a field. The fields are determined by the child object in the cross filter. For example, if your cross filter is *Accounts with Cases*, you can use case fields for your subfilter.

   c. Choose a filter operator.

   d. Enter a value.

7. Click **OK**.

8. Click **Save**.

The report filters based on the primary-secondary object relationship and any additional filters on the secondary object.

> **Note:** In Lightning Experience, although cross filters still filter the report, they do not appear in the Filters panel on the report run page. If you have permission to edit a report, you can view cross filters in the report builder.

SEE ALSO:

Example: Multiple Cross Filters and a Subfilter

Example: Use WITH in Cross Filters

Example: Use WITHOUT in Cross Filters

Edit a Cross Filter

To change how a cross filter filters the primary or secondary object, edit it.

Cross filters are available in both Lightning Experience and Salesforce Classic. Cross filters that you create in Lightning Experience work in Salesforce Classic, and vice versa. You can edit cross filters in either builder.

The steps for editing a cross filter are a little different depending on whether you're using Lightning Experience or Salesforce Classic. Follow the instructions based on whether you're using the Lightning Experience or the Salesforce Classic report builder.

Edit a Cross Filter in the Lightning Experience Report Builder

1. Edit a report.

2. From the FILTERS tab, under Cross Filters, find the cross filter you want to edit.

3. To edit the primary or secondary object, click the cross filter.

    a. From the Show Me dropdown, change the primary object.

    b. From the Operator dropdown, choose whether to show your primary object **with** or **without** a secondary object. Then, choose the secondary object.

    c. From the Secondary Object dropdown, change the secondary object.

    d. Click **Apply**.

4. To edit a subfilter on the secondary object, click the sub filter.

    a. From the Edit Filter menu, change operator and values.

    b. To change the field of the subfilter, remove the subfilter by clicking **X**. Then, add another subfilter based on the desired field.

    c. Click **Apply**.

5. To save the report, click **Save**.

Edit a Cross Filter in the Salesforce Classic Report Builder

1. Edit a report.

2. From the Filters pane, click find the cross filter you want to edit. Click **Edit**.

    a. Change primary object, operator, or secondary object.

    b. Edit subfilters by changing field, operator, and value.

    c. To delete a subfilter, click **Remove**.

    d. To add another subfilter, click **Add** `Secondary Object` **Filter**, where `Secondary Object` is the name of the secondary object.

3. Click **OK**.

4. To save the report, click **Save**.

The report filters based on the primary-secondary object relationship and any additional filters on the secondary object.

> 📝 **Note:** In Lightning Experience, although cross filters still filter the report, they do not appear in the Filters panel on the report run page. If you have permission to edit a report, you can view cross filters in the report builder.

Delete a Cross Filter

When you no longer need a cross filter, or a subfilter in a cross filter, delete it.

Cross filters are available in both Lightning Experience and Salesforce Classic. Cross filters that you delete in Lightning Experience are also deleted in Salesforce Classic, and vice versa.

The steps for deleting a cross filter are a little different depending on whether you're using Lightning Experience or Salesforce Classic. Follow the instructions based on whether you're using the Lightning Experience or the Salesforce Classic report builder.

Delete a Cross Filter in the Lightning Experience Report Builder

1. Edit a report.
2. From the FILTERS tab, under Cross Filters, find the cross filter you want to delete. To delete the cross filter, and all its subfilters, click **X**.
3. To delete a subfilter on the cross filter, but not the cross filter, find the subfilter you want to delete. From the subfilter, click **X**.
4. To save the report, click **Save**.

Delete a Cross Filter in the Salesforce Classic Report Builder

1. Edit a report.
2. From the Filters pane, find the cross filter you want to delete. To delete the cross filter, and all its subfilters, click **Remove**.
3. To delete a subfilter on the cross filter, but not the cross filter, click **Edit**.
    a. Find the subfilter you want to delete, and click **Remove**.
    b. Click **OK**.
4. To save the report, click **Save**.

The report filters based on the primary-secondary object relationship and any additional filters on the secondary object.

📝 Note: In Lightning Experience, although cross filters still filter the report, they do not appear in the Filters panel on the report run page. If you have permission to edit a report, you can view cross filters in the report builder.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

USER PERMISSIONS

To add edit, or delete a cross filter in private reports:

- **Legacy Folder Sharing**
    Create and Customize Reports

    Enhanced Folder Sharing
    Create and Customize Reports

To add edit, or delete a cross filter in private and public reports:

- **Legacy Folder Sharing**
    Report Builder

    Enhanced Folder Sharing
    Report Builder OR Report Builder (Lightning Experience)

Example: Use WITH in Cross Filters

Use cross filters to filter a report by an object's child objects using WITH conditions. For example, filter a report to show just accounts with cases.

Let's say a recent campaign in California won you a lot of new customers. You want to ensure that their customer cases get resolved quickly. You can create a report to see which of those accounts currently have cases.

Find Accounts with Cases in the Lightning Experience Report Builder

1. Edit or create a report based on the Accounts report type.

2. Create a filter where `Billing State/Province equals CA` and click **OK**.

   Now your report shows all accounts in California. To see Californian accounts with cases, let's add a cross filter: `Accounts with Cases`.

3. Click ▼ > **Add Cross Filter**.

4. From the Show Me dropdown, select **Accounts**.

5. From the Operator dropdown, select **with**.

6. From the Secondary Object dropdown, select **Cases**.

7. Click **Apply**.

8. Click **Save**.

Find Accounts with Cases in the Salesforce Classic Report Builder

1. Edit or create a report based on the Accounts report type.

2. Create a field filter where `Billing State/Province equals CA` and click **OK**.

   Now your report shows all accounts in California. To see Californian accounts with cases, let's add a cross filter: `Accounts with Cases`.

3. From the Filters pane, click **Add** > **Cross Filter**.

4. From the Primary Object dropdown (left of the operator), choose **Accounts**.

5. From the operator dropdown, choose **with**.

6. From the Secondary Object dropdown (right of the operator), choose **Cases**.

7. Click **OK**.

8. Click **Save**.

When you run the report, the results include only California accounts with associated cases.

SEE ALSO:

[Create a Cross Filter](#)

Example: Use WITHOUT in Cross Filters

Use cross filters to filter a report by an object's child objects using WITHOUT conditions. For example, filter a report to show just contacts without activities.

Let's say that you've just imported a list of California accounts and you want to find which ones are missing contacts before you assign owners:

Find Accounts with Missing Contacts in the Lightning Experience Report Builder

1. Edit or create a report based on the Accounts report type.

2. Create a filter where `Billing State/Province equals CA` and click **OK**.

    Now your report shows all accounts in California. To see Californian accounts without contacts, let's add a cross filter: `Accounts without Contacts`.

3. Click ⏷ > **Add Cross Filter**.

4. From the Show Me dropdown, select **Accounts**.

5. From the Operator dropdown, select **without**.

6. From the Secondary Object dropdown, select **Contacts**.

7. Click **Apply**.

8. Click **Save**.

Find Accounts with Missing Contacts in the Salesforce Classic Report Builder

1. Edit or create a report based on the Accounts report type.

2. Create a field filter where `Billing State/Province equals CA` and click **OK**.

    Now your report shows all accounts in California. To see Californian accounts without contacts, let's add a cross filter: `Accounts without Contacts`.

3. From the Filters pane, click **Add** > **Cross Filter**.

4. From the Primary Object dropdown (left of the operator), choose **Accounts**.

5. From the operator dropdown, choose **without**.

6. From the Secondary Object dropdown (right of the operator), choose **Contacts**.

7. Click **OK**.

8. Click **Save**.

When you run the report, the results include only California accounts without an associated contact.

SEE ALSO:

[Create a Cross Filter](#)

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

USER PERMISSIONS

To add edit, or delete a cross filter in private reports:

- **Legacy Folder Sharing**
  Create and Customize Reports

  Enhanced Folder Sharing
  Create and Customize Reports

To add edit, or delete a cross filter in private and public reports:

- **Legacy Folder Sharing**
  Report Builder

  Enhanced Folder Sharing
  Report Builder OR Report Builder (Lightning Experience)

Example: Multiple Cross Filters and a Subfilter

Use cross filters to filter a report by an object's child objects using both WITH and WITHOUT conditions. For example, filter a report to show accounts that have cases but don't have activities.

Say you're a salesperson who wants to see which customer accounts have unresolved escalations because you want to ensure that your support team takes care of them.

Find Accounts with Escalated Cases and without Activities in the Lightning Experience Report Builder

1. Edit or create a report based on the Accounts report type.

2. Click ▼ > **Add Cross Filter**.

3. From the Show Me dropdown, select **Accounts**.

4. From the Operator dropdown, select **with**.

5. From the Secondary Object dropdown, select **Cases**.

6. Click **Apply**.

   The report now shows Accounts that have Cases.

7. Refine the cross filter so that the report shows Accounts that have Escalated Cases:

   a. Click **Add Cases Filter**.

   b. Select **Status**.

   c. Select **Escalated**.

   d. Click **Apply**.

   Now, add another cross filter so that the report shows only Accounts without Activities.

8. Click ▼ > **Add Cross Filter**.

9. From the Show Me dropdown, select **Accounts**.

10. From the Operator dropdown, select **without**.

11. From the Secondary Object dropdown, select **Activities**.

12. Click **Apply**.

13. Click **Save**.

Find Accounts with Escalated Cases and without Activities in the Salesforce Classic Report Builder

1. Edit or create a report based on the Accounts report type.

2. From the Filters pane, click **Add** > **Cross Filter**.

3. From the Primary Object dropdown (left of the operator), choose **Accounts**.

4. From the operator dropdown, choose **with**.

5. From the Secondary Object dropdown (right of the operator), choose **Cases**.

6. Refine the cross filter so that the report shows Accounts that have Escalated Cases:

   a. Click **Add Cases Filter**.

   b. From the Case Field dropdown, select **Status**.

   c. From the Operator dropdown, select **equals**.

    d. From the picklist lookup, enter or select **Escalated**.

7. Click **OK**.

    Now, add another cross filter so that the report shows only Accounts without Activities.

8. From the Filters pane, click **Add** > **Cross Filter**.

9. From the Primary Object dropdown (left of the operator), choose **Accounts**.

10. From the operator dropdown, choose **with**.

11. From the Secondary Object dropdown (right of the operator), choose **Cases**.

12. Click **Save**.

When you run the report, it will include only customer accounts with escalated cases and without activities.

SEE ALSO:

[Create a Cross Filter](#)

Cross Filter Considerations and Limits

As you create and work with cross filters, take note of these considerations and limits.

Considerations

- Adding cross filters can potentially slow down your report. To avoid having the report or preview time out, limit the data returned by setting filters. For example, select `My opportunities` for `Show` and `Current FQ` for `Range` instead of viewing all opportunities for all time.

- Since the objects available in cross filters depend on the parent object of the report type you choose, consider the related child objects before selecting a report type. For example, choose the Accounts report type to filter on `Accounts with Partners` because Partner is a child object of Account.

- Cross filters work in conjunction with your report type selection. Cross filters have an AND relationship with the report type you select. Choosing a report type of Accounts with Partners and adding a cross filter for `Accounts without Partners` will yield no results.

- In Lightning Experience, although cross filters appear on the run page, you cannot change their values on the run page. If you have permission to edit a report, you can change cross filter values in the report builder.

- When you filter on a Salesforce object ID, the ID must be 15 or 18 characters in length and the ID is not case-sensitive.

Limits

- Each report can have up to 3 cross filters.
- Each cross filter can have up to 5 subfilters.
- Filter logic doesn't apply to cross filters.

SEE ALSO:

Example: Multiple Cross Filters and a Subfilter

Create a Cross Filter

Example: Use WITH in Cross Filters

Example: Use WITHOUT in Cross Filters

Filter Report Data by Role Hierarchy

Want to see records based on org structure or job function? Get records owned by everyone in a job role (like sales manager) and their subordinate roles (like sales person) by filtering your report on a role.

For example, to see opportunities owned by sales team members in California, filter your opportunity report by the role *Sales Manager - California*. Optionally, drill down on opportunities owned by a specific sales manager in California, narrow your results by a specific person in the role.

> **Note:** Role hierarchy filters are available for reports based on these standard report types:
> - Activity
> - Task
> - Event
> - Campaigns with Opportunities
> - Forecasts
> - Opportunity, except for:
>   - Opportunities with Competitors
>   - Opportunities with Contact Roles
>   - Opportunities with Contact Roles and Products
>   - Opportunity History
>   - Opportunities with Partners
>   - Opportunities with Teams
>   - Opportunities with Teams and Products

Filter by Role Hierarchy in Lightning Experience

In Lightning Experience, you can filter by role hierarchy while reading or editing a report.

While Editing a Report in the Lightning Experience Report Builder

1. Edit or create a report.
2. From the FILTERS tab, click **Show Me**.
3. From the Show Me menu, filter by a role.
4. Optionally, further filter the report by narrowing results by a person in your selected role. Narrowing by a person shows records that belong to that person, and to people in roles that report to that person.
5. Optionally, to reset your selection and show results rolling up to the top-most role, click **Reset Role to Top Level**.
6. Click **Done**.
7. Click **Save**.

While Viewing a Report

1. If necessary, from the report run page, click ▼ to open the Filters panel.
2. From the Filters panel, click **Role Hierarchy**.
3. From the Role Hierarchy menu, filter by a role.

4. Optionally, further filter the report by narrowing results by a person in your selected role. Narrowing by a person shows records that belong to that person, and to people in roles that report to that person.

Filter by Role Hierarchy in Salesforce Classic

1. From the report run page, click **Show Hierarchy**.

2. Drill down to a role.

3. If you'd like the report to open already filtered by a role, then drill down to the role and then click **Customize** to open the report builder. Then, click **Report Properties** and check **Save Hierarchy Level**. From Report Properties, click **Save**, then, from the report builder, click **Save** again.

In Lightning Experience and Salesforce Classic, if you previously saved a report without selecting a role, the report is filtered by your role when you run it. If you don't have an assigned role, then the report is filtered by the top-most role.

Filter Reports Via URL Parameters in Lightning Experience

No need to futz with filters! Pass URL parameters to set filter values in Lightning Experience reports. When linking to reports or when bookmarking a report, add filter value parameters to the URL to customize how the report filters when opened. For example, bookmark your opportunities report and add a filter value parameter to specify whether you see New Business or Existing Business.

Watch a Demo: ⊙ [Dynamically Filter Reports Using URL Parameters (Lightning Experience)](#)

1. Append the parameter `&fv0=`***`Filter Value`*** to the end of a report's URL.

   Let's take a closer look at what the parameter means.

   - *&* — Denotes a new parameter in the URL. If no other parameters are present in the URL, then substitute `?` in place of `&`.

   - *fv0* — The *fv* stands for "filter value," and is the name of the parameter. The `0` is the numerical order in which the filter appears in the report. (The first filter is 0, the second filter is 1, the third is 2, and so forth.) Standard filters don't count in this order, and can't be filtered using URL parameters, although they appear as the first three filters on any report. To set the value of the fifth filter in the report, specify `fv4`. In our example, we're filtering the first field filter in the report.

     The order in which filters appear in Lightning Experience on the report run page isn't necessarily the order that filters appear in the report. Locked filters are listed beneath unlocked filters on the run page, but aren't necessarily ordered after all the unlocked filters. To see the order of filters in your report, refer to their order in the report builder. Alternatively, make a GET call to `/services/data/`***`v39.0`***`/analytics/reports/`***`<report ID>`***`/describe` and note their order in the JSON response.

   - ***`Filter Value`*** — The criteria that the filter operates on.

     The filter value must be URI encoded, which means certain characters (such as spaces) must be written in a format that URLs can understand. A space (' ') becomes `%20` when URI encoded.

2. Navigate to the report's URL with the parameter appended.

When the report opens, it opens with filters applied as specified with parameters in the URL.

👁 **Example:** Periodically throughout your day you check your opportunities report, and after opening it, you always edit a filter to show `New Business` or `Existing Business`. (The **Type** filter includes both `New Business` and `Existing Business` by default.)

Instead of loading the report, editing the filter, and reloading the report, create a bookmark with a filter value parameter for `New Business` in the URL. Take note of the unfiltered opportunities report URL.

```
https://MyDomainName.my.salesforce.com/lighting/r/Report/00OR0000000PCHYMA4/view?t=1479844235107
```

Edit the URL to set the **Type** filter value to `New Business` by appending the parameter `&fv0=New%20Business`. Remember, the parameter value must be URI encoded, which means certain characters (such as spaces) must be written in a format that URLs can understand. In our example, the space (' ') between "New" and "Business" becomes `%20` when URI encoded. The full, bookmarked URL reads:

```
https://MyDomainName.my.salesforce.com/lighting/r/Report/00OR0000000PCHYMA4/view?t=1479844235107&fv0=New%20Business
```

When you navigate to your opportunity report through the URL with a filter value parameter, the report opens filtered and ready to read.

Take note of these limitations to setting filter values via the URL.

- Only field filters support edits from URL parameters. These filters don't support edits from URL parameters. Because they don't support edits, they don't count when listing filters — `fv0` is the first field filter.

  - Standard filters (role hierarchy filters, scope filters, date filters)
  - Cross filters
  - Row-limit filters
  - Chart filters
  - Locked filters

- Filter operators (like `equals` and `greater than`) can't be modified via URL parameters.
- You can't change the field being filtered via URL parameters.
- You can't add new filters to reports using filter value URL parameters. You can only modify existing filters.
- You can't delete filters from reports using filter value URL parameters. Setting a blank value filters by no text or numerals, but doesn't remove the filter.
- You can't use filter value URL parameters with historical tracking reports.
- Locked filters

Filter Field History Reports by Old and New Values

Field history reports let you track changes in fields by adding an "Old Value" and a "New Value" column. With a field history report, each report row represents a change to a record. By filtering a field history report, you can answer questions like "Which opportunities have we closed so far this financial quarter?"

Before filtering on Old Value and New Value, ensure that field tracking is turned on for your Salesforce org. If you don't see any field history report types (such as Opportunity Field History), or if your field history report isn't returning any records even though you know that records have changed, then ask your admin to turn on field history tracking. For information about field tracking, including how to set it up, see Field History Tracking in Salesforce help.

All filters on the Old Value and New Value fields are text filters, which sometimes means that special consideration is needed. Even if New Value captures a change to, say, a numeric field - like Amount - it captures the number as text. That means that filter operators like "greater than" and "less than" won't work like you'd expect. If a field is represents a currency - like USD 1,000.50 - and you filter the report to show `New Value equals USD 1,000.50`, then the filter may not return any records for a colleague who works in a different locale. For example, for your colleague in Paris to get the same report results, the filter would probably need to read `New Value equals USD 1 000,50` because numer formats vary widely by locale.

To see which opportunities closed this financial quarter, create an report based on the report type Opportunity Field History, then add these filters:

- Ensure that the Edit Date standard filter is set to `Current FQ`.
- `New Value contains Closed`
- `Old Value does not contain Closed`

Filter Knowledge Reports by Category

Return information about entire categories and subcategories of knowledge articles with category filters.

Say you have categories defined by geographic region. The root category lists countries like United States, Mexico, and Canada. Under United States, categories include individual states like Arizona, California, and Oregon. Under each state, categories include cities. Category filters have their own operators:

- **at**—Return knowledge articles in the specified category.
- **above**—Return knowledge articles above and in the specified category.
- **below**—Return knowledge articles below and in the specified category.
- **above or below**—Return all knowledge articles above, below, and in the specified category. If there is more than one root category, then the filter returns only knowledge articles that are part of the parent root category.

So with a category filter on `United States below Arizona`, your report returns knowledge articles about cities like Tucson and Phoenix. Or, with a category filter on `United States above Arizona`, your report instead excludes knowledge articles about cities in Arizona.

Add a category filter while editing a Knowledge report.

1. From the Lightning Experience report builder, open the filter menu by clicking 🔻 **Filters**.

2. Click 🔽 > **Add Category Filter** (1).

    A category filter section appears in the filter pane (2).

3. Choose a category to filter by (3).

4. Choose an operator. Options are:

    - **at**—Return knowledge articles in the specified category.
    - **above**—Return knowledge articles above and in the specified category.
    - **below**—Return knowledge articles below and in the specified category.
    - **above or below**—Return all knowledge articles above, below, and in the specified category.

5. Click **Apply**.

6. To save and read your filtered report, click **Save & Run**.

Report data filters based on the criteria you set.

> 📝 Note: Data category filters are only available on report types that include the Knowledge object, such as Knowledge Articles or Knowledge with Cases.

Filters Type Reference

Several different types of filters help you scope your report data: standard filters, field filters, cross filters, and row limit filters. Each filter type filters your report in different ways. This list of filter types helps you choose the right filter types for your report.

Filter Type	Description
Standard Filter	Standard filters are applied by default to most objects. Different objects have different standard filters, but most objects include the standard filters `Show Me` and `Date Field`. `Show Me` filters the object around common groupings (like "My accounts" or "All accounts"). `Date`

Filter Type	Description
	`Field` filters by a field (such as `Created Date` or `Last Activity`) and a date range (such as "All Time" or "Last Month").
Field Filter	Field filters are available for reports, list views, workflow rules, and other areas of the application. For each filter, set the field, operator, and value. For all report types except joined reports, you can add a report filter by dragging a field from the Fields pane to the Filters pane.
Filter Logic	Add Boolean conditions to control how field filters are evaluated. Add at least 1 field filter before applying filter logic.
Cross Filter	Filter a report by the child object using WITH or WITHOUT conditions. Add subfilters to further filter by fields on the child object. For example, if you have a cross filter of `Accounts with Opportunities`, click **Add Opportunity Filter** and create the `Opportunity Name equals ACME` subfilter to only include those opportunities.
Row Limit	For tabular reports, select the maximum number of rows to display, then choose a field to sort by and the sort order. You can use a tabular report as the source report for a dashboard table or chart component, if you limit the number of rows it returns.

To add a report with a row limit filter, specify a "name" and "value" in Dashboard Settings in the Report Builder.

Example: Using Row Limits in Report Filters
Here is where you can see a sample of a report filter using a field filter, filter logic, and a row limit.

Example: Using Row Limits in Report Filters

Here is where you can see a sample of a report filter using a field filter, filter logic, and a row limit.

Say you are a sales executive who wants to see which California accounts currently have the most potential to generate revenue:

1. Click **New Report** from the Reports tab.

2. Open the **Accounts & Contacts** report type, choose **Accounts**, and click **Create**.

3. To find California accounts that either have over $10 million in revenue or are public companies, click **Add** > **Field Filter** and create these filters:

   a. *Billing State/Province equals CA*

   b. *Annual Revenue greater than 10000000*

   c. *Ownership equals Public*

4. To ensure that your results include all California accounts with $10 million in revenue OR that are public, click **Add** > **Filter Logic** and enter *1 AND (2 OR 3)*.

5. To limit the number of results for a tabular report to 10, click **Add** > **Row Limit** and enter 10. Choose your sort field and sort order. Click **OK**.

When you click **Run Report**, your results will contain ten rows and include public companies in California with revenues of more than $10 million.

SEE ALSO:

Filters Type Reference

Limit Report Results

Filter Operators Reference

The operator in a filter is like the verb in a sentence. Operators specify how filter criteria relate to one another. Refer to this list of filter operators when setting filters on list views, reports, dashboards, and some custom fields.

Operator	Uses
equals	Use for an exact match. For example, "Created equals today."
less than	Use for results that are less than the value you enter. For example, "Quota less than 20000" returns records where the quota field ranges from 0 to 19,999.99.
greater than	Use when you want results that exceed the value you enter; for example, "Quota greater than 20000" returns records where the quota amount begins at 20,000.01.
less or equal	Use for results that match or are less than the value you enter.
greater or equal	Use for results that match or exceed the value you enter.

Operator	Uses
not equal to	Shows results that don't have the value you enter. This operator is useful for eliminating empty fields. For example, "Email not equal to <blank>." **Note:** If evaluating more than one value, none of the specified values can exist on the record. For example, if you specify `not equal to` A & C, only records where the value is neither A nor C are returned.
contains	Use for fields that include your search string, but sometimes also include other information. For example, "Account contains California" would find California Travel, California Pro Shop, and Surf California. Keep in mind that if you enter a short search string, it can match a longer word. For example, "Account contains pro" would find California Pro Shop and Promotions Corporation. This operator is not case-sensitive.
does not contain	Eliminates records that don't contain the value you enter. For example, "Mailing Address Line 2 does not contain P. O. Box." **Note:** When specifying filter criteria on roll-up summary fields, `does not contain` uses "or" logic on comma-separated values. On list views, reports, and dashboards, `does not contain` uses "and" logic. This operator is not case-sensitive.
starts with	Use when you know what your value starts with, but not the exact text. This operator is a narrower search term than "contains." For example, if you enter "Account starts with California," you find California Travel and California Pro Shop, but not Surf California.
includes	Available when you choose a multi-select picklist as the selected field. Use this operator to find records that include one or more of the values you enter. For example, if you enter "Interests includes hockey, football, baseball," you find records that have only hockey selected, and records that have two or three of the values entered. Results do not include partial matches of values.
excludes	Available when you choose a multi-select picklist as the selected field. Use this operator to find records that do not contain any values that match the ones entered. For example, if you enter "Interests exclude wine, golf," your report lists records that contain any other values from that picklist, including values that are blank. Results do not include partial matches of values. Enter values on separate lines.
between	Available for dashboard filters only. Use to filter on ranges of values. For each range, the filter returns results that are greater than or equal to the minimum value and less than the maximum value. For example, if you enter "Number of Employees from 100 through 500," your results include accounts with 100 employees up to those with 499 employees. Accounts with 500 employees aren't included in the results.
within	Available when you create list views based on a Geolocation custom field. Shows results that are within the specified radius from a fixed latitude and longitude. For example, if you enter "Warehouse location within 50 miles 37.775° −122.418°,", your list view includes all warehouses within a 50–mile radius of San Francisco, California.

Relative Date Filter Reference

Relative date filters let you filter on date fields using easy-to-understand, human-speech-inspired syntax.

For example, instead of filtering on `Close Date greater than Jan 1, 2017`, filter using a relative date: `Close Date equals this year`.

For Enterprise, Unlimited, Performance, Professional, and Developer Editions, the week is defined by the Locale dropdown list on your personal information page. For Contact Manager, Group, and Personal Editions, the week is defined by the Locale setting in the company profile. For example, when the locale is US English, a week runs Sunday to Saturday, whereas with UK English, a week spans Monday to Sunday.

Capitalization doesn't matter in relative date filter operators. *THIS YEAR* works, as do *This Year* and *this year*.

Notes:

- Standard filters and custom field filters interpret relative date filters that begin with NEXT *n* DAYS - differently. Standard date filters return records that are time-stamped on the day that the report is run continuing for *n* days. Custom field filters return records time-stamped on the day after the day the report is run continuing for *n* days.

  For example, say that your opportunity report's standard date filter is *CLOSE DATE equals NEXT 5 DAYS* and you run it on October 24. The report returns opportunities that close sometime between October 24 through October 28.

  Say that your opportunity report has a custom field filter (instead of a standard date filter) for *CLOSE DATE equals NEXT 5 DAYS* and you run it on October 24. The report returns opportunities that close sometime between October 25 through October 29.

- The drill down option for reports in Lightning Experience works only if the date field is grouped by any of the options available in the list of relative date filters in the following table. For example, if the date field on the report is grouped by Fiscal Period or Fiscal Week, you cannot drill down in the report.

Relative Date Value	Range
YESTERDAY	Starts at 12:00:00 AM on the day before the current day and continues for 24 hours.
TODAY	Starts at 12:00:00 AM on the current day and continues for 24 hours.
TOMORROW	Starts at 12:00:00 AM. on the day after the current day and continues for 24 hours.
LAST WEEK	Starts at 12:00:00 AM on the first day of the week before the current week and continues for seven days.
THIS WEEK	Starts at 12:00:00 AM on the first day of the current week and continues for seven days.
NEXT WEEK	Starts at 12:00:00 AM on the first day of the week after the current week and continues for seven days.
LAST *n* WEEKS	Starts at 12:00:00 AM on the first day of the week that started *n* weeks before the current week, and continues up to 11:59 PM on the last day of the week before the current week.
NEXT *n* WEEKS	Starts at 12:00:00 AM on the first day of the week after the current week and continues for *n* times seven days.
n WEEKS AGO	Starts at 12:00:00 AM on the first day of the week that started *n* weeks before the start of the current week and continues for seven days.

Relative Date Value	Range
LAST MONTH	Starts at 12:00:00 AM on the first day of the month before the current month and continues for all the days of that month.
THIS MONTH	Starts at 12:00:00 AM on the first day of the current month and continues for all the days of that month.
NEXT *n* MONTHS	Starts at 12:00:00 AM on the first day of the month after the current month and continues until the end of the *n*th month.
LAST *n* MONTHS	Starts at 12:00:00 AM on the first day of the month that started *n* months before the current month and continues up to 11:59 PM on the last day of the month before the current month.
n MONTHS AGO	Starts at 12:00:00 AM on the first day of the month that started *n* months before the start of the current month and continues for all the days of that month.
NEXT MONTH	Starts at 12:00:00 AM on the first day of the month after the current month and continues for all the days of that month.
LAST n DAYS	Starts at 12:00:00 AM n days before the current day and continues up to the current second. (The range includes today. Using this date value includes records from $n + 1$ days ago up to the current day.) In standard filters, *n* can be 7, 30, 60, 90, or 120.
NEXT n DAYS	For standard date filters, starts at 12:00:00 AM on the day that the report is run and continues for n days. (The range includes today.) In standard filters, *n* can be 7, 30, 60, 90, or 120.
	For custom field filters, starts at 12:00:00 AM on the next day and continues for the next n days. (The range does not include today.)
n DAYS AGO	Starts at 12:00:00 AM on the day *n* days before the current day and continues for 24 hours. (The range does not include today.)
LAST QUARTER	Starts at 12:00:00 AM on the first day of the calendar quarter before the current calendar quarter and continues to the end of that quarter.
THIS QUARTER	Starts at 12:00:00 AM on the first day of the current calendar quarter and continues to the end of the quarter.
NEXT QUARTER	Starts at 12:00:00 AM on the first day of the calendar quarter after the current calendar quarter and continues to the end of that quarter.
LAST n QUARTERS	Starts at 12:00:00 AM on the first day of the calendar quarter n quarters ago and continues to the end of the calendar quarter before the current quarter. (The range does not include the current quarter.)
NEXT n QUARTERS	Starts at 12:00:00 AM on the first day of the calendar quarter after the current quarter and continues to the end of the calendar quarter n quarters in the future. (The range does not include the current quarter.)
n QUARTERS AGO	Starts at 12:00:00 AM on the first day of the calendar quarter n quarters before the current calendar quarter and continues to the end of that quarter.

Relative Date Value	Range
LAST YEAR	Starts at 12:00:00 AM on January 1 of the year before the current year and continues through the end of December 31 of that year.
THIS YEAR	Starts at 12:00:00 AM on January 1 of the current year and continues through the end of December 31 of the current year.
NEXT YEAR	Starts at 12:00:00 AM on January 1 of the year after the current year and continues through the end of December 31 of that year.
n YEARS AGO	Starts at 12:00:00 AM on January 1 of the calendar year *n* years before the current calendar year and continues through the end of December 31 of that year.
LAST *n* YEARS	Starts at 12:00:00 am on January 1, `n+1` years ago. The range ends on December 31 of the year before the current year.
NEXT *n* YEARS	Starts at 12:00:00 AM on January 1 of the year after the current year and continues through the end of December 31 of the *n*th year.
LAST FISCAL QUARTER	Starts at 12:00:00 AM on the first day of the fiscal quarter before the current fiscal quarter and continues through the last day of that fiscal quarter. The fiscal quarter is defined on the Fiscal Year page in Setup. 📝 Note: None of the FISCAL literal date values are supported when creating mobile custom views.
THIS FISCAL QUARTER	Starts at 12:00:00 AM on the first day of the current fiscal quarter and continues through the end of the last day of the current fiscal quarter. The fiscal quarter is defined on the Fiscal Year page in Setup.
NEXT FISCAL QUARTER	Starts at 12:00:00 AM on the first day of the fiscal quarter after the current fiscal quarter and continues through the last day of that fiscal quarter. (The range does not include the current quarter.) The fiscal quarter is defined on the Fiscal Year page in Setup.
LAST *n* FISCAL QUARTERS	Starts at 12:00:00 AM on the first day of the fiscal quarter *n* fiscal quarters ago and continues through the end of the last day of the previous fiscal quarter. (The range does not include the current fiscal quarter.) The fiscal quarter is defined on the Fiscal Year page in Setup.
NEXT *n* FISCAL QUARTERS	Starts at 12:00:00 AM on the first day of the fiscal quarter after the current fiscal quarter and continues through the end of the last day of the *n*th fiscal quarter. (The range does not include the current fiscal quarter.) The fiscal quarter is defined on the Fiscal Year page in Setup.
n FISCAL QUARTERS AGO	Starts at 12:00:00 AM on the first day of the fiscal quarter *n* fiscal quarters before the current fiscal quarter and continues through the end of the last day of that fiscal quarter.
LAST FISCAL YEAR	Starts at 12:00:00 AM on the first day of the fiscal year before the current fiscal year and continues through the end of the last day of that fiscal year. The fiscal quarter is defined on the Fiscal Year page in Setup.

Relative Date Value	Range
THIS FISCAL YEAR	Starts at 12:00:00 AM on the first day of the current fiscal year and continues through the end of the last day of the fiscal year. The fiscal quarter is defined on the Fiscal Year page in Setup.
NEXT FISCAL YEAR	Starts at 12:00:00 AM on the first day of the fiscal year after the current fiscal year and continues through the end of the last day of that fiscal year. The fiscal quarter is defined on the Fiscal Year page in Setup.
LAST n FISCAL YEARS	Starts at 12:00:00 AM on the first day of the fiscal year n fiscal years ago and continues through the end of the last day of the fiscal year before the current fiscal year. (The range does not include the current fiscal year.) The fiscal quarter is defined on the Fiscal Year page in Setup.
NEXT n FISCAL YEARS	Starts at 12:00:00 AM on the first day of the fiscal year after the current fiscal year and continues through the end of the last day of the nth fiscal year. (The range does not include the current fiscal year.) The fiscal quarter is defined on the Fiscal Year page in Setup.
n FISCAL YEARS AGO	Starts at 12:00:00 AM on the first day of the fiscal year n fiscal years ago and continues through the end of the last day of that fiscal year.

Notes about Filtering on Types of Fields and Values

Keep these tips in mind when filtering on text fields, date fields, numeric values, picklist values, and blank or null values.

Filtering on Text Fields

- Separate search terms by commas to filter by more than one value. For example, to search for accounts in California, New York, or Washington, use `State contains CA,NY,WA`.

- Filtering isn't case sensitive. For example, searching `State contains ID` returns all matches for "ID", but also returns any instances of "Florida" and "Idaho" because they contain "id" in their names.

- When you filter on standard long text area fields, such as Description or Solution Details, only the first 1000 characters of the field are searched for matches in reports. Only the first 255 characters are shown for custom long text area fields in list views.

- If your org has **Enable Middle Names for Person Names** turned on, which adds a "Middle Name" field to standard objects like users, leads, and contacts, then you may notice that the "Full Name" field on reports excludes middle names.

  For example, if a user's first, middle, and last name are Nadia Nancy Smith, then in reports the full name appears as "Nadia Smith".

  As a work-around, consider adding a person's middle name to their "First Name" field; "Nadia" would become "Nadia Nancy".

  A missing middle name could cause issues if you are filtering reports on the "Full Name" field, because filtering by `Full Name equals Nadia Nancy Smith` doesn't return the user. Instead, filter the report by `Full Name equals Nadia Smith` or add three filters:

  - `First Name equals Nadia`
  - `Middle Name equals Nancy`
  - `Last Name equals Smith`

  📝 Note: Case reports and Lead reports are affected by the discrepancy between `Full Name` and `Middle Name` described in this article. Other reports may also be affected, but not all reports are.

Filtering on Date Fields

- If entering a date, use the format allowed by your `Locale` setting. You can also use special date values like TODAY, NEXT WEEK, NEXT YEAR, LAST <number> DAYS, and so on.

- The standard Birthdate field allows you to filter the birthdate based only on the month and day, ignoring the year. The year is always included when you filter on any custom date fields.

Filtering on Numeric Values

- Place quotation marks around numbers or other data that includes commas. For example `Amount equals "10,000"` returns records that have an amount of $10,000 but `Amount equals 10,000` returns $10,000 as well as $10 and $0.

- To search for phone numbers, include the exact phone number formatting or example, `Phone starts with (561).`

Filtering on Picklist Values

- When filtering on multi-select picklist fields, use a semicolon between values to specify an exact match.

  For example, selecting the "equals" operator and a semicolon between two values includes records with both values specified, excluding all other values.

- If your organization uses record types, the lookup dialog lists picklist values for all record types. Use the "equals" or "not equal to" operators for these filters.

  > 📝 Note: If you change the label for a picklist value that's used as a filter criterion, the picklist value is automatically removed from the filter criteria. For example, if your report contains a filter where `Lead Source equals Email or Web` and you change the picklist value Web to Referral, your report filter changes to `Lead Source equals Email`. If the changed picklist value was the only value specified for a particular filter, it continues to show up in your filters, but an error appears.

Filtering on Blank or Null Values

- When you use the "less than," "greater than," "less or equal," or "greater or equal" operators on fields that have numeric values, records with blank or "null" values are returned as if their value is zero (0). For example, if you create a workflow rule or a lead assignment rule for accounts with the criteria `Annual Revenue less than 100000`, account records match if their `Annual Revenue` is blank.

  However, records with blank field values are not considered matches in report filters, custom list views, and account assignment rules (which assign accounts to territories).

- To limit results to records that are blank or contain "null" values for a particular field, choose the field and the "equals" or "not equal to" operators, leaving the third field blank. For example, `Amount equals` returns records with blank amount fields. You can search for blank values or other specified values at the same time. For example, `Amount equals 1,,2` returns records where the `Amount` is blank or contains the value "1" or "2".

SEE ALSO:

Tips for Filtering on Multiple Currencies

Tips for filtering on currency fields when your organization uses multiple currencies.

If your organization uses multiple currencies, follow these tips to create more effective filters:

- Use the `Currency` field to find items with a particular currency. For example, `Opportunity Currency equals AUD` finds opportunities with amounts in Australian dollars.

- Prefix currency amounts with a currency code, such as `Annual Revenue greater than USD 50000000`.

  Without the currency code, all amounts are assumed to be in the user's currency. For example, if the user's currency is U.S. dollars, `Annual Revenue greater than 50000000` means 50 million U.S. dollars.

  If the user's currency is invalid, the corporate currency is used.

  From Setup, enter `Manage Currencies` in the `Quick Find` box, then select **Manage Currencies** to obtain the currency codes.

- All amounts are converted to the corporate currency for comparison. For example, `Annual Revenue greater than USD 50000000` finds accounts with revenue greater than 50 million U.S. dollars. This would include an account with revenue of 114 million Australian dollars, which is the equivalent of 60 million U.S. dollars, assuming a conversion rate of 1.9.

- Amounts in reports are shown in their original currencies, and report totals are displayed in your personal currency. You can change the currency used for report totals by clicking **Show** > **Currencies**. For any amount, you can also choose to display the "converted" column (for example, "Annual Revenue (converted)"), which will show amounts in the currency you select from the Display Currencies Using drop-down list.

SEE ALSO:

  Notes about Filtering on Types of Fields and Values

Schedule and Subscribe to Reports

Schedule and Subscribe to Reports and receive notifications that keep you informed about metrics you care most about without having to manually run reports. In Salesforce Classic, you can specify criteria that trigger report notifications.

▶ Watch a video:

- ⊙ Subscribe to Reports and Dashboards (Lightning Experience)
- ⊙ Report Notifications (Salesforce Classic)
- ⊙ Tips for Scheduling Reports (Salesforce Classic)

1. Subscribe Users, Groups, and Roles to Reports in Lightning Experience

   Subscribe yourself and other users, groups, or roles to receive refreshed report results by email on a schedule that you set.

2. Get Notified When Report Conditions Are Met

   Set report conditions so that recipients are notified when a total in your Salesforce report reaches a meaningful threshold.

3. Attach Files to Report Subscriptions

   When subscribing to a report, choose to receive report results as a formatted spreadsheet (`.XLSX`) or a comma-separated (`.CSV`) file attached to the subscription email.

4. Subscribe to Get Report Notifications in Salesforce Classic

In Salesforce Classic, subscribe to a report to receive report notifications periodically when the metrics you care about most meet certain conditions. Set the conditions that trigger notification, and specify if you want to be notified via Salesforce app notifications, Chatter, or email.

5. Schedule Reports in Salesforce Classic

You can set up a report to run itself daily, weekly, or monthly and send the results automatically to the people who need to see them, so that you don't have to remember to log in and do it yourself.

Subscribe Users, Groups, and Roles to Reports in Lightning Experience

Subscribe yourself and other users, groups, or roles to receive refreshed report results by email on a schedule that you set.

▶ Watch a video: ▶ Subscribe to Reports and Dashboards (Lightning Experience)

For example, instead of manually running and emailing your weekly sales report to your team, you can subscribe the team to the report and have it emailed to everyone each Monday morning.

1. From the **Reports** tab or from the report run page, click ▼ > **Subscribe**.

   If you've already subscribed to a report but want to change the schedule, take the same action.

2. From the Edit Subscription menu, set the subscription schedule.

   For example for the weekly sales report, choose a weekly subscription with delivery Monday 8:00 AM.

3. Optionally, add conditions.

   The conditions are evaluated when the report is run according to the schedule you set. The report is only emailed if all conditions are met.

4. Under `Send To`, you are automatically selected as a recipient. To add others or remove yourself, click **Edit Recipients**.

5. Select from the available entity types and start typing to see all the matching names.

   Only the users, groups, or roles with permission to access the report are shown in the list of matches.

6. Select from the matching options and click **Add** to add to the list of subscribers. Add more users, groups, or roles as needed and then close the **Edit Recipients** window.

   When the subscription emails the refreshed report to each recipient, it sends to the email address set in **Settings** > **Email** > **My Email Settings**. If no email is set in **My Email Settings**, then the refreshed report is sent to the recipient's email address set on their Salesforce User record.

   ⊘ **Important:** Recipients see emailed report data as the person running the report. Consider that they may see more or less data than they normally see in Salesforce.

7. Under `Run Report As`, specify who runs the report.

   - **Me** — You run the report, and recipients see report data in the emailed report as you.

   - **Another Person** — Specify someone who has permission to run reports and who has access to the report.

8. Click **Save**.

EDITIONS

Available in: Lightning Experience

Available in: **Essentials**, **Group**, **Platform**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

USER PERMISSIONS

To subscribe to reports:
- Subscribe to Reports

To subscribe other people to reports:
- Subscribe to Reports: Add Recipients

To subscribe other people to reports by groups or roles:
- Subscribe to Reports: Send to Groups and Roles

To specify who runs the report in a report subscription:
- Subscribe to Reports: Set Running User

The report subscription starts, and recipients begin receiving report results by email according to the schedule and conditions you set. For the weekly sales report, everyone who is on the team receives the report, so you don't have to change the setup if people join or leave the team.

Note: As you subscribe to reports, take note of these notes, considerations, and limitations:

- Each user can set up subscriptions for up to 5 reports.
- Subscription recipients aren't listed on the report subscription emails.
- For Platform users, the admin must enable.
- To subscribe a user, group, or role to a report, the folder containing the report must be shared with the user, group, or role.
- Each subscription supports up to 500 recipients. Each recipient is a single user, role, role and subordinates, or group. Roles, roles and subordinates, and groups can each have more than 500 users, but subscriptions send a maximum of 500 emails. If a recipient role, role and subordinates, or group has lots of users, some of them don't receive subscription emails.

  After including all users from roles, roles and subordinates, and groups, if subscriptions have more than 500 users as recipients, users are prioritized over roles, roles are prioritized over roles and subordinates, and roles and subordinates are prioritized over groups. Each time the subscription sends an email, the role and group users who receive the email are chosen again and can be different each time a subscription email sends.

  For example, say that a subscription has 100 recipients: 98 users, 1 role which includes 500 users, and one group which includes 400 users. The total number of users associated with the subscription is 998, so when the subscription email sends, 98 users from the role don't receive subscription emails and none of the 400 users in the group receive emails.

- Each Salesforce organization can schedule up to 500 dashboard subscriptions and 500 report subscriptions on a given hour of a given day, such as Monday at 9:00am.
- Lightning Experience report subscriptions don't support these features:
  - Historical tracking reports
  - Joined reports
  - Conditional highlighting (You can subscribe to reports with conditional highlighting, but conditional highlighting doesn't appear in the subscription email.)

To unsubscribe, open the **Edit Subscription** window by clicking **Subscribe** again. Then, click **Unsubscribe**.

Get Notified When Report Conditions Are Met

Set report conditions so that recipients are notified when a total in your Salesforce report reaches a meaningful threshold.

1. From the **Reports** tab or from the report run page, click ▼ > **Subscribe**.

2. In the **Schedule** area of the setup screen, set the schedule for updating your report.

3. In the **Conditions** area, select **Add conditions to this report**.

4. Select the aggregate to which the threshold will be applied, such as `Record Count`.

5. Select the operator for comparison, such as `Greater Than`.

6. Enter the threshold value, such as `400`.

7. To add additional conditions, click **+ Add Condition**.

   With multiple conditions, the threshold is met if all conditions are met (AND match).

8. To send a summary of the conditions that were met plus the full report, select **Summary + Report**. To send only a summary of the conditions that were met, select **Summary only**.

9. In the **Edit Recipients** select the users, groups, or roles to receive the notifications.

10. Click **Save**.

Each time your report runs, the report conditions are checked. If the threshold conditions are met, email is sent to the specified recipients with a summary of the conditions that were met. If you select **Summary + Report**, a copy of the full report is sent along with a summary of the conditions that were met.

Attach Files to Report Subscriptions

USER PERMISSIONS

To subscribe to reports:	Subscribe to Reports
To subscribe other people to reports:	Subscribe to Reports: Add Recipients
To subscribe other people to reports by groups or roles:	Subscribe to Reports: Send to Groups and Roles
To specify who runs the report in a report subscription:	Subscribe to Reports: Set Running User
To attach files to report subscriptions	Export Reports

When subscribing to a report, choose to receive report results as a formatted spreadsheet (`.XLSX`) or a comma-separated (`.CSV`) file attached to the subscription email.

1. From the **Reports** tab or from the report run page, click ▼ > **Subscribe**.

2. Click **Attach File**.

3. Choose to attach a **Formatted Report** (`.XLSX`) or a **Details Only** (`.CSV`) file.

4. If attaching a details only (`.CSV`) file, optionally choose an encoding.

5. Click **Save**.

The Edit Subscription menu shows that report results are included as an attachment.

6. Click **Save**.

The subscription email arrives with an attached file containing report results.

Limits, Limitations, and Allocations

The attached report can include up to 15,000 rows, 30 columns, and 3 MB file size. Extra data is clipped or not sent. To view the entire report in Salesforce, click the report title in the formatted report email or sign in to Salesforce and search for the report.

SEE ALSO:

Let Users Attach Files to Report Subscriptions (Beta)

Subscribe to Get Report Notifications in Salesforce Classic

In Salesforce Classic, subscribe to a report to receive report notifications periodically when the metrics you care about most meet certain conditions. Set the conditions that trigger notification, and specify if you want to be notified via Salesforce app notifications, Chatter, or email.

For example, you could subscribe to an open-issue report and get notified every morning if there are over 20 open issues. You can subscribe to notifications for up to five reports.

Watch a demo: ▶ Report Notifications (Salesforce Classic)

> 📝 **Note:** Report Notifications doesn't appear as an available option for unsaved reports. Save the report before subscribing to it.
>
> Personal Report Notifications aren't related to the Schedule Future Runs feature, which enables you to email reports at specified times without specifying conditions. To schedule emailed reports, select **Schedule Future Runs** from the Run Report drop-down menu.

1. On the Report Run page, click **Subscribe**.

2. On the Report Subscription page, choose whether to be notified each time conditions are met or only the first time.

3. Specify each condition in three parts: aggregate, operator, value.

For example, trigger notifications whenever the sum of amount is less than $1 million.

- Aggregate is the metric that's the basis of your condition. It can be Record Count, Average Amount, Smallest Amount, Largest Amount, or Sum of Amount.

- Operator is the basis of comparison, such as Equal, Not Equal, Greater Than, and so on.

- Value is the number that you want the aggregate compared to.

Your conditions are evaluated when the report is run, and notifications are sent if all conditions are met (up to five conditions per report).

4. Schedule how often (every weekday, daily, or weekly) and when to evaluate for your conditions.

For example, run the report every weekday at 7 a.m.

5. Select one or more notification types.

- Send a Salesforce in-app notification

- Post to Chatter

- Send an email notification

EDITIONS

Available in: Salesforce Classic

Available in: **Group**, **Platform**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

USER PERMISSIONS

To subscribe to reports in Salesforce Classic:
- Run Reports

To enable or disable Salesforce Classic report notification subscriptions:
- Customize Application

- Execute a custom Apex action, such as creating tasks or escalating cases

  For more information about developing a custom Apex class, see the Apex Developer Guide.

6. Make sure that the subscription is active if you're ready to start receiving notifications.

7. Click **Save** to schedule the notifications.

👁 **Example:** A report is scheduled to be run every weekday at 7 a.m. If the sum of the amount is less than 1,000,000, a Salesforce app notification is sent and a Chatter post is made.

SEE ALSO:

Schedule Reports in Salesforce Classic

Schedule Reports in Salesforce Classic

You can set up a report to run itself daily, weekly, or monthly and send the results automatically to the people who need to see them, so that you don't have to remember to log in and do it yourself.

▶ Tips for Scheduling Reports (Salesforce Classic)

1. Schedule a Report for Refresh

   Schedule a report to run daily, weekly, or monthly. An HTML version of the report can be sent by email to users in your organization.

2. View a Report's Schedule

   View a report's schedule on the Schedule Report page or from the Reports tab. View all report schedules for the organization under Setup.

3. Manage a Report's Schedule

   Create, change, view or delete a scheduled report from the Schedule Report page.

4. Change a Report's Schedule

   You can make changes to an already scheduled report on the Schedule Report page.

5. Delete a Report's Schedule

   Select a scheduled report and unschedule it to delete its scheduled run.

6. Tips on Scheduling Reports

   Some tips to keep in mind about timings, limits, and email notifications when scheduling a report.

SEE ALSO:

    Schedule a Report for Refresh

    Change a Report's Schedule

    Delete a Report's Schedule

    Manage a Report's Schedule

    Tips on Scheduling Reports

    View a Report's Schedule

    Subscribe to Get Report Notifications in Salesforce Classic

---

EDITIONS

Available in: Salesforce Classic (not available in all orgs)

Available in: **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

USER PERMISSIONS

To create, edit, and delete reports:

- **Legacy Folder Sharing**
  Schedule Reports

  Enhanced Folder Sharing
  Schedule Reports

Schedule a Report for Refresh

Schedule a report to run daily, weekly, or monthly. An HTML version of the report can be sent by email to users in your organization.

Watch a Demo: ▶ Tips for Scheduling Reports (Salesforce Classic)

1. On the Reports tab, click a report name.

2. Click **Schedule Future Runs** from the **Run Report** drop-down menu.

   If you're creating a new report, you are asked to save the report in a folder before scheduling. If the report folder is shared with a group, you can schedule the report only for the entire group. To schedule the report for an individual group member, share the report folder with that member.

3. On the Schedule Report page, specify a running user who has access to the folder where the report is stored.

   The access level of the running user determines what other users, including portal users, see when they receive the scheduled report run results. This means that not all users can see every field, and the running user can see fields that others may not be able to see. When it's sent, the report indicates which fields the running user can see that can't be viewed by others. You need the "View All Data" permission to specify a running user other than yourself.

   ✎ Note: If the running user becomes inactive, the report doesn't run. Salesforce sends an email notification to either activate the user, delete the report schedule, or change the running user to an active one. Salesforce sends the notification to users with the "Manage Users," "Modify All Data," and "Manage Billing" permissions. If no user has all these user permissions, Salesforce sends the notification to users with the "Manage Users" and "Modify All Data" user permissions.

4. Select an email setting.

Select	To
To me	Send the report to your email address specified on your user profile.
To me and/or others	Email the report to additional users.

You can send reports only to email addresses included on Salesforce user records. When portal users receive emailed reports, they see the same data as the running user set in the report schedule. If you have information you'd rather not share, schedule the report to run with a portal user as the running user.

✎ Note: Portal users receive report and dashboard refresh email notifications when the `Allow Reports and Dashboards to Be Sent to Portal Users` option is enabled.

5. Set the frequency, duration, and time for running the report:

   - In the `Frequency` field, select `Daily`, `Weekly`, or `Monthly` and then refine the frequency criteria.
   - Using the `Start` and `End` fields, specify the dates during which you want to schedule the report. To enter the current date, click the link showing the date.
   - Next to `Preferred Start Time`, click **Find available options** to choose a start time.

   Your preferred start time might not be available if other users have already selected that time to schedule a report.

6. Click **Save Report Schedule**. You can choose:

Choose	To
Save report modifications with this schedule	Save both the report schedule and changes you made to the report.
Discard report modifications	Save the schedule only. Changes you made to the report are discarded.

Report recipients can click the report name in emailed reports to log in to Salesforce and view the report directly.

👁 Example: **Example**

A report is scheduled to run every Friday at midnight, and its results are emailed to a selected group and user.

1. All users, including portal users, viewing the scheduled report see the report data that Sales Director Chet's access level allows.

2. Report run results are set to be emailed to a public user group called All Internal Users and the admin user. You can only send emails to users and groups with access to the report folders. The `Search` drop-down displays all available categories based on your search criteria in the `Running User` field.

3. The report is scheduled to run every Friday.

4. The report run is scheduled to start on the current date.

5. The schedule is saved without saving prior changes made to the report.

SEE ALSO:

Manage a Report's Schedule

View a Report's Schedule

View a report's schedule on the Schedule Report page or from the Reports tab. View all report schedules for the organization under Setup.

▶ Tips for Scheduling Reports (Salesforce Classic)

- To see the schedule for a report on the Reports tab, hover over ✔ in the schedule 🖻 column. This shows the frequency and the date of the next run.

  Users without the "Schedule Reports" permission can't see the icon and information.

- To see a report's run schedule on the Schedule Report page:

  1. Click a scheduled report name on the Reports tab.

  2. Click **Schedule Future Runs** from the **Run Report** drop-down menu..

- To see all scheduled reports for your organization, from Setup, enter `Scheduled Jobs` in the `Quick Find` box, then select **Scheduled Jobs**.

  Only users with the "View Setup and Configuration" permission can view this information.

SEE ALSO:

Manage a Report's Schedule

Manage a Report's Schedule

Create, change, view or delete a scheduled report from the Schedule Report page.

▶ Tips for Scheduling Reports (Salesforce Classic)

On the Schedule Report page you can:

- Schedule a new or existing report to run in the future and have its results emailed to others.
- Change the schedule on a previously scheduled report.
- View scheduled jobs for all reports in your organization or view the schedule for just a selected report.
- Delete a scheduled run for a selected report.

Additional scheduled reports may be available for purchase.

SEE ALSO:

Schedule a Report for Refresh

Change a Report's Schedule

You can make changes to an already scheduled report on the Schedule Report page.

▶ Tips for Scheduling Reports (Salesforce Classic)

1. On the Reports tab, click the name of the scheduled report.

2. Click **Schedule Future Runs** from the **Run Report** drop-down menu.

3. Make the required changes on the Schedule Report page.

4. Click **Save Report Schedule**.

SEE ALSO:

Manage a Report's Schedule

Delete a Report's Schedule

Select a scheduled report and unschedule it to delete its scheduled run.

⏺ Tips for Scheduling Reports (Salesforce Classic)

1. On the Reports tab, click the name of the scheduled report.

2. Click **Schedule Future Runs** from the **Run Report** drop-down menu.

3. Click **Unschedule Report**.

The run schedule for the report is canceled and *not* sent to the Recycle Bin.

SEE ALSO:

Manage a Report's Schedule

Tips on Scheduling Reports

Some tips to keep in mind about timings, limits, and email notifications when scheduling a report.

Scheduling Report Runs

- On the Reports tab, hover over ✔ in the schedule column (🖫) to view a report's schedule. Note that users without the "Schedule Reports" permission can't see this icon and information.
- You can't create schedules for joined reports.
- Your organization is limited to no more than 200 scheduled reports. Daily limits differ by edition. Additional scheduled reports may be available for purchase.
- Scheduled reports run in the time zone of the user who set up the schedule. For example, if the `Time Zone` field on your user record is set to Pacific Standard Time, and you schedule a report to run every day at 2:00 PM, then the report runs every day between 2:00 PM and 2:29 PM Pacific Standard Time.
- If you view and save a schedule in a time zone different from the one in which it was previously scheduled, the time slot could potentially change.
- If you schedule a report to run on a specific day of every month, the report runs only on months that have that specific day. For example, if you schedule a report to run on the 31st day of every month, then the report runs only on months that have 31 days. To schedule a report on the last day of every month, choose last from the `On day of every month` drop-down list.

- The report runs within 30 minutes of the time you select for `Preferred Start Time`. For example, if you select 2:00 PM as your preferred start time, the report runs any time between 2:00 PM and 2:29 PM, depending on how many other reports are scheduled at that time.

- Scheduling reports is not tracked in the audit trail history.

Emailing Scheduled Reports

- You can't email scheduled reports to Chatter Free users.

  > **Note:** Previously, Chatter Free users could receive scheduled reports by email, even though they can't access the report in Salesforce. Scheduled reports that were set up to email Chatter Free users continue to email to Chatter Free users.

- Emailed reports don't include report charts. To email a chart of the report, create a dashboard and schedule a dashboard refresh.

- Reports display only the first 2,000 records (same as in a browser).

- Outlook 2007 limitation: Report emails containing tables more than 22 inches wide or with more than 63 columns might not display properly.

- The maximum size for emailed reports is 10 MB. Try the following techniques to reduce the amount of data in your report:

  - Filter for your own records, rather than all records.

  - Limit the scope of the data to a specific date range.

  - Exclude unnecessary columns from your report.

  - Hide the report details.

SEE ALSO:

Schedule a Dashboard Refresh in Salesforce Classic

Manage a Report's Schedule

Export and Connect Reports to Other Tools

Export or connect a report to another tool, such as Quip, to work with report data outside of Salesforce.

Export a Report

To work with report data in a dedicated tool, such as a spreadsheet, export report data as a Microsoft Excel (`.xlsx` or `.xls`) file or comma-separated values (`.csv`) file.

Salesforce Reports in Quip

Work with connected report data in a document to collaborate on live Salesforce data in context. @Mention colleagues to discuss data inline or in the document conversations pane. Update filter values from Quip to customize your Salesforce report views without altering the base report in Salesforce.

Report on Salesforce Data with Excel

Connect for Office includes an Excel add-in that enables you to securely access your Salesforce reports with Microsoft® Excel®. You create the reports you need in Salesforce, then pull them into an Excel worksheet, and use Excel's formulas, charts, and pivot tables to customize and analyze your data. When Salesforce disables TLS 1.0, we're ending support for Connect for Office.

> EDITIONS
>
> Available in: Salesforce Classic (not available in all orgs) and Lightning Experience
>
> Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
>
> Available in: Enhanced Folder Sharing and Legacy Folder Sharing

Export a Report

To work with report data in a dedicated tool, such as a spreadsheet, export report data as a Microsoft Excel (`.xlsx` or `.xls`) file or comma-separated values (`.csv`) file.

Export a Report in Lightning Experience

1. From Reports, next to the report you want to export, click ▼ > **Export**.

2. Choose an Export View.

   Formatted Report exports the report as it appears in Salesforce, with the report header, groupings, and filter details. Because formatting is retained, you can't choose an encoding and the only supported formatted export file type is `.xlsx`.

   Details Only exports each detail row without formatting and is useful for doing further calculations in a spreadsheet.

3. If exporting as **Details Only**, select **Excel Format .xlsx** , **Excel Format .xls**, or **Comma Delimited .csv** as the format. For **.xls** and **.csv**, select the encoding option appropriate for your language and locale.

4. Click **Export**.

5. If prompted by a browser dialog, select a location and save the file.

Export a Report in Salesforce Classic

1. From Reports, next to the report you want to export, click ▼ > **Export**.

   If you set the "Don't save encrypted pages to disk" option in Internet Explorer, you can't open your report online in Excel. Instead, save the exported report to your computer, and then open it in Excel. To change this setting in Internet Explorer, deselect "Don't save encrypted pages to disk" under Internet options.

2. Set an `Encoding` appropriate for your language.

3. Set the `Format` to **Excel Format .xlsx** , **Excel Format .xls** , or **Comma Delimited .csv**.

4. Click **Export**.

5. If prompted by a browser dialog, select a location and save the file.

Your exported report downloads.

As you get ready to export report data, take note of these limits and considerations.

Limits

When exporting a report in Salesforce Classic or in Lightning Experience as Details Only, you can export an unlimited number of report rows and columns. Most spreadsheet tools have a per-file row and column limit. Common tool-imposed limits are 1,048,576 rows by 16,384 columns, or 65,536 rows by 256 columns.

When exporting a report in Lightning Experience as Formatted Report, you can export up to 100,000 rows and 100 columns.

When you export a report with a date/time field using the **Formatted Report** option, the field in the exported report shows only the date, not the time. To include the time information in the formatted export, add a row level formula field or custom formula field.

Sometimes reports with lots of rows and columns can take a few minutes to export. If a report takes 10 minutes to export, then the reports export times out and fails. If a report export times out, create a copy of the report. Filter both reports so that they each return half the data of the original. Then export each report.

Each user can export up to 5 reports at once. Formatted report exports from Lightning Experience don't count against this limit.

Considerations

When exporting reports in the comma-separated values (`.csv`) format, the locale settings on your user detail page determine the field separator (delimiter) included in the exported file. For example, if your locale setting is English (United States), then the decimal separator is a period ("."), If your locale setting is French (France), then the decimal separator is a comma (","). You can override the default separator for your locale by choosing Comma Delimited (non-locale) `.csv` from the `Format` dropdown list.

Excel doesn't display the field separator in `.csv` format. If you want to work with `.csv` files in Excel, we recommend that your locale setting in Salesforce match your Regional Options setting in Windows.

Joined reports always export as **Formatted Report**.

Joined report exports can include a maximum of 2000 rows. Additional rows are clipped.

SEE ALSO:

Attach Files to Report Subscriptions

Run Reports in the Background

Print a Report

Salesforce Reports in Quip

Work with connected report data in a document to collaborate on live Salesforce data in context. @Mention colleagues to discuss data inline or in the document conversations pane. Update filter values from Quip to customize your Salesforce report views without altering the base report in Salesforce.

Apply your Salesforce report's custom filters to tailor your report view in Quip. Use filters to sort your reps' open opportunities by region or track your reps' progress. You can even dynamically filter live Salesforce reports added to a document based on the record it's embedded in.

To add a Salesforce report to a document, enter `@Salesforce Report` and select your report. For Salesforce Classic users, click **Open in Quip** from the report in Salesforce to open your report in a new document.

EDITIONS

Available in: Lightning Experience

Available in: **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Report on Salesforce Data with Excel

Connect for Office includes an Excel add-in that enables you to securely access your Salesforce reports with Microsoft® Excel®. You create the reports you need in Salesforce, then pull them into an Excel worksheet, and use Excel's formulas, charts, and pivot tables to customize and analyze your data. When Salesforce disables TLS 1.0, we're ending support for Connect for Office.

🛇 **Important:** The Microsoft® Excel Add-On, the Microsoft® Word Add-On, and the standard Mail Merge feature will be retired in February 2019. If your sales reps still use these features, check out the linked product retirement articles for more details.

The Excel add-in provides the same access to reports and fields that you normally experience in Salesforce. You can distribute your customized Excel worksheets via the Documents tab, allowing all users to track customized analytics in real time. You can reference data from multiple reports in one worksheet to create a single-page overview of key metrics.

Communication between Excel and Salesforce uses the same secure HTTPS protocol as when you log in via your Web browser.

1. Install Connect for Office

    When Salesforce disables TLS 1.0, we're ending support for Connect for Office.

2. Log Into Connect for Office

    You need to log in to Salesforce before you can request data from your Salesforce reports. When Salesforce disables TLS 1.0, we're ending support for Connect for Office.

3. Import Reports Into Excel with Connect for Office

    Import your custom or standard Salesforce reports into Excel so you can further analyze the data using Excel's formulas, charts, and pivot tables. When Salesforce disables TLS 1.0, we're ending support for Connect for Office.

EDITIONS

Available in: Salesforce Classic (not available in all orgs)

Available in: **All** Editions

USER PERMISSIONS

To access Salesforce reports from Excel:
*   Run Reports

    AND

    Export Reports

4. Refresh and Update Data with Connect for Office

Keep your Salesforce reports up to date in Excel by periodically refreshing the report data and any pivot tables you have created. When Salesforce disables TLS 1.0, we're ending support for Connect for Office.

Install Connect for Office

When Salesforce disables TLS 1.0, we're ending support for Connect for Office.

🛑 **Important:** The Microsoft® Excel Add-On, the Microsoft® Word Add-On, and the standard Mail Merge feature will be retired in February 2019. If your sales reps still use these features, check out the linked product retirement articles for more details.

The system requirements for Connect for Office are:

- Microsoft® Excel 2007 to use the Excel add-in. (The Word add-in is not supported for Microsoft Word 2007.)
- Microsoft® Windows Vista® (32-bit only) - Until Salesforce disables TLS 1.0.

1. Close all Microsoft® Office programs, including Word, Excel®, and Outlook®.

2. From your personal settings, enter `Office` in the `Quick Find` box, then select **Connect for Office**.

   💡 **Tip:** If you can't see the download page, ask your administrator for access.

3. Click **Install Now**.

4. Click **Yes** when prompted to install Connect for Office. We recommend that you install Connect for Office to the default folder suggested by the installer.

5. After the installation completes, open Excel or Word, and select the **Salesforce** menu to begin using Connect for Office.

6. The first time you open Word, you are prompted to enable macros from Salesforce. You must enable the macros and accept Salesforce as a macro publisher to use the Word add-in.

📝 **Note:** The Connect for Office installer edits the registry on your computer. If your organization imposes security that prevents you from editing the registry, log in as the administrator of your machine before installing Connect for Office or contact your IT department for assistance.

SEE ALSO:

Personalize Your Salesforce Experience

Log Into Connect for Office

You need to log in to Salesforce before you can request data from your Salesforce reports. When Salesforce disables TLS 1.0, we're ending support for Connect for Office.

🛑 **Important:** The Microsoft® Excel Add-On, the Microsoft® Word Add-On, and the standard Mail Merge feature will be retired in February 2019. If your sales reps still use these features, check out the linked product retirement articles for more details.

If your organization restricts IP addresses, logins from untrusted IPs are blocked until they're activated. Salesforce automatically sends you an activation email that you can use to log in. The email contains a security token that you add to the end of your password. For example, if your password is *mypassword*, and your security token is *XXXXXXXXXX*, you must enter *mypasswordXXXXXXXXXX* to log in.

1. Open Excel.

2. In Microsoft Office 2003 and earlier, select **Log In** from the **Salesforce** drop-down menu on the Excel toolbar. In Microsoft Office 2007, select the Salesforce tab on the Ribbon, click the **Reporting** drop-down menu, and then select **Log In**.

3. Enter your Salesforce username and password.

4. Click **Login**.

Import Reports Into Excel with Connect for Office

Import your custom or standard Salesforce reports into Excel so you can further analyze the data using Excel's formulas, charts, and pivot tables. When Salesforce disables TLS 1.0, we're ending support for Connect for Office.

> ⊘ **Important:** The Microsoft® Excel Add-On, the Microsoft® Word Add-On, and the standard Mail Merge feature will be retired in February 2019. If your sales reps still use these features, check out the linked product retirement articles for more details.

1. Create a custom report in Salesforce. You can also use any of the standard reports.

2. Open a blank worksheet in Excel.

3. Select the Salesforce tab on the Ribbon, click the **Reporting** drop-down menu, and then select **Import a Report...**.

> 📝 **Note:** In Microsoft Office 2003 and earlier, select **Import a Report...** from the **Salesforce** drop-down menu on the Excel toolbar.

4. Select a report from the list of standard and custom Salesforce reports available to you.

5. Specify where you want to put the report data in your Excel file.

   a. Enter the name of your Excel worksheet in the `Destination worksheet` field.

   b. In the `Cell` field, enter the uppermost cell where you want to begin putting the data.

   If the specified worksheet and cells already contain report data, Connect for Office moves the existing data over to make room for the new report data.

   > 💡 **Tip:** Avoid renaming worksheets that contain imported reports. When you do that the connection between the worksheet and your report is lost. You must import the report again to refresh the data.

6. Choose **Raw Data** to import the data without formatting, subtotals, or grand totals. Choose **Formatted** to keep the colors, fonts, subtotals, and grand totals from the Salesforce report.

   This is useful for importing large matrix reports with the data already summarized into a small table.

   > 💡 **Tip:**
   > - The **Raw Data** option is best if you're importing summary or matrix reports for use with Excel formulas and pivot tables.
   > - Use the **Formatted** option if you're importing large matrix reports with the data already summarized into a small table.

7. Click **OK**.

> 💡 **Tip:** You can copy and paste data from Excel into other Office applications. Use the **Paste Special** option, rather than **Paste**, to reference the Excel data as a worksheet object. If the data then changes in Excel, you can right-click the object and update it automatically. See the Microsoft Word help for more information.

Refresh and Update Data with Connect for Office

Keep your Salesforce reports up to date in Excel by periodically refreshing the report data and any pivot tables you have created. When Salesforce disables TLS 1.0, we're ending support for Connect for Office.

🛑 **Important:** The Microsoft® Excel Add-On, the Microsoft® Word Add-On, and the standard Mail Merge feature will be retired in February 2019. If your sales reps still use these features, check out the linked product retirement articles for more details.

After logging into Salesforce, select any of the following options from the **Salesforce** Ribbon tab (or toolbar in Office 2003 and earlier):

- **Refresh Existing Reports...** - Allows you to choose which reports you want to update in Excel.

  1. From the list of reports you have imported, select the reports to update.

  2. Optionally, select **Update Pivot Tables** to update any pivot tables you have created in Excel for the selected reports. The **Refresh All Reports** menu choice does this automatically.

  3. Click **Refresh Selected** to update the report data.

     To remove reports from this list, select the report names and click **Delete Selected**. The reports are not removed from your Excel worksheet or from Salesforce, only from the list of reports available for refreshing.

- **Refresh All Reports** - Refreshes all of the reports that you have imported into your Excel file, including the pivot tables referenced by those reports.

💡 Tip:

  - If you write a formula, select an entire column rather than a range of cells, because the number of rows in your report may change when you refresh the report data. For example, use `=Sum(Sheet2!E:E)` to sum column E rather than `=Sum(Sheet2!E1:E200)`.

  - If you use the `VLOOKUP` and `HLOOKUP` functions in Excel to join data across different cell ranges, these functions may make report record IDs, which are 15-character alphanumeric IDs, case-sensitive. Make sure to use the correct case when identifying report records. See the Microsoft Excel help for more information.

Drill Down into Your Reports to Learn Even More

Drill-down helps you take a closer look at records in a report. For example, as a sales manager, drill-down can help you track the progress of just a few of your reps or review the breakdown of current opportunities based on type.

The drill-down feature is available on the reports run page for all reports with row groupings (summary and matrix reports). When you drill down into a grouped column, the report filters by that column. You can also modify the primary report grouping when you select drill-down options.

To drill down into a report:

1. Run a report that includes at least one-row grouping.

2. On the report run page, select checkboxes for the particular values that you want to drill in to. You can only drill in to values from the primary row grouping.

3. Click **Drill Down**.

   📝 Note: The number on the **Drill Down** button indicates the number of values you selected.

4. The primary row group for your report is preselected in the **Group by** field. To change the grouping, click **X** and select a different field.

5. Click **Apply**.

The report now displays data only for the values and grouping that you specified. If you edit the report and open the **Filters** tab, you can see that a filter was added for the values you selected. To revert to the original report, or to select another value to filter by, delete the filter that was added.

EDITIONS

Available in: Lightning Experience

Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

USER PERMISSIONS

To drill down into a report from the reports run page:
- Run Reports

To specify settings for the report:
- Create and Customize Reports

Organize Reports

Keep your reports at your fingertips by sorting them into folders and deleting unused reports. If you have a lot of reports, you can use the search field to find the one you need.

1. Search for Reports and Dashboards in Lightning Experience

   When using the report and dashboard search in Lightning Experience, look for a report by name. To refine your results, sort or search within a selected folder.

2. Search for Reports and Dashboards from the Reports Tab in Salesforce Classic

   When using the report and dashboard search, look for a report by name, description, or who created it or modified it last. Filter, sort, or search within a selected folder to refine your results.

3. Get the Information You Need From the Reports Tab List View in Salesforce Classic

   To see the information you want to see about your reports, you can resize, hide, reorder, sort columns, and select the number of records to display in your list view on the Reports tab.

4. Customize Report and Dashboard Lists in Lightning Experience

   You can customize the columns in the list on the Reports or Dashboards tab.

5. Rename a Report

   What's in a name? A report by any other name still answers your business questions. When it's time to change the name of a report, do so from report properties.

6. Describe a Report

   Give your report a description to help people (and yourself) distinguish it.

7. Print a Report

   Print a report from the run page of a report using your browser's print function.

8. Keep Favorite Report Folders In View

   Pin your most-used report and dashboard folders to the top of the folder list so you don't have to scroll down every time you need them.

9. Hide Unused Report Types

   If your users use only a subset of the available standard report types, you can hide the ones they don't need.

10. Run Reports in the Background

    Run large reports in the background so you can keep working in Salesforce without waiting for results to display. This is a good way to run data-intensive reports that might otherwise time out due to the large number of report results.

11. Deliver Your Report

    To get the information in your report to the people who need it, you can share the report's URL, make the report available for Chatter feeds, or export the data to another tool, such as Excel. You can also set the report to run on a schedule so that viewers always have the latest information.

12. Embedded Report Charts

    Typically, users have had to navigate to the Reports tab to find data. But you can give them valuable information directly on the pages they visit often. To do that, embed report charts in detail pages for objects. When users see charts on pages, they are empowered to make decisions based on the data they see in the context of the page without going elsewhere. For example, an opportunity record shows important data directly on its detail page.

13. Delete a Report

    You can delete a selected report from the Reports tab or from its run page.

Search for Reports and Dashboards in Lightning Experience

When using the report and dashboard search in Lightning Experience, look for a report by name. To refine your results, sort or search within a selected folder.

Search doesn't find partial strings within a longer word or number except at the start of the word or number. A word or number starts at the beginning of the field label or following a space. For example:

- Searching on "commun" finds "Edge Communications" but searching on "dge" or "cation" doesn't.
- Searching on "12" finds "Leads Report 12" but not "Leads Report-12".

Search for Reports and Dashboards from the Reports Tab in Salesforce Classic

When using the report and dashboard search, look for a report by name, description, or who created it or modified it last. Filter, sort, or search within a selected folder to refine your results.

1. On the Reports tab, type in the search box below **All Folders**.

   Search looks up all folders. Filters are set to `All Items` and `All Types` to display all the folder's contents.

   Salesforce searches these fields: **Name**, **Description**, **Last Modified By**, or **Created By**. Results are based on an exact match of what you typed with those fields.

   If tagging is enabled and added to a custom report or dashboard, search by its tag in the global search box at the top of every page.

2. Refine results using these options:

To refine results by	Do this
Type	Select a filter: • `All Types` • `Reports` • `Dashboards` Type filters don't apply to report templates in standard report folders.
View	Select a filter: • `All Items` • `Recently Viewed` • `Items I'm Following` • `Items I Created` • `Items I'm Subscribed to` View filters.

To refine results by	Do this
A selected folder	Select a folder, then type in the search box. Filters are set to *All Items* and *All Types* to display all the folder's contents.
Sorting	Click ▼ for the column and select **Sort Ascending** or **Sort Descending**. Sorting applies to all items in the list view, including the ones on subsequent pages.

Get the Information You Need From the Reports Tab List View in Salesforce Classic

To see the information you want to see about your reports, you can resize, hide, reorder, sort columns, and select the number of records to display in your list view on the Reports tab.

- Customize your list view as follows:

Option	Description
To resize a column	Click and drag its right margin to the preferred size.
To hide a column	Click ▼ > **Columns** on any column and deselect the column you want to hide.
To reorder a column	Drag it to where you'd like it to appear.
To sort a column	Click ▼ for the column and select **Sort Ascending** or **Sort Descending**. Sorting applies to all items in the list view, including the ones on subsequent pages.
To change the number of records displayed per page	Click ▼ in the lower left corner of the list and select the desired setting. You can view 10, 25, 50, 100, or 200 records at a time. When you change this setting, you return to the first page of list results. Once set, however, you view the same number of records throughout the list. The record display setting on the Reports tab doesn't affect list views elsewhere in Salesforce.

Select Column Header for Sorting and Hiding Columns
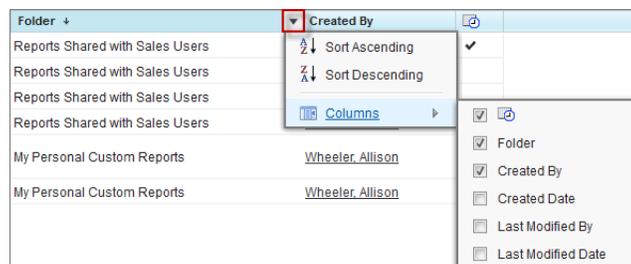
Customize Report and Dashboard Lists in Lightning Experience

You can customize the columns in the list on the Reports or Dashboards tab.

Report Last Run Date

Report Last Run Date is now included as a default column on the Reports tab, so you can easily determine when each report was last run.

Sort on a Selected Column

To see the current sort order for a column, move your cursor to the column header. An up arrow indicates A to Z or lowest to highest ordering; a down arrow indicates the reverse. To change the sort order for a column, click the column header.

Resize Columns

You can resize columns in either of the following ways:

- Move your cursor to the column divider and drag the divider to the left or right.
- Click the gear icon and choose **Select Fields to Display**. Then move your cursor to the column divider and drag the divider to the left or right. Your changes are retained if you leave the page and return later.

> 📝 Note: If you have resized the columns in the Reports or Dashboards list, you might need to scroll to the right to see the Actions arrow ▼ .

Wrap or Clip Column Text

Some columns have an option to choose how content is displayed if it extends beyond the column width. Click the down arrow in the column header and select one of the following options:

- **WRAP TEXT**. Continue text on additional lines as needed so all content is shown.
- **CLIP TEXT**. Display only the content that fits in the column.

Select Columns to Display

Click the gear icon and choose **Select Fields to Display**.

Select columns on the left and click the right-facing arrow to move them to the **Visible Fields** area on the right. To hide a field, select it in the **Visible Fields** area and click the left-facing arrow to move it to the **Available Fields** area on the left.

Click **Save** to save the changes and redisplay the Reports or Dashboards list. Your changes are retained if you leave the page and return later.

EDITIONS

Available in: Lightning Experience

Available in: **Essentials**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

Rename a Report

What's in a name? A report by any other name still answers your business questions. When it's time to change the name of a report, do so from report properties.

1. Edit a report.

2. Open the properties menu.

   In Lightning Experience, click ▼ > **Properties**.
   In Salesforce Classic, click **Report Properties**.

3. Under Report Name, enter a new name for the report.

4. Click **Save**.

The report is renamed.

Describe a Report

Give your report a description to help people (and yourself) distinguish it.

1. Edit a report.

2. Open the properties menu.

   In Lightning Experience, click ▼ > **Properties**.

   In Salesforce Classic, click **Report Properties**.

3. Under Report Description, give your report a helpful description.

4. Click **Save**.

The report's description is updated.

Print a Report

Print a report from the run page of a report using your browser's print function.

- Click **Printable View** from the report's run page.

- From the browser dialog, do one of the following:

  - Open the report with your browser and use your browser's print function.

  - Save the file in Excel and use the print option in Excel.

SEE ALSO:

[Export a Report](#)

Keep Favorite Report Folders In View

Pin your most-used report and dashboard folders to the top of the folder list so you don't have to scroll down every time you need them.

1. In the list of report and dashboard folders, hover over any folder, then click 📌.

Reports & Dashboards
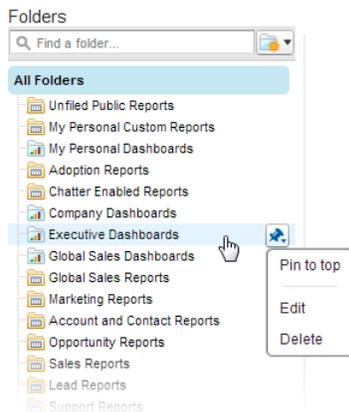
2. Select **Pin to top**.

    Your folder moves to the top of the folders list.

Each new folder that you pin goes to the top of the list, above any folders you have already pinned. To move a pinned folder back to the top, just pin it again.

Hide Unused Report Types

If your users use only a subset of the available standard report types, you can hide the ones they don't need.

1. On the Reports tab, click **New Report**.

2. Select **Select Report Types to Hide**.

    The green check mark next to a report means it's visible to everyone.

3. To hide the report type, click the check mark to change it to an X.
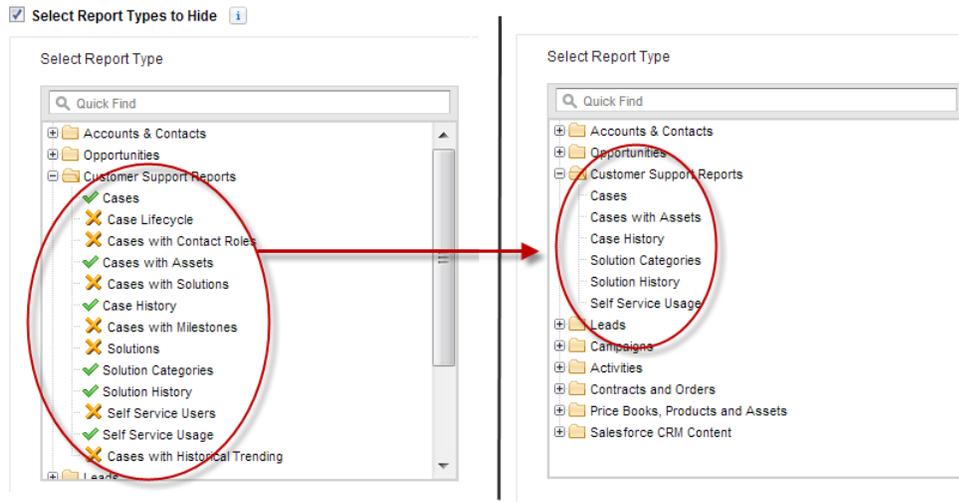
Hidden report types don't show up when you use the search box on the Create New Report page.

If you hide all the report types in a folder, the folder is also hidden. However, if you later unhide the report type in the "Select Report Types to Hide" dialog box, users can see the folder.

Run Reports in the Background

Run large reports in the background so you can keep working in Salesforce without waiting for results to display. This is a good way to run data-intensive reports that might otherwise time out due to the large number of report results.

Reports exported to the background run sequentially, one at a time. Therefore, the latest report exported to the background runs after all reports previously exported to the background finish running.

You can export an unlimited number of reports to the background.

📝 **Note:** Running reports in the background is only supported in the Report Wizard. The Report Builder doesn't support running reports in the background.

1. Export a Report to Run in the Background

   To set up a report to run in the background, you have to export it in a special way.

2. View and Manage Background Reports

   Once you've set up a report to run in the background, you can view and manage it from the background report exports list.

SEE ALSO:

Create a Custom Report in Accessibility Mode

Create a Report

Customizing Reports

Export a Report

Export a Report to Run in the Background

To set up a report to run in the background, you have to export it in a special way.

📝 **Note:** Running reports in the background is only supported in the Report Wizard. The Report Builder doesn't support running reports in the background.

1. Create a report in the Report Wizard.

2. Click **Export Details** and choose **Run Background Export** from the drop-down button.

   💡 **Tip:** When running an existing report, click this button to avoid waiting for report results to display.

3. Select a file encoding setting and a file format in which to export the report.

4. Choose how to save the report parameters.

   - Click `Save revisions` to save any parameter changes you made to the report before exporting. Selecting this option overwrites the parameters of the original report.

   - Click `Save revisions to a copy` to create a new report before exporting. If you select this option, enter a name and description for the report, and choose a report folder in which to save the report parameters.

   📝 **Note:** Only parameters are saved, not results. To view report results, you must view report details from the background report exports list.

5. Click **Start Background Export**.

When your report has finished running and its results are ready for viewing, a link to the report details is emailed to you. You can download report results from the report exports list. You can only download data that you have permission to view.

📝 **Note:** If commas aren't appropriate for your locale, use a tab or other delimiter. Specify your delimiter in Data Loader Settings (**Settings** | **Settings**).

In the unlikely event that a background export of a report fails, an email notification is sent to you. Reports exported to the background can fail for a number of reasons. For example, between the time you exported the report and the export process began, fields may have been deleted from the report, the report may have been deleted, or the status of the custom report type from which the report was created may have been changed to "In Development."

View and Manage Background Reports

Once you've set up a report to run in the background, you can view and manage it from the background report exports list.

📝 **Note:** Running reports in the background is only supported in the Report Wizard. The Report Builder doesn't support running reports in the background.

1. From Setup, enter `Background Report Exports` in the `Quick Find` box, then select **Background Report Exports**. The background report exports list appears.

2. View the details of a report exported to the background, such as its name, status, and location, by clicking the name of a report in the `Job Name` column.

3. Click **Cancel** or **Del** next to the name of a report to cancel a pending export or to delete a report from the background report exports list.

When a report exported to the background finishes running, it is available for viewing in the background report exports list for 48 hours. After 48 hours, the report is automatically deleted.

> ⊘ **Important:** Reports deleted from the background report exports list are permanently deleted and *not* sent to the Recycle Bin.

4. Click **Refresh List** to view any reports that have been exported to the background since you began viewing the background report exports list.

Deliver Your Report

To get the information in your report to the people who need it, you can share the report's URL, make the report available for Chatter feeds, or export the data to another tool, such as Excel. You can also set the report to run on a schedule so that viewers always have the latest information.

Administrators, or users with the "Manage Public Reports" and "Create and Customize Reports" permissions, can create custom reports that all users can view. They can also organize reports by creating custom report folders and configuring which groups of users have access to them.

To make a report public, run the report and click **Save As**. Give the report a name and choose a public report folder.

SEE ALSO:

Embedded Report Charts

Typically, users have had to navigate to the Reports tab to find data. But you can give them valuable information directly on the pages they visit often. To do that, embed report charts in detail pages for objects. When users see charts on pages, they are empowered to make decisions based on the data they see in the context of the page without going elsewhere. For example, an opportunity record shows important data directly on its detail page.

1. Add a Report Chart to a Page Layout

   To embed a report chart on object pages, edit the object's page layout with the enhanced page layout editor, then add the chart.

2. Customizing a Report Chart in a Page Layout

   After you add a report chart to a page, you can customize it in the Chart Properties dialog box on the page layout editor.

3. Example of Report Charts on an Account Page

   In this example, we've embedded two report charts on an important account page that show deals in the pipeline and open support cases for the account. From looking at the charts, the account executive can quickly gauge the account's activity and health.

4. Limits on Report Charts in Pages

   Consider these limits when embedding charts in detail pages.

Add a Report Chart to a Page Layout

To embed a report chart on object pages, edit the object's page layout with the enhanced page layout editor, then add the chart.

Before you add a chart, check that:

- Its source report is in a folder that's shared with users who need access. Reports in personal report folders are unavailable to add to a page.

- The source report format is summary or matrix.

- The source report has a chart.

1. Go to the page layout editor for the object that you're adding a chart to.

2. Click **Edit** next to the page layout.

3. Click **Report Charts**.
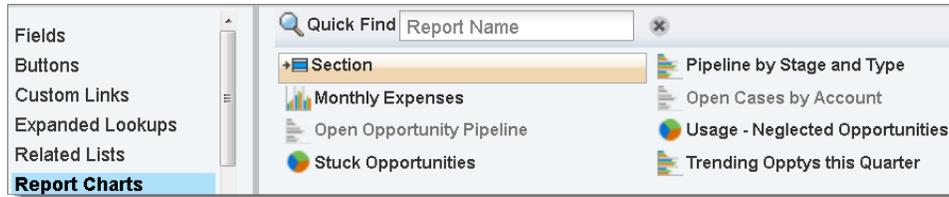
4. In the Quick Find box, type the name of the report and click to find and select the report chart.

   You can browse up to 200 recently viewed reports by chart type in the Report Charts palette.

5. Drag the chart to a new or existing section of the layout.

6. Click on the chart to customize it.

7. Click **Save**.

   The **Preview As** option is unavailable for report charts.

SEE ALSO:

Customizing a Report Chart in a Page Layout

Example of Report Charts on an Account Page

Limits on Report Charts in Pages

Find Object Management Settings

Page Layouts

Customizing a Report Chart in a Page Layout

After you add a report chart to a page, you can customize it in the Chart Properties dialog box on the page layout editor.

1. Hiding a Report Chart that Shows an Error

   A chart can sometimes, for whatever reason, show an error on the detail page instead of data. Since a chart with an error is not that useful, you have the option to hide it from viewers.

2. Filtering Report Charts to Show Data Relevant to the Page

   Charts usually filter to show relevant data when the chart's report type has a matching ID field for the record, such as Account ID for an account record. Occasionally, charts show data unfiltered. But you can set up a chart to be filterable if its report type has a lookup to the matching ID field.

3. Refreshing Report Chart Data

   Normally, charts refresh data once every 24 hours. But you can change a chart to refresh each time someone opens the page it's on. The refresh option is under the Chart Properties dialog box of the page layout editor. However, we recommend daily refresh over selecting the option, because users will soon reach the refresh limit or will wait for chart data to show until refresh is complete.

Hiding a Report Chart that Shows an Error

A chart can sometimes, for whatever reason, show an error on the detail page instead of data. Since a chart with an error is not that useful, you have the option to hide it from viewers.

To hide such a chart, select **Hide chart with error** from the Chart Properties dialog box in the page layout editor.

A chart can show an error on a page for one or more of these reasons.

- The viewer doesn't have access to the field that's used for filtering.
- The viewer doesn't have access to the report folder.
- The report chart has been deleted from the report.
- The report definition has changed.
- The report itself is no longer available.

Filtering Report Charts to Show Data Relevant to the Page

Charts usually filter to show relevant data when the chart's report type has a matching ID field for the record, such as Account ID for an account record. Occasionally, charts show data unfiltered. But you can set up a chart to be filterable if its report type has a lookup to the matching ID field.

Make a Chart Filterable

When the chart's report type doesn't have a matching ID field for the record, the chart is not filtered. Instead, it shows all data.

A report chart on open cases shows information for all accounts on an account record. To make the chart filterable, edit the layout of the chart's cases custom report type, and add the `Account ID` field via lookup. Now when you view an account record, the chart is filtered.

Pick a Relevant Filter

Sometimes when there are multiple ID fields available to filter, the chart may not show any data. In such situations, select the most relevant ID field from the **Filtered by** drop-down in the Chart Properties dialog box. A chart on open opportunities on account records is filtered by `Partner Account ID`, which isn't that useful. But we can edit the chart properties and choose `Account ID` from the drop-down to filter by the account record.
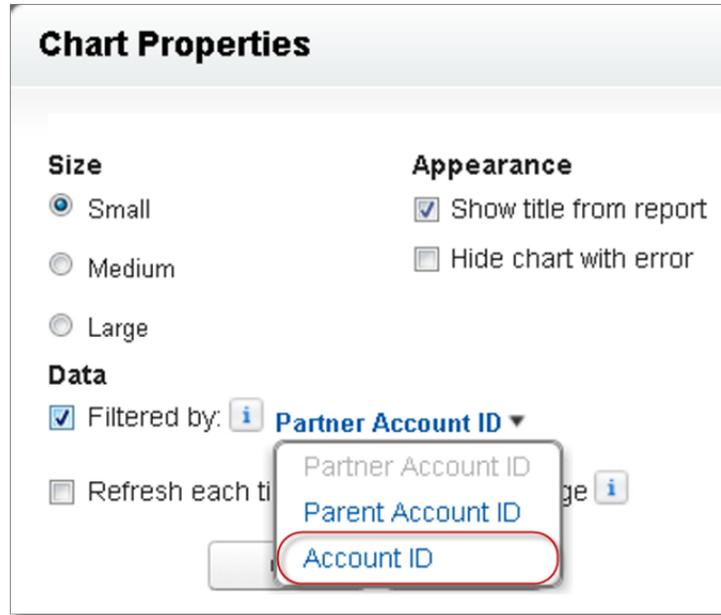
Note: Charts can't filter for relevant data if the source report has reached the limit of 20 field filters.

Refreshing Report Chart Data

Normally, charts refresh data once every 24 hours. But you can change a chart to refresh each time someone opens the page it's on. The refresh option is under the Chart Properties dialog box of the page layout editor. However, we recommend daily refresh over selecting the option, because users will soon reach the refresh limit or will wait for chart data to show until refresh is complete.

Daily Refresh

Charts refresh data once every 24 hours. If within that time users want the latest, they can click **Refresh** on the chart.

Refresh When User Opens the Page

To change a chart's normal refresh, select **Refresh each time a user opens the page** in the Chart Properties dialog box of the page layout editor. This option triggers a chart refresh each time someone opens the page the chart is on. Selecting the option is not recommended for two reasons.

- There's a risk of reaching the chart refresh limit faster. Refreshes count towards the hourly limit for each user and organization.
- For reports that take longer to run, selecting this option can make users wait to see chart data.

EDITIONS

Available in: Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

USER PERMISSIONS

To create, edit, and delete reports:

- **Legacy Folder Sharing**
  Create and Customize Reports

  AND

  Report Builder

  Enhanced Folder Sharing
  Create and Customize Reports

  AND

  Report Builder

Example of Report Charts on an Account Page

In this example, we've embedded two report charts on an important account page that show deals in the pipeline and open support cases for the account. From looking at the charts, the account executive can quickly gauge the account's activity and health.

Report Charts on an Account Page

1. The pipeline chart shows opportunities for the account in various stages including two that were won. The account executive can hover on the chart to get the value of each opportunity (Values are highlighted if the source report chart has hover enabled). She is able to click the chart, go to the report, and get more details for these opportunities.

2. From the charts on open cases, she can see there are a couple of unresolved support issues for the account. She can click the chart, drill into the report for case details and follow up with the support engineers to get them resolved before a crucial meeting with the client.

Limits on Report Charts in Pages

Consider these limits when embedding charts in detail pages.

- You can have two report charts per page.

- You can only add report charts from the enhanced page layout editor. The mini console and the original page layout editor are not supported.

- For historical trend reports in Lightning Experience, you must set snapshot date as the primary row grouping.

- In Lightning Experience, embedded report charts display the source report table's groupings, not the report chart's. In Salesforce Classic, embedded report charts display the source report chart's grouping, not the report table's groupings.

- On detail pages, users can refresh up to 100 report charts every 60 minutes.

- Your org can refresh up to 3,000 report charts every 60 minutes.

SEE ALSO:

Add a Report Chart to a Page Layout

Example of Report Charts on an Account Page

Delete a Report

You can delete a selected report from the Reports tab or from its run page.

1. Delete a report in one of two ways:

Option	Description
To delete from the Reports tab	Next to the report, click ▼ > **Delete**.
To delete from the report's run page	Click **Delete**.

2. Click **OK**.

    Deleted reports are moved to the Recycle Bin. You can't delete reports in others' personal folders. You also can't delete reports used by dashboard components or reporting snapshots unless you first delete the dashboard component or reporting snapshot.

Analyze Reports with Einstein Data Insights

Einstein Data Insights scans your report data - quickly and thoroughly - using artificial intelligence and comprehensive statistical analysis powered by Einstein Discovery. Einstein Data Insights goes deep into the report data, explores underlying patterns, identifies insights, and surfaces those insights with charts and explanations that are easy to understand. Einstein Data Insights works with Tabular and Summary reports.

> **Note:** Einstein Data Insights produces descriptive insights for report data in Tabular and Summary reports. To make predictions and improvements, or to analyze Einstein Analytics datasets, see Explain, Predict, and Take Action with Einstein Discovery .

Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

Set Up Einstein Data Insights

Einstein Data Insights is activated automatically when Tableau CRM is enabled. The only setup task is to create and assign permission sets to users. Setup uses the permission sets that come with the Tableau CRM Plus license.

Analyze Report Data

Let Einstein Data Insights augment your report with AI and comprehensive statistical analysis to produce Einstein Discovery-powered insights into your data. Einstein does the analytic heavy lifting for you. Your efforts aren't required to group report data, or summarize it, or chart it (unless you really want to).

Report Tips for Einstein Data Insights

Consider these tips to optimize your report analysis. To learn more, see Einstein Data Insights: Limits and Allocations on page 274.

Einstein Data Insights: Limits and Allocations

Learn about the limits, allocations, and considerations for Einstein Data Insights.

Monitor Usage Statistics for Einstein Data Insights

Einstein Data Insights monitors usage statistics in real time. You can view the usage statistics in your org.

Set Up Einstein Data Insights

Einstein Data Insights is activated automatically when Tableau CRM is enabled. The only setup task is to create and assign permission sets to users. Setup uses the permission sets that come with the Tableau CRM Plus license.

To access Einstein Data Insights, users need the **Can Run Einstein Data Insights** permission. The Tableau CRM Plus license, which gives your org access to Einstein Data Insights, comes with two standard permission sets:

- **Tableau CRM Plus User** permission set for restricted access users
- **Tableau CRM Plus Admin** permission set for admin-level access

These permission sets are automatically created when the Tableau CRM Plus license is provisioned in your org. Both of these permission sets include the **Can Run Einstein Data Insights** permission, as well as permissions for other Tableau CRM functionality.

To give someone access to Einstein Data Insights, assign one of these permission sets to them:

1. From Setup, in the Quick Find box, enter `Permission Sets`, and select **Permission Sets**.

2. Click **Tableau CRM Plus User** or **Tableau CRM Plus Admin**.

3. Click **Manage Assignments** > **Add Assignments**.

4. Select everyone you want to grant Einstein Data Insights access to.

5. Click **Assign**.

Everyone you selected now has access to Einstein Data Insights. The next time they run a report with sufficient data, they'll see the **Analyze** button, and they'll be able to start an analysis.

You have the option to create a custom permission set that gives users access to Einstein Data Insights but not to other Tableau CRM features in the standard permission sets. If you create a custom permission set:

- For Licenses, select **Tableau CRM Plus**.
- For System Permissions, select the **Can Run Einstein Data Insights** permission.

EDITIONS

Available in Lightning Experience

Available in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To assign a permission set to users:
- Assign Permission Sets

To create a custom permission set:
- Manage Profiles and Permission Sets

Analyze Report Data

Let Einstein Data Insights augment your report with AI and comprehensive statistical analysis to produce Einstein Discovery-powered insights into your data. Einstein does the analytic heavy lifting for you. Your efforts aren't required to group report data, or summarize it, or chart it (unless you really want to).

> 📝 Note:
> - For an introduction, check out the Einstein Data Insights: Quick Look Trailhead module.
> - Consider Report Tips for Einstein Data Insights on page 273.

To get insights into your report data:

1. Run a Tabular and Summary report that has at least 2 columns and 50 rows of data.

   For help with building a report, see Build a Report in Salesforce Help.

2. In the Toolbar, click **Analyze**.

EDITIONS

Available in Lightning Experience

Available in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To start an analysis:
- Can Run Einstein Data Insights

📝 Note: If the **Analyze** button is visible but disabled, then be sure to enable **Detail Rows** at the bottom of the run page. The **Insights** button is enabled only when **Detail Rows** is enabled.

If there are no existing analyses on a report, Einstein Data Insights automatically runs an initial analysis. Einstein counts the number of values (frequency count) in the report and looks for statistically significant occurrences of values in the report data.
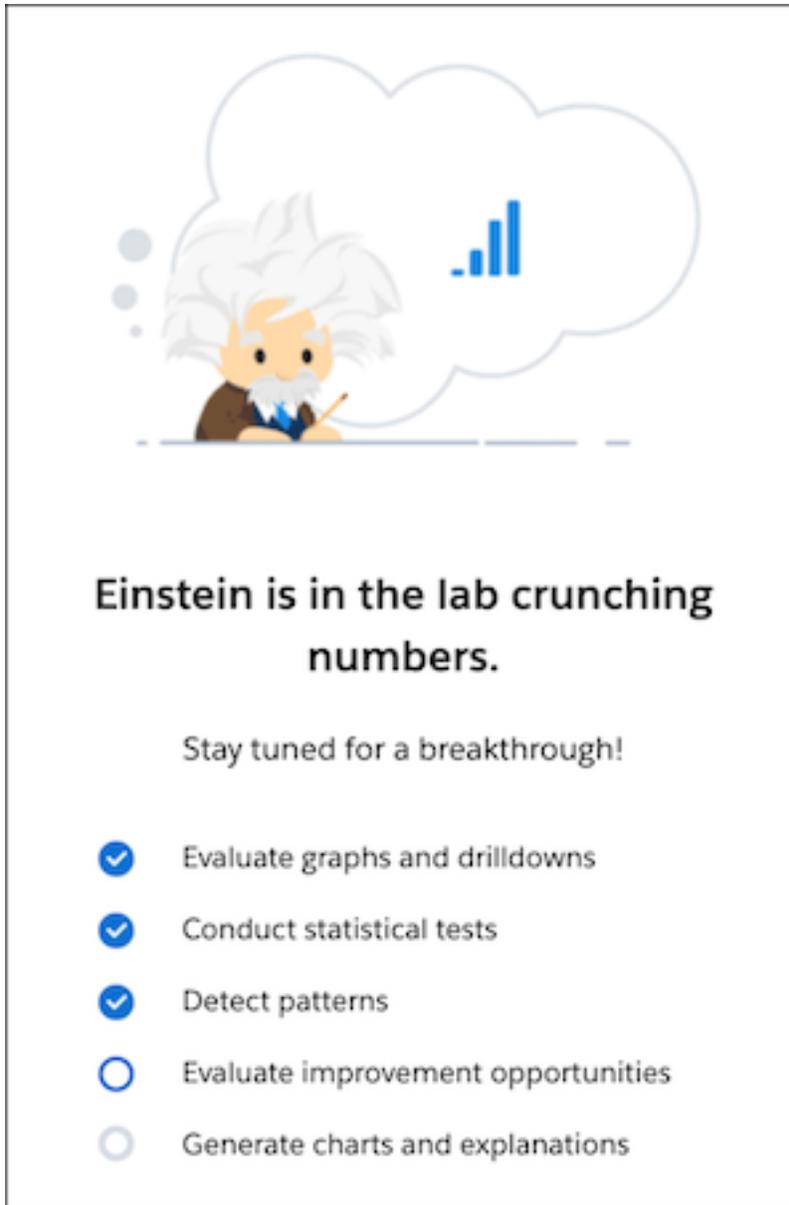
Einstein displays the results of its Occurrences analysis.

When you finish scrolling through the Occurrences insights, click Home on the toolbar to create other analyses.

The Einstein Data Insights panel appears.

If you see a **My Past Analyses** for this report, you can click one to run a past analysis, and then skip the next two steps.

3. If you want, change the focus of Einstein's analysis. From **I want to**, select whether you want Einstein's analysis to **Maximize** or **Minimize** the outcome. From **the variable** dropdown, select a different outcome from the list of candidate report columns, or select **Occurrences** if you want to analyze the number of times (frequency count) a value has occurred. If you can't find the variable you want, edit the report and add the desired column. Outcomes can be numeric fields (measures) or text fields with two values (such as won/lost, churn/not-churn, or public/private).

4. Click **Create Analysis**.

Einstein begins analyzing your report data. When finished, Einstein shows you the report columns, ranked by correlation to the goal you selected.
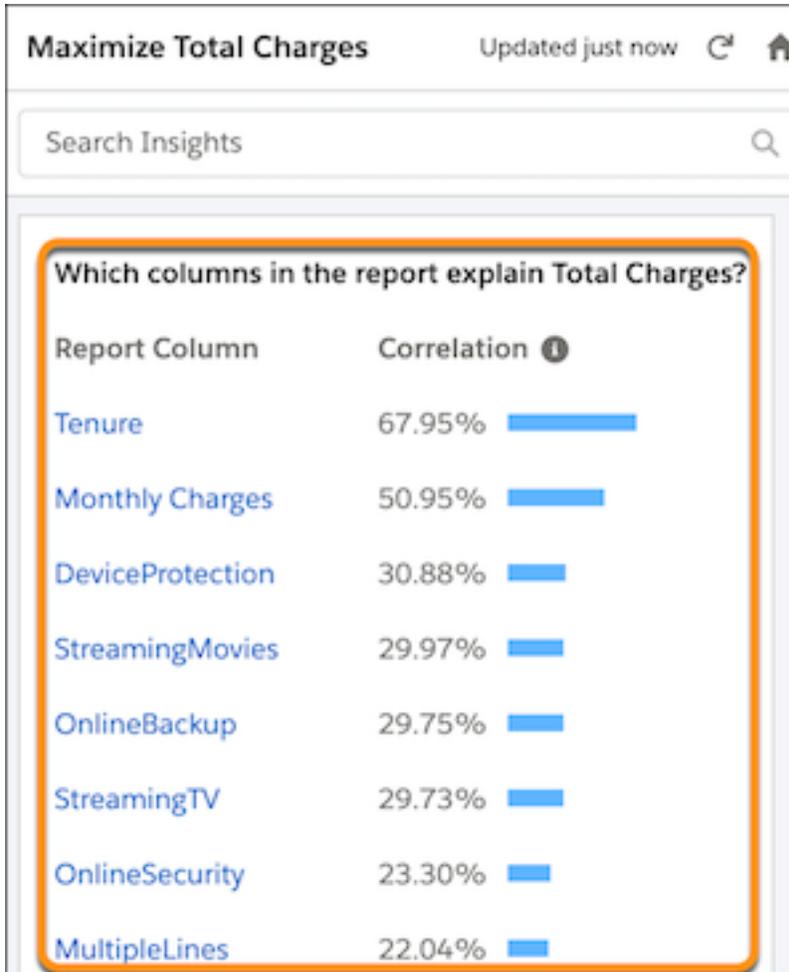
The higher the correlation, expressed as a percentage, the stronger the statistical relationship, and the more likely it's worth a closer look.

> ✎ Note: Keep in mind that correlation is not causation. Correlation merely describes the strength of association between variables, not whether they causally affect each other.

5. Scroll down to review the list of insights that Einstein uncovered in its analysis.
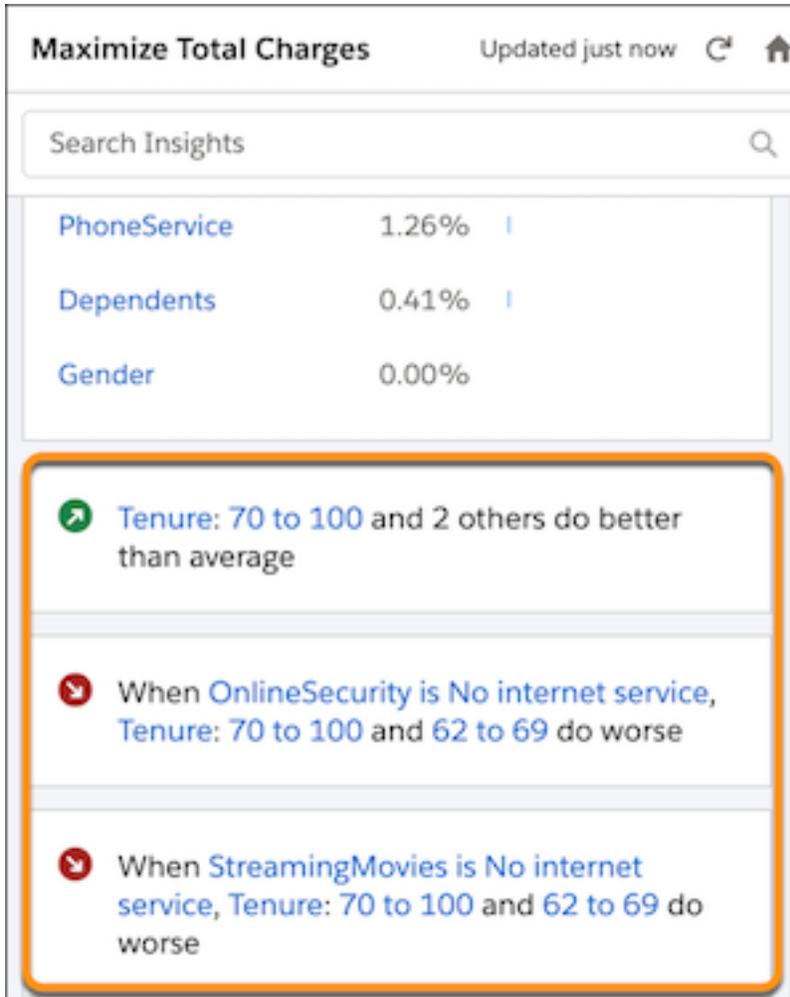
Einstein lists insights in order of statistical significance, starting with the insights that have the highest impact on the goal.

6. Click an insight that you're interested in learning more about.

Each insight contains a name, a chart, a summary headline, and descriptive text (insight title and supporting details).

- To display details about a particular segment in the chart, hover over it.

- To show where details appear in the chart, hover over a hypertext link in the insight description.

7. To refresh your analysis and create a new analysis version, click the refresh button.

Einstein begins analyzing your report data. When finished, Einstein updates the list of insights resulting from its analysis.

Report Tips for Einstein Data Insights

Consider these tips to optimize your report analysis. To learn more, see Einstein Data Insights: Limits and Allocations on page 274.

Suggestions for Report Columns and Rows

To get the best results for your analysis, consider the following suggestions:

- Avoid using report columns that contain unique ID fields, data with high cardinality (over 100 unique values), and data with high correlations (greater than 90%) to the selected analytical outcome.
- Exclude rows that haven't yet reached an outcome. For example, use closed opportunities or cases only.

Capabilities

Einstein Data Insights can analyze up to 500,000 rows and 50 columns of report data.

Minimum Requirements and Qualified Data

Einstein requires a report with at least 50 qualified rows and 2 qualified columns. When you first open a report, you see the **Analyze** button if the report contains 50 rows and 2 columns, whether they are qualified or not.

During analysis, Einstein Data Insights reviews your data to determine which rows and columns are suitable for analysis.

- Rows are disqualified if the outcome field selected for analysis is null.
- Columns are disqualified if a column contains only one value, contains no values (all values are null), or has too many unique values with very little repetition (for example, unique identifier fields).

If your data contains fewer than 50 qualified rows or 2 qualified columns, Einstein Data Insights displays an error indicating insufficient data.

Additional Processing Considerations for Report Columns

- If a non-outcome column contains null values, Einstein Data Insights categorizes the null values into a separate group (Null/Unspecified).
- If a numeric column contains 10 or fewer unique values, Einstein Data Insights treats the data as text, and applies the qualification requirements described for text fields.

Report Scope

Einstein Data Insights analyzes:

- All the data in your report, not just the maximum of 2,000 rows that are displayed.
- All the data within the scope of your report, and ignores any data that is outside of its scope. Therefore, be sure to include everything you're interested in. For example, if you create a report that evaluates all opportunities, then Einstein analyzes all opportunities as well. If you filter the report so that it doesn't return open opportunities, then Einstein doesn't analyze open opportunities.

Other Considerations

- You can encounter **NaN** in place of numerical values, indicating that the value is "Not a Number". This value is often returned when Einstein's calculation produces an undefined number, like dividing a number by 0.
- Einstein Data Insights is supported on Summary and Tabular reports. It is not currently supported on Matrix reports or Joined reports.
- Trend Reports are shown on a monthly basis. They are currently not shown by fiscal quarter, fiscal year, or other time period.

- Localization is currently not supported.

Einstein Data Insights: Limits and Allocations

Learn about the limits, allocations, and considerations for Einstein Data Insights.

Limits

Limit	Number
Maximum number of report rows analyzed	500,000
Minimum number of report rows required for an analysis	50
Maximum number of report columns analyzed	50
Minimum number of report columns required for an analysis	2

Allocations

Allocation	Number
Maximum number of EDI analysis creations per person per day	50
Maximum number of EDI analysis creations per org per day	1,000
Maximum number of concurrent EDI analyses per org	10

If you have questions about your allocations, contact your Salesforce representative.

Usage Statistics

Einstein monitors usage statistics in real time. To learn more, see Monitor Usage Statistics for Einstein Data Insights on page 275.

Monitor Usage Statistics for Einstein Data Insights

Einstein Data Insights monitors usage statistics in real time. You can view the usage statistics in your org.

📝 **Note:** Before you begin, Analytics must be enabled in your org. To enable Analytics, see Basic Analytics Platform Setup on page 551.

To view usage statistics:

1. From Setup, in the Quick Find box, enter *Discovery*.

2. Under **Einstein Discovery and Einstein Data Insights**, select **Usage**.

> **Note:** Times and dates are based on your org's default time zone.

Statistic	Description
Number of analysis created today	Total number of analyses created in your org since 12:00am today.

Statistic	Description
Number of analysis currently being created	Total number of analyses that are currently being run in your org.

SEE ALSO:

[Einstein Data Insights: Limits and Allocations](#)

Troubleshoot Reports

Use these tips to help solve problems that arise when you're working with reports.

1. [Why can't I run a report on a custom or external object?](#)
   Sometimes an object, its report types, or your access isn't set up completely.

2. [Why do my chart labels overlap?](#)
   Sometimes the labels for your charts can overlap and be difficult to read. There may be too much data in the same space, or the segments or wedges of the chart may be too small.

3. [Why doesn't my report return any data?](#)
   Check with your administrator to make sure that you have access to the records you're trying to report on. If you're still not seeing any results in your report, try casting a wider net.

4. [Why can't I see formula options in report builder?](#)
   If you can't see the **Add Formula** or **Add Summary Formula** option in the Fields pane of the report builder, add a row grouping to the report. (In Salesforce Classic, change your report format to summary, matrix, or joined.) Formulas require at least one row grouping.

5. [Why doesn't my report return the data I expect?](#)
   Check your filters, groupings, fields, report type, and role or user hierarchy to make sure that you are asking the right questions of your data.

6. [What are some common report limits?](#)
   Here is a full list of limits that apply to reports and dashboards.

7. [Why am I getting an "obsolete report" error message?](#)
   You may be trying to report on data that isn't available.

8. [Why is an old role name appearing in the "role hierarchy" trail of my report?](#)
   Changes to role names aren't automatically updated in reports. To see the latest role name, change the `Role name as displayed on reports` field for the role record.

9. [Why aren't middle names appearing in full name fields on reports?](#)
   If your org has **Enable Middle Names for Person Names** turned on, which adds a "Middle Name" field to standard objects like users, leads, and contacts, then you may notice that the "Full Name" field on reports excludes middle names.

SEE ALSO:

[Report Considerations for Salesforce Connect—All Adapters](#)

Why can't I run a report on a custom or external object?

Sometimes an object, its report types, or your access isn't set up completely.

If you can't run a report on a custom or external object, check with your admin that the following conditions are true.

- Reports are enabled on the custom or external object.
- You have access to the object and its records.
- The object and its custom report types are deployed.

Why do my chart labels overlap?

Sometimes the labels for your charts can overlap and be difficult to read. There may be too much data in the same space, or the segments or wedges of the chart may be too small.

To fix overlapping labels, try the following:

- **Enlarge the chart.** Change the chart size in the report to large or extra large or make the dashboard column wider.
- **Remove extra grouping levels.** Reduce the number of grouping levels by using a different chart type. For example, horizontal bar charts have fewer values than grouped horizontal bar charts.
- **Use horizontal charts.** Horizontal charts use different spacing than vertical charts. The advantage of the horizontal bar charts is that the chart can be extended vertically to show numerous groupings, though the width is fixed. Depending on chart settings, you can also display Chatter photos.
- **Change the scale.** Set the chart to use larger units. For example, show values as multiples of 1000 or 1000000.
- **Group small values.** Select the `Combine Small Groups into 'Others'` option for pie, donut, and funnel charts.

Why doesn't my report return any data?

Check with your administrator to make sure that you have access to the records you're trying to report on. If you're still not seeing any results in your report, try casting a wider net.

- Show more than your own records. For example, select **Show** > **All accounts**.
- Expand your time frame filters. For example, select All Time for `Range`, or select a broader custom range.
- Choose field filter operators carefully. If you select `Account Owner equals John James`, you limit potential results to exactly "John James." If you don't see the results you expect, consider expanding the selection by using `Account Owner contains James` instead.
- Check your filter logic. Make sure your combination of conditions isn't excluding all data.

Limiting your report can improve performance, but make sure you're not filtering out the data you want to see.

If your report includes an external object, the results probably don't reflect the full data set. External objects behave similarly to custom objects, except that they map to data that's stored outside your Salesforce org. A report that includes an external object fetches up to 20,000 records for the primary object and can encounter callout limits while fetching external object data. If the report results in few or no rows, try customizing the report to obtain more relevant external object rows.

Why can't I see formula options in report builder?

If you can't see the **Add Formula** or **Add Summary Formula** option in the Fields pane of the report builder, add a row grouping to the report. (In Salesforce Classic, change your report format to summary, matrix, or joined.) Formulas require at least one row grouping.

Why doesn't my report return the data I expect?

Check your filters, groupings, fields, report type, and role or user hierarchy to make sure that you are asking the right questions of your data.

Filters

Make sure to include all the data you want.

> 💡 **Tip:** To change a report filter, hover over it and click **Edit** or **Remove**. Your filters display when you run your report; click **Edit** on that page to make additional changes.

Groupings

When you group by a field, you remove it from the details of the report. If you export the report, you see the field, though it does not appear in the detail area.

Fields

After checking your groupings, check your fields. If your report contains multiple related records—for example, an account, its opportunities, and their products—look at the detail pages for the account record, opportunity record, and product line item to ensure that the fields contain data. If the data isn't available in the format you want, work with your Salesforce administrator to add formula fields to get the data. When formula fields are added to an object, they appear in record detail pages and in reports.

Report Type

The report type selected may not be appropriate or the records anticipated may not share the relationship between objects required for the report type. When choosing a report, be sure you understand which fields are available in the report type.

If your report type includes both parent and child objects, but no child fields are used in a report, the report shows parent records whether they have a child record or not.

Hierarchy

Hierarchy options let you drill down to different data sets based on the role or user hierarchy. The selected hierarchy level, or its default role, may affect the data shown.

> 📝 **Note:** Hierarchy applies to activity and opportunity reports.

External Objects

If your report includes an external object, the results probably don't reflect the full data set. External objects behave similarly to custom objects, except that they map to data that's stored outside your Salesforce org. A report that includes an external object fetches up to 20,000 records for the primary object and can encounter callout limits while fetching external object data. If the report results in few or no rows, try customizing the report to obtain more relevant external object rows.

What are some common report limits?

Here is a full list of limits that apply to reports and dashboards.

Salesforce Reports and Dashboards Allocations

Feature	Personal Edition	Contact Manager	Group Edition	Professional Edition	Enterprise Edition	Unlimited and Performance Edition	Developer Edition
Custom report types (Limits apply to all custom report types regardless of development status.)	N/A			50	200	2,000	400

Feature	Personal Edition	Contact Manager	Group Edition	Professional Edition	Enterprise Edition	Unlimited and Performance Edition	Developer Edition
Dashboard filters				3 per dashboard			
Dynamic dashboards per org	N/A					Up to 5	Up to 10	Up to 3
Field filters per report[1]	20							
Formulas per report	5							
Reporting snapshots	N/A				1^{234}	1^2	2^2	1^{234}
Scheduled dashboard refreshes	N/A					1^2	2^2	N/A
Scheduled reports per hour (Emailed reports can be up to 10 MB.)	N/A				1^{235}	1^2	2^2	1^{235}

[1] These allocations apply to the report builder. If you're using the report wizard, the allocation is 10.

[2] Up to 200 total.

[3] Off-peak hours (between 6 PM and 3 AM local time) only.

[4] Limited to one preferred start time per day.

[5] Limited to three preferred start times per day.

Salesforce retains historical data for the previous three months, plus the current month.

The following Salesforce Reports and Dashboards limits, limitations, and allocations apply to all supported editions.

Report Limits, Limitations, and Allocations

- The report builder preview shows a maximum of 20 rows for summary reports (grouped by rows) and matrix reports (grouped by columns), and 50 rows for tabular reports (no groupings).
- In Salesforce Classic, you can't have more than 250 groups or 4,000 values in a chart. If you see an error message saying that your chart has too many groups or values to plot, adjust the report filters to reduce the number. In combination charts, all groups and values count against the total.
- In Lightning Experience, a report chart can have at most 2000 groups. If a report has more than 2000 groups, the action 'Combine Small Groups into Others' applies only to the small groups within the 2000 that are included in the report chart. Any additional small groups are ignored.
- Reports display a maximum of 2,000 rows. To view more the rows, export the report to Excel or use the printable view for tabular and summary reports. For joined reports, printable view displays a maximum of 20,000 rows. Printable view is only available in Salesforce Classic.
  - Summary reports (grouped by rows) and matrix reports (grouped by columns) display the first 2,000 groupings when Show Details is disabled.
  - Matrix reports display a maximum of 400,000 summarized values.

- Matrix reports display a maximum of 2,000 groupings in the vertical axis when Show Details is disabled. If there are more than 400,000 summarized values, rows are removed until the 2,000 groupings limit is met. Then columns are removed until the number of summarized values moves below 400,000.

- Matrix reports that return more than 2,000 rows don't show details. If you click **Show Details**, nothing happens. You can only view the report with details hidden.

- Because a matrix report includes multiple groupings, the maximum of 2000 values is typically reported in fewer than 2000 groups.

- Up to five metrics display in the Lightning Experience report header. Metrics such as summarized fields appear in the order that they appear in the report, left to right. The grand total, when shown, always displays.

- When reports that have groupings are viewed in the Salesforce mobile app, they're converted to tabular reports.

- The Salesforce mobile app supports a maximum of 25 report columns.

- By default, reports time out after 10 minutes.

- In a joined report, each block can have up to 100 columns. A joined report can have up to 5 blocks.

- When you add a block to a joined report and the block has multiple entities in common with the report, only the first entity (in alphabetical order) is shown. Only the fields from the first entity are shown in the common fields area.

- You can add up to 10 custom summary formulas to each block in a joined report. A joined report can have a total of 50 custom summary formulas.

- Each joined report can have up to 10 cross-block custom summary formulas.

- In a non-joined report, if you click a bar in a report chart, the report results are filtered according to the selected bar. In a joined report, clicking a bar doesn't apply the filter.

- If you filter on standard long text area fields, such as Description or Solution Details, only the first 1000 characters of the field are searched for matches in reports.

- Some filters (such as date range) are constructed using multiple custom filters, each of which counts towards the total of 20.

- Field-to-field filtering isn't available on currency fields for orgs that have multi-currency enabled.

- The first 999 characters in a standard rich text area or a long text area are displayed in a report. For custom fields, only the first 255 characters are displayed.

- Summary fields on tabular, summary, and matrix reports can display up to 21-digits.

- Reports can't be filtered on custom long text area fields.

- Joined reports require that the new user interface theme is enabled. Users without the new theme are unable to create, edit, or run joined reports.

- Forecast reports include only opportunities that are set to close within the forecast period, except those assigned to the Omitted forecast category.

- Internet Explorer 6 isn't supported for joined reports.

- Acceptable range for values: The maximum value allowed is 999999999999999. The minimum value allowed is -99999999999999999.

- Each person in your org can subscribe to up to 5 reports.

- Up to 500 individual recipients can be added. A recipient is a user, role, or group.

- If a role or group contains more than 500 users, users sometimes don't receive the updated report.

- The State/Province picklist filter converts the selected state or province to a two-digit code (example: MO for Missouri). If a state or province in another country covered by the report has the same code (example: MO for Morales, Mexico), filtering on one of the states or provinces can return data for the other.

- In Lightning Experience, embedded report charts display the source report table's groupings, not the report chart's. In Salesforce Classic, embedded report charts display the source report chart's grouping, not the report table's groupings.

Dashboard Limits, Limitations, and Allocations

- A dashboard filter can have up to 50 values.
- Each dashboard can have up to 20 components.
- It's not possible to filter on bucket fields. However, it's possible to use a report filtered on a bucket field on the dashboard page.
- Filtering is restricted in some dashboards that contain multiple components based on different report types:
  - If a dashboard has a component based on Cases or Leads and another component based on a different report type, you can't filter the dashboard on the Case Owner or Lead Owner field. In addition, filtering on other Owner fields doesn't display Case Owner or Lead Owner as equivalent fields.
  - If a dashboard has a component based on the Tasks and Events, Activities with Accounts, or Activities with Contacts report type and another component based on a different report type, you can't filter the dashboard on the Assigned field.
- A dashboard table or chart can display up to 20 photos.
- Wait at least one minute between dashboard refreshes.
- File attachments for report subscriptions are limited to 15,000 rows, 30 columns, and 3 MB file size. Extra data is clipped or not sent.
- Each person in your org can subscribe to up to 5 dashboards.
- Up to 500 individual recipients can be added. A recipient is a user, role, or group.
- If a role or group contains more than 500 users, some users don't receive the updated dashboard.
- Downloaded and shared images of dashboard component tables have a maximum height of 3000 pixels or approximately 100 rows. Extra rows beyond the limit are clipped. To avoid clipping, filter the chart to fewer than 100 rows.
- For funnel charts, the total value isn't included in subscription emails.

Report Type Limits, Limitations, and Allocations

- A custom report type can contain up to 60 object references. For example, if you select the maximum limit of four object relationships for a report type, then you could select fields via lookup from an extra 56 objects. However, users receive an error message if they run a report from a custom report type and the report contains columns from more than 20 different objects.
- You can add up to 1000 fields to each custom report type.

Reporting Snapshot Limits, Limitations, and Allocations

- The maximum number of rows you can insert into a custom object is 2,000.
- The maximum number of runs you can store is 200.
- The maximum number of source report columns you can map to target fields is 100.

Filtering Limits, Limitations, and Allocations

- Only the first 255 characters in a custom text field count for filtering purposes.
- You can enter up to 1,333 characters for filter criteria, including commas used as OR operators.

Embedded Report Charts Limits, Limitations, and Allocations

- You can have two report charts per page.
- You can only add report charts from the enhanced page layout editor. The mini console and the original page layout editor aren't supported.
- For historical trend reports in Lightning Experience, you must set snapshot date as the primary row grouping.
- In Lightning Experience, embedded report charts display the source report table's groupings, not the report chart's. In Salesforce Classic, embedded report charts display the source report chart's grouping, not the report table's groupings.
- On detail pages, users can refresh up to 100 report charts every 60 minutes.

- Your org can refresh up to 3,000 report charts every 60 minutes.

List View Limits, Limitations, and Allocations

- Only the first 255 characters are shown for custom long text area fields in list views.

Bucket and Bucket Field Limits, Limitations, and Allocations

- Each report can include up to five bucket fields.
- Each bucket field can contain up to 20 buckets.
- Each bucket can contain up to 20 values.
- Bucket fields are available for use only in the report where they're generated. To use a bucket in multiple reports, create the field for each report, or create a separate formula field for the object that's dependent on the bucket.

  📝 **Note:** These limits don't apply to the use of Other as permitted within the bucket field's setup.

- Buckets and bucket fields aren't available for reports that include external objects.
- If a bucket field's source column has a custom index, and you filter by the bucket field, then the performance gains from the custom index are lost.

Historical Trend Report Limits, Limitations, and Allocations

- Salesforce retains historical data for the previous three months, plus the current month.
- Up to 5 million rows of historical trending data can be stored for each object. Historical data capture stops when the limit is exceeded. The admin is alerted by email when any object reaches 70 percent of the limit, and again if the limit is exceeded.
- Each historical trend report can contain up to 100 fields. In Opportunities reports, the fields include standard preselected fields, which can't be disabled.
- Formula fields aren't supported.
- Row limit filters aren't supported.
- The summary report format isn't supported.
- You can specify up to five historical snapshot dates in each historical trend report.
- You can use up to four historical filters on each historical trend report.
- These field types are supported: Number, Currency, Date, Picklist, Lookup.
- Dynamic exchange rates aren't supported. When you run a historical trend report, it uses a static exchange rate, which could be outdated.
- Internet Explorer 6 isn't supported.
- You can't subscribe to historical trend reports.
- The Report Wizard isn't supported. Historical trend reports can only be created with the Report Builder.
- Historical trend reporting with charts is supported in Lightning Experience, but tabular views of historical trend reports aren't available.

External Object Report Limits, Limitations, and Allocations

If your report includes an external object, the results probably don't reflect the full data set. External objects behave similarly to custom objects, except that they map to data that's stored outside your Salesforce org. A report that includes an external object fetches up to 20,000 records for the primary object and can encounter callout limits while fetching external object data. If the report results in few or no rows, try customizing the report to obtain more relevant external object rows.

Cross Filters

- Each report can have up to three cross filters.
- Each cross filter can have up to five subfilters.

- Filter logic applies only to field filters, not cross filters.

Why am I getting an "obsolete report" error message?

You may be trying to report on data that isn't available.

Data may be unavailable because:

- An object in the report isn't enabled for reporting anymore.
- A lookup relationship used by objects in the report has been deleted or modified.
- An object in the report has been deleted.
- You don't have "View" permissions for an object in the report.

Why is an old role name appearing in the "role hierarchy" trail of my report?

Changes to role names aren't automatically updated in reports. To see the latest role name, change the `Role name as displayed on reports` field for the role record.

Why aren't middle names appearing in full name fields on reports?

If your org has **Enable Middle Names for Person Names** turned on, which adds a "Middle Name" field to standard objects like users, leads, and contacts, then you may notice that the "Full Name" field on reports excludes middle names.

For example, if a user's first, middle, and last name are Nadia Nancy Smith, then in reports the full name appears as "Nadia Smith".

As a work-around, consider adding a person's middle name to their "First Name" field; "Nadia" would become "Nadia Nancy".

A missing middle name could cause issues if you are filtering reports on the "Full Name" field, because filtering by `Full Name equals Nadia Nancy Smith` doesn't return the user. Instead, filter the report by `Full Name equals Nadia Smith` or add three filters:

1. `First Name equals Nadia`
2. `Middle Name equals Nancy`
3. `Last Name equals Smith`

> **Note:** Case reports and Lead reports are affected by the discrepancy between `Full Name` and `Middle Name` described in this article. Other reports may also be affected, but not all reports are.

Improve Report Performance: Best Practices

When you run a report, the report looks for and then returns data. If a report is running slowly, it's because parts of the report that take a long time to find and return data. By optimizing the slow parts of a report, the report can be made to run much faster. Follow the tips in this guide to speed up sluggish reports.

Watch this video for quick tips on ▶ Making Your Reports Run Faster (Salesforce Classic).

Use efficient filters

By limiting the number of records returned, adding a filter can significantly speed up reports.

Because they limit the amount of data returned by reports, you'd think filters would always make reports run faster. But because each record must get evaluated against each report filter, filters with

complex logic can slow down reports. Apart from the performance-slowing filters described in this guide, adding a filter to return fewer records is a sure-fire way to speed up a sluggish report.

Here's how you can apply filters to optimize report performance.

Filter out unnecessary data to return fewer records

Add a field filter from the Lightning Report Builder.

1. Click **Filters**.

2. From the **Add filter...** lookup, select a field to filter on.

   A filter menu opens.

3. From the **Operator** dropdown menu, choose an operator like `equals`, `greater than`, or `less than`.

4. From the **Type** dropdown menu, choose to filter by a specific `Value` or by another `Field`.

5. In **Value**, enter a value or select a field name.

6. Click **Apply**.

Your report now runs faster because it returns fewer records.

Say that you run a sales team and want to see how many opportunities each of your salespeople has. You build a report that returns all the opportunities in your company. Now you have all the data you need, but also some that you don't, and your report takes a long time to run. To make your report run faster, filter it so that it only returns opportunities assigned to your team (1). If you need your report to run even faster, consider filtering by time (2). Do you need to see opportunities from last financial quarter, or the quarter before? If not, add a second filter which scopes your report to this financial quarter only. Now your report runs much faster!

SEE ALSO

- [Filter Report Data](#)

Filter with defined date ranges instead of open-ended relative date ranges

If a sluggish report has an open-ended relative date filter, like `Close Date LESS THAN Yesterday`, replace the relative date with a defined date range to speed up the report. For example, if you only need opportunities closed this week, edit the filter so that it reads `Close Date EQUALS This Week`.

SEE ALSO

- [Relative Date Filter Reference](#)

Filter with `EQUALS` instead of `CONTAINS`

When you set a filter with `CONTAINS` the report executes many steps to return data, which slows down the report. `EQUALS` executes in fewer steps, resulting in a faster report.

`CONTAINS` takes the value you enter in the filter and evaluates each character in each record's corresponding field to see if there is a match. `CONTAINS` can result in many steps per record. `EQUALS` compares the full value that you enter to the full field value of each record, which results in one step per record.

For example, say that you have a cases report. Each case record includes a subject. Some subjects include:

- Widgets not delivered.
- Widgets too small.
- Widgets too large.

You want to filter the report to return cases where widgets were not delivered, so you add a filter on the Subject field. Then you set the operator to `CONTAINS`, and the value to `delivered`. Your report returns the cases you want, but it takes a long time to run.

Speed up the report by editing the filter and using `EQUALS` instead of `CONTAINS`. Your edited filter reads `Subject EQUALS Widgets not delivered.` Now your report returns the cases you need, quickly.

SEE ALSO

- [Filter Operators Reference](#)

Filter with `NOT EQUAL TO` instead of `DOES NOT CONTAIN`

When you set a filter with `DOES NOT CONTAIN` the report executes many steps to return data, which slows down the report. `NOT EQUAL TO` executes in fewer steps, resulting in a faster report.

`DOES NOT CONTAIN` takes the value you enter in the filter and evaluates each character in each record's corresponding field to see if there is a match. `DOES NOT CONTAIN` can result in many steps per record. `NOT EQUAL TO` compares the full value that you enter to the full field value of each record, which results in one step per record.

For example, say that you have a cases report. Each case record includes a subject. Some subjects include:

- Widgets not delivered.
- Widgets too small.
- Widgets too large.

You want to filter the report to return cases where widgets are too large or too small, so you add a filter on the Subject field. Then you set the operator to `DOES NOT CONTAIN`, and set the value to `delivered`. Your report returns the cases you want, but it takes a long time to run.

Speed up the report by editing the filter and using `NOT EQUAL TO` instead of `DOES NOT CONTAIN`. Your edited filter reads `Subject NOT EQUAL TO Widgets not delivered.` Now your report returns the cases you need, quickly.

SEE ALSO

- [Filter Operators Reference](#)

Don't filter by row-level formulas

Filtering by a row-level formula slows down reports. If you have a slow-running report that filters by row-level formulas, speed it up by removing the row-level formula filters.

SEE ALSO

- Evaluate Each Record in Reports with Row-Level Formulas

Remove unnecessary columns

Much like returning lots of records can slow down a report, returning many columns can slow down a report. Many report types add a dozen or more columns to a report when it is created. Removing columns you don't need speeds up the report.

Administrators can improve report performance by removing default columns from commonly used report types.

SEE ALSO

- Build a Report in Lightning Experience
- Design the Field Layout for Reports Created from Your Custom Report Type

Hide details

Improve report performance of reports with grouped data by hiding detail rows. By hiding detail rows, individual records don't display in the report, but you still see key details like groupings, totals, subtotals, and record counts.

To hide report details, click the **Detail Rows** switch at the bottom of the report.

SEE ALSO

- Show and Hide Report Details

Write efficient formulas

Formulas are powerful tools for evaluating report data, but if they are written inefficiently then they can really slow down report runs. Here are a few tips for optimizing report formula performance.

Add formula fields to report types instead of writing report formulas

Both summary and row-level formulas are evaluated when reports run, which extends the time it takes for a given report to run. Formula fields run with workflows, so they don't slow down reports at run time. If a report with formulas is taking a long time to run, you can speed it up by moving the formula from the report builder into the report type via a formula field.

SEE ALSO

- Calculate Field Values With Formulas
- Set Up a Custom Report Type
- Workflow

Avoid referencing objects outside a report type in a formula field

When a formula field references an object outside of a report's report type, Salesforce joins the outside object to the objects within the report type. The join procedure can significantly slow down an otherwise performant report.

Rather than reference an outside object in a formula field, create a report type that joins the requisite objects together, then add your formula field. The reports based on that report type runs much more quickly than before.

Say that you have a report listing accounts and want to add a formula field to it that calculates average number of cases per account. Before adding the formula field, check to see if the report type includes the Cases object. If it does not, consider adding the formula field to a report type that has both Accounts and Cases, such as Accounts with Cases. Then create a report using that report type.

SEE ALSO

- Calculate Field Values With Formulas
- Set Up a Custom Report Type

Write row-level formulas sparingly

Row-level formulas are wonderful tools for evaluating each record in a report, but they can slow down reports. If you have a slow-running report that has a row-level formula, redesigning the report without a row-level formula can improve performance.

Some common equations are built into the report builder and don't require row-level formulas. Finding the sum, min, max, or average value doesn't require row-level formulas. It is much more performant to add a summary instead. While editing a report, add a summary from the column heading by clicking ▼ > **Summarize** and selecting a summary function like **Average**.

If a more sophisticated formula is called for, consider adding a formula field to the report type.

SEE ALSO

- Calculate Field Values With Formulas
- Set Up a Custom Report Type
- Evaluate Each Record in Reports with Row-Level Formulas

Don't group data by row-level formulas

Grouping data by a row-level formula slows down reports. If you have a slow-running report that groups by row-level formulas, speed it up by removing the row-level formula groups.

SEE ALSO

Bucket data sparingly

Because buckets evaluate every record returned by reports to see whether they belong in the bucket or out, they can slow down reports. If a report is running slowly, consider redesigning it to use fewer buckets.

SEE ALSO

- Categorize Data with Bucket Columns on page 39

Simplify sharing rules throughout Salesforce

Sharing rules govern access to objects. Reports access objects. Complex sharing rules can therefore slow down reports as each sharing rule has to be evaluated each time the report gets data from an object.

Common ways sharing rules become complicated include:

- Giving each user permission its own permission set and then assigning each user many permission sets. Instead, try to add as many user permissions to each permission set as possible and assign fewer permission sets to users.
- Adding complex sharing and access rules on objects.

With sharing rules, the best advice is to keep it simple...both for performance and your own sanity if you ever troubleshoot sharing settings!

SEE ALSO

- Control Who Sees What

"Hard delete" records

After deleting a record in Salesforce, it remains in the recycle bin for up to a month, and can be restored during that time. If many recently deleted records are in the recycle bin, report query performance can slow. Improve report performance by emptying the recycle bin.

In some cases, a log of deleted records may still exist and reports may query it, even if records have been emptied from the recycle bin. If an org has deleted lots of records, then querying this log could slow report performance.

SEE ALSO

- View, Restore, and Manage the Recycle Bin in Salesforce Classic
- Delete Unwanted Data in an Organization

Run slow reports during off-peak hours

If a report is still taking a long time to run, plan ahead. Subscribe to the report and schedule it to run during less busy times, then have Salesforce email the report to you and your team. Salesforce caches recently run report data, so you can view the report data in Salesforce without re-running the report.

1. From the **Reports** tab or from the report run page, click ▼ > **Subscribe**.

2. From the Edit Subscription menu, set the subscription schedule.

3. Optionally, add conditions. The conditions are evaluated when the report is run according to the schedule you set. The report is only emailed if all conditions are met.

4. Under `Send To`, you are automatically selected as a recipient. To add others or remove yourself, click **Edit Recipients**.

5. Under `Run Report As`, specify who runs the report.

   - **Me** — You run the report, and recipients see report data in the emailed report as you.
   - **Another Person** — Specify someone who has permission to run reports and who has access to the report.

6. Click **Save**.

The report subscription starts, and recipients begin receiving report results by email according to the schedule and conditions you set. For example, subscribe to a Sales Overview report and have it run daily at 6:00 AM (1). By the time you sit down at your desk at 9:00 AM, a recently refreshed copy of the report is waiting in your email inbox. And cached report data is waiting for you in Salesforce.
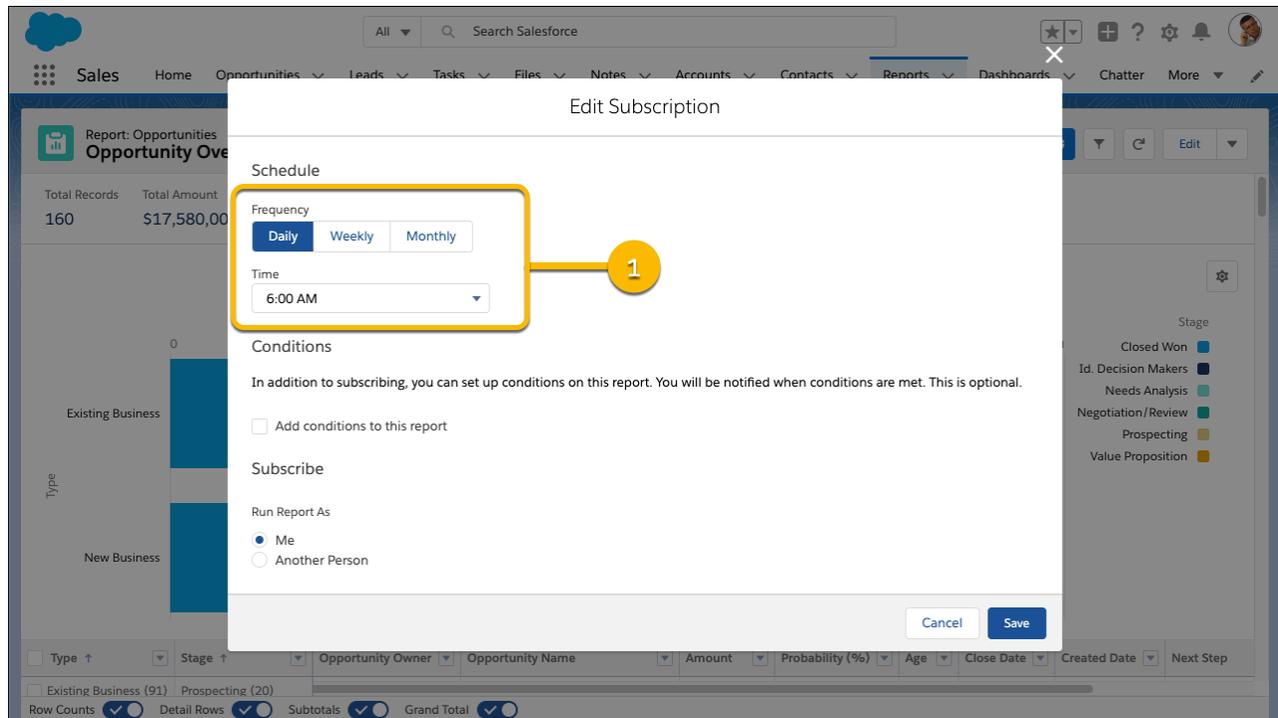
SEE ALSO

- Schedule and Subscribe to Reports on page 223

Create a Report

Create a report to get an up-to-the-minute view of an aspect of your business, such as the status of customers, opportunities, support cases, and so on.

To create a new report:

1. From the Reports tab, click **New Report**.

2. Select the report type for the report, and click **Create**.

3. Customize your report, then save or run it.

SEE ALSO:

Customizing Reports

Creating a Custom Report

You can customize standard reports, or you can build custom reports from scratch to suit the exact needs of your organization.

Users with the "Manage Custom Report Types" permission can define custom report types that extend the types of reports from which all users in their organization can create or update custom reports. A *report type* defines the set of records and fields available to a report based on the relationships between a primary object and its related objects. Reports display only records that meet the criteria defined in the report type. Salesforce provides a set of pre-defined standard report types; administrators can create custom report types as well.

Users with the "Manage Reporting Snapshots" permission can create and schedule snapshots. A reporting snapshot lets you report on historical data. Authorized users can save tabular or summary report results to fields on a custom object, then map those fields to corresponding fields on a target object. They can then schedule when to run the report to load the custom object's fields with the report's data. Reporting snapshots enable you to work with report data similarly to how you work with other records in Salesforce.

Create a Custom Report in Accessibility Mode

📝 **Note:** This topic only applies if you're not using report builder. *Report builder* is a visual editor for reports.

To create a new custom report using the custom report wizard:

1. From the Reports tab, choose the **Create New Custom Report** button.

2. Select the type of data for the report, and click **Next**.

   To create reports on custom objects, choose the **Other Reports** report type category, unless the custom object has a relationship with a standard object. When the custom object has a master-detail relationship with a standard object or is a lookup object on a standard object, select the standard object for the report type category instead.

   To create reports on an external object, choose the **Other Reports** category. To create reports on an external object with a child standard or custom object, select the category containing the custom report type that you created for that relationship.

3. Choose the report format.

4. Follow the steps of the wizard using the **Next** button. For each report, customize the following:

   - **Specify Row and Column Headers:** On the Select Grouping page for summary and matrix reports, choose the fields by which you want to group and subtotal the data. In a summary report, choosing more than one sort field allows you to subsort your data. For matrix reports, select summary fields for the row labels and column headings. When grouping by a date field, you can further group the data by a specific time period such as days, weeks, or months.

     📝 **Note:** On the Select Grouping page, if you set `Group Dates By` to "Calendar Month in Year" or "Calendar Day in Month," you won't be able to drill down to those date groupings in reports or dashboards. Users are taken to the unfiltered report instead.

   - **Summarize Data:** On the Select Columns to Total page, choose the types of summary information to display for numeric fields.

   - **Build Custom Summary Formulas:** On the Select Columns to Total page for summary and matrix reports, create custom summary formulas to calculate additional totals based on existing report summaries. A formula is an algorithm that derives its value from other fields, expressions, or values. See Add a Summary Formula Column to a Report on page 67.

   - **Choose Fields:** On the Select Columns page, choose the fields to display in the report. You can display only those fields that are visible in your page layout and field-level security settings. If you choose the `Description` field or any other long text field, only the first 255 characters are displayed.

   - The first 999 characters in a standard rich text area or a long text area are displayed in a report. For custom fields, only the first 255 characters are displayed. If you download the report as **Details Only**, the entire field is available.

   - **Order Columns:** On the Order Columns page, select the order for displaying the chosen fields.

   - **Limit Report Results:** On the Select Criteria page, choose the appropriate settings from the drop-down lists, then use the filter options filter options to limit the report to records with specific data.

     The report wizard supports up to 10 filters. On reports with more than 10, additional filters are dropped and the report shows an error for any filter logic.

293

> 💡 Tip: To use a tabular report on a dashboard, first limit the row count, by setting the `Rows to Display` option, the sort column, and the order on the Select Criteria page of the report. You can't use gauge or metric components on dashboards using tabular reports.

- **Chart Settings:** On the Select the Chart Type and Report Highlights step of the report wizard, set chart properties to display your report data in a chart. Charts are available only for summary and matrix reports.

5. Click **Run Report** to view the report, or click **Export Details** to save the report as an Excel file or other format.

> 💡 Tip: Customizing your reports can require running them a few times as you adjust the report criteria and options. We recommend using a filter that gives you a smaller sampling of data until you are finished customizing the report and ready to save.

Edit a Numeric Bucket Field

A numeric bucket helps you sort data that can be described in terms of numbers.

In the Edit Bucket Field overlay for a numeric field:

1. For `Source Column`, select the field you want to bucket.

2. Enter a bucket field name. This appears as the column name in the report.

   Since a bucket field is intended to have multiple buckets (known as "ranges" in numeric bucket fields) within it, a good name for a bucket field describes the scope of the ranges. For example, a bucket field named "Size" could have ranges of "Small," "Medium," and "Large".

3. Define your ranges by entering a number and a name. The range names appear as values in your new column.

   Each range is greater than the lower number up to and including the higher number.

   To remove all ranges and start over, click **Clear All**.

4. To move all empty values to the bucket containing the value zero, enable `Treat empty` *`source column`* `values in the report as zeros.`

   If this is disabled, unbucketed values appear as a dash (-) in the column.

5. Click **OK**.

In a report, numeric bucket columns are sorted by range values.

SEE ALSO:

Numeric Bucketing Example: Deal Size

Numeric Bucketing Example: Deal Size

To gain insight into your deals, use bucketing to group by deal size instead of looking at individual deals. This lets you concentrate on the large deals that affect your quota the most.

1. Create or edit a standard opportunity report.

2. In the Fields pane of the report builder, double-click **Add Bucket Field** or drag it into the report.

3. For Source Column, select `Amount`.

4. For Bucket Field Name, enter *Deal Size*.

5. Under Define Ranges, enter *1000* in the first row. This represents the maximum for a small deal. Name this range *Small*.

6. Click **Add** and enter *25000* in the second row. This represents the maximum for a medium-size deal. Name this range *Medium*.

7. By default, the last range is any amount over the previous range. You don't need to enter a number for this range. Name this range *Large*.

8. Click **OK**.

With numeric bucket fields, each range is greater than the lower number, up to and including the higher number. Once you've set up this bucket field, amounts will be bucketed as follows:

Amounts	Bucket
1000 or less	Small
1001 to 25000	Medium
25001 or more	Large

SEE ALSO:

Edit a Numeric Bucket Field

Edit a Picklist Bucket Field

A picklist bucket field contains items that can be selected from a list.

> ✏️ **Note:** The following picklist types can't be bucketed.
>
> - Record types
> - Divisions
> - Multi-value picklists
> - The `Type` picklist in Activity reports

In the Edit Bucket Field overlay for a picklist field:

1. For `Source Column`, select the field you want to bucket.

2. Enter a bucket field name. This appears as the column name in the report.

   Since a bucket field is intended to have multiple buckets within it, a good name for a bucket field describes the scope of the buckets. For example, a bucket field named "Priority" could have "High," "Medium," and "Low" buckets.

   > ⛔ **Important:** Picklist bucket names must include one or more letters or symbols. If a picklist bucket field includes a bucket whose name contains only numbers, the bucket field cannot be saved.

3. To create a bucket, click **New Bucket** and enter a bucket name. Create multiple buckets to group your report records.

4. To find a particular value in the list of values, type the first few characters of its name in the Quick Find box. As you type, items that match your search terms appear in the menu.

5. Select values and drag them into a bucket. Alternatively, select values, click **Move To**, and select a bucket or enter a new bucket name.

   While you're bucketing values, use these functions as needed:

   - To show all the values in the report, click **All Values**.
   - To show the values for a particular bucket, click the bucket name.
   - To remove values from a bucket, select the values and drag them to another bucket, or drag them to **Unbucketed Values**.

   > ✏️ **Note:** You can bucket only active picklist values. Inactive picklist values aren't shown.

6. To move all unbucketed values into a bucket named "Other", enable `Show unbucketed values as "Other"`. If this is disabled, unbucketed values appear in the bucket column with the value name.

7. Click **OK**.

In a report, picklist bucket columns are sorted by the bucket position as shown in the Edit Bucket Field dialog box, followed by "Other" if `Show unbucketed values as "Other"` is enabled, or the unbucketed picklist item names if `Show unbucketed values as "Other"` is disabled.

SEE ALSO:

Picklist Bucketing Example: Industry Types

Picklist Bucketing Example: Industry Types

Use a picklist bucket field to sort your accounts by their industry.

1. Create or edit a standard accounts report, making sure at least a few records appear in the report.

2. In the Fields pane of the report builder, double-click **Add Bucket Field** or drag it into the report.

3. For Source Column, select `Industry`.

4. For Bucket Field Name, enter `Industry Types`.

5. Click **New Bucket** and name the bucket `IT`.

6. Click **New Bucket** and name the bucket `Bio Tech`.

7. Click **New Bucket** and name the bucket `Telecom`.

8. Select values and drag them into the appropriate buckets.

9. Enable `Show unbucketed values as "Other"`.

10. Click **OK**.
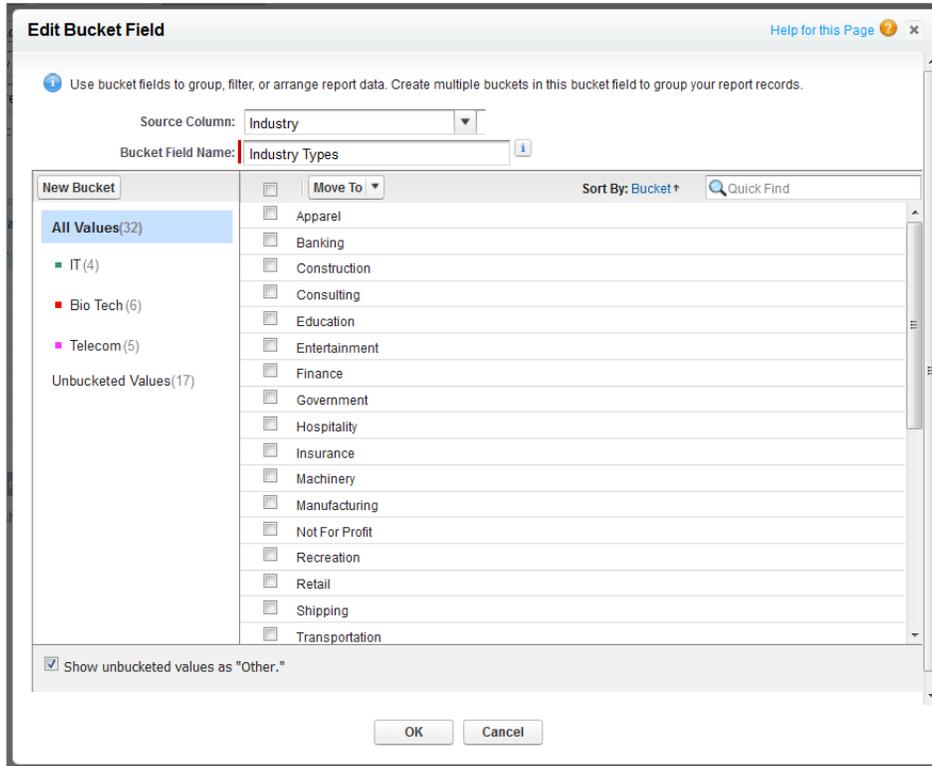
SEE ALSO:

Edit a Picklist Bucket Field

Edit a Text Bucket Field

A text bucket helps you sort out values that consist of words or phrases.

> 📝 Note: The following text types can't be bucketed.
> - Text area
> - Text area—long
> - Text area—rich
> - Text area—encrypted
> - URL
> - Date
> - Date/Time

In the Edit Bucket Field overlay for a text field:

1. For `Source Column`, select the field you want to bucket.

2. Enter a bucket field name. This appears as the column name in the report.

   Since a bucket field is intended to have multiple buckets within it, a good name for a bucket field describes the scope of the buckets. For example, a bucket field named "Region" could have "East," "West," and "Central" buckets.

3. To create a bucket, click **New Bucket** and enter a bucket name. Create multiple buckets to group your report records.

4. To find a particular value in the list of values, type all or part of its name in the **Search for values...** box and click **Search**, or leave the box empty and click **Search**.

   The search returns up to 200 values.

5. Select values and drag them into a bucket. Alternatively, select values, click **Move To**, and select a bucket or enter a new bucket name.

   While you're bucketing values, use these functions as needed:

   - Use Enter Values to enter the exact name of a value you want to bucket, or to bucket values that may appear in your report later.
   - To show the values for a particular bucket, click the bucket name.
   - To remove values from a bucket, select the values, select **Move To**, and select a bucket or enter a new bucket name.

6. To move all unbucketed values into a bucket named "Other," enable `Show unbucketed values as "Other"`. If this is disabled, unbucketed values appear in the bucket column with the value name.

7. Click **OK**.

In a report, text bucket columns are sorted in alphanumeric order.

SEE ALSO:

    Enter Values for Text Buckets

    Text Bucketing Example: Strategic Accounts

Enter Values for Text Buckets

A text bucket field needs a descriptive name and two or more buckets containing text strings.

If you know the exact value you want to bucket, you can use the Enter Values function to quickly bucket it without searching. This is useful if your report has a large number (such as millions) of values and searching for a value is slow. You can also use this method to enter and bucket values that may appear in your report later.

In the Edit Bucket Field overlay for a text field:

1. Click **Enter Values**.

2. Select the bucket where you want to move the values. If you select **New Bucket**, enter a name for the bucket.

3. Type one or more values in the box. Enter multiple values on separate lines.

4. Click **Move**.

SEE ALSO:

[Edit a Text Bucket Field](#)

Text Bucketing Example: Strategic Accounts

Use a text bucket to sort accounts into general categories that you can act on.

1. Create or edit a standard account report, making sure at least a few records appear in the report.

2. In the Fields pane of the report builder, double-click **Add Bucket Field** or drag it into the report.

3. For Source Column, select `Account Name`.

4. For Bucket Field Name, enter *`Strategic`*.

5. Click **New Bucket** and name the bucket *`IBM`*.

6. Click **New Bucket** and name the bucket *`Dell`*.

7. Click **New Bucket** and name the bucket *`HP`*.

8. To show the available values, leave the **Search for values...** box empty and click **Search**.

9. Select values and drag them into the appropriate buckets.

10. Enable `Show unbucketed values as "Other"`.

11. Click **OK**.
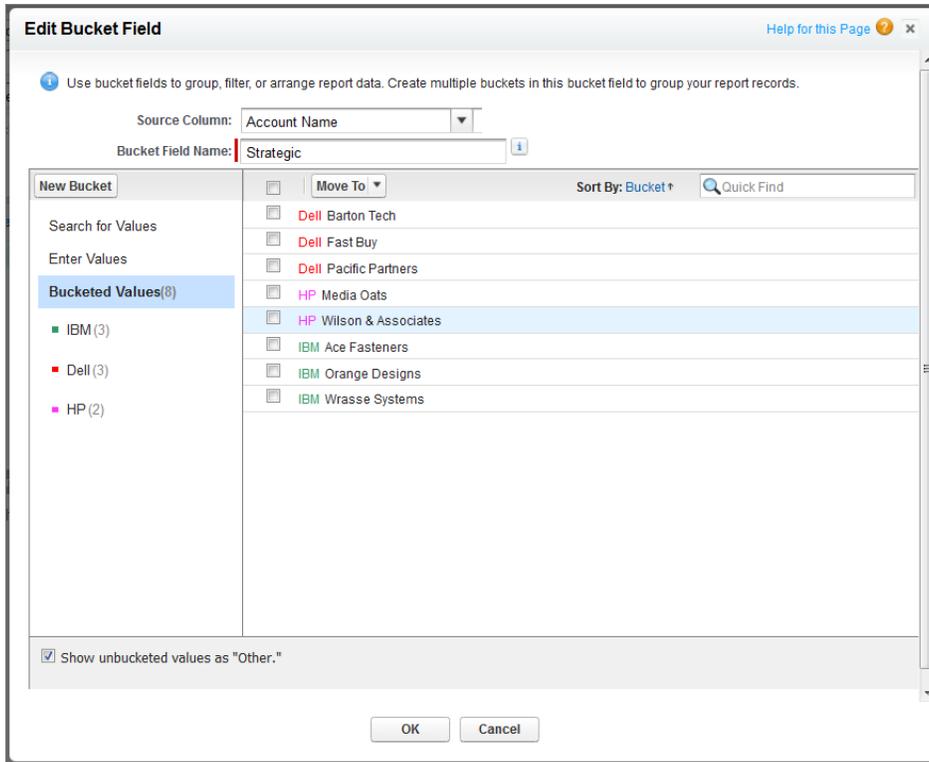
SEE ALSO:

Edit a Text Bucket Field

Use a Summary Function in a Custom Summary Formula

Summary functions let you use grouping values in custom summary formulas for summary, matrix, and joined reports. There are two summary functions: `PARENTGROUPVAL` and `PREVGROUPVAL`.

1. Double-click **Add Formula** in the Fields pane.

2. In the Custom Summary Formula dialog, under Functions, select *Summary*.

3. Select `PARENTGROUPVAL` or `PREVGROUPVAL`.

4. Select the grouping level and click **Insert**.

5. Define the formula, including where to display the formula.

6. Click **OK**.

SEE ALSO:

    Evaluate Groups and Totals with Summary Formulas

    PARENTGROUPVAL and PREVGROUPVAL

EDITIONS

Available in: Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

---

USER PERMISSIONS

To create, edit, and delete reports:

- **Legacy Folder Sharing**
  Create and Customize Reports

  AND

  Report Builder

  Enhanced Folder Sharing
  Create and Customize Reports

  AND

  Report Builder

Run and Read a Report

Click on a report's name to run it. After running a report, there are a series of tools you can use to help you read the information.

From the report run page in Lightning Experience, you can:

- Show or hide a report chart.
- Edit report filters (except for locked filters).
- Launch the report builder to edit the report.
- Refresh the report.
- Subscribe to the report.
- Export the report.
- Delete the report.

From the report run page in Salesforce Classic, you can:

- Display a Chatter feed of updates and posts about the report.
- View `Report Generation Status`.
- Choose hierarchy options.
- Change top-level groupings using the `Summarize information by` drop-down.
- Change `Time Frame` and `View` options.
- Click **Run Report** to immediately run or schedule the report.
- Click **Show Details** to view all data or **Hide Details** to show only summary information.
- Click **Customize** to open the report in report builder.
- Print or export the report.
- Click **Subscribe** to set up report notifications, which notify you whenever certain metrics in the report meet your conditions.
- Click **Edit** next to the chart to change its properties, or click **Large**, **Medium**, or **Small** to change its size.
- Sort report results.
- Edit or clear applied filter criteria.
- For summary and matrix reports, filter selected rows by a particular field. Select the rows you want to view, choose a field to group by, then click **Drill Down**. For example, if you created an opportunity report that displays deals by industry, you can select the industries that are doing well and group by `Product Name` to see which products are selling in that industry. Click **Clear** to remove the drill-down filter.

Dashboards

Dashboards help you visually understand changing business conditions so you can make decisions based on the real-time data you have gathered with reports. Use dashboards to help users identify trends, sort out quantities, and measure the impact of their activities. Before building, reading, and sharing dashboards, review these dashboard basics.

Watch a Demo: ▶ An Overview of Dashboards (Salesforce Classic)

📝 **Note:** Dashboards in Group Edition organizations are view-only.

📝 **Note:** Dashboads do not support localization.

As you prepare to curate your Salesforce data with dashboards, keep these tips in mind:

- Reports provide all the data shown in a dashboard. Dashboards can show data from more than one report.
- When refreshing a dashboard, all the data-providing reports must run. If the reports take a long time to run, then the dashboard does too.
- Dashboards are shared via folders. Whomever has permission to the folder your dashboard is saved in also has access to your dashboard. Ensure that you save your dashboard in an appropriate folder.

Before creating or reading a dashboard, familiarize yourself with these features and concepts.

Dashboard Editor

The Dashboard Editor is a visual, drag-and-drop tool which you use to create dashboards and edit existing ones. The Dashboard Editor is where you add, edit, and arrange dashboard components.

To launch the Dashboard Editor, click **New Dashboard**.

Components

Dashboards are made up of components. Each component contains a chart or metric which shows data from one report. Different components can show data from different reports.

Running User (Viewing As)

In Salesforce, different users have different permissions to access data. A dashboard only displays data that the dashboard's running user can access.

For example, say you're viewing a dashboard describing Leads. Emily is a sales operations manager who sees all Leads, and Marcus is a direct sales specialist who sees only the leads he owns. If Emily is the dashboard's running user, then the dashboard shows data about all the Leads in Salesforce. If Marcus is the running user, then the dashboard only shows data about Leads that Marcus owns.

1. Build a Dashboard

   When you're ready to share Salesforce data with colleagues, build a dashboard. Dashboards let you curate data from reports using charts, tables, and metrics. If your colleagues need more information, then they're able to view your dashboard's data-supplying reports.

2. Filter a Dashboard

   Dashboard filters make it easy to provide different combinations of data from a single dashboard. You don't need separate dashboards for different sets of users — give each group a filter that makes sense for them. When you use a filter on a dashboard, the filtered view is shown again the next time you visit the dashboard.

3. Dynamic Dashboards: Choose Who People View a Dashboard As

   Say that your sales people can only view their own opportunities, but you'd like to review all opportunities closed in the last quarter. Create a dashboard and let people view the dashboard as you (or anyone else who can see all opportunities). When your sales people open the dashboard, they see info about all opportunities instead of only their opportunities. (Their data access in Salesforce remains unchanged. They can only see more data in your dashboard.)

4. Subscribe to or Schedule Dashboards

   Set up automatic dashboard refreshes and receive refreshed dashboard results by email on a schedule that you set. If you start your week by reviewing your Sales Overview dashboard, expedite your morning by subscribing to the dashboard. After subscribing, you can have it automatically refreshed and in your email inbox each Monday morning at 8:00 AM.

5. Link from Dashboard Components in Lightning Experience

   A sales overview dashboard is a great way for teams to collaborate on opportunities because it provides a complete picture of your pipeline. Dashboard components already link to source reports, so you can get more details, but what if you're ready to take action?

6. Work with Dashboards

   Click on a dashboard's name to run it. Dashboard charts are interactive, so be sure to hover and click on them to get more info! From refreshing dashboard data to sharing dashboard components, there are lots of ways to use dashboards to keep yourself and your team informed and collaborating.

7. Organize Dashboards

   Keep your dashboards at your fingertips by embedding them around Salesforce, printing them, or deleting dashboards that you don't need anymore.

8. Why Doesn't My Dashboard Display the Data I Expect?

   If you're not seeing data you expect, refresh for latest data, check that you have the right running user, and verify your dashboard data sources.

9. Improve Dashboard Performance: Best Practices

   If a dashboard is running slowly, it likely because of inefficient source reports or because many people are refreshing it at once. By optimizing source reports and planning dashboard refreshes, a slow dashboard can be sped up. Follow the tips in this guide to speed up sluggish dashboards!

10. Improve Dashboard Performance

    Dashboard refresh times can vary based on available resources. Since Salesforce is built on a multi-tenant structure, dashboard refreshes are processed in a queue. The refresh tend to complete faster off peak hours, when more resources are available. Other factors like caching can also affect dashboard refresh times.

Build a Dashboard

When you're ready to share Salesforce data with colleagues, build a dashboard. Dashboards let you curate data from reports using charts, tables, and metrics. If your colleagues need more information, then they're able to view your dashboard's data-supplying reports.

1. Build a Lightning Experience Dashboard
   Build a Lightning Experience dashboard to provide a graphical view of the Salesforce data in reports.
2. Build a Salesforce Classic Dashboard
   Build a dashboard to provide a graphical view of the data in your reports.

Build a Lightning Experience Dashboard

USER PERMISSIONS

To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards

	Enhanced Folder Sharing Edit My Dashboards
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data
	Enhanced Folder Sharing Manage Dashboards in Public Folders
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards
	Enhanced Folder Sharing Manage Dynamic Dashboards

Build a Lightning Experience dashboard to provide a graphical view of the Salesforce data in reports.

Dashboards are built with source reports, filters, and components.

Source reports are reports that provide data for components. Each component has one source report. Different components can have different source reports. The fields available for dashboard filters are the fields available in the objects that source reports are based on. Create source reports in the Report Builder.

Filters let dashboard readers scope the data they see in the dashboard to a particular view.

Components are the visual "blocks" of a dashboard. Each component is either a chart, gauge, metric, or table.

	Use a chart when you want to show data graphically. You can choose from a variety of ...
	Use a gauge when you have a single value that you want to show within a range of custom values
	Use a metric when you have one key value to display.
	Use a table to show a set of report data in column form.

This topic is about creating dashboards for reports in Lightning Experience. For information on creating dashboards in Salesforce Classic or Analytics Cloud, review these articles:

- Build a Salesforce Classic Dashboard
- Build Tableau CRM Dashboards

💡 **Tip:** You can clone a dashboard to quickly create a dashboard with the same properties and components as the one you're viewing. Click **Save As** (or **Clone**), modify the dashboard settings, and save.

1. If necessary, create the source reports containing the data you want to display.

   ⛔ **Important:** Be sure to store these reports in folders that intended dashboard viewers can access.

2. From the Dashboards tab, click **New Dashboard**.

3. Name the dashboard.

    Optionally, provide a short description. Then, place the dashboard in a folder. Private Dashboards is the default folder.

4. To add a component to the dashboard, click **+ Component**.

    a. Choose a source report for the component, then click **Select**.

    b. Customize how the component displays data, then click **Add**.

    c. Arrange and resize the component as necessary.

    d. To edit an existing component, click the pencil icon ( ). To remove a component, click the X icon (✕).

5. To add a filter to the dashboard, click **+ Filter**.

    a. From the `Field` drop-down, select a field to filter on. The drop-down shows fields that can be used to filter all the dashboard's components. If there are equivalent fields for your selection, hover over the info icon (🛈) to see them.

    b. Give the filter a `Display Name` to identify it. If the filter has many equivalent fields, we consider using a name that works for all components.

    c. Assign values to the filter by clicking **Add Filter Value**.

6. To specify who people view the dashboard as, click the gear icon (⚙) to open the Properties menu.

    a. Under `Name`, optionally rename the dashboard.

    b. Under `Description`, optionally describe the dashboard.

    c. Under `Folder`, optionally move the dashboard into another folder. To switch folders, first click the X icon (✕), then select another folder.

    d. Under `View Dashboard As`, choose who people view the dashboard (and all source reports) as. People could see more or less data than they normally see elsewhere in Salesforce. Be careful not to reveal sensitive information to too broad an audience.

        * **Me** — People view the dashboard as you.

        * **Another person** — People view the dashboard as whomever you choose. You are the default person. To choose someone else, click the X icon (✕).

        * **The dashboard viewer** — People see data as themselves, according to their own access to data. These types of dashboards are often called *dynamic dashboards*.

            Your organization can have up to 5 dynamic dashboards for Enterprise Edition, 10 for Unlimited and Performance Edition, and 3 for Developer Edition. Dynamic dashboards aren't available in other editions. Additional dynamic dashboards may be available for purchase.

            Take note of these dynamic dashboard limitations:

            – Dynamic dashboards don't support following components.

            – You can't save dynamic dashboards in private folders.

            – You can't schedule refreshes for dynamic dashboards. They must be refreshed manually.

        * **Let dashboard viewers choose whom they view the dashboard as** — Optionally, select **Let dashboard viewers choose whom they view the dashboard as** to enable a reader with appropriate user permissions to choose who they view the dashboard as. With the "View My Team's Dashboards" user permission, the reader can view the dashboard as themself or as anyone beneath them in the role hierarchy. With the "View All Data" user permission, the reader can view the dashboard as anyone.

    e. Click **Save**.

7. To customize dashboard theme or color palette, open the Properties menu by clicking the gear icon (⚙).

Customize colors in your dashboard by choosing a light or dark theme and one of 14 color palettes. Themes and color palettes help you organize and curate data in your dashboard. For example, highlight a set of summary metrics by giving them a dark theme and setting them against a light-themed dashboard. This feature is new in Lightning Experience.

    a. From Dashboard Theme, choose **Light** or **Dark** to set the theme for the entire dashboard. To set a single component's theme, edit the component by clicking ✏️ .

    b. From Dashboard Palette, choose one of 14 color palettes. **Wildflowers** is the default color palette. If you need an accessible color palette, choose **Mineral**.

    c. Click **Save**.

8. To switch between a 12-column and 9-column layout, open the Properties menu by clicking the gear icon (⚙).

    a. Under `Dashboard Grid Size`, choose **12 columns** or **9 columns**.

    b. Click **Save**.

9. Click **Save**, then click **Done**.

Your dashboard is done. Review and admire your handiwork!

Edit and Customize Lightning Experience Dashboard Components

Get your Lightning Experience dashboard components to show exactly what you want.

Sort Data in Dashboard Components

Sort dashboard components by groups and measures. Sort ascending or descending. Dashboard components support two-level sorting. So you can sort by Stage and then by Sum of Amount, or by Average Age and then Record Count.

Color Dashboard Data with Themes and Palettes

Customize colors in your dashboard by choosing a light or dark theme and one of 14 color palettes. Themes and color palettes help you organize and curate data in your dashboard. For example, highlight a set of summary metrics by giving them a dark theme and setting them against a light-themed dashboard.

Add a Report to a Dashboard from the Report

Reading a report that you'd like to add to a dashboard? Go from report to dashboard with only a few clicks!

Use Source Report Charts in Dashboard Components

You've already charted data just the way you like in a report. Instead of recreating the chart from scratch, surface it in a dashboard component.

Show Tabular Information with a Lightning Table

Show up to 200 records and 10 columns and from any field in a source report's report type. After grouping data in a Lightning table, you can add measures (sum, average, minimum, and maximum) to the grouped data. If a Lightning table's source report has a summary formula column, then you can show it on the Lightning table.

Set Decimal Places for Numbers in Dashboard Charts, Tables, and Metrics

Because 14.11B is greater than 14B and the difference between 90%, 90.1%, and 90.10001% matters, choose how many decimal places appear after numbers in dashboard components.

Set Axis Range for Dashboard Charts

Choose the range of the x- and y-axes on dashboard charts, or let Salesforce automatically set them.

Switch Between a 12 or 9-Column Dashboard Layout

Both 12-column and 9-column dashboards give you plenty of room to lay out data in easy-to-consume ways.

From the Dashboard tab, you can create, edit, or delete a dashboard using nothing but your keyboard. Use keystrokes to edit dashboards quickly and easily in Lightning Experience.

Edit and Customize Lightning Experience Dashboard Components

USER PERMISSIONS		EDITIONS
To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: Lightning Experience
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: **Essentials**, **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Edit My Dashboards	Available in: both Legacy Folder Sharing and Enhanced Folder Sharing
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data **Enhanced Folder Sharing** Manage Dashboards in Public Folders	
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards **Enhanced Folder Sharing** Manage Dynamic Dashboards	

Get your Lightning Experience dashboard components to show exactly what you want.

Each component has a series of properties that help you customize the component until it shows exactly the data that you want. Refer to this table for help customizing each component's properties, and for ideas about when to use each type of component:

1. Add a new dashboard component by clicking **+ Component**, or edit an existing one by clicking the pencil icon (✏) on the component.

2. Set dashboard component properties.

3. If adding a new component, click **Add**. Or, if editing an existing component, click **Update**.

4. Click **Save**.

For help setting dashboard component properties, refer to this table.

Component	Description
Bar Carts 	**Use** A bar chart shows values as horizontal lengths, so this format can be good for comparing distance or time. Use a bar chart when you have a summary report with a single grouping, or you only want to display one grouping. For example, to see the amount in each sales stage in a report, select `Sum of Amount` as the X-axis and `Stage` as the Y-axis. The chart displays 1 bar for each stage, with the length proportional to the total opportunity amount. **Properties** • **Use chart settings from report** – Display the source report's chart. To customize the chart, either deselect **Use chart settings from report** or edit the report chart. • **Y-Axis** and **X-Axis** – Choose a group (horizontal bar charts) or a measure (vertical bar chart) to display on each axis. If applicable, click **+ Group** or **+ Measure** to chart more groups or measures. • **Display Units** – Choose how to display values. Shortened Number automatically chooses a unit appropriate for your data and abbreviates it. Shorten numbers to display approximations that take up less space (for example, 1,876 becomes 1.9k), or view the full number for to-the-digit accuracy. Full Number displays the actual values. Other options are Hundreds, Thousands, Millions, Billions, and Trillions. Choose what makes most sense for your data. • **Show Values** – Specify whether chart segments display values, or not. • **Show Chatter Photos** (Horizontal bar charts only) – Choose whether to display Chatter photos, or not. Table must be grouped by a user or group name field, such as Opportunity Owner. • **Y-Axis Range** or **X-Axis** – Set the axis range. To let Salesforce automatically set axis range, select **Automatic**. To set your own range, select **Customize** and then enter the range limits in **Min** and **Max**. • **Decimal Places** – Choose how many decimal places appear on numbers. To let Salesforce automatically set how many decimal places appear, select **Automatic**. To set your own, choose a number between 0 and 5.

Component	Description

- **Sort By** – Choose how to sort groups or measures displayed in the bar chart. Click the arrow icon to change sort direction between ascending (↑) and descending (↓).

- **Then Sort By** – If applicable, choose how to sort second-level groups or measures displayed in the bar chart. Click the arrow icon to change sort direction between ascending (↑) and descending (↓).

- **Custom Link** – Link from a dashboard component to a Salesforce Record or another website. For more information, see Link a Dashboard Component to a Website or Salesforce Record in Salesforce help.

- **Max Values Displayed** – Specify how many groups display in the bar chart.

- **Title** – Give the chart a title.

- **Subtitle** – Give the chart a subtitle.

- **Footer** – Give the chart a footer.

- **Component Theme** – Choose a **Light** or a **Dark** theme. Customize chart colors from the dashboard properties menu. Individual components support unique themes, but colors are only customizable from the dashboard properties menu.

Stacked Bar Charts

Use

Use a stacked bar chart when you have multiple groupings and are interested in the proportions between values in each grouping and each grouping's total.

For example, compare the status of leads by campaign, and also to compare the totals for each status. Set record count as the `X-axis`, status as the `Y-axis`, and campaign as the `Stack By` value. The chart displays a single bar for each status, broken down by campaign, with each campaign shown in a different color.

Properties

- **Use chart settings from report** – Display the source report's chart. To customize the chart, either deselect **Use chart settings from report** or edit the report chart.

- **Y-Axis** and **X-Axis**– Choose a group (horizontal bar charts) or a measure (vertical bar chart) to display on each axis. If applicable, click **+ Group** or **+ Measure** to chart more groups or measures.

- **Stack By** – Choose which group to stack by.

Component	Description

- **Display Units** – Choose how to display values. Shortened Number automatically chooses a unit appropriate for your data and abbreviates it. Shorten numbers to display approximations that take up less space (for example, 1,876 becomes 1.9k), or view the full number for to-the-digit accuracy. Full Number displays the actual values. Other options are Hundreds, Thousands, Millions, Billions, and Trillions. Choose what makes most sense for your data.

- **Stack to 100%** – Specify whether to stack the bar chart to 100%, or not.

  Stack to 100% when you have multiple groupings and are interested in the proportions between values in each grouping and each grouping's total.

- **Y-Axis Range** or **X-Axis** – Set the axis range. To let Salesforce automatically set axis range, select **Automatic**. To set your own range, select **Customize** and then enter the range limits in **Min** and **Max**.

- **Decimal Places** – Choose how many decimal places appear on numbers. To let Salesforce automatically set how many decimal places appear, select **Automatic**. To set your own, choose a number between 0 and 5.

- **Sort By** – Choose how to sort groups or measures displayed in the bar chart. Click the arrow icon to change sort direction between ascending (↑) and descending (↓).

- **Then Sort By** – If applicable, choose how to sort second-level groups or measures displayed in the bar chart. Click the arrow icon to change sort direction between ascending (↑) and descending (↓).

- **Custom Link** – Link from a dashboard component to a Salesforce Record or another website. For more information, see Link a Dashboard Component to a Website or Salesforce Record in Salesforce help.

- **Max Values Displayed** – Specify how many groups display in the bar chart.

- **Title** – Give the chart a title.

- **Subtitle** – Give the chart a subtitle.

- **Footer** – Give the chart a footer.

- **Component Theme** – Choose a **Light** or a **Dark** theme. Customize chart colors from the dashboard properties menu. Individual components support unique themes,

Component	Description
	but colors are only customizable from the dashboard properties menu.

Line Charts

Use

Line charts are good for showing changes in the value of an item over a series of points in time, such as week to week. Use a line chart when you have one important grouping representing an ordered set of data and one value to show.

Line charts are useful for showing data over time. For example, to see the numbers of leads created each month in a report, set record count as the `Y-axis` and created month for the `X-axis`. The chart displays a line connecting the record count totals for each month. Salesforce does not plot missing (null) values.

Properties

- **Use chart settings from report** – Display the source report's chart. To customize the chart, either deselect **Use chart settings from report** or edit the report chart.

- **X-Axis** – Choose a group to display on the x-axis. If applicable, click **+ Group** to chart another group.

- **Y-Axis** – Choose a measure to display on the y-axis. If applicable, click **+ Measure** to chart another measure.

- **Cumulative** – Specify whether the line chart is cumulative, or not.

  Use a cumulative line chart when you have one important grouping representing an ordered set of data and one value to show, summed over time.

  For example, see the total number of closed opportunities by day for each of the last three months. Set amount as the `Y-axis`, closing day as the `X-axis`, and closing month as the `Group` value. The chart displays a line for each month, with the line's height representing the cumulative number of closed opportunities up to and including that day.

- **Display Units** – Choose how to display values. Shortened Number automatically chooses a unit appropriate for your data and abbreviates it. Shorten numbers to display approximations that take up less space (for example, 1,876 becomes 1.9k), or view the full number for to-the-digit accuracy. Full Number displays the actual values. Other options are Hundreds, Thousands, Millions, Billions, and Trillions. Choose what makes most sense for your data.

Component	Description

- **Y-Axis Range** – Set the axis range. To let Salesforce automatically set axis range, select **Automatic**. To set your own range, select **Customize** and then enter the range limits in **Min** and **Max**.

- **Decimal Places** – Choose how many decimal places appear on numbers. To let Salesforce automatically set how many decimal places appear, select **Automatic**. To set your own, choose a number between 0 and 5.

- **Show Values** – Specify whether points on the line display values, or not.

- **Sort By** – Choose how to sort groups or measures displayed in the bar chart. Click the arrow icon to change sort direction between ascending (↑) and descending (↓).

- **Then Sort By** – If applicable, choose how to sort second-level groups or measures displayed in the bar chart. Click the arrow icon to change sort direction between ascending (↑) and descending (↓).

- **Custom Link** – Link from a dashboard component to a Salesforce Record or another website. For more information, see Link a Dashboard Component to a Website or Salesforce Record in Salesforce help.

- **Max Values Displayed** – Specify how many groups display in the line chart.

- **Title** – Give the chart a title.

- **Subtitle** – Give the chart a subtitle.

- **Footer** – Give the chart a footer.

- **Component Theme** – Choose a **Light** or a **Dark** theme. Customize chart colors from the dashboard properties menu. Individual components support unique themes, but colors are only customizable from the dashboard properties menu.

Donut Charts

Use

Use a donut chart when you have multiple groupings and want to show both the proportion of a single value for each grouping against the total, and the total amount itself.

For example, see the breakdown of your case queue by case status in a report, and the total number of cases. Set record count for `Value` and status for `Sliced By`. The chart displays a donut made up of wedges, each wedge representing a case status. Wedge size is proportional to the

Component	Description
	numbers of cases. The total number of cases for all statuses is shown in the middle.

Properties

- **Use chart settings from report** – Display the source report's chart. To customize the chart, either deselect **Use chart settings from report** or edit the report chart.

- **Value** – Choose a measure to display on the donut chart.

- **Sliced By** – Choose a group to slice the donut by.

- **Display Units** – Choose how to display values. Shortened Number automatically chooses a unit appropriate for your data and abbreviates it. Shorten numbers to display approximations that take up less space (for example, 1,876 becomes 1.9k), or view the full number for to-the-digit accuracy. Full Number displays the actual values. Other options are Hundreds, Thousands, Millions, Billions, and Trillions. Choose what makes most sense for your data.

- **Show Values** – Show values on each donut group (slice). Each value is representative of a summary value, like average or record count, for the group.

- **Show Percentages** – Show percentages on each donut group (slice).

- **Combine Small Groups into "Others"** – Show each group (slice) that represents 3% or less of the total as a new group called "Others".

- **Show Total** – Show a total in the center of the donut chart.

- **Decimal Places** – Choose how many decimal places appear on numbers. To let Salesforce automatically set how many decimal places appear, select **Automatic**. To set your own, choose a number between 0 and 5.

- **Label Groups** – Specify whether to label sliced wedges with values, percentages, or no labels.

- **Sort By** – Choose how to sort groups or measures displayed in the bar chart. Click the arrow icon to change sort direction between ascending (↑) and descending (↓).

- **Custom Link** – Link from a dashboard component to a Salesforce Record or another website. For more information, see Link a Dashboard Component to a Website or Salesforce Record in Salesforce help.

- **Max Values Displayed** – Specify how many groups display in the donut chart.

Component	Description
	• **Title** – Give the chart a title.
	• **Subtitle** – Give the chart a subtitle.
	• **Footer** – Give the chart a footer.
	• **Component Theme** – Choose a **Light** or a **Dark** theme. Customize chart colors from the dashboard properties menu. Individual components support unique themes, but colors are only customizable from the dashboard properties menu.
Funnel Charts	**Use**
	Use a funnel chart when you have multiple groupings in an ordered set and want to show the proportions among them.
	For example, to see the number of opportunities in each stage in a report, set amount for `Value` and stage for `Color By`. Since the `Opportunity: Stage` field is an ordered picklist, the stages are sorted in the same order as the picklist. Funnel charts are useful for showing the flow of opportunities through the stages; a substantially larger segment can indicate a bottle-neck at that stage.
	Properties
	• **Use chart settings from report** – Display the source report's chart. To customize the chart, either deselect **Use chart settings from report** or edit the report chart.
	• **Value** – Choose a measure to display on the funnel chart.
	• **Display Units** – Choose how to display values. Shortened Number automatically chooses a unit appropriate for your data and abbreviates it. Shorten numbers to display approximations that take up less space (for example, 1,876 becomes 1.9k), or view the full number for to-the-digit accuracy. Full Number displays the actual values. Other options are Hundreds, Thousands, Millions, Billions, and Trillions. Choose what makes most sense for your data.
	• **Color By** – Choose a group to display in the funnel chart.
	• **Label Groups** – Specify whether to label sliced wedges with values, percentages, or no labels.
	• **Show Values** – Specify whether chart segments display values, or not.
	• **Show Percentages** – Specify whether chart segments display percentages, or not.
	• **Combine Small Groups into "Others"** – Show each group (slice) that represents 3% or less of the total as a new group called "Others".

Component	Description
	• **Decimal Places** – Choose how many decimal places appear on numbers. To let Salesforce automatically set how many decimal places appear, select **Automatic**. To set your own, choose a number between 0 and 5.
	• **Sort By** – Choose how to sort groups or measures displayed in the bar chart. Click the arrow icon to change sort direction between ascending (↑) and descending (↓).
	• **Custom Link** – Link from a dashboard component to a Salesforce Record or another website. For more information, see Link a Dashboard Component to a Website or Salesforce Record in Salesforce help.
	• **Max Values Displayed** – Specify how many groups display in the funnel chart.
	• **Title** – Give the chart a title.
	• **Subtitle** – Give the chart a subtitle.
	• **Footer** – Give the chart a footer.
	• **Component Theme** – Choose a **Light** or a **Dark** theme. Customize chart colors from the dashboard properties menu. Individual components support unique themes, but colors are only customizable from the dashboard properties menu.

Scatter Charts

Use

Use scatter charts to show meaningful information using one or two groups of report data plus measures.

For example, see how stage duration correlates with the number of activities for opportunities. Group your report by `Opportunity Name`. Then set `X-Axis` on the chart to `Record Count` and `Y-Axis` to `Stage Duration`. The chart shows a dot for each opportunity. At a glance, you can tell if the stage duration is shorter for opportunities that have more activities.

Properties

• **Use chart settings from report** – Display the source report's chart. To customize the chart, either deselect **Use chart settings from report** or edit the report chart.

• **X-Axis** – Choose a measure for the x-axis.

• **Y-Axis** – Choose a measure for the y-axis.

• **Detail** – Choose a group to plot on the scatter chart.

• **Decimal Places** – Choose how many decimal places appear on numbers. To let Salesforce automatically set

Component	Description
	how many decimal places appear, select **Automatic**. To set your own, choose a number between 0 and 5.
	• **Display Units** – Choose how to display values. Shortened Number automatically chooses a unit appropriate for your data and abbreviates it. Shorten numbers to display approximations that take up less space (for example, 1,876 becomes 1.9k), or view the full number for to-the-digit accuracy. Full Number displays the actual values. Other options are Hundreds, Thousands, Millions, Billions, and Trillions. Choose what makes most sense for your data.
	• **Add** – Choose a second group to plot on the scatter chart.
	• **Sort By** – Choose how to sort groups or measures displayed in the bar chart. Click the arrow icon to change sort direction between ascending (↑) and descending (↓).
	• **Then Sort By** – If applicable, choose how to sort second-level groups or measures displayed in the bar chart. Click the arrow icon to change sort direction between ascending (↑) and descending (↓).
	• **Custom Link** – Link from a dashboard component to a Salesforce Record or another website. For more information, see Link a Dashboard Component to a Website or Salesforce Record in Salesforce help.
	• **Max Values Displayed** – Specify how many groups display in the scatter chart.
	• **Title** – Give the chart a title.
	• **Subtitle** – Give the chart a subtitle.
	• **Footer** – Give the chart a footer.
	• **Component Theme** – Choose a **Light** or a **Dark** theme. Customize chart colors from the dashboard properties menu. Individual components support unique themes, but colors are only customizable from the dashboard properties menu.

Gauge Charts	Use
	A gauge is used to see how far you are from reaching a goal. It displays a single value, such as closed deals.

For example, use a gauge to see how close you are to meeting your target revenue. |

Component	Description
	Properties

Properties

- **Use chart settings from report** – Display the source report's chart. To customize the chart, either deselect **Use chart settings from report** or edit the report chart.
- **Measure** – Choose a measure for the gauge chart.
- **Display Units** – Choose how to display values. Shortened Number automatically chooses a unit appropriate for your data and abbreviates it. Shorten numbers to display approximations that take up less space (for example, 1,876 becomes 1.9k), or view the full number for to-the-digit accuracy. Full Number displays the actual values. Other options are Hundreds, Thousands, Millions, Billions, and Trillions. Choose what makes most sense for your data.
- **Shorten Number** – Choose whether to shorten numbers, or not.

  Shorten numbers to display approximations that take up less space (for example, 1,876 becomes 1.9k), or view the full number for to-the-digit accuracy.

- **Show Range** – Choose whether to show or hide a range.
- **Decimal Places** – Choose how many decimal places appear on numbers. To let Salesforce automatically set how many decimal places appear, select **Automatic**. To set your own, choose a number between 0 and 5.
- **Segment Ranges** – Choose breakpoints and colors for segments on the gauge.
- **Custom Link** – Link from a dashboard component to a Salesforce Record or another website. For more information, see Link a Dashboard Component to a Website or Salesforce Record in Salesforce help.
- **Title** – Give the chart a title.
- **Subtitle** – Give the chart a subtitle.
- **Footer** – Give the chart a footer.
- **Component Theme** – Choose a **Light** or a **Dark** theme. Customize chart colors from the dashboard properties menu. Individual components support unique themes, but colors are only customizable from the dashboard properties menu.

Metric Components	Use
	Use a metric when you have one key value to display.

Component	Description
	For example, show the total number of customer support cases filed today.

Properties

- **Use chart settings from report** – Display the source report's chart. To customize the chart, either deselect **Use chart settings from report** or edit the report chart.
- **Measure** – Choose a measure for the metric chart.
- **Display Units** – Choose how to display values. Shortened Number automatically chooses a unit appropriate for your data and abbreviates it. Shorten numbers to display approximations that take up less space (for example, 1,876 becomes 1.9k), or view the full number for to-the-digit accuracy. Full Number displays the actual values. Other options are Hundreds, Thousands, Millions, Billions, and Trillions. Choose what makes most sense for your data.
- **Shorten Number** – Choose whether to shorten numbers, or not.

  Shorten numbers to display approximations that take up less space (for example, 1,876 becomes 1.9k), or view the full number for to-the-digit accuracy.

- **Show Range** – Choose whether to show or hide a range.
- **Segment Ranges** – Choose breakpoints and colors for segments on the component.
- **Decimal Places** – Choose how many decimal places appear on numbers. To let Salesforce automatically set how many decimal places appear, select **Automatic**. To set your own, choose a number between 0 and 5.
- **Custom Link** – Link from a dashboard component to a Salesforce Record or another website. For more information, see Link a Dashboard Component to a Website or Salesforce Record in Salesforce help.
- **Title** – Give the component a title.
- **Subtitle** – Give the component a subtitle.
- **Footer** – Give the component a footer.
- **Component Theme** – Choose a **Light** or a **Dark** theme. Customize chart colors from the dashboard properties menu. Individual components support unique themes, but colors are only customizable from the dashboard properties menu.

Component	Description

Lightning Dashboard Table

Use

Show up to 200 records and 10 columns and from any field in a source report's report type. The available fields in a Lightning dashboard table come from the source report's report type, so you can show fields as columns in Lightning dashboard tables that aren't included as columns in the source report.

Lightning tables aren't available in Salesforce Classic. You can't add them from the Classic dashboard builder, nor can you see them when you view a dashboard in Salesforce Classic.

If a dashboard hasn't been refreshed for 7 days, then Lightning dashboard tables display no data. Instead, they prompt dashboard viewers to refresh the dashboard. After refreshing the dashboard the Lightning dashboard table shows data as normal.

Properties

- **Use chart settings from report** – Display the source report's chart. To customize the chart, either deselect **Use chart settings from report** or edit the report chart.

- **Columns** – Choose columns from fields in the source report's report type. To add a column, search for it and select it from the **Add column...** field.

  To reorder columns, drag them up or down. The top columns appear on the left-hand side of the table, and the bottom columns appear on the right.

  To remove a column, find it in the Columns list and click ✕ . To remove all columns, click 🗑 .

- **Sort By** – Choose how to sort groups or measures displayed in the bar chart. Click the arrow icon to change sort direction between ascending (↑) and descending (↓).

- **Then Sort By** – If applicable, choose how to sort second-level groups or measures displayed in the bar chart. Click the arrow icon to change sort direction between ascending (↑) and descending (↓).

- **Display Units** – Choose how to display values. Shortened Number automatically chooses a unit appropriate for your data and abbreviates it. Shorten numbers to display approximations that take up less space (for example, 1,876 becomes 1.9k), or view the full number for to-the-digit accuracy. Full Number displays

Component	Description
	the actual values. Other options are Hundreds, Thousands, Millions, Billions, and Trillions. Choose what makes most sense for your data.

- **Show Chatter Photos** – Choose whether to display Chatter photos, or not. Table must be grouped by a user or group name field, such as Opportunity Owner.
- **Show Total** – For numeric columns, Display the sum of all rows in the table.
- **Show Subtotal** – Available when tables have at least two groups. Show subtotals of first-level groups.
- **Add Conditional Highlighting** – Highlight values in the table based on breakpoints and colors set in segment ranges. Click each color to customize it.
- **Decimal Places** – Choose how many decimal places appear on numbers. To let Salesforce automatically set how many decimal places appear, select **Automatic**. To set your own, choose a number between 0 and 5.
- **Max Values Displayed** – Specify how many groups display in the component.
- **Title** – Give the component a title.
- **Subtitle** – Give the component a subtitle.
- **Footer** – Give the component a footer.
- **Component Theme** – Choose a **Light** or a **Dark** theme. Customize chart colors from the dashboard properties menu. Individual components support unique themes, but colors are only customizable from the dashboard properties menu.

Legacy Table Components	Use

🗹 Note: Beginning with the Spring '19 release, you can edit an existing legacy table component, but you can no longer create legacy tables. When creating new dashboard tables, use the Lightning Dashboard Table component

A legacy table component shows columns of data from a custom report in a dashboard. You can use color and scale to help users interpret the report data the legacy table displays.

The default two-column table uses the first grouping and summary field from the chart in the source report. If the report has no chart, default columns are based on the first grouping and summary field in the report.

Two-column legacy tables do not allow null values.

Component	Description
	To use a tabular report as the source report, `Rows to Display` must be set for that report.

Properties

- **Use chart settings from report** – Display the source report's chart. To customize the chart, either deselect **Use chart settings from report** or edit the report chart.

- **Columns** – The columns from the source report displayed in the table component.

- **Display Units** – Choose how to display values. Shortened Number automatically chooses a unit appropriate for your data and abbreviates it. Shorten numbers to display approximations that take up less space (for example, 1,876 becomes 1.9k), or view the full number for to-the-digit accuracy. Full Number displays the actual values. Other options are Hundreds, Thousands, Millions, Billions, and Trillions. Choose what makes most sense for your data.

- **Show Chatter Photos** – Choose whether to display Chatter photos, or not.

  Display Chatter photos for up to 20 records in a horizontal bar chart component whose source report is grouped by a user or group name field. If there are more than 20 records with photos, record names are shown instead of photos.

- **Add Conditional Highlighting** – Highlight values in the table based on breakpoints and colors set in segment ranges. Click each color to customize it.

- **Decimal Places** – Choose how many decimal places appear on numbers. To let Salesforce automatically set how many decimal places appear, select **Automatic**. To set your own, choose a number between 0 and 5.

- **Segment Ranges** – Choose breakpoints and colors for segments on the gauge. Only available when conditional highlighting is included.

- **Sort Rows By** – Choose how to sort groups displayed in the chart.

- **Max Values Displayed** – Specify how many groups display in the component.

- **Title** – Give the component a title.

- **Subtitle** – Give the component a subtitle.

- **Footer** – Give the component a footer.

- **Component Theme** – Choose a **Light** or a **Dark** theme. Customize chart colors from the dashboard properties

Component	Description
	menu. Individual components support unique themes, but colors are only customizable from the dashboard properties menu.

Sort Data in Dashboard Components

To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards	
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards	
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	
	Enhanced Folder Sharing Edit My Dashboards	
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data	
	Enhanced Folder Sharing Manage Dashboards in Public Folders	
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards	
	Enhanced Folder Sharing Manage Dynamic Dashboards	

EDITIONS

Available in: Lightning Experience

Available in: **Essentials**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing

Sort dashboard components by groups and measures. Sort ascending or descending. Dashboard components support two-level sorting. So you can sort by Stage and then by Sum of Amount, or by Average Age and then Record Count.

To sort data in a dashboard component, first edit the dashboard component.

1. To set the first-level sorting behavior, choose an eligible group or measure from the **Sort By** dropdown (1).

   > **Note:** Don't see the group or measure that you want to sort by? Make sure it's in the component.

2. Click the arrow icon to change sort direction (2) between ascending (↑) and descending (↓).

3. If adding a new table, click **Add**. Otherwise, click **Update**.

4. To save the dashboard, click **save**.

The table sorts data as you specified.

As you get ready to sort dashboard data, review these notes and limitations:

- Two-level sorting isn't available in Salesforce Classic. When you view a dashboard with two level sorting in Salesforce Classic, components with two-level sorting sort by label ascending. Existing components keep their original sort order.

- In table components with no groups, you can sort by any column added to the table.

- Charts with two grouping levels always default to sorting by the sum of values, not by the labels for the secondary grouping.

- You can sort only by groups and measures that are visible in the dashboard component. For example, say that a source report is grouped by type, stage, and lead source and measured by record count and sum of amount. If the component only shows type, stage, and record count, then you can't sort by lead source and sum of amount.

- These components don't support dashboard sorting:

  - Metric

  - Gauge

- Two-level sorting is only available for dashboard components that can show two groups or measures:

  - Bar charts

  - Line charts

- – Scatter charts
- – Tables

- You can sort by summary formulas, but take note of these summary formula-specific limitations:

  - – If a summary formula is evaluated on a specified group in the source report, then the order of the groupings in the dashboard component must match the order of groups in the source report. Otherwise, the summary formula isn't sortable.

    For example, say that there are two summary formulas in a dashboard component's source report. One is applied to Stage and one is applied to Type. The dashboard component is grouped first by Stage and then by Type. The first level sorting can use the summary formula that applies to Stage but not the summary formula that applies to Type.

  - – If a source report is grouped by both rows and columns (matrix report) and it has a summary formula that applies to a specific group, then sorting behavior in the component editor preview is incorrect. A warning in the editor appears noting that the preview is incorrect. Once you save and refresh the dashboard, sorting behaves as expected.

  - – If a dashboard table's source report is grouped by both rows and columns (matrix report), then only summary formulas applied at all summary levels are sortable. But, if you add a summary formula to the table, then no summary formulas are sortable and only groups appear as sort options.

📝 Note:

A Word about Sorting Behavior Past and Present

For dashboard components created before the Summer '19 release, there is an extra sort option: **Report Grouping**. The primary purpose of the **Report Grouping** option is to provide forwards compatibility for existing dashboards without disrupting how each component is sorted. Choosing to sort by **Report Grouping** ascending (↑) means "match the sort order of the source report". Choosing to sort by **Report Grouping** descending (↓) means "invert the sort order of the source report".

Report Grouping isn't available for dashboard components created in Summer '19 and later. If you change the sort order from **Report Grouping** to a named group or measure and save the dashboard, then the **Report Grouping** sort option disappears. If a source report has a row limit filter (only available in Salesforce Classic), then the **Report Grouping** option is always available.

Color Dashboard Data with Themes and Palettes

Customize colors in your dashboard by choosing a light or dark theme and one of 14 color palettes. Themes and color palettes help you organize and curate data in your dashboard. For example, highlight a set of summary metrics by giving them a dark theme and setting them against a light-themed dashboard.

1. Edit a dashboard.

2. Open the properties menu by clicking the gear icon ( ).

3. From Dashboard Theme, choose **Light** or **Dark** to set the theme for the entire dashboard.

To set a single component's theme, edit the component by clicking ✏️ .

4. From Dashboard Palette, choose one of 14 color palettes. **Wildflowers** is the default color palette. If you need an accessible color palette, choose **Mineral**.

Every component in a dashboard uses the same color palette. Individual components don't support unique color palettes.

5. Click **Save**.

Your dashboard now features a custom color palette and theme. Review and admire your dashboard's new look!

If you subscribe to a dashboard which contains a custom color palette or theme, the dashboard subscription email doesn't display the custom palette or theme.

Add a Report to a Dashboard from the Report

Reading a report that you'd like to add to a dashboard? Go from report to dashboard with only a few clicks!

User Permissions Needed	
To add a report to a dashboard:	Create and Customize Dashboards
To add a report to a Dynamic Dashboard:	Manage Dynamic Dashboards

1. From either the Reports tab or while reading a report, click 🔽 > **Add to Dashboard**.

2. Choose whether you want to add the report to an existing dashboard (**Add to existing dashboard)**, or create a new dashboard from the report (**Add to new dashboard**). If necessary, name, describe, and choose a folder for the new dashboard.

3. Click **Add**.

4. Customize the dashboard component.

5. Click **Add**.

6. Optionally, continue customizing the dashboard.

7. Click **Save**.

There you go. Report to dashboard in record time!

Use Source Report Charts in Dashboard Components

USER PERMISSIONS		EDITIONS
To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: Lightning Experience Available in: **Essentials**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: Enhanced Folder Sharing
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Edit My Dashboards	
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data **Enhanced Folder Sharing** Manage Dashboards in Public Folders	
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards **Enhanced Folder Sharing** Manage Dynamic Dashboards	

You've already charted data just the way you like in a report. Instead of recreating the chart from scratch, surface it in a dashboard component.

1. Create or edit a dashboard.

2. Add a new dashboard component by clicking **+ Component**, or edit an existing one by clicking the pencil icon (✐) on the component.

3. From the Add Component or Edit Component menu, select **Use chart settings from report**.

The dashboard component displays the source report chart. To customize the chart, either deselect **Use chart settings from report** or edit the report chart.

> 📝 Note: Dashboard charts and report charts have certain differences. For example, report charts display more data points than dashboard charts. Dashboard charts are still subject to dashboard-specific limitations even when using chart settings from a source report.

Show Tabular Information with a Lightning Table

USER PERMISSIONS		EDITIONS
To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: Lightning Experience Available in: **Essentials**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: Enhanced Folder Sharing

To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing**
	Run Reports AND Manage Dashboards
	Enhanced Folder Sharing
	Edit My Dashboards
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing**
	Run Reports, Manage Dashboards AND View All Data
	Enhanced Folder Sharing
	Manage Dashboards in Public Folders
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing**
	Manage Dynamic Dashboards
	Enhanced Folder Sharing
	Manage Dynamic Dashboards

Show up to 200 records and 10 columns and from any field in a source report's report type. After grouping data in a Lightning table, you can add measures (sum, average, minimum, and maximum) to the grouped data. If a Lightning table's source report has a summary formula column, then you can show it on the Lightning table.

Ungrouped Lightning tables display detail columns, but not measure or formula columns. Grouped Lightning tables display measure and formula columns, but not detail columns.

Detail columns show per-record information about each individual record. The available fields in a Lightning dashboard table come from the source report's report type. You can show fields as detail columns in Lightning dashboard tables that aren't included as columns in the source report.

Measure columns display either the sum, average, maximum, or minimum value of grouped records. Because measure columns are the result of mathematics, they are always based on numeric fields.

Formula columns let you show a summary formula column from the Lightning table's source report on the Lightning table. If a source report's summary formula column is evaluated at **All Summary Levels** or **Grand Total Only**, it is compatible with all Lightning tables. If a summary formula column is evaluated at **Specific Groups**, group data in the Lightning table the same way as the source report to show the formula column.

Reports support up to four groups, but Lightning tables support two groups. So, you can't add a formula column based on the third or fourth grouping of a report. Lightning tables with source reports grouped by both rows and columns only support formula columns based on the report's first-level row and column groupings.

To add a Lightning table to a dashboard, or to edit one:

1. Create or edit a dashboard.

2. Add a Lightning table by clicking **+ Component** and choosing ▦ , or edit an existing one by clicking the pencil icon (✏) on the table.

3. From **Add column...**, choose columns to add.

   Remove a column by clicking **X**. Remove all columns at once by clicking 🗑 .

4. From **Add group...**, select a field to group by.

   You can group the table by any field from the source report type. Group by a field even if it isn't part of the report.

After grouping, detail columns disappear and measure columns appear. To display record-level detail columns again, ungroup the table by clicking **X** or 🗑 .

5. To add a measure or formula column, choose a field from **Add measure column...**. After adding a measure column, click it and choose up to four measures: **Sum**, **Average**, **Max**, or **Min**. You can't add measures to formula columns.

Add up to 9 measure and formula columns. Remember, you can only measure numeric fields, so date, text, picklist, and Boolean fields aren't available.

Formula columns are listed before measure columns. Formula columns are identifiable by the formula icon that appears before their name (fx).

> 📝 Note:
>
> - The default measure column is record count. You can't remove record count, but after adding another measure column, you can hide it by clicking the eye icon (👁).
> - Lightning table dashboard components don't support unique counts.

6. When satisfied with the Lightning table, click **Add** or **Update**.

7. To save the dashboard, click **Save**.

For example, as seen on the left-hand side of this Sales Overview dashboard, you can use a Lightning table to list details about individual opportunity records. The right-hand Lightning table, grouped by stage, provides a handy reference of key metrics like average probability and sum of amount. The right-most column is a formula column, showing percent of total.

Show Subtotals on First-Level Groups in Lightning Tables

Lightning tables are a powerful tool for summarizing data on a dashboard. Second-level groups automatically show subtotals for each measure column. Show subtotals to add subtotals for each measure column to first-level groups.

Show Subtotals on First-Level Groups in Lightning Tables

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

To edit and delete dashboards you didn't create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

To create, edit, and delete dynamic dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards

Lightning tables are a powerful tool for summarizing data on a dashboard. Second-level groups automatically show subtotals for each measure column. Show subtotals to add subtotals for each measure column to first-level groups.

To show subtotals on a Lightning table, first edit the component.

1. If necessary, group data and add measure columns. To show subtotals, the table requires at least two groups.

2. Select **Show Subtotals** (1).

    > 📝 Note: Subtotals always match the summary type of their associated metric column. If the metric column is $sum\ of$ $amount$, then the subtotal is a sum (2).

3. Click **Update**.

4. Click **Save**.

The table now features subtotals for the first-level group.

Depending on how summary formulas are applied to source report data, you can show subtotals for summary formulas that appear on Lightning table as metric columns. Here are the conditions.

- If a summary formula is applied at all levels, then it is subtotal-able.

- If a summary formula is applied to a specific group, then it is subtotal-able if the table also has the specific group. If the specific group is a second-level group, then the table must match both the source report's first-level and second-level groups. Because Lightning tables support up to two groups, summary formulas applied to third and fourth-level groups are not subtotal-able.

  For example, say that a source report is grouped first by Type and then by Stage. It has a summary formula applied to Stage. The summary formula is subtotal-able only on Lightning tables grouped first by Type and then by Stage. If you reverse the group order in the Lightning table - grouping first by Stage and then by Type - then the subtotal reads "Incalculable" because the group order prevents the summary formula from running.

- If a summary formula is applied only to the grand total, then it is not subtotal-able.

Set Decimal Places for Numbers in Dashboard Charts, Tables, and Metrics

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize
Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize
Dashboards

To edit and delete dashboards you created
in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

To edit and delete dashboards you didn't
create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND
View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

To create, edit, and delete dynamic
dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards

Because 14.11B is greater than 14B and the difference between 90%, 90.1%, and 90.10001% matters, choose how many decimal places appear after numbers in dashboard components.

1. Edit or create a dashboard.

2. Edit a dashboard component, or add one by clicking **+ Component**.

3. From **Decimal Places**, select a number from 0 through 5. To let Salesforce set decimal precision for you, choose **Automatic**.

4. Click **Update**. (If you're adding a component, click **Add**.)

5. Save the dashboard by clicking **Save**.

The dashboard component now displays up to 5 decimal places.

> 📝 Note: Some components, such as donut charts, feature the option to display percentages with the **Show Percentage** setting.
> Setting decimal places does not change how many decimal places appear after these percentages.
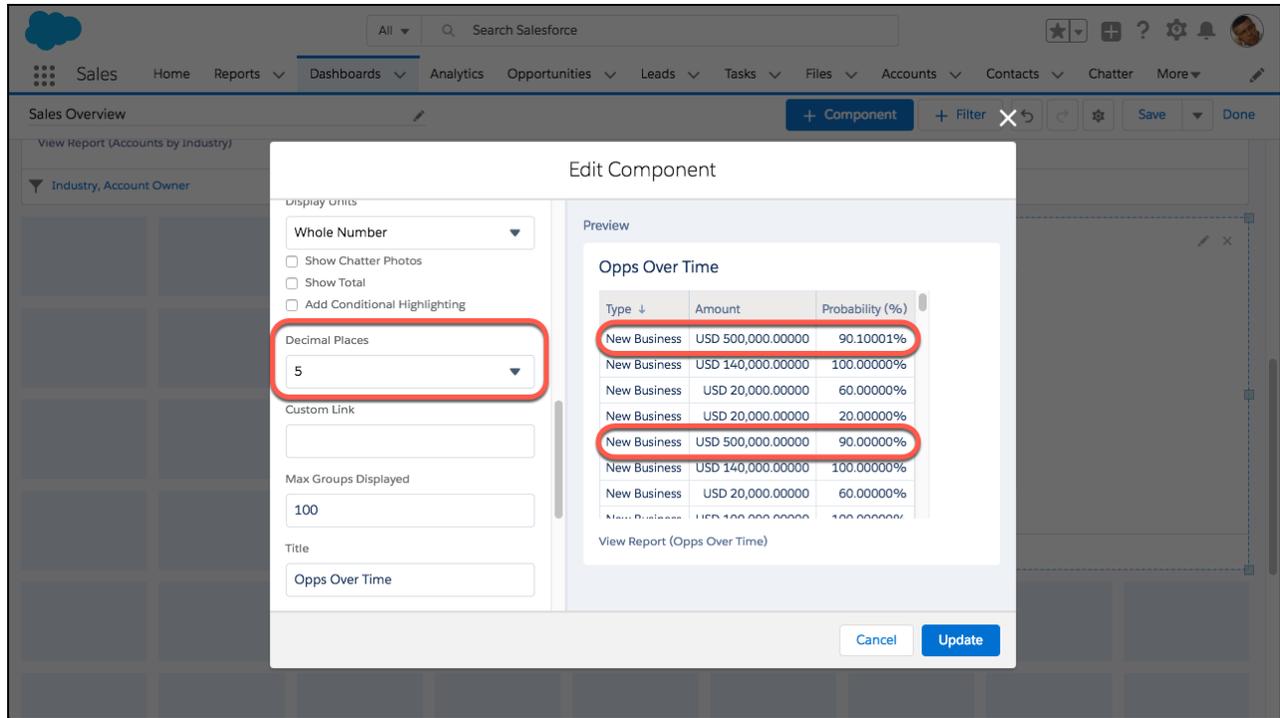
Set Axis Range for Dashboard Charts

USER PERMISSIONS		EDITIONS

To create dashboards:	**Legacy Folder Sharing**
	    Run Reports AND Manage Dashboards
	Enhanced Folder Sharing
	    Run Reports AND Create and Customize
	    Dashboards
		Available in: Lightning Experience

		Available in: **Essentials**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

To edit and delete dashboards you created:	**Legacy Folder Sharing**
	    Run Reports AND Manage Dashboards
	Enhanced Folder Sharing
	    Run Reports AND Create and Customize
	    Dashboards
		Available in: Enhanced Folder Sharing

To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing**
	    Run Reports AND Manage Dashboards
	Enhanced Folder Sharing
	    Edit My Dashboards	

To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing**
	    Run Reports, Manage Dashboards AND
	    View All Data	

| | **Enhanced Folder Sharing** |
| | Manage Dashboards in Public Folders |

To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing**
	Manage Dynamic Dashboards
	Enhanced Folder Sharing
	Manage Dynamic Dashboards

Choose the range of the x- and y-axes on dashboard charts, or let Salesforce automatically set them.

Some charts only let you customize the x- or y-axis. For example, horizontal bar charts only let you customize the range of the x-axis. You can only set custom ranges on axes that display a numeric field.

1. Edit or create a dashboard.

2. Edit a dashboard component, or add one by clicking **+ Component**.

3. To set the axis range yourself, from **X-Axis Range** or **Y-Axis Range**, select **Custom**. Then set a minimum range in **Min** and a maximum range in **Max**. To let Salesforce set the axis range for you, choose **Automatic**.

    📝 Note: Optionally, set only a **Min** or a **Max** range and Salesforce chooses the corresponding range automatically.

4. Click **Update**. (If you're adding a component, click **Add**.)

5. Save the dashboard by clicking **Save**.

The dashboard component now displays data within the range you set.
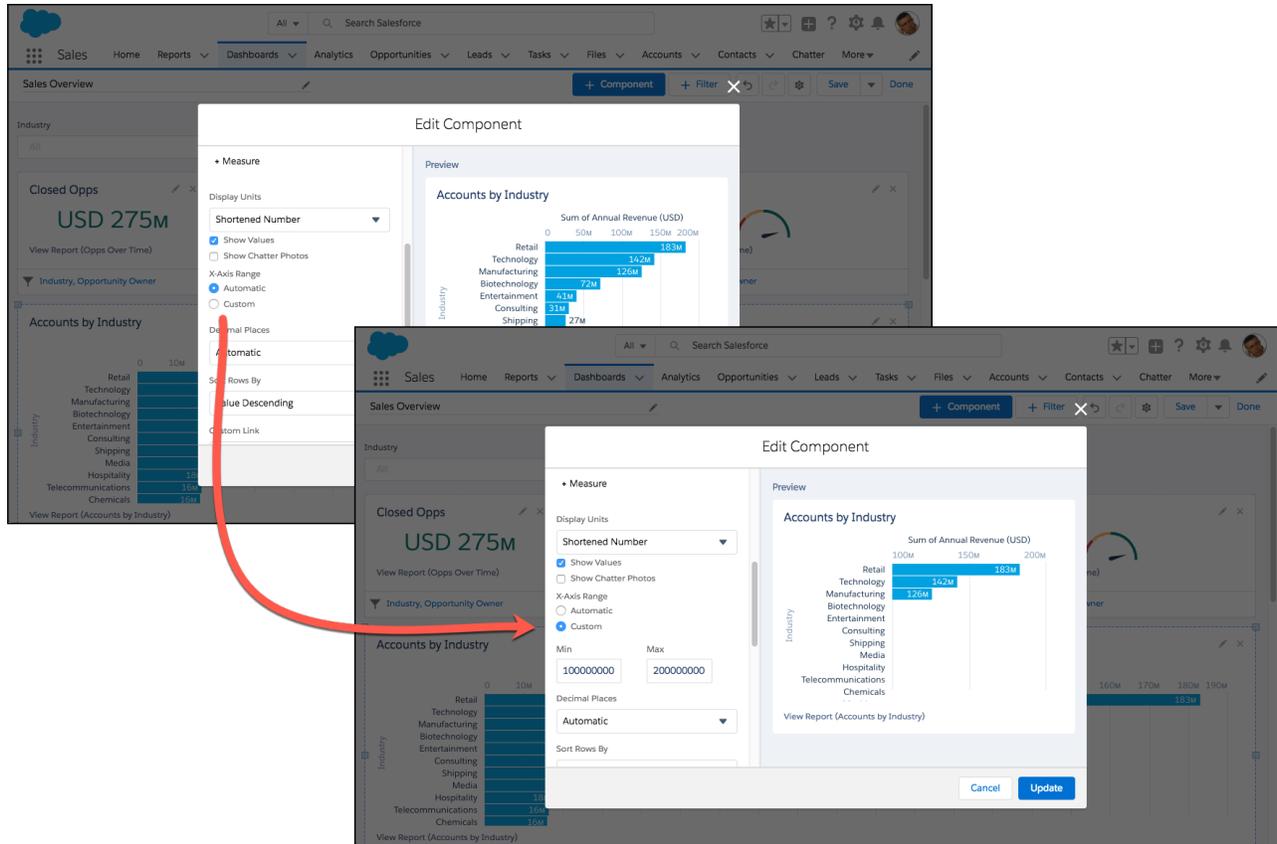
Switch Between a 12 or 9-Column Dashboard Layout

Both 12-column and 9-column dashboards give you plenty of room to lay out data in easy-to-consume ways.

User Permissions Needed

To switch between a 12-column and 9-column dashboard:	Create and Customize Dashboards
To switch between a 12-column and 9-column dynamic dashboard:	Manage Dynamic Dashboards

Any dashboard can have 12 or 9 columns.

When switching to a 9-column dashboard from a 12-column dashboard, even with 12 components in a single row, the dashboard automatically updates its layout. No need to remove or move components in advance.

When switching to a 12-column dashboard from a 9-column dashboard, three empty columns appear on the right-hand side of the dashboard.

1. Create or edit a dashboard.

2. To switch between a 12-column and 9-column layout, open the Properties menu by clicking the gear icon (⚙).

    a. Under `Dashboard Grid Size`, choose **12 columns** or **9 columns**.

b. Click **Save**.

3. Click **Save**.

The dashboard layout updates.

Edit Dashboards with Keyboard Shortcuts in Lightning Experience

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize
Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize
Dashboards

To edit and delete dashboards you created
in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

To edit and delete dashboards you didn't
create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND
View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

To create, edit, and delete dynamic
dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards

EDITIONS

Available in: Lightning
Experience

Available in: **Essentials**,
Group (View Only),
Professional, **Enterprise**,
Performance, **Unlimited**,
and **Developer** Editions

Available in: both Legacy
Folder Sharing and
Enhanced Folder Sharing

From the Dashboard tab, you can create, edit, or delete a dashboard using nothing but your keyboard. Use keystrokes to edit dashboards quickly and easily in Lightning Experience.

Use these handy keyboard shortcuts while building or editing a dashboard.

Keyboard Shortcut	Description
Tab	Focus on next item in the dashboard
Shift+Tab	Focus on previous item in the dashboard

Keyboard Shortcut	Description
Spacebar	Select a component to move or a component corner to resize
Arrow keys	Move or resize a widget
Enter	Click a button
Ctrl+Z	Undo
Ctrl+Y	Redo
Ctrl+S	Save
Ctrl+N	Add component
Esc	Cancel pending component move or resize

As you tab through dashboard components, notice that the option to have a screen reader read each chart's dataset appears, **Read Chart Dataset (Screen Reader)**.

For more information about using screen readers with Salesforce, see ***Recommendations for Salesforce Accessibility*** in the Salesforce help.

Build a Salesforce Classic Dashboard

USER PERMISSIONS

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

To edit and delete dashboards you didn't create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

EDITIONS

Available in: Salesforce Classic (not available in all orgs)

Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

To create, edit, and delete dynamic dashboards:

Legacy Folder Sharing
  Manage Dynamic Dashboards

Enhanced Folder Sharing
  Manage Dynamic Dashboards

Build a dashboard to provide a graphical view of the data in your reports.

This topic is about creating dashboards from reports in Salesforce Classic. For information on creating dashboards in Salesforce Classic or Analytics Cloud, review these articles:

- Build a Lightning Experience Dashboard
- Build Tableau CRM Dashboards

💡 **Tip:** You can clone a dashboard to quickly create a dashboard with the same properties and components as the one you're viewing. Click **Clone**, modify the dashboard settings, and save.

1. Create the custom reports containing the data you want to display.

   🛑 **Important:** Be sure to store these reports in folders that your intended dashboard viewers can access.

2. Click the Dashboards tab.

3. Click **Go To Dashboard List**.

4. Click **New Dashboard**.

   To modify an existing dashboard, click its name from the list.

5. Customize your dashboard and click **Save**.

Work with Salesforce Classic Dashboards
A *dashboard* shows data from source reports as visual components, which can be charts, gauges, tables, metrics, or Visualforce pages. The components provide a snapshot of key metrics and performance indicators for your organization.

Edit Dashboards in Accessibility Mode in Salesforce Classic
In Salesforce Classic, you can edit dashboards in Accessibility mode.

SEE ALSO:

Delete a Dashboard

Install the CRM Sample Dashboards from AppExchange

Work with Salesforce Classic Dashboards

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

To edit and delete dashboards you didn't create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

To create, edit, and delete dynamic dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards

Available in: Salesforce Classic (not available in all orgs)
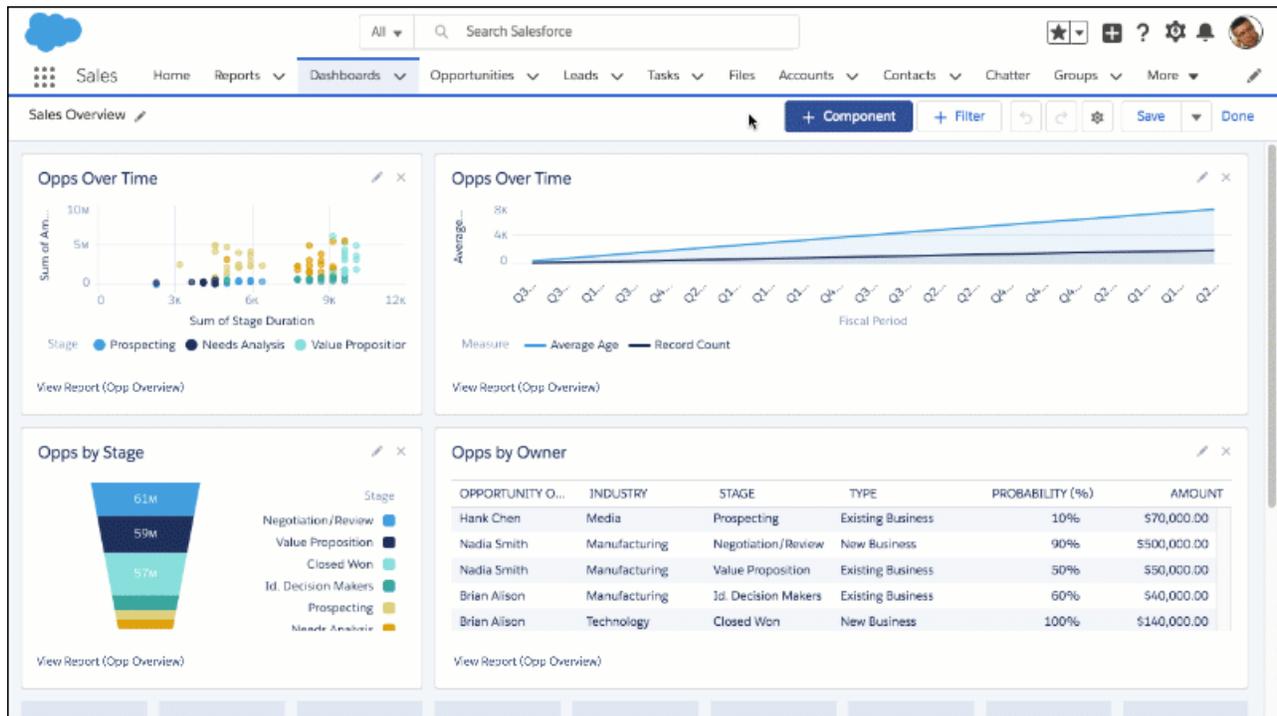
Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

A *dashboard* shows data from source reports as visual components, which can be charts, gauges, tables, metrics, or Visualforce pages. The components provide a snapshot of key metrics and performance indicators for your organization.

Dashboard builder is a drag-and-drop interface for creating and modifying dashboards. You can use it to customize the whole dashboard, a column in the dashboard, or a component in the dashboard.

> 💡 Tip: Dashboard builder uses a compressed page header to let you see more of the screen. To view your application tabs, simply close the builder or click the Salesforce logo.

To customize a dashboard, view it and click **Edit**.

Add a Dashboard Component in Salesforce Classic
Add components by dragging a component type onto the dashboard, then dropping a data source (report, s-control, or Visualforce page) onto it.

Modify a Dashboard Component in Salesforce Classic

A dashboard component is a visual representation of the data in a report. You can change where the component's data comes from, what the data looks like in the component, and what kind of component it is.

SEE ALSO:

Add a Dashboard Filter

Build a Salesforce Classic Dashboard

Add a Dashboard Component in Salesforce Classic

<table>
<tr><td>USER PERMISSIONS</td><td></td><td>EDITIONS</td></tr>
<tr>
<td>To create dashboards:</td>
<td>Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards</td>
<td>Available in: Salesforce Classic (not available in all orgs)</td>
</tr>
<tr>
<td>To edit and delete dashboards you created:</td>
<td>Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards</td>
<td>Available in: Group (View Only), Professional, Enterprise, Performance, Unlimited, and Developer Editions</td>
</tr>
<tr>
<td>To edit and delete dashboards you created in public folders:</td>
<td>Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards</td>
<td>Available in: both Legacy Folder Sharing and Enhanced Folder Sharing</td>
</tr>
<tr>
<td>To edit and delete dashboards you didn't create in public folders:</td>
<td>Legacy Folder Sharing
Run Reports, Manage Dashboards AND View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders</td>
<td></td>
</tr>
<tr>
<td>To create, edit, and delete dynamic dashboards:</td>
<td>Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards</td>
<td></td>
</tr>
</table>

Add components by dragging a component type onto the dashboard, then dropping a data source (report, s-control, or Visualforce page) onto it.

1. On the dashboard where you want to add a component, click **Edit**.

   You can continue to edit the dashboard while components and data sources are loading.

2. Drag the component type you want from the Components tab onto your dashboard.

> 💡 **Tip:** You can also drop the data source first, then drop a component type onto it.

To select the kind of component you need, consider the type of data you're showing and the uses it will serve.

	Use a chart when you want to show data graphically. You can choose from a variety of chart types.
	Use a gauge when you have a single value that you want to show within a range of custom values
	Use a metric when you have one key value to display. • Enter metric labels directly on components by clicking the empty text field next to the grand total. • Metric components placed directly above and below each other in a dashboard column are displayed together as a single component.
	Use a table to show a set of report data in column form.
Visualforce Page	Use a Visualforce page when you want to create a custom component or show information not available in another component type. Visualforce Pages are only available in Salesforce Classic.
Custom S-Control	Custom s-controls can contain any type of content that you can display in a browser, for example a Java applet, an Active-X control, an Excel file, or a custom HTML Web form. Custom S-Controls are only available in Salesforce Classic.

3. Drag a report from the Data Sources tab onto the component you just dropped on the dashboard.

- You can show a joined report that includes a chart on a dashboard. Edit the joined report dashboard component and select **Use chart as defined in the source report**.
- If the dashboard has a filter, the data source must contain the filter field or an equivalent. If it doesn't, the filter may not work.
- Some custom forecast and lead reports aren't available for dashboards.
- For Visualforce components, the data source must be a Visualforce page.

> 💡 **Tip:** Each folder can contain up to 200 data sources. To home in on the right data source quickly, try Quick Find or the **Recent**, **My**, and **All** filters.

4. Click 🔧 on your dashboard component.

5. On the Component Data tab, choose which summary fields and groupings in the underlying report you want to display in your component.

> **Tip:** Make more fields available for a dashboard component by adding them to the source report chart.For example, create a combination chart on the report using vertical columns and lines. The additional groupings in the report chart are available to use in dashboard components.

6. On the Formatting tab, specify how your component shows its data.

   Your formatting choices depend on the component type you choose.

   > **Note:** For Visualforce pages and s-controls, set the `Height`.

7. Click **OK**.

8. Drag, drop and click to rearrange components on the dashboard.

   - Grab components by the header bar and drag them to the right location on the dashboard.
   - Click ✖ for a data source to remove it from the component.
   - Click ✖ for a component to remove it from the dashboard.
   - Click component header, title, and footer fields to edit them.
   - Change colors for picklist values displayed in dashboard components. You need the "Customize Application" permission to update picklists.
   - Optionally, for filtered dashboards, choose a different field in the `Filtered By` drop-down.

Modify a Dashboard Component in Salesforce Classic

USER PERMISSIONS		EDITIONS
To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: Salesforce Classic ([not available in all orgs](#))
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Edit My Dashboards	Available in: both Legacy Folder Sharing and Enhanced Folder Sharing
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data **Enhanced Folder Sharing** Manage Dashboards in Public Folders	

To create, edit, and delete dynamic dashboards:

Legacy Folder Sharing
   Manage Dynamic Dashboards

Enhanced Folder Sharing
   Manage Dynamic Dashboards

A dashboard component is a visual representation of the data in a report. You can change where the component's data comes from, what the data looks like in the component, and what kind of component it is.

📝 Note: If you are using legacy folder sharing, you might see the **Edit** option for dashboards that you are not permitted to edit. To resolve this issue, turn on enhanced folder sharing on page 432.

Dashboard builder is a drag-and-drop interface for creating and modifying dashboards. Drag, drop and click to rearrange components on the dashboard. Hover details and drill-down are available when you view and not when you edit a dashboard.

1. Make Fields Available in a Dashboard Component

   Make more fields available for a dashboard component by adding them to the source report chart.

2. Choose Where Users Go When Clicking a Dashboard Component

   You can edit a dashboard component so that when users click the component, they can drill down to the source report, filtered report, record detail page, or other URL.

3. Custom Four-Column Table

   Table visuals are useful to list values of several items against some criteria. The fields you add to the source report chart are the ones available for the table columns in dashboards.

4. Dashboard Component Types

   Dashboard components can be charts, tables, gauges, metrics, or other components that you can create with VisualForce.

5. Chart Types

   You can show data in reports and dashboards in the form of bars, columns, lines, shapes, or other elements. Which is right depends on what the data is about and what you want to show with it.

6. Data Settings for Dashboard Chart Components

   Choose the data you want to show on your report or dashboard chart. The chart automatically selects groupings and summary values from your report. You can override some of those choices to focus on the data you need to share.

7. Visual Settings for Dashboard Chart Components

   Choose the type of chart that fits the data you are sharing, then apply the visual settings that will communicate the data most effectively.

SEE ALSO:

   Add a Dashboard Filter

Make Fields Available in a Dashboard Component

Make more fields available for a dashboard component by adding them to the source report chart.

For example, create a combination chart on the report using vertical columns and lines. The additional groupings in the report chart are available to use in dashboard components.

Choose Where Users Go When Clicking a Dashboard Component

You can edit a dashboard component so that when users click the component, they can drill down to the source report, filtered report, record detail page, or other URL.

Edit a component and set the `Drill Down to` option on the Component Data tab. Choose one of these options:

- Source Report—Takes the user to the full source report for the dashboard component.
- Filtered Source Report—When users click individual groups, X-axis values, or legend entries, they are taken to the source report filtered by what they clicked.

  For example, if you had a stacked vertical column chart of opportunities grouped by stage, with months as the X-axis, you could click an individual stage in a bar, a month on the X-axis, or a legend entry for a stage to drill down to the filtered source report. (Not available for gauges, metrics, or tables.)

- Record Detail Page—When users click chart or table elements, axis values, or legend entries, they are taken to the detail page for that record. You can only choose this option for tables and charts that use a source report grouped by record name, record owner, or feed post. (Not available for gauges or metrics.)
- Other URL—Takes the user to the URL that you specify. You can't add URLs that begin with "mailto:" or "javascript:" to dashboard components.

Custom Four-Column Table

Table visuals are useful to list values of several items against some criteria. The fields you add to the source report chart are the ones available for the table columns in dashboards.

Tables show two columns of data by default. They can show totals and up to four columns of data if you customize. You can also personalize the table and show users' Chatter photos as long as the table doesn't have more than 20 rows.

Before you customize a table, make sure your source report is in matrix or summary format and *contains* a chart.

1. To create a four-column table, edit a table component on a dashboard and click the **Customize Table** link . Customized tables allow null values in the results. Default two-column tables do not.
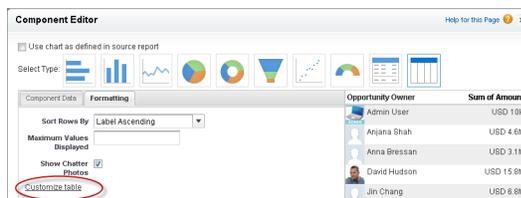
2. Which fields you can use in the table depends on the fields you picked in the source report chart:
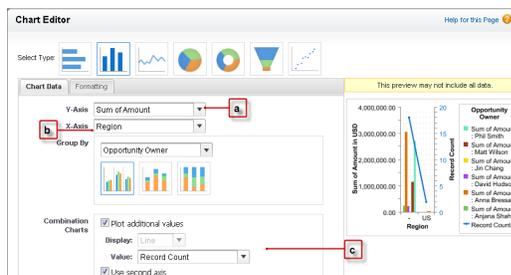
   a. The chart must contain groupings and at least one summary.

350

b. The first grouping in the report chart becomes the first column of the table. To show a different field as the first column in the table, select the desired grouping in the report chart.

c. Use a combination vertical bar or a scatter chart in the report to make more fields available for the table.

Example

You can use a table to show:

- Top sales reps for the quarter with the number of opportunities each won.
- Number of cases by product in the last three months.
- Number of accounts by region, the total number of opportunities available, and those won.

Take a look at this customized four-column table that shows the top sales reps by region with the number and value of opportunities they won.

Dashboard Component Types

To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards	
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards	
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	
	Enhanced Folder Sharing Edit My Dashboards	
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data	

	Enhanced Folder Sharing Manage Dashboards in Public Folders
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards
	Enhanced Folder Sharing Manage Dynamic Dashboards

Dashboard components can be charts, tables, gauges, metrics, or other components that you can create with VisualForce.

Component Type	Image	Description
Chart		Use a chart when you want to show data graphically. You can choose from a variety of chart types.
Gauge		Use a gauge when you have a single value that you want to show within a range of custom values. For example, to create a dashboard that measures where your current closed opportunity amounts fall within a range of values, set the `Minimum Value`, `Breakpoint #1 Value`, `Breakpoint #2 Value`, and `Maximum Value` for the gauge. The ranges that you set can indicate poor, acceptable, and good performance. Set appropriate colors for each of these ranges to visually indicate progress. To create a gauge with only two ranges, leave `Breakpoint #2 Value` blank. Select `Show Percentage` or `Show Total` to display those values on the gauge. Values exceeding the maximum are shown as greater than 100%.
Metric		Use a metric when you have one key value to display. For example, if you have a report showing the total amount for all opportunities in the `Closed`, `Commit`, and `Base Case` stages in the current month, you can name that value and use it as a revenue target for the month displayed on the dashboard.
Table		Use a table to show a set of report data in column form. For example, to see the top 20 opportunities by amount, set `Maximum Values Displayed` to *20*, click **Customize Table** and select opportunity name, amount, and other columns to display, choose the sort order, and set conditional highlighting. Available columns include all chart groupings and report summary fields, as well as the second-level grouping defined in the report.
Visualforce Page	N/A	Use a Visualforce page when you want to create a custom component or show information not available in another component type. For example, a Visualforce page can display data from an external system or show Salesforce data in a custom way. Visualforce pages must meet certain requirements to be displayed in dashboards; otherwise, they don't appear in the `Visualforce Page` drop-down list. See Creating Visualforce Dashboard Components.

Component Type	Image	Description
Custom S-Control	N/A	Custom s-controls can contain any type of content that you can display in a browser, for example a Java applet, an Active-X control, an Excel file, or a custom HTML Web form.

⊘ **Important:** Visualforce pages supersede s-controls. Organizations that haven't previously used s-controls can't create them. Existing s-controls are unaffected, and can still be edited. |

SEE ALSO:

[Modify a Dashboard Component in Salesforce Classic](#)

Chart Types

You can show data in reports and dashboards in the form of bars, columns, lines, shapes, or other elements. Which is right depends on what the data is about and what you want to show with it.

1. [Bar Charts](#)

   A bar chart shows values as horizontal lengths, so this format can be good for comparing distance or time. Use a bar chart when you have a summary report with a single grouping, or you only want to display one grouping.

2. [Column Charts](#)

   A column chart is very much like a bar chart, but it can be a better format for showing relative counts of things, such as leads or dollars. Use a column chart when you have a summary report with a single grouping, or you only want to display one grouping.

3. [Line Charts](#)

   Line charts are good for showing changes in the value of an item over a series of points in time, such as week to week or quarter to quarter. Use a line chart when you have one important grouping representing an ordered set of data and one value to show.

4. [Pie Charts](#)

   Use a pie chart when you have multiple groupings and want to show the proportion of a single value for each grouping against the total.

5. [Donut Charts](#)

   Use a donut chart when you have multiple groupings and want to show not only the proportion of a single value for each grouping against the total, but also the total amount itself.

6. [Funnel Charts](#)

   Use a funnel chart when you have multiple groupings in an ordered set and want to show the proportions among them.

7. [Scatter Charts](#)

   Use scatter charts to show meaningful information using one or two groups of report data plus summaries.

SEE ALSO:

[Scatter Charts](#)

[Modify a Dashboard Component in Salesforce Classic](#)

[Create a Custom Report in Accessibility Mode](#)

[Show Multiple Sets of Data in One Chart](#)

[Data Settings for Dashboard Chart Components](#)

[Visual Settings for Dashboard Chart Components](#)

Bar Charts

A bar chart shows values as horizontal lengths, so this format can be good for comparing distance or time. Use a bar chart when you have a summary report with a single grouping, or you only want to display one grouping.

For example, to see the amount in each sales stage in a report, select `Sum of Amount` as the `X-axis` and `Stage` as the `Y-axis`. The chart displays one bar for each stage, with the length proportional to the total opportunity amount.

The advantage of the horizontal bar charts is that the chart can be extended vertically to show numerous groupings, though the width is fixed. Depending on chart settings, you can also display Chatter photos.

1. Grouped Bar Charts

   Use a grouped bar chart when you have multiple groupings, and you want to compare values within a secondary grouping, but not the totals.

2. Stacked Bar Charts

   Use a stacked bar chart when you have multiple groupings and are interested in the proportions between values in each grouping, as well as each grouping's total.

3. Bar Charts Stacked to 100%

   Use a bar chart stacked to 100 percent when you have multiple groupings and are interested in the proportions between values in each grouping, as well as each grouping's total.

SEE ALSO:

Data Settings for Dashboard Bar and Column Chart Components

Formatting Settings for Dashboard Bar Chart Components

Grouped Bar Charts

Use a grouped bar chart when you have multiple groupings, and you want to compare values within a secondary grouping, but not the totals.

For example, to compare the amount of deals closed each month by lead source in a report, set amount as the `X-axis`, source as the `Y-axis`, and closing month as the `Groupings` value. The chart displays a set of bars for each source, one bar for each month. The monthly differences within a particular source are easy to compare.

You can also compare a given month across sources, but comparing the total number of leads for each source may be difficult.

Tip: Use a stacked chart to compare totals.

Stacked Bar Charts

Use a stacked bar chart when you have multiple groupings and are interested in the proportions between values in each grouping, as well as each grouping's total.

For example, to compare the status of leads by campaign in a report, and also to compare the totals for each status, set record count as the `X-axis`, status as the `Y-axis`, and campaign as the `Groupings` value. The chart displays a single bar for each status, broken down by campaign, with each campaign shown in a different color.

The proportion of each campaign in each status is easy to compare, as are the totals for each status, but comparing a single campaign's contribution to different statuses, or to the total, may be difficult.

Bar Charts Stacked to 100%
Use a bar chart stacked to 100 percent when you have multiple groupings and are interested in the proportions between values in each grouping, as well as each grouping's total.

For example, to compare the status of leads by campaign in a report, and also to compare the totals for each status, set record count as the `X-axis`, status as the `Y-axis`, and campaign as the `Groupings` value. The chart displays a single bar for each status, broken down by campaign, with each campaign shown in a different color.

The proportion of each campaign in each status is easy to compare, as are the totals for each status, but comparing a single campaign's contribution to different statuses, or to the total, may be difficult.

Column Charts
A column chart is very much like a bar chart, but it can be a better format for showing relative counts of things, such as leads or dollars. Use a column chart when you have a summary report with a single grouping, or you only want to display one grouping.

For example, to see the number of leads by lead source in a report, set record count as the `Y-axis` and source as the `X-axis`. The chart displays one column for each source, with the height proportional to the total number of leads. The width of column charts is constrained by dashboard column size and report chart size. Horizontal bar charts may be better for large numbers of groupings. Column charts are good when showing values by date, since dates traditionally run along the X-axis.

1. Grouped Column Charts
   Use a grouped column chart when you have multiple groupings, and you want to compare values within a secondary grouping, but not the totals.

2. Stacked Column Charts
   Use a stacked column chart when you have multiple groupings and you're interested in the proportions between values in each grouping, as well as each grouping's total.

3. Stacked Bar Charts
   Use a stacked bar chart when you have multiple groupings and are interested in the proportions between values in each grouping, as well as each grouping's total.

4. Column Chart Stacked to 100 Percent

   Use a column chart stacked to 100 percent when you have multiple groupings and are interested in the proportions between values in each grouping, as well as each grouping's total.

Grouped Column Charts

Use a grouped column chart when you have multiple groupings, and you want to compare values within a secondary grouping, but not the totals.

For example, to compare the number of opportunities created each month by campaign source in a report, set record count as the `Y-axis`, created month as the `X-axis`, and source as the `Groupings` value. The chart displays a set of bars for each month, one bar for each campaign source. The differences between sources within a particular month are easy to compare.

You can also compare a particular source across months, but comparing the total number of opportunities for each month may be difficult.

💡 Tip: Use a stacked chart to compare totals.

Stacked Column Charts

Use a stacked column chart when you have multiple groupings and you're interested in the proportions between values in each grouping, as well as each grouping's total.

For example, to compare the number of opportunities created each month by campaign source in a report, and also to compare the totals for each month, set record count as the `Y-axis`, created month as the `X-axis`, and source as the `Groupings` value. The chart displays a single bar for each month, broken down by source, with each source shown in a different color.

The proportion of each source in each month is easy to compare, as are the monthly totals, but comparing a single source's contribution to different months, or to the total, may be difficult.

Stacked Bar Charts

Use a stacked bar chart when you have multiple groupings and are interested in the proportions between values in each grouping, as well as each grouping's total.

For example, to compare the status of leads by campaign in a report, and also to compare the totals for each status, set record count as the `X-axis`, status as the `Y-axis`, and campaign as the `Groupings` value. The chart displays a single bar for each status, broken down by campaign, with each campaign shown in a different color.

The proportion of each campaign in each status is easy to compare, as are the totals for each status, but comparing a single campaign's contribution to different statuses, or to the total, may be difficult.

Column Chart Stacked to 100 Percent

Use a column chart stacked to 100 percent when you have multiple groupings and are interested in the proportions between values in each grouping, as well as each grouping's total.

For example, to compare the number of opportunities created each month by campaign source in a report, and also to compare the totals for each month, set record count as the `Y-axis`, created month as the `X-axis`, and source as the `Groupings` value. The chart displays a single bar for each month, broken down by source, with each source shown in a different color. The proportion of each source in each month is easy to compare, as are the monthly totals, but comparing a single source's contribution to different months, or to the total, may be difficult.

Line Charts

Line charts are good for showing changes in the value of an item over a series of points in time, such as week to week or quarter to quarter. Use a line chart when you have one important grouping representing an ordered set of data and one value to show.

Line charts are useful for showing data over time. For example, to see the numbers of leads created each month in a report, set record count as the `Y-axis` and created month for the `X-axis`. The chart displays a line connecting the record count totals for each month. Salesforce does not plot missing (null) values.

If a missing value occurs in the middle of a data set, Salesforce displays a gap in the line.

1. Grouped Line Charts

   Use a grouped line chart when you have multiple groupings—each with one important secondary grouping representing an ordered set of data—and one value to show.

2. Cumulative Line Charts

   Use a cumulative line chart when you have one important grouping representing an ordered set of data and one value to show, summed over time.

3. Grouped Cumulative Line Charts

   Use a cumulative line chart when you have one important grouping representing an ordered set of data and one value to show, summed over time.Use a grouped line chart when you have multiple groupings—each with one important secondary grouping representing an ordered set of data—and one value to show.

Grouped Line Charts

Use a grouped line chart when you have multiple groupings—each with one important secondary grouping representing an ordered set of data—and one value to show.

For example, to see monthly leads by lead source in a report, set record count as the `Y-axis`, created month as the `X-axis`, and source as the `Groupings` value. The chart displays a line for each source, connecting that source's record count totals for each month. Each line spans the earliest to the latest month containing data. Comparing the total numbers for each month may be difficult.

Salesforce does not plot missing (null) values. If a missing value occurs in the middle of a data set, Salesforce displays a gap in the line.

Cumulative Line Charts

Use a cumulative line chart when you have one important grouping representing an ordered set of data and one value to show, summed over time.

For example, to see the total amount of closed opportunities by day in the current month in a report, set amount as the `Y-axis` and closing day as the `X-axis`. The chart displays one line, with the line's height representing the cumulative amount of closed opportunities up to and including that day. You can't see the amount for any single day—only the cumulative amount.

If the data set contains a missing (null) value, Salesforce continues the line using the previous value in the data set.

Grouped Cumulative Line Charts

Use a cumulative line chart when you have one important grouping representing an ordered set of data and one value to show, summed over time.Use a grouped line chart when you have multiple groupings—each with one important secondary grouping representing an ordered set of data—and one value to show.

For example, to see the total amount of closed opportunities by day for each of the last three months in a report, set amount as the `Y-axis`, closing day as the `X-axis`, and closing month as the `Groupings` value. The chart displays a line for each month, with the line's height representing the cumulative amount of closed opportunities up to and including that day.

Pie Charts

Use a pie chart when you have multiple groupings and want to show the proportion of a single value for each grouping against the total.

For example, to see the breakdown of your case queue by case status in a report, set record count for `Values` and status for `Wedges`. The chart displays a circle made up of wedges, each wedge representing the cases in a case status. Wedge size is proportional to the numbers of cases.

Pie charts are not ideal for comparing values that are close together or numerous small values.

Select `Show Labels`, `Show Values`, or `Show Wedge %` to include that information on the chart. (Only available with Chart Analytics 2.0.)

Donut Charts

Use a donut chart when you have multiple groupings and want to show not only the proportion of a single value for each grouping against the total, but also the total amount itself.

For example, to see the breakdown of your case queue by case status in a report, as well as the total number of cases, set record count for `Values` and status for `Wedges`. The chart displays a donut made up of wedges, each wedge representing a case status. Wedge size is proportional to the numbers of cases. The total number of cases for all statuses is shown in the middle.

Select `Show Labels`, `Show Values`, `Show Wedge %`, or `Show Total` to include that information on the chart.

Funnel Charts
Use a funnel chart when you have multiple groupings in an ordered set and want to show the proportions among them.

For example, to see the amount of opportunities in each stage in a report, set amount for `Values` and stage for `Segments`. Since the `Opportunity: Stage` field is an ordered picklist, the stages are sorted in the same order as the picklist, with each segment representing the amount for that stage. Funnel charts are useful for showing the flow of opportunities through the stages; a substantially larger segment may indicate a bottle-neck at that stage.

Select `Show Labels`, `Show Values`, or `Show Segment %` to include that information on the chart.

Scatter Charts
Use scatter charts to show meaningful information using one or two groups of report data plus summaries.

For example, to see how stage duration correlates with the number of activities for opportunities, group your report by `Opportunity Name` and plot the scatter chart by the grouping. Then set `X-Axis` on the chart to `Record Count` and `Y-Axis` to `Stage Duration`. The chart will show a dot for each opportunity. You can tell at a glance if the stage duration is shorter for opportunities that have more activities.

Because a scatter chart shows data grouped by summarized values, you need at least one grouping in your report. Choose a report format that allows groupings, such as, summary, matrix, or joined. You also need at least one summarized field in the report to show data on the axes of the chart. Otherwise, the chart will show record count on the axes. Scatter charts automatically show data from the source report or you can manually choose what information to display for groupings and summaries.

If your source report has	The X-axis automatically displays	The Y-axis automatically displays
No summary fields	Record count.	Record count.
One or more summary fields	First summary.	• Either record count or the second summary in report charts.

EDITIONS

Available in: Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

EDITIONS

Available in: Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

EDITIONS

Available in: Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

- The first summary in dashboard charts
  even if the source report has multiple
  summaries. But you can manually
  choose a different summary to show on
  the axis.

This is a report on closed won opportunities grouped by account and opportunity owner. A scatter chart can reveal the potential for tapping into accounts with a higher annual revenue.

This chart builds on some key fields:

1. **Plot By** automatically chooses the first report grouping. We manually chose the second grouping to show opportunities won by `Account Name`.

2. **X-axis** shows record count when there are no summarized fields or autoselects the first summary field. Since summary is more useful, the chart shows `Sum of Won Amount`.

3. **Y-axis** here shows a manually selected summary field, `Average Annual Revenue`.

4. **Group By** is not set by default. Since the report has another grouping, the chart groups data further by the manually selected `Opportunity Owner` grouping. The colored dots in the chart and legend show this grouping.

The scatter chart component type in dashboards has these limitations:

- It doesn't show tabular reports.
- You can't sort by labels or values.
- You can only change how Y-axis units are displayed.
- You can manually define the range for Y-axis alone.

SEE ALSO:

Chart Types

Modify a Dashboard Component in Salesforce Classic

Data Settings for Dashboard Scatter Chart Components

Formatting Settings for Dashboard Scatter Chart Components

Data Settings for Dashboard Scatter Chart Components

Formatting Settings for Dashboard Scatter Chart Components

Data Settings for Dashboard Chart Components

Choose the data you want to show on your report or dashboard chart. The chart automatically selects groupings and summary values from your report. You can override some of those choices to focus on the data you need to share.

1. Scatter Charts

   Use scatter charts to show meaningful information using one or two groups of report data plus summaries.

2. Data Settings for Dashboard Bar and Column Chart Components

   Horizontal bar and vertical column charts are useful for comparing the values of one or more report groupings. Use the Component Data tab to select the groupings and summaries you want your bar or column chart to display.

3. Data Settings for Dashboard Funnel Chart Components

   Funnel charts are useful for showing the flow of opportunities through stages.

4. Data Settings for Dashboard Scatter Chart Components

   Scatter charts are useful to show one or two groups of report data plus summaries. These data settings for the scatter chart are available in the dashboard component editor under **Component Data**.

5. Data Settings for Dashboard Gauge Components

   A gauge is used to see how far you are from reaching a goal. It displays a single value, such as closed deals.

6. Data Settings for Dashboard Line Chart Components

   Line charts are useful for showing data over time.

7. Data Settings for Dashboard Metric Components

   The settings on the Component Data tab define how your dashboard metric component gets and manages the data it displays.

8. Data Settings for Dashboard Pie and Donut Chart Components

   A pie chart or a donut chart is good for showing the relative shares of different quantities. Use the component data tab to select the values your pie or donut chart will compare.

9. Data Settings for Dashboard Table Components

   A table component shows columns of data from a custom report in a dashboard. The settings on the Component Data tab control how a table component gets and manages the data it displays.

Scatter Charts

Use scatter charts to show meaningful information using one or two groups of report data plus summaries.

For example, to see how stage duration correlates with the number of activities for opportunities, group your report by `Opportunity Name` and plot the scatter chart by the grouping. Then set `X-Axis` on the chart to `Record Count` and `Y-Axis` to `Stage Duration`. The chart will show a dot for each opportunity. You can tell at a glance if the stage duration is shorter for opportunities that have more activities.

Because a scatter chart shows data grouped by summarized values, you need at least one grouping in your report. Choose a report format that allows groupings, such as, summary, matrix, or joined. You also need at least one summarized field in the report to show data on the axes of the chart. Otherwise, the chart will show record count on the axes. Scatter charts automatically show data from the source report or you can manually choose what information to display for groupings and summaries.

If your source report has	The X-axis automatically displays	The Y-axis automatically displays
No summary fields	Record count.	Record count.
One or more summary fields	First summary.	• Either record count or the second summary in report charts. • The first summary in dashboard charts even if the source report has multiple summaries. But you can manually choose a different summary to show on the axis.

This is a report on closed won opportunities grouped by account and opportunity owner. A scatter chart can reveal the potential for tapping into accounts with a higher annual revenue.

This chart builds on some key fields:

1. **Plot By** automatically chooses the first report grouping. We manually chose the second grouping to show opportunities won by `Account Name`.

2. **X-axis** shows record count when there are no summarized fields or autoselects the first summary field. Since summary is more useful, the chart shows `Sum of Won Amount`.

3. **Y-axis** here shows a manually selected summary field, `Average Annual Revenue`.

4. **Group By** is not set by default. Since the report has another grouping, the chart groups data further by the manually selected `Opportunity Owner` grouping. The colored dots in the chart and legend show this grouping.

The scatter chart component type in dashboards has these limitations:

- It doesn't show tabular reports.
- You can't sort by labels or values.
- You can only change how Y-axis units are displayed.

- You can manually define the range for Y-axis alone.

SEE ALSO:

Chart Types

Modify a Dashboard Component in Salesforce Classic

Data Settings for Dashboard Scatter Chart Components

Formatting Settings for Dashboard Scatter Chart Components

Data Settings for Dashboard Scatter Chart Components

Formatting Settings for Dashboard Scatter Chart Components

Data Settings for Dashboard Bar and Column Chart Components

USER PERMISSIONS		EDITIONS
To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: Salesforce Classic (not available in all orgs)
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Edit My Dashboards	Available in: both Legacy Folder Sharing and Enhanced Folder Sharing
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data **Enhanced Folder Sharing** Manage Dashboards in Public Folders	
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards **Enhanced Folder Sharing** Manage Dynamic Dashboards	

Horizontal bar and vertical column charts are useful for comparing the values of one or more report groupings. Use the Component Data tab to select the groupings and summaries you want your bar or column chart to display.

Choose what values to display on the axes of your chart. Depending on the chart type, axis values can be record count, summary fields, or groupings defined in the report. For example, to see the amount in each sales stage in a report, select `Sum of Amount` as the `X-axis` and `Stage` as the `Y-axis`. The chart displays one bar for each stage, with the length proportional to the total opportunity amount.

Setting	Description
X-Axis	Choose what values to display on the horizontal axis of your bar, column, scatter, or line chart. Depending on the chart type, axis values can be summary fields or groupings. To use the second grouping or summary field defined in the source report's chart, select `Auto`. When the source report has a chart, Auto picks the values used by the chart. If the X-axis corresponds to a custom summary formula that has the `Where Will this Formula Be Displayed?` option set to a grouping level other than `All summary levels`, then the Y-axis and Groupings selection must correspond to that custom summary formula's grouping level.
Y-Axis	Choose what values to display on the vertical axis of your bar, column, scatter, or line chart. Depending on the chart type, axis values can be summary fields or groupings. To use the second grouping or summary field defined in the source report's chart, select `Auto`. When the source report has a chart, Auto picks the values used by the chart. If the Y-axis corresponds to a custom summary formula that has the `Where Will this Formula Be Displayed?` option set to a grouping level other than `All summary levels`, then the X-axis and Groupings selection must correspond to that custom summary formula's grouping level.
Group By	Choose how to group information on your chart. This option is available only if the underlying report has more than one grouping. To use the second grouping or summary field defined in the source report's chart, select `Auto`.
Combination Chart	Select this option to plot additional values on this chart. The chart type you chose must allow combination charts.
Display Units	Choose a scale for displaying your chart values. For table components, this setting applies only to the first column. For best results, choose Auto to let Salesforce select appropriate units.
Drill Down to	Select where users go when they click a dashboard component: the full source report for the dashboard component; the source report filtered by the group, X-axis value, or legend entry they clicked; the detail page for a chart or table element, axis value, or legend entry; or a URL that you specify. (You can't use URLs that begin with "mailto:" or "javascript:".) Filtered and record detail page drill-down are disabled when viewing dashboard charts with more than 200 values.

SEE ALSO:

Formatting Settings for Dashboard Bar Chart Components

Data Settings for Dashboard Funnel Chart Components

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize
Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize
Dashboards

To edit and delete dashboards you created
in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

To edit and delete dashboards you didn't
create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND
View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

To create, edit, and delete dynamic
dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards

Funnel charts are useful for showing the flow of opportunities through stages.

Choose what values to display on the axes of your chart. Depending on the chart type, axis values can be record count, summary fields, or groupings defined in the report.

Setting	Description
Display Units	Choose a scale for displaying your chart values. For table components, this setting applies only to the first column. For best results, choose Auto to let Salesforce select appropriate units.
Drill Down to	Select where users go when they click a dashboard component: the full source report for the dashboard component; the source report filtered by the group, X-axis value, or legend entry they clicked; the detail page for a chart or table element, axis value, or legend entry; or a URL that you specify. (You can't use URLs that begin with "mailto:" or "javascript:".) Filtered and record detail page drill-down are disabled when viewing dashboard charts with more than 200 values.

Setting	Description
Values	Choose what to display as values for your pie chart, donut chart, funnel chart, gauge, or metric. To use the second grouping or summary field defined in the source report's chart, select *Auto*. When the source report has a chart, Auto picks the values used by the chart. In gauges and metrics, Auto shows the value of the first summary field.
Segments	Choose what to display as segments for your funnel chart. To use the second grouping or summary field defined in the source report's chart, select *Auto*. When the source report has a chart, Auto picks the values used by the chart.

SEE ALSO:

[Formatting Settings for Funnel Dashboard Components](#)

Data Settings for Dashboard Scatter Chart Components

USER PERMISSIONS		EDITIONS
To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: Salesforce Classic ([not available in all orgs](#))
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Edit My Dashboards	Available in: both Legacy Folder Sharing and Enhanced Folder Sharing
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data **Enhanced Folder Sharing** Manage Dashboards in Public Folders	
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards **Enhanced Folder Sharing** Manage Dynamic Dashboards	

Scatter charts are useful to show one or two groups of report data plus summaries. These data settings for the scatter chart are available in the dashboard component editor under **Component Data**.

Group and set the scale to help users make sense of the data you are displaying.

Setting	Description
Plot By	Choose the grouping to display on your chart. To always use the first grouping or use what's in the source report's chart, pick Auto.
X-Axis	Choose the values to display on the horizontal axis.
Y-Axis	Choose what values to display on the vertical axis of your bar, column, scatter, or line chart. Depending on the chart type, axis values can be summary fields or groupings. To use the second grouping or summary field defined in the source report's chart, select `Auto`. When the source report has a chart, Auto picks the values used by the chart. If the Y-axis corresponds to a custom summary formula that has the `Where Will this Formula Be Displayed?` option set to a grouping level other than `All summary levels`, then the X-axis and Groupings selection must correspond to that custom summary formula's grouping level.
Group By	Choose how to group information on your chart. This option is available only if the underlying report has more than one grouping. To use the second grouping or summary field defined in the source report's chart, select `Auto`.
Display Units	Change the measure for values displayed on the Y-axis of the chart.
Drill Down to	Select where users go when they click the chart. Options include the source report, the filtered source report, record detail page, or a URL you specify. Filtered report and record detail page options are unavailable when the chart has more than 200 values.

SEE ALSO:

Scatter Charts

Modify a Dashboard Component in Salesforce Classic

Formatting Settings for Dashboard Scatter Chart Components

Data Settings for Dashboard Gauge Components

USER PERMISSIONS		EDITIONS
To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	Available in: Salesforce Classic (not available in all orgs)
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards	Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	Available in: both Legacy Folder Sharing and Enhanced Folder Sharing
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards	

To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing**
	Run Reports AND Manage Dashboards
	Enhanced Folder Sharing
	Edit My Dashboards
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing**
	Run Reports, Manage Dashboards AND View All Data
	Enhanced Folder Sharing
	Manage Dashboards in Public Folders
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing**
	Manage Dynamic Dashboards
	Enhanced Folder Sharing
	Manage Dynamic Dashboards

A gauge is used to see how far you are from reaching a goal. It displays a single value, such as closed deals.

Setting	Description
Values	Choose what to display as values for your pie chart, donut chart, funnel chart, gauge, or metric. To use the second grouping or summary field defined in the source report's chart, select *Auto*. When the source report has a chart, Auto picks the values used by the chart. In gauges and metrics, Auto shows the value of the first summary field.
Display Units	Choose a scale for displaying your chart values. For table components, this setting applies only to the first column. For best results, choose Auto to let Salesforce select appropriate units.
Drill Down to	Select where users go when they click a dashboard component: the full source report for the dashboard component; the source report filtered by the group, X-axis value, or legend entry they clicked; the detail page for a chart or table element, axis value, or legend entry; or a URL that you specify. (You can't use URLs that begin with "mailto:" or "javascript:".) Filtered and record detail page drill-down are disabled when viewing dashboard charts with more than 200 values.

SEE ALSO:

[Formatting Settings for Dashboard Gauge Components](#)

Data Settings for Dashboard Line Chart Components

To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards
	Enhanced Folder Sharing Edit My Dashboards
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data
	Enhanced Folder Sharing Manage Dashboards in Public Folders
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards
	Enhanced Folder Sharing Manage Dynamic Dashboards

EDITIONS

Available in: Salesforce Classic (not available in all orgs)

Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

Line charts are useful for showing data over time.

For example, to see the numbers of leads created each month in a report, set record count as the `Y-axis` and created month for the `X-axis`. The chart displays a line connecting the record count totals for each month.

Setting	Description
`Y-Axis`	Choose what values to display on the vertical axis of your bar, column, scatter, or line chart. Depending on the chart type, axis values can be summary fields or groupings. To use the second grouping or summary field defined in the source report's chart, select `Auto`. When the source report has a chart, Auto picks the values used by the chart. If the Y-axis corresponds to a custom summary formula that has the `Where Will this Formula Be Displayed?` option set to a grouping level other than `All summary levels`, then the X-axis and Groupings selection must correspond to that custom summary formula's grouping level.

Setting	Description
X-Axis	Choose what values to display on the horizontal axis of your bar, column, scatter, or line chart. Depending on the chart type, axis values can be summary fields or groupings. To use the second grouping or summary field defined in the source report's chart, select *Auto*. When the source report has a chart, Auto picks the values used by the chart. If the X-axis corresponds to a custom summary formula that has the `Where Will this Formula Be Displayed?` option set to a grouping level other than `All summary levels`, then the Y-axis and Groupings selection must correspond to that custom summary formula's grouping level.
Group By	Choose how to group information on your chart. This option is available only if the underlying report has more than one grouping. To use the second grouping or summary field defined in the source report's chart, select *Auto*.
Combination Chart	Select this option to plot additional values on this chart. The chart type you chose must allow combination charts.
Display Units	Choose a scale for displaying your chart values. For table components, this setting applies only to the first column. For best results, choose Auto to let Salesforce select appropriate units.
Drill Down to	Select where users go when they click a dashboard component: the full source report for the dashboard component; the source report filtered by the group, X-axis value, or legend entry they clicked; the detail page for a chart or table element, axis value, or legend entry; or a URL that you specify. (You can't use URLs that begin with "mailto:" or "javascript:".) Filtered and record detail page drill-down are disabled when viewing dashboard charts with more than 200 values.

SEE ALSO:

> Formatting Settings for Dashboard Line Chart Components

Data Settings for Dashboard Metric Components

USER PERMISSIONS			EDITIONS
To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards		Available in: Salesforce Classic (not available in all orgs)
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards		Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards		
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards		Available in: both Legacy Folder Sharing and Enhanced Folder Sharing
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards		

<div align="right">

Enhanced Folder Sharing
Edit My Dashboards

</div>

To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data **Enhanced Folder Sharing** Manage Dashboards in Public Folders
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards **Enhanced Folder Sharing** Manage Dynamic Dashboards

The settings on the Component Data tab define how your dashboard metric component gets and manages the data it displays.

Setting	Description
Values	Choose what to display as values for your pie chart, donut chart, funnel chart, gauge, or metric. To use the second grouping or summary field defined in the source report's chart, select *Auto*. When the source report has a chart, Auto picks the values used by the chart. In gauges and metrics, Auto shows the value of the first summary field.
Display Units	Choose a scale for displaying your chart values. For table components, this setting applies only to the first column. For best results, choose Auto to let Salesforce select appropriate units.
Drill Down to	Select where users go when they click a dashboard component: the full source report for the dashboard component; the source report filtered by the group, X-axis value, or legend entry they clicked; the detail page for a chart or table element, axis value, or legend entry; or a URL that you specify. (You can't use URLs that begin with "mailto:" or "javascript:".) Filtered and record detail page drill-down are disabled when viewing dashboard charts with more than 200 values.

SEE ALSO:

[Formatting Settings for Dashboard Metric Components](#)

Data Settings for Dashboard Pie and Donut Chart Components

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

To edit and delete dashboards you didn't create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

To create, edit, and delete dynamic dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards

Available in: Salesforce Classic (not available in all orgs)

Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

A pie chart or a donut chart is good for showing the relative shares of different quantities. Use the component data tab to select the values your pie or donut chart will compare.

Use a pie chart when you have multiple groupings and want to show the proportion of a single value for each grouping against the total.

For example, to see the breakdown of your case queue by case status in a report, set record count for `Values` and status for `Wedges`. The chart displays a circle made up of wedges, each wedge representing the cases in a case status.

Setting	Description
Values	Choose what to display as values for your pie chart, donut chart, funnel chart, gauge, or metric. To use the second grouping or summary field defined in the source report's chart, select `Auto`. When the source report has a chart, Auto picks the values used by the chart. In gauges and metrics, Auto shows the value of the first summary field.

Setting	Description
Wedges	Choose what to display as wedges for your pie or donut chart. To use the second grouping or summary field defined in the source report's chart, select `Auto`. When the source report has a chart, Auto picks the values used by the chart.
Display Units	Choose a scale for displaying your chart values. For table components, this setting applies only to the first column. For best results, choose Auto to let Salesforce select appropriate units.
Drill Down to	Select where users go when they click a dashboard component: the full source report for the dashboard component; the source report filtered by the group, X-axis value, or legend entry they clicked; the detail page for a chart or table element, axis value, or legend entry; or a URL that you specify. (You can't use URLs that begin with "mailto:" or "javascript:".) Filtered and record detail page drill-down are disabled when viewing dashboard charts with more than 200 values.

SEE ALSO:

> [Formatting Settings for Pie and Donut Dashboard Components](#)

Data Settings for Dashboard Table Components

USER PERMISSIONS

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

To edit and delete dashboards you didn't create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

To create, edit, and delete dynamic dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

EDITIONS

Available in: Salesforce Classic ([not available in all orgs](#))

Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

A table component shows columns of data from a custom report in a dashboard. The settings on the Component Data tab control how a table component gets and manages the data it displays.

Field	Description
Display Units	Choose a scale for displaying your chart values. For table components, this setting applies only to the first column. For best results, choose Auto to let Salesforce select appropriate units.
Drill Down to	Select where users go when they click a dashboard component: the full source report for the dashboard component; the source report filtered by the group, X-axis value, or legend entry they clicked; the detail page for a chart or table element, axis value, or legend entry; or a URL that you specify. (You can't use URLs that begin with "mailto:" or "javascript:".) Filtered and record detail page drill-down are disabled when viewing dashboard charts with more than 200 values.

SEE ALSO:

    Formatting Settings for Dashboard Table Components

Visual Settings for Dashboard Chart Components

Choose the type of chart that fits the data you are sharing, then apply the visual settings that will communicate the data most effectively.

1. Formatting Settings for Dashboard Bar Chart Components

   Horizontal bar and vertical column charts are useful for comparing the values of one or more report groupings. Set the scale and sorting to help you make sense of the data you are displaying.

2. Formatting Settings for Funnel Dashboard Components

   Use color to show the status of a value through stages in a funnel chart.

3. Formatting Settings for Dashboard Scatter Chart Components

   Scatter charts are useful to show one or two groups of report data plus summaries. These settings for the scatter chart are available in the dashboard component editor under **Formatting**.

4. Formatting Settings for Dashboard Gauge Components

   Set the breakpoints and colors on your gauge component to help users interpret the current value of the field you are tracking.

5. Formatting Settings for Dashboard Line Chart Components

   Line charts are useful for showing data over time. Use sorting and scale to help your users make sense of the data on your line chart.

6. Formatting Settings for Dashboard Metric Components

   A metric component displays one value at a point in time. Use color to help users make sense of the data in the metric component.

7. Formatting Settings for Pie and Donut Dashboard Components

   A pie chart is good for showing the relative shares of different quantities. Use the formatting tab to choose how to configure and label the divisions in your pie chart.

8. Formatting Settings for Dashboard Table Components

   A table component shows columns of data from a custom report in a dashboard. You can use color and scale to help users interpret the report data the table displays.

Formatting Settings for Dashboard Bar Chart Components

Horizontal bar and vertical column charts are useful for comparing the values of one or more report groupings. Set the scale and sorting to help you make sense of the data you are displaying.

Setting	Description
Sort Rows By	Choose a sorting element to determine what element you want displayed first in the horizontal axis of any horizontal chart or the vertical axis of any vertical chart. For a table, choose the sort order for the default two-column table to be ascending or descending by row labels or values.
Maximum Values Displayed	Set the maximum number of elements to include in the top-level grouping of the horizontal axis of a horizontal chart, vertical axis of a vertical chart, or selected axis of a stacked bar chart. For a table, set the maximum number of rows to include. For example, if you want to list only your top five salespeople, create an opportunity report that lists total opportunity amounts by owner and enter *5* in this field.
Legend Position	Choose a place to display the legend in relation to your chart.

Setting	Description
Show Chatter Photos	Display Chatter photos for up to 20 records in a horizontal bar chart component whose source report is grouped by a user or group name field. If there are more than 20 records with photos, record names are shown instead of photos. Set `Grouping Display` to *None* to show photos. Set the `Drill Down to` option to *Record Detail Page* to take users directly to user profile or group pages when they click photos. Chatter must be enabled for photos to be displayed. Depending on your organization's setup, you may not see photos on tables and charts.
Show Values	Display the values of individual records or groups on the chart. This only applies to certain chart types.
Enable Hover	Display values, labels, and percentages when hovering over charts. Hover details depend on chart type. Percentages apply to pie, donut, and funnel charts only. Hover is disabled if your chart has more than 200 data points.

Formatting Settings for Funnel Dashboard Components

USER PERMISSIONS		EDITIONS
To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: Salesforce Classic (not available in all orgs)
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Edit My Dashboards	Available in: both Legacy Folder Sharing and Enhanced Folder Sharing
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data **Enhanced Folder Sharing** Manage Dashboards in Public Folders	
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards **Enhanced Folder Sharing** Manage Dynamic Dashboards	

Use color to show the status of a value through stages in a funnel chart.

Setting	Description
`Sort Rows By`	Choose a sorting element to determine what element you want displayed first in the horizontal axis of any horizontal chart or the vertical axis of any vertical chart. For a table, choose the sort order for the default two-column table to be ascending or descending by row labels or values.
`Maximum Values Displayed`	Set the maximum number of elements to include in the top-level grouping of the horizontal axis of a horizontal chart, vertical axis of a vertical chart, or selected axis of a stacked bar chart. For a table, set the maximum number of rows to include. For example, if you want to list only your top five salespeople, create an opportunity report that lists total opportunity amounts by owner and enter *5* in this field.
`Legend Position`	Choose a place to display the legend in relation to your chart.
`Combine Small Groups into "Others"`	Click this link to create a custom table. The `Maximum Values Displayed` field is populated with the value you entered, and the first two columns are prepopulated with the default columns. To customize a table, the source report must be summary or matrix format and contain a chart.
`Show Values`	Display the values of individual records or groups on the chart. This only applies to certain chart types.
`Show Segment %`	Display the percentage value for each segment of funnel charts.
`Enable Hover`	Display values, labels, and percentages when hovering over charts. Hover details depend on chart type. Percentages apply to pie, donut, and funnel charts only. Hover is disabled if your chart has more than 200 data points.

Formatting Settings for Dashboard Scatter Chart Components

USER PERMISSIONS

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

EDITIONS

Available in: Salesforce Classic (not available in all orgs)

Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing**
	Run Reports, Manage Dashboards AND View All Data
	Enhanced Folder Sharing
	Manage Dashboards in Public Folders
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing**
	Manage Dynamic Dashboards
	Enhanced Folder Sharing
	Manage Dynamic Dashboards

Scatter charts are useful to show one or two groups of report data plus summaries. These settings for the scatter chart are available in the dashboard component editor under **Formatting**.

Tweak data displayed on the dashboard scatter chart using these settings.

Setting	Description
Sort Rows By	Currently sorting is unavailable.
Maximum Values Displayed	Set the maximum number of dots to show on the chart. When set to 5 for example, the chart shows 5 dots. These are the top 5 ascending values in the grouping used by the Plot By field of the scatter chart.
Axis Range	Keep automatic or choose manual to enter minimum and maximum values for the Y-axis range. If there are values outside the manual range, Y-axis automatically extends to include them.
Legend Position	Choose a place to display the legend in relation to your chart.
Show Details on Hover	Display values or labels when hovering over charts.

Formatting Settings for Dashboard Gauge Components

USER PERMISSIONS		EDITIONS
To create dashboards:	**Legacy Folder Sharing**	Available in: Salesforce Classic (not available in all orgs)
	Run Reports AND Manage Dashboards	
	Enhanced Folder Sharing	Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
	Run Reports AND Create and Customize Dashboards	
To edit and delete dashboards you created:	**Legacy Folder Sharing**	
	Run Reports AND Manage Dashboards	
	Enhanced Folder Sharing	Available in: both Legacy Folder Sharing and Enhanced Folder Sharing
	Run Reports AND Create and Customize Dashboards	
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing**	
	Run Reports AND Manage Dashboards	

| | **Enhanced Folder Sharing** |
| | Edit My Dashboards |

To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing**
	Run Reports, Manage Dashboards AND View All Data
	Enhanced Folder Sharing
	Manage Dashboards in Public Folders

To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing**
	Manage Dynamic Dashboards
	Enhanced Folder Sharing
	Manage Dynamic Dashboards

Set the breakpoints and colors on your gauge component to help users interpret the current value of the field you are tracking.

Optionally, set conditional highlighting by defining up to three value ranges and colors. You need to also set the minimum and maximum for the scale. You must set highlighting to follow a component and receive alerts in your Chatter feed when the value crosses a threshold.

Setting	Description
Minimum	The lowest value on the chart.
Low Range Color	Select a color to represent the low range, up to the first breakpoint.
Breakpoint 1	The value that separates the middle and high range colors on the dashboard.
Middle Range Color	Select a color to represent the middle range, between the first and second breakpoints.
Breakpoint 2	Place the chart above or below your report.
High Range Color	Select a color to represent the high range, beyond the second breakpoint.
Maximum	The highest value on the chart.
Show %	Display the percentage value for each wedge of pie and donut charts, or for each segment of funnel charts. On a gauge, show the percentage value of the point where the needle is pointing.
Show Total	Display the total value for the chart. For a table, include the sum total for number and currency summary fields.

SEE ALSO:

   Modify a Dashboard Component in Salesforce Classic

Formatting Settings for Dashboard Line Chart Components

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

To edit and delete dashboards you didn't create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

To create, edit, and delete dynamic dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards

Available in: Salesforce Classic (not available in all orgs)

Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

Line charts are useful for showing data over time. Use sorting and scale to help your users make sense of the data on your line chart.

Setting	Description
Sort Rows By	Choose a sorting element to determine what element you want displayed first in the horizontal axis of any horizontal chart or the vertical axis of any vertical chart. For a table, choose the sort order for the default two-column table to be ascending or descending by row labels or values.
Maximum Values Displayed	Set the maximum number of elements to include in the top-level grouping of the horizontal axis of a horizontal chart, vertical axis of a vertical chart, or selected axis of a stacked bar chart. For a table, set the maximum number of rows to include. For example, if you want to list only your top five salespeople, create an opportunity report that lists total opportunity amounts by owner and enter *5* in this field.
Y-Axis Range	Choose a manual or automatic axis range for the vertical axis of a bar, line, or column chart. If you choose manual, enter numbers for the minimum and maximum axis values to be displayed. If there are data points outside the range that you set, the axis automatically extends to include those values when you generate the chart.

Setting	Description
Legend Position	Choose a place to display the legend in relation to your chart.
Enable Hover	Display values, labels, and percentages when hovering over charts. Hover details depend on chart type. Percentages apply to pie, donut, and funnel charts only. Hover is disabled if your chart has more than 200 data points.

Formatting Settings for Dashboard Metric Components

USER PERMISSIONS

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

To edit and delete dashboards you didn't create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

To create, edit, and delete dynamic dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards

EDITIONS

Available in: Salesforce Classic (not available in all orgs)

Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

A metric component displays one value at a point in time. Use color to help users make sense of the data in the metric component.

Optionally, set conditional highlighting by defining up to three value ranges and colors. You must set highlighting to follow a component and receive alerts in your Chatter feed when the value crosses a threshold.

📝 Note:

- Enter metric labels directly on components by clicking the empty text field next to the grand total.

- Metric components placed directly above and below each other in a dashboard column are displayed together as a single component.
- If you don't define breakpoint values or if you leave them blank, alerts won't be sent for the component.

Setting	Description
Low Range Color	Select a color to represent the low range, up to the first breakpoint.
Breakpoint 1	The value that separates the middle and high range colors on the dashboard.
Middle Range Color	Select a color to represent the middle range, between the first and second breakpoints.
Breakpoint 2	Place the chart above or below your report.
High Range Color	Select a color to represent the high range, beyond the second breakpoint.

SEE ALSO:

Modify a Dashboard Component in Salesforce Classic

Formatting Settings for Pie and Donut Dashboard Components

USER PERMISSIONS		EDITIONS

USER PERMISSIONS

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

To edit and delete dashboards you didn't create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

EDITIONS

Available in: Salesforce Classic (not available in all orgs)

Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

To create, edit, and delete dynamic dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards

A pie chart is good for showing the relative shares of different quantities. Use the formatting tab to choose how to configure and label the divisions in your pie chart.

Setting	Description
Sort Rows By	Choose a sorting element to determine what element you want displayed first in the horizontal axis of any horizontal chart or the vertical axis of any vertical chart. For a table, choose the sort order for the default two-column table to be ascending or descending by row labels or values.
Maximum Values Displayed	Set the maximum number of elements to include in the top-level grouping of the horizontal axis of a horizontal chart, vertical axis of a vertical chart, or selected axis of a stacked bar chart. For a table, set the maximum number of rows to include. For example, if you want to list only your top five salespeople, create an opportunity report that lists total opportunity amounts by owner and enter 5 in this field.
Legend Position	Choose a place to display the legend in relation to your chart.
Combine Small Groups into "Others"	Combine all groups less than or equal to 3% of the total into a single "Others" wedge or segment. Deselect to show all values individually on the chart. This only applies to pie, donut, and funnel charts. This option is on by default for pie and donut charts, and off for funnel.
Show Values	Display the values of individual records or groups on the chart. This only applies to certain chart types.
Show %	Display the percentage value for each wedge of pie and donut charts.
Enable Hover	Display values, labels, and percentages when hovering over charts. Hover details depend on chart type. Percentages apply to pie, donut, and funnel charts only. Hover is disabled if your chart has more than 200 data points.
Show Total	Display the total value for the chart. For a table, include the sum total for number and currency summary fields.

Formatting Settings for Dashboard Table Components

To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To edit and delete dashboards you created in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards

To edit and delete dashboards you didn't create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

To create, edit, and delete dynamic dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards

A table component shows columns of data from a custom report in a dashboard. You can use color and scale to help users interpret the report data the table displays.

You can use the default two-column table or create a custom table with up to four columns and totals.

The default two-column table uses the first grouping and summary field from the chart in the source report. If the report has no chart, default columns are based on the first grouping and summary field in the report.

Customized tables allow null values in the results. Default two-column tables do not.

To use a tabular report as the source report, `Rows to Display` must be set for that report.

Optionally, set conditional highlighting by defining up to three value ranges and colors. Highlighting only applies to the first summary field column in the table.

Setting	Description
Sort Rows By	Choose a sorting element to determine what element you want displayed first in the horizontal axis of any horizontal chart or the vertical axis of any vertical chart. For a table, choose the sort order for the default two-column table to be ascending or descending by row labels or values.
Maximum Values Displayed	Set the maximum number of elements to include in the top-level grouping of the horizontal axis of a horizontal chart, vertical axis of a vertical chart, or selected axis of a stacked bar chart. For a table, set the maximum number of rows to include. For example, if you want to list only your top five salespeople, create an opportunity report that lists total opportunity amounts by owner and enter *5* in this field.
Show Chatter Photos	Display Chatter photos for up to 20 records in a table component whose source report is grouped by a user or group name field. If there are more than 20 records with photos, record names are shown instead of photos. Set `Grouping Display` to *None* to show photos. Set the `Drill Down to` option to *Record Detail Page* to take users directly to user profile or group pages when they click photos. Chatter must be enabled for photos to be displayed. Depending on your organization's setup, you may not see photos on tables and charts.
Customize Table	Click this link to create a custom table. The `Maximum Values Displayed` field is populated with the value you entered, and the first two columns are prepopulated with the default columns. To customize a table, the source report must be summary or matrix format and contain a chart.
Table Columns	Specify up to four columns to display in the table. Available columns can be any grouping or summary field used in the chart. Update the report's chart or groupings to make more columns available for the dashboard table.
Sort Ascending	Sort the custom table in A-to-Z or smallest-to-largest order in a column. You can't sort on second-level groupings.
Sort Descending	Sort the custom table in Z-to-A or largest-to-smallest order in a column. You can't sort on second-level groupings.
Show Total	Display the total value for the chart. For a table, include the sum total for number and currency summary fields.
Reset Table to Defaults	Go back to the default two-column table.
Low Range Color	Select a color to represent the low range, up to the first breakpoint.
Breakpoint 1	The value that separates the low and middle range colors on the dashboard.
Middle Range Color	Select a color to represent the middle range, between the first and second breakpoints.
Breakpoint 2	Breakpoint 2
High Range Color	Select a color to represent the high range, beyond the second breakpoint.

SEE ALSO:

[Modify a Dashboard Component in Salesforce Classic](#)

Edit Dashboards in Accessibility Mode in Salesforce Classic

USER PERMISSIONS		EDITIONS
To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	Available in: Salesforce Classic (not available in all orgs)
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards	Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	Available in: both Legacy Folder Sharing and Enhanced Folder Sharing
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards	
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	
	Enhanced Folder Sharing Edit My Dashboards	
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data	
	Enhanced Folder Sharing Manage Dashboards in Public Folders	
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards	
	Enhanced Folder Sharing Manage Dynamic Dashboards	

In Salesforce Classic, you can edit dashboards in Accessibility mode.

🛑 **Important:** This topic applies only if you're *not* using the dashboard builder. *Dashboard builder* is a drag-and-drop interface for creating and modifying dashboards.

A *dashboard* shows data from source reports as visual components, which can be charts, gauges, tables, metrics, or Visualforce pages. To customize a dashboard, view it and click **Edit**.

From the Dashboard Edit page, you can:

- See the running user for the dashboard in the `Displaying data as` field.
- Click **Dashboard Properties** to change the title, folder, running user, and more.
- Click **Done** to view the dashboard. All changes you make to the dashboard are saved as you make them.
- Click the **Delete** button to delete the entire dashboard.
- Click **Add Component** in any column.

- Click **Narrow**, **Medium**, or **Wide** to set a column's width.

  > 📝 Note: If your component is a pie or donut chart with `Show Values` or `Show Percentages` enabled and `Legend Position` set to Right, the dashboard column width must be `Wide` for values and percentages to show on the dashboard.

- Click ◀, ▲, ▼, and ▶ to rearrange components in the dashboard.
- Click **Edit** to modify component properties like the component type, display units, source report, and more.
- Click **Delete** to remove a component from the dashboard. Deleted components don't get stored in the Recycle Bin.
- Click a dashboard component or its elements to drill down to the source report, filtered report, record detail page, or other URL. If you drill down on a filtered component, the dashboard filters are applied to the source report.

1. Set Dashboard Properties in Accessibility Mode
2. Adding and Editing Dashboard Components in Accessibility Mode
3. Dashboard Component Properties in Accessibility Mode

SEE ALSO:

Adding and Editing Dashboard Components in Accessibility Mode
Build a Salesforce Classic Dashboard
Set Dashboard Properties in Accessibility Mode

Set Dashboard Properties in Accessibility Mode

> **⊘ Important:** This topic applies only if you're *not* using the dashboard builder. *Dashboard builder* is a drag-and-drop interface for creating and modifying dashboards.

1. Edit a dashboard and click **Dashboard Properties**.

2. Do the following:

   - Enter a title and description for the dashboard.

   - If you have the "Customize Application" permission, enter a unique name to be used by the API and managed packages.

   - Select the number of columns for this dashboard. Each dashboard can have two or three columns.

     > **⊘ Important:** Before removing a column, move its components to another column; otherwise, they may not be visible.

   - Select a folder to store the dashboard. The folder should be accessible by all of your intended viewers.

   - Choose the `Dashboard Running User` to set visibility settings for the dashboard:

     - Select `Run as specified user` and set the `Running User` field to show all dashboard users the same data, regardless of their personal security settings. If you don't have "View All Data," you can only choose yourself.

     - Select `Run as logged-in user` to show data to each user according to his or her access level.

       - Set the `View Edit Page as` field to preview the dashboard edit page from the point of view of the selected user.

       - If you have the "View My Team's Dashboards" or "View All Data" permission, select `Let authorized users change running user` to enable those with permission to change the running user on the dashboard edit page. Users with "View My Team's Dashboards" can view the dashboard as any user below them in the role hierarchy. Users with "View All Data" can edit the dashboard and view it as any user in their organization.

   - Under Component Settings, select the title color and size, text color, and background fade. If you don't want a gradient, choose the same color for both `Starting Color` and `Ending Color`.

3. Click **Save**.

SEE ALSO:

[Set Up Dynamic Dashboards in Salesforce Classic](#)

[Choose a Running User in Salesforce Classic](#)

EDITIONS

Available in: Salesforce Classic ([not available in all orgs](#))

Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

USER PERMISSIONS

To create dashboards:

- **Legacy Folder Sharing**
  Run Reports AND Manage Dashboards

  Enhanced Folder Sharing
  Run Reports AND Create and Customize Dashboards

To create, edit, and delete dynamic dashboards:

- **Legacy Folder Sharing**
  Manage Dynamic Dashboards

  Enhanced Folder Sharing
  Manage Dynamic Dashboards

To enable choosing a different running user for the dashboard:

- View My Team's Dashboards OR View All Data

Adding and Editing Dashboard Components in Accessibility Mode

<table>
<tr><td>USER PERMISSIONS</td><td></td></tr>
<tr><td>To create dashboards:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards</td><td rowspan="5">EDITIONS

Available in: Salesforce Classic (not available in all orgs)

Available in: Group (View Only), Professional, Enterprise, Performance, Unlimited, and Developer Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing</td></tr>
<tr><td>To edit and delete dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards</td></tr>
<tr><td>To edit and delete dashboards you created in public folders:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Edit My Dashboards</td></tr>
<tr><td>To edit and delete dashboards you didn't create in public folders:

Legacy Folder Sharing
Run Reports, Manage Dashboards AND View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders</td></tr>
<tr><td>To create, edit, and delete dynamic dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards</td></tr>
</table>

🛑 **Important:** This topic applies only if you're *not* using the dashboard builder. *Dashboard builder* is a drag-and-drop interface for creating and modifying dashboards.

A *dashboard* shows data from source reports as visual components, which can be charts, gauges, tables, metrics, or Visualforce pages. The components provide a snapshot of key metrics and performance indicators for your organization. Each dashboard can have up to 20 components.

To add a dashboard component:

1. Edit a dashboard.

2. Click **Add Component** from the top of any column and define component properties.

3. Choose the `Custom Report` to use for your dashboard. If you chose the `Visualforce Page`, or `Custom S-Control` component, select a page or s-control and enter the display height. You can show a joined report that includes a chart on a dashboard. Edit the joined report dashboard component and select **Use chart as defined in the source report**.

> **Note:**
> - Custom forecast and lead reports that you created using a standard report may not be available in the `Custom Report` list.
> - To use a tabular report on a dashboard, first limit the row count, by setting the `Rows to Display` option, the sort column, and the order on the Select Criteria page of the report. You can't use gauge or metric components on dashboards using tabular reports.

4. Enter the appropriate settings for the component type you selected:Choose settings on the Formatting tab for the component type you selected:

- Formatting Settings for Dashboard Bar Chart Components
- Formatting Settings for Dashboard Scatter Chart Components
- Formatting Settings for Dashboard Gauge Components
- Formatting Settings for Dashboard Metric Components
- Formatting Settings for Dashboard Table Components

5. Click **Save**.

> **Note:** Metric components placed directly above and below each other in a dashboard column are displayed together as a single component.

SEE ALSO:

Chart Types

Modify a Dashboard Component in Salesforce Classic

Dashboard Component Properties in Accessibility Mode

USER PERMISSIONS		EDITIONS
To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: Salesforce Classic (not available in all orgs)
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards	Available in: **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions Available in: both Legacy Folder Sharing and Enhanced Folder Sharing
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Edit My Dashboards	

To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data
	Enhanced Folder Sharing Manage Dashboards in Public Folders
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards
	Enhanced Folder Sharing Manage Dynamic Dashboards

🛇 **Important:** This topic applies only if you're *not* using the dashboard builder. *Dashboard builder* is a drag-and-drop interface for creating and modifying dashboards.

The following settings may vary according to the component type you select.

Field	Description
Component Type	Select vertical or horizontal bar chart, line chart, pie or donut chart, table, metric, gauge, Visualforce page, or custom s-control.
Header	Enter text to display at the top of the dashboard component.
Footer	Enter text to display at the bottom of the dashboard component.
Title	Enter a title to identify the dashboard component.
Display Units	Choose a scale for displaying your chart values. For table components, this setting applies only to the first column. For best results, choose Auto to let Salesforce select appropriate units.
Drill Down to	Select what happens when users click a dashboard component: • Source Report—Takes the user to the full source report for the dashboard component. • Filtered Source Report—When users click individual groups, X-axis values, or legend entries, they are taken to the source report filtered by what they clicked. For example, if you had a stacked vertical column chart of opportunities grouped by stage, with months as the X-axis, you could click an individual stage in a bar, a month on the X-axis, or a legend entry for a stage to drill down to the filtered source report. (Not available for gauges, metrics, or tables.) • Record Detail Page—When users click chart or table elements, axis values, or legend entries, they are taken to the detail page for that record. You can only choose this option for tables and charts that use a source report grouped by record name, record owner, or feed post. (Not available for gauges or metrics.) • Other URL—Takes the user to the URL that you specify. You can't add URLs that begin with "mailto:" or "javascript:" to dashboard components. 📝 **Note:** Filtered and record detail page drill-down are disabled for dashboard charts with more than 200 values.

Field	Description
Drill Down URL	Specify the URL that users go to when they click the dashboard component. Use this option to send users to another dashboard, report, record detail page, or other system that uses a Web interface.

SEE ALSO:

[Modify a Dashboard Component in Salesforce Classic](#)

[Dashboard Component Types](#)

Filter a Dashboard

<table>
<tr><td colspan="2" style="background:#7a93a8;color:white">USER PERMISSIONS</td></tr>
<tr>
<td>To add filters to dashboards you created:</td>
<td>Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards</td>
</tr>
<tr>
<td>To add filters to dashboards you created in public folders:</td>
<td>Legacy Folder Sharing
Only available in Enhanced Folder Sharing

Enhanced Folder Sharing
Edit My Dashboards OR Manage Dashboards in Public Folders</td>
</tr>
<tr>
<td>To add filters to dashboards you didn't create:</td>
<td>Legacy Folder Sharing
Run Reports, Manage Dashboards, and View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders</td>
</tr>
<tr>
<td>To add filters to dynamic dashboards:</td>
<td>Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards</td>
</tr>
<tr>
<td>To view, refresh, and apply filters to dashboards you created:</td>
<td>Legacy Folder Sharing
Run Reports

Enhanced Folder Sharing
Run Reports</td>
</tr>
<tr>
<td>To view, refresh, and apply filters to dashboards in public folders:</td>
<td>Legacy Folder Sharing
Run Reports</td>
</tr>
</table>

EDITIONS

Available in: both Salesforce Classic ([not available in all orgs](#)) and Lightning Experience

Available in: **Essentials**, **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

Dashboard filters make it easy to provide different combinations of data from a single dashboard. You don't need separate dashboards for different sets of users — give each group a filter that makes sense for them. When you use a filter on a dashboard, the filtered view is shown again the next time you visit the dashboard.

Each filter has a name, a filter operator and up to 50 values.

You can filter on picklists, lookups, and checkboxes, and on text, numeric, date, and datetime fields.

When you change filters on a dashboard, the dashboard shows previously cached data, if it exists. If no data exists, the dashboard fetches the latest. Either way, the dashboard's "Last Refreshed" date shows you the timestamp for the data you're viewing. Click **Refresh** to get the most recent data.

Note: As you prepare to filter dashboards, keep these dashboard filter limitations in mind.

- You can't add filters to dashboards that contain Visualforce or s-control components.
- It's not possible to filter on bucket fields. However, it is possible to use a report filtered on a bucket field on the dashboard page.
- Filters aren't applied when you schedule or email a dashboard.
- You can't filter data on a joined report in dashboard view or add a filter to a dashboard that only has joined reports.
- You can't use custom summary formulas in dashboard filters.

1. Add a Dashboard Filter

   To create a dashboard filter, select a field that contains the type of information you want to filter, then define how the filter returns the data.

2. Apply a Dashboard Filter

   Filter a dashboard to analyze the information interactively. After you filter a dashboard, the filtered view is preserved so that the next time you see the dashboard, data is filtered in the same view.

3. Filter Dashboards via URL Parameters in Lightning Experience

   There's no need to set up the same filters each time you view your dashboard in Lightning Experience. Create a custom URL that includes the filter values. Then when you access the URL, your dashboard opens with the filters already set. For example, you can customize your Opportunity Stages dashboard with a different URL for each account owner or create a URL to zero in on the closed deals for a particular account.

4. Dashboard Filter Examples

   Filters on a dashboard allow you to choose different views of data. After you filter a dashboard, the filtered view is preserved so that the next time you see the dashboard, data is filtered in the same view. Without dashboard filters, you'd have to create multiple dashboards, each with its own set of filtered reports.

Add a Dashboard Filter

To add filters to dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards
To add filters to dashboards you created in public folders:	**Legacy Folder Sharing** Only available in Enhanced Folder Sharing
	Enhanced Folder Sharing Edit My Dashboards OR Manage Dashboards in Public Folders
To add filters to dashboards you didn't create:	**Legacy Folder Sharing** Run Reports, Manage Dashboards, and View All Data
	Enhanced Folder Sharing Manage Dashboards in Public Folders
To add filters to dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards
	Enhanced Folder Sharing Manage Dynamic Dashboards

To create a dashboard filter, select a field that contains the type of information you want to filter, then define how the filter returns the data.

The field you select for the filter could have *equivalent fields*. Equivalent fields share the same underlying object as the field you select for the filter. You can use equivalent fields to filter components that don't have the exact field you selected for the filter, or to filter some components differently. For example, if you filter on the `Account Owner` field, equivalent fields include `Opportunity Owner` or `Opportunity Created by`, as all three are part of the User object.

When you edit a dashboard, you can see how each component is being filtered under the filter icon (🔻) in Lightning Experience or the `Filtered By` in Salesforce Classic. If equivalent fields are available for a component, you can select a different field to filter the component.

1. Edit a dashboard, then click **+ Filter** in Lightning Experience or **Add Filter** in Salesforce Classic.

2. From the `Field` drop-down, select a field to filter on. The drop-down shows fields that can be used to filter all the dashboard's components. If there are equivalent fields for your selection, hover over the info icon (🛈) to see them.

3. Give the filter a `Display Name` to identify it. If the filter has many equivalent fields, we consider using a name that works for all components.

4. Assign values to the filter.

In Lightning Experience, click **Add Filter Value**, choose an `Operator`, and set a `Value`. Optionally, set custom `Display Text`. Then, click **Apply**.
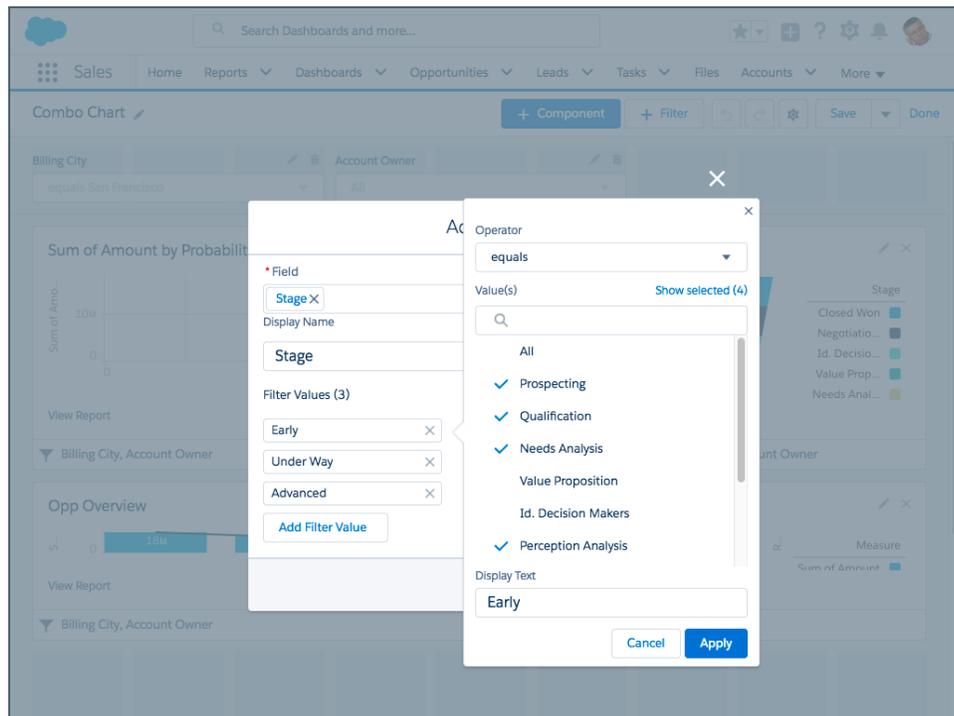
In Salesforce Classic, under Filter Options, select an operator and provide one or more values to filter by.

Use the `Display Text` field to give your filter a name. For example, on a sales dashboard, you might gather several stages of Opportunities into a group called "Early." This lets users quickly filter the dashboard to show only data relevant to deals that are in the early stages of development.
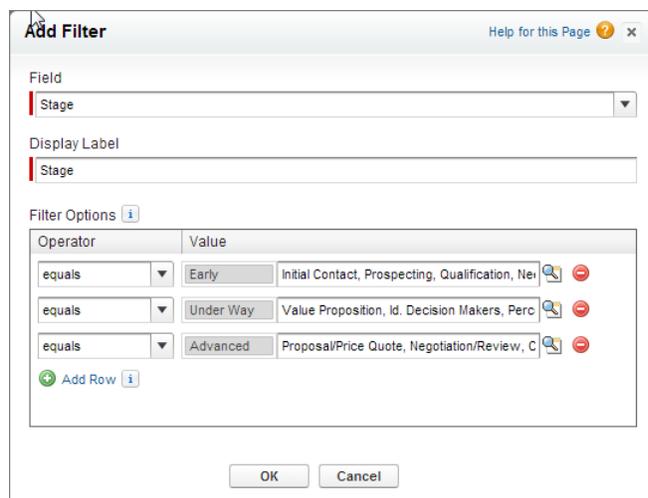
5. In Lightning Experience, click **Add**. In Salesforce Classic, click **OK**. If equivalent fields are available for a component, you can select which one to use to filter the component.

In Lightning Experience, to modify an existing filter, click the pencil icon (🖊). To delete an existing filter, click the garbage icon (🗑).

In Salesforce Classic, to modify or delete an existing filter, from the filter drop-down, select **Edit Filter** or **Remove Filter**.

Apply a Dashboard Filter

Filter a dashboard to analyze the information interactively. After you filter a dashboard, the filtered view is preserved so that the next time you see the dashboard, data is filtered in the same view.

> **Note:** All components on the dashboard aren't necessarily filtered on the same field. The person who created or edited the dashboard specifies which field is used.

1. Open a dashboard.

2. Select an option from the filter drop-down.
   Each filter has one or more options you can choose to narrow your selection further.

3. To see data unfiltered, select **Clear Filter** or **All** from the filter drop-down.

Filter Dashboards via URL Parameters in Lightning Experience

There's no need to set up the same filters each time you view your dashboard in Lightning Experience. Create a custom URL that includes the filter values. Then when you access the URL, your dashboard opens with the filters already set. For example, you can customize your Opportunity Stages dashboard with a different URL for each account owner or create a URL to zero in on the closed deals for a particular account.

1. In the dashboard editor, create up to 3 filters with the values that you want to include in the custom URL.

2. Identify the dashboard URL in your browser's address bar.

3. Add the filter to the end of the URL. For example, the following URL sets Stage (the first filter, or *fv0*) to Closed Won and Account Name (the second filter, or *fv1*) to Acme.

`https://`**`MyDomainName`**`.my.salesforce.com/lightning/r/Dashboard/01ZRM0000005jrv2AA/view?queryScope=userFolders`**`&fv0=Closed%20Won&fv1=Acme`**

Let's take a closer look at what each parameter means.

- *&* — Denotes a new parameter in the URL. If no other parameters are present in the URL, then substitute `?` in place of `&`.

- *fv0* — *fv* stands for "filter value." The number that follows represents the order in which the filter appears in the dashboard. The first filter is `0`, the second filter is `1`, and the third is `2`.

- *=Closed&20Won* — Specifies the filter value. The parameter value must be URI encoded, which means that certain characters (such as spaces) must be written in a format that URLs can understand. For the filter value "Closed Won", the space (' ') between "Closed" and "Won" becomes `%20` when it's URI encoded.

4. Navigate to the dashboard URL with the parameter appended.

When the dashboard opens, it opens with filter values applied.

These limitations apply to dashboard filter URLs:

- Dashboard filter URLs with the `equals` operator are supported, but you can't create filter URLs with other operators such as `less than`, `starts with`, or `contains`.

- You can't change the field being filtered using URL parameters.
- You can assign at most one value per filter. For example, you can include *fv0=Acme* in a filter URL but not *fv0=Acme&fv0=Universal*.
- You can't add new filters to dashboards using filter value URL parameters. You can only modify existing filters.
- You can't delete filters from dashboards using filter value URL parameters. Setting a blank value filters by no text or numerals, but doesn't remove the filter.
- Dashboard filter URLs aren't supported in Salesforce Classic.

Dashboard Filter Examples

USER PERMISSIONS

To add filters to dashboards you created:

Legacy Folder Sharing
Run Reports AND Manage Dashboards

Enhanced Folder Sharing
Run Reports AND Create and Customize Dashboards

To add filters to dashboards you created in public folders:

Legacy Folder Sharing
Only available in Enhanced Folder Sharing

Enhanced Folder Sharing
Edit My Dashboards OR Manage Dashboards in Public Folders

To add filters to dashboards you didn't create:

Legacy Folder Sharing
Run Reports, Manage Dashboards, and View All Data

Enhanced Folder Sharing
Manage Dashboards in Public Folders

To add filters to dynamic dashboards:

Legacy Folder Sharing
Manage Dynamic Dashboards

Enhanced Folder Sharing
Manage Dynamic Dashboards

EDITIONS

Available in: both Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Essentials**, **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

Filters on a dashboard allow you to choose different views of data. After you filter a dashboard, the filtered view is preserved so that the next time you see the dashboard, data is filtered in the same view. Without dashboard filters, you'd have to create multiple dashboards, each with its own set of filtered reports.

Sales Performance by Products

Create a sales dashboard that lets viewers track sales performance by products. To do so, create a dashboard with key performance indicators, like closed revenue. Add a filter on the `Product Name` field so viewers can see performance by product.
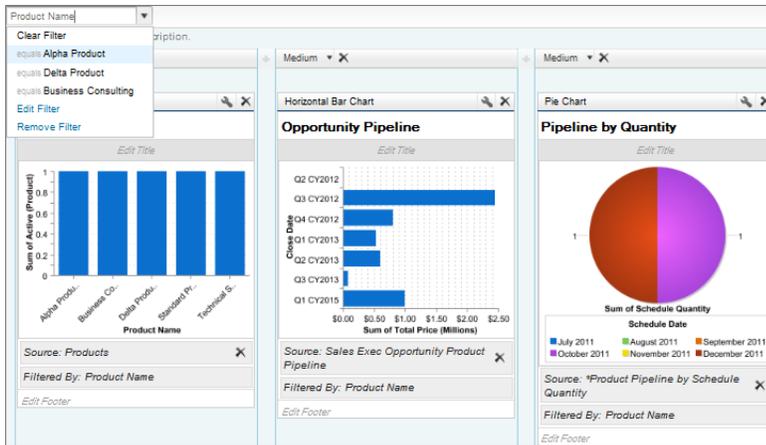
Deals for Different Account Segments

Show how different types of deals are performing across different account segments by creating a dashboard with two filters. Create a filter for `Annual Revenue` that contains several ranges that reflect how your organization segments accounts, then create a filter on opportunity `Type`, and finally add a filter to show deals by `Region`. Create this as a dynamic dashboard to let the entire sales organization use it: managers can use it to view the performance of their teams, while reps can use it to monitor their own performance.
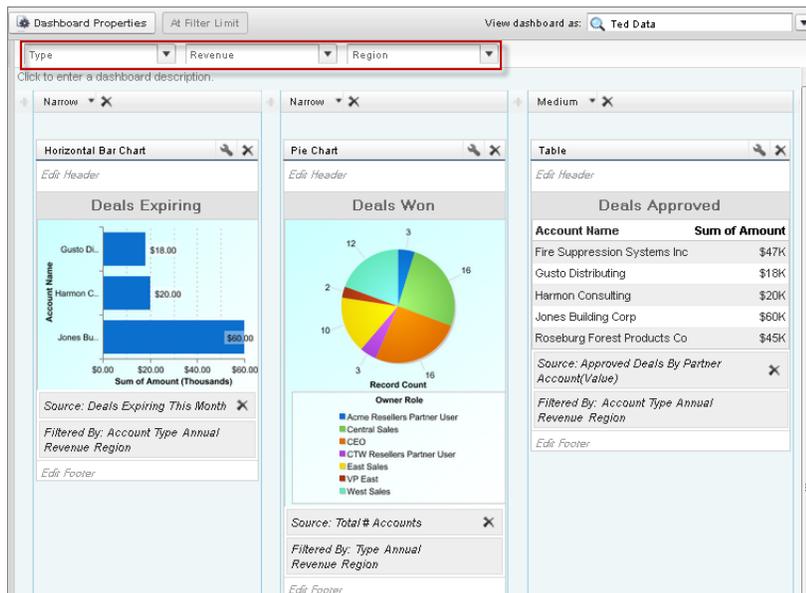
Dynamic Dashboards: Choose Who People View a Dashboard As

Say that your sales people can only view their own opportunities, but you'd like to review all opportunities closed in the last quarter. Create a dashboard and let people view the dashboard as you (or anyone else who can see all opportunities). When your sales people open the dashboard, they see info about all opportunities instead of only their opportunities. (Their data access in Salesforce remains unchanged. They can only see more data in your dashboard.)

Available in: both Salesforce Classic and Lightning Experience

Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

[Dynamic Dashboards: Choose Who People View a Dashboard as in Lightning Experience](#)
Scope the data that dashboard readers see by specifying who they view the dashboard as. Remember, when you specify who people view the dashboard as you can give dashboard readers a broader view of data than they normally have. Be sure to control access to your dashboard by saving it in an appropriate dashboard folder.

[Choose a Running User in Salesforce Classic](#)
Select a *running user* to specify which data to display in a dashboard.

Dynamic Dashboards: Choose Who People View a Dashboard as in Lightning Experience

USER PERMISSIONS

To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Run Reports AND Create and Customize Dashboards
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards **Enhanced Folder Sharing** Edit My Dashboards
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data **Enhanced Folder Sharing** Manage Dashboards in Public Folders
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards **Enhanced Folder Sharing** Manage Dynamic Dashboards
To specify who people view a Lightning Experience dashboard as:	**Legacy Folder Sharing** View My Team's Dashboards OR View All Data

	Enhanced Folder Sharing
	View My Team's Dashboards OR View All Data
To enable dashboard readers to choose who they view the dashboard as:	**Legacy Folder Sharing**
	View My Team's Dashboards OR View All Data
	Enhanced Folder Sharing
	View My Team's Dashboards OR View All Data

Scope the data that dashboard readers see by specifying who they view the dashboard as. Remember, when you specify who people view the dashboard as you can give dashboard readers a broader view of data than they normally have. Be sure to control access to your dashboard by saving it in an appropriate dashboard folder.

| Available in: Lightning Experience |
| Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions |
| Available in: Enhanced Folder Sharing |

People view dashboards as a specified user — whomever is listed above the dashboard next to "Viewing as." Specify who people view the dashboard as while creating or editing a dashboard.

If you have "View All Data," you can choose any user in your organization to be a running user of the dashboard. If you have "View My Team's Dashboards," you can choose any user below you in the role hierarchy.

1. Edit a dashboard.

2. Open the Properties menu by clicking ⚙ .

3. Under **View Dashboard As**, select who people view the dashboard as:

   • **Me** — Dashboard readers see data in the dashboard according to your access to data.

      For example, if you can only see Opportunities in Canada, then dashboard readers only see data about Opportunities in Canada.

   • **Another person** — Dashboard readers see data in the dashboard according to the data access level of whomever you specify.

      For example, if you choose someone who can see Opportunities from any country, then dashboard readers see data about Opportunities from all countries.

   • **The dashboard viewer** — Dashboard readers see data as themselves, according to their own access to data. These types of dashboards are often called *dynamic dashboards*.

      Your organization can have up to 5 dynamic dashboards for Enterprise Edition, 10 for Unlimited and Performance Edition, and 3 for Developer Edition. Dynamic dashboards aren't available in other editions. Additional dynamic dashboards may be available for purchase.

      Take note of these dynamic dashboard limitations:

      – You can't follow components on dynamic dashboards.

      – You can't save dynamic dashboards in private folders.

      – You can't schedule refreshes for dynamic dashboards. They must be refreshed manually.

4. Optionally, select **Let dashboard viewers choose whom they view the dashboard as** to enable a reader with appropriate user permissions to choose who they view the dashboard as. With the "View My Team's Dashboards" user permission, the reader can view the dashboard as themself or as anyone beneath them in the role hierarchy. With the "View All Data" user permission, the reader can view the dashboard as anyone.

5. From the Properties window, click **Save**. Then, from the Dashboard Builder, click **Save** again.

When people open your dashboard, they see data as the person that you specified.

Choose a Running User in Salesforce Classic

Select a *running user* to specify which data to display in a dashboard.

Available in: Salesforce Classic (not available in all orgs)

Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

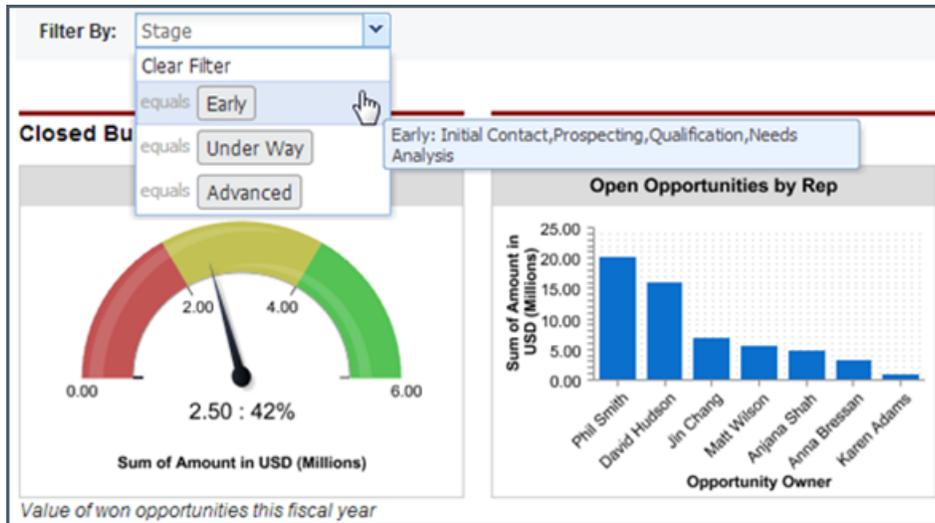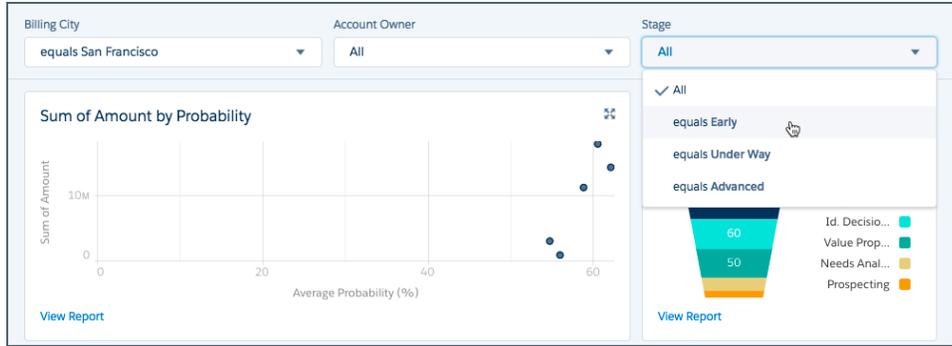Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

Each dashboard has a *running user*, whose security settings determine which data to display in a dashboard.

If you have "View All Data," you can choose any user in your organization to be a running user of the dashboard. If you have "View My Team's Dashboards," you can choose any user below you in the role hierarchy.

> **Note:** Dashboard components that use Visualforce ignore the running user; content displays only if the viewing user has access to the Visualforce page. Other components in the dashboard are not affected.

To select a dashboard's running user in Salesforce Classic:

1. Edit a dashboard.

2. Click the ▾ button next to the `View dashboard as` field.

> **Note:** If you don't have "Manage Dynamic Dashboards" permission, just enter a running user and skip to the final step. Enter "*" to see all available users.

3. Choose a running user setting.

- **Run as specified user.** The dashboard runs using the security settings of that single, specific user. All users with access to the dashboard see the same data, regardless of their own personal security settings. This approach is perfect for sharing the big picture across a hierarchy or motivating team members by showing peer performance within a team. If you don't have "View All Data," you can only choose yourself. If you have "View My Team's Dashboards," you can choose any user below you in the role hierarchy.

- **Run as logged-in user.** A *dynamic dashboard* runs using the security settings of the user viewing the dashboard. Each user sees the dashboard according to his or her own access level. This approach helps administrators share one common set of dashboard components to users with different levels of access.

4. Optionally, select `Let authorized users change running user` to enable those with permission to change the running user on the dashboard view page.

- Users with "View My Team's Dashboards" can view the dashboard as any user below them in the role hierarchy.

- Users with "View All Data" can edit the dashboard and view it as any user in their organization.

> 📝 **Note:** If you have "View All Data" or "View My Team's Dashboards," you can preview the dashboard *edit* page as a different user, but you must select `Let authorized users change running user` to change the running user from the dashboard *view* page.

5. Click **OK**.

6. In the `View dashboard as` field, enter a running user.

7. Save your dashboard.

[Provide Individualized Views of a Dashboard in Salesforce Classic with Dynamic Dashboards](#)

Dynamic dashboards enable each user to see the data they have access to. With a dynamic dashboard, you can control data visibility without having to create a separate dashboard, with its own running user and folder, for each level of data access.

[Set Up Dynamic Dashboards in Salesforce Classic](#)

Create a folder to hold the dashboard and its underlying reports, then create the dashboard.

Provide Individualized Views of a Dashboard in Salesforce Classic with Dynamic Dashboards

Dynamic dashboards enable each user to see the data they have access to. With a dynamic dashboard, you can control data visibility without having to create a separate dashboard, with its own running user and folder, for each level of data access.

Available in: Salesforce Classic

Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

USER PERMISSIONS
To create, edit, and delete dynamic dashboards:
• **Legacy Folder Sharing** Manage Dynamic Dashboards
Enhanced Folder Sharing Manage Dynamic Dashboards

Administrators control access to dashboards by storing them in folders with certain visibility settings. Dashboard folders can be public, hidden, or restricted to groups, roles, or territories. If you have access to a folder, you can view its dashboards.

Folders control access to the dashboard, but the *running user* determines access to data. The running user options are:

- **Run as specified user.** The dashboard runs using the security settings of that single, specific user. All users with access to the dashboard see the same data, regardless of their own personal security settings. This approach is perfect for sharing the big picture across a hierarchy or motivating team members by showing peer performance within a team. If you don't have "View All Data," you can only choose yourself. If you have "View My Team's Dashboards," you can choose any user below you in the role hierarchy.

- **Run as logged-in user.** A *dynamic dashboard* runs using the security settings of the user viewing the dashboard. Each user sees the dashboard according to his or her own access level. This approach helps administrators share one common set of dashboard components to users with different levels of access.

With a dynamic dashboard, you can control data visibility without having to create a separate dashboard, with its own running user and folder, for each level of data access. A single dynamic dashboard can display a standard set of metrics across all levels of your organization.

You can create up to three filters for each dynamic dashboard. Filtering dynamic dashboards gives administrators additional flexibility in creating dashboards. For example, you can create an organization-wide sales scorecard that contains sales rep and product filters. This allows individual sales managers to view their reps' performance collectively as well as individually. It also lets them view sales by product to understand which products specific reps are or aren't selling.

Managers with the "View My Team's Dashboards" or "View All Data" permission can set an option to preview the dashboard from the point of view of users under them in the role hierarchy.

Your organization can have up to 5 dynamic dashboards for Enterprise Edition, 10 for Unlimited and Performance Edition, and 3 for Developer Edition.Additional dynamic dashboards may be available for purchase.

> **Note:**
> - Dynamic dashboards don't support following components.
> - You can't save dynamic dashboards to personal folders.
> - You can't schedule refreshes for dynamic dashboards. They must be refreshed manually.

Example Business Scenario

Let's say that your opportunity team consists of one vice president, four sales managers, and 40 sales reps—10 reps per manager. You've been asked to create dashboards that display the following metrics, restricted by role and hierarchy:

Role	Total Bookings	Close Rates by Competitor	Number of Activities by Meeting Type
Sales Rep	✅		✅
Sales Manager	✅	✅	
VP of Sales	✅	✅	

Sales reps should only see their own data; managers should only see data for the reps they manage; and the VP should see data across the entire team. In this scenario, you'd typically have to create *45* different dashboards—one for every single person. You'd also have to create multiple folders to manage access rights.

With dynamic dashboards, you can create just *two* dashboards and store them in a single folder:

- Create a dynamic dashboard for sales reps with the following components:
  - A gauge of total bookings
  - A table of activities by meeting type

- Create a dynamic dashboard for managers and the VP with the following components:
  - A gauge of total bookings
  - A column chart of close rates by competitor

- Optionally, create filters that let viewers further refine their dashboard views. For example, create a filter on key accounts to let viewers focus on bookings, activities, and competitive threats for each account.

All users only see data that they can access. Sales reps see their own bookings and activities. Managers see bookings and close rates for the reps they manage. The VP sees bookings and close rates for the whole team. Because the metrics are the same for managers and the VP, you can use the same dynamic dashboard for both roles. The dynamic dashboards feature reduces the number of required dashboards from 45 to two!

SEE ALSO:

Filter a Dashboard

Build a Salesforce Classic Dashboard

Set Up Dynamic Dashboards in Salesforce Classic

Set Up Dynamic Dashboards in Salesforce Classic

Create a folder to hold the dashboard and its underlying reports, then create the dashboard.

Available in: Salesforce Classic
Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

Your organization can have up to 5 dynamic dashboards for Enterprise Edition, 10 for Unlimited and Performance Edition, and 3 for Developer Edition. Additional dynamic dashboards may be available for purchase.

Take note of these dynamic dashboard limitations:

- Dynamic dashboards don't support following components.
- You can't save dynamic dashboards in private folders.
- You can't schedule refreshes for dynamic dashboards. They must be refreshed manually.

1. Create folders accessible to all dashboard viewers to store dynamic dashboards and corresponding component source reports.

2. From the Dashboards tab, create a new dashboard or edit an existing one.

3. Click the ▼ button next to the `View dashboard as` field.

   📝 Note: If you don't have "Manage Dynamic Dashboards" permission, just enter a running user and skip to the final step. Enter "*" to see all available users.

4. Select `Run as logged-in user`.

5. Optionally, select `Let authorized users change running user` to enable those with permission to change the running user on the dashboard view page.

   - Users with "View My Team's Dashboards" can view the dashboard as any user below them in the role hierarchy.
   - Users with "View All Data" can edit the dashboard and view it as any user in their organization.

6. Click **OK**.

7. In the `View dashboard as` field, enter a running user.

8. Save your dashboard.

9. Set the appropriate `Show` option on the report run page.
   For example, if you choose "My Team's Opportunities," each dynamic dashboard viewer can see all opportunities for the team.

   💡 Tip: To avoid restricting the dashboard's view of the data:

   - Make sure advanced filters don't include specific record owners (for example, `Opportunity Owner equals Frank Smith`).

- Don't click `Save Hierarchy Level` when saving opportunity reports.

SEE ALSO:

Add a Dashboard Filter

Provide Individualized Views of a Dashboard in Salesforce Classic with Dynamic Dashboards

Choose a Running User in Salesforce Classic

Subscribe to or Schedule Dashboards

Set up automatic dashboard refreshes and receive refreshed dashboard results by email on a schedule that you set. If you start your week by reviewing your Sales Overview dashboard, expedite your morning by subscribing to the dashboard. After subscribing, you can have it automatically refreshed and in your email inbox each Monday morning at 8:00 AM.

▶ Watch a video: ◉ Subscribe to Reports and Dashboards (Lightning Experience)

Subscribe to Dashboards in Lightning Experience

Subscribe to dashboards to refresh them on a schedule that you set (daily, weekly, or monthly), and receive the refreshed dashboards results by email. You can subscribe to dashboards yourself and also add other users, groups, and roles.

Schedule a Dashboard Refresh in Salesforce Classic

Schedule dashboards to refresh them on a schedule that you set (daily, weekly, or monthly), and receive refreshed dashboards results by email.

EDITIONS
Available in: both Salesforce Classic and Lightning Experience
Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Subscribe to Dashboards in Lightning Experience

Subscribe to dashboards to refresh them on a schedule that you set (daily, weekly, or monthly), and receive the refreshed dashboards results by email. You can subscribe to dashboards yourself and also add other users, groups, and roles.

▶ Watch a video: ⊙ Subscribe to Reports and Dashboards (Lightning Experience)

For example, suppose that every Monday morning you arrive at your desk and run your case overview dashboard to determine case priority. Instead of manually refreshing the dashboard, subscribe to it and have it emailed to you and anyone else you specify every Monday morning at 8:00am.

1. To set up dashboard subscriptions, do either of the following on the Dashboards page:

   - Open the dashboard, and then click **Subscribe**.

   - Find the dashboard you want to subscribe to and select the **Subscribe** row level action. (If you have customized columns on the page, you might have to scroll to the right to see the row-level action arrow ▼ .)

2. In the **Edit Subscription** window, set the subscription schedule.

   For example, to review a dashboard at the start of business every week, set the schedule to receive the dashboard weekly on Monday at 8:00 AM.

3. Under `Send To`, you are automatically selected as a recipient. To add others or remove yourself, click **Edit Recipients**.

4. Select from the available entity types and start typing to see all the matching names.

   Only users, roles, and groups who have permission to access the dashboard are shown in the list of matches.

5. Select from the matching options and click **Add** to add to the list of subscribers. Add recipients as needed, and then close the **Edit Recipients** window.

   When the subscription emails the refreshed report to each recipient, it sends to the email address set in **Settings** > **Email** > **My Email Settings**. If no email is set in My Email Settings, then the refreshed report sends to the recipient's Salesforce User record email address.

   🛑 Important: Recipients see emailed report data as the person running the report. Consider that they can see more or less data than they normally see in Salesforce.

6. Click **Save** in the settings window.

The dashboard subscription starts. You and any other designated subscribers begin receiving refreshed dashboard results by email according to the schedule you set. On the Dashboard page, the **Subscribe** column has a checkmark for each dashboard that has subscriptions.

To unsubscribe, open the Edit Subscription window by clicking **Subscribe** again. Then, click **Unsubscribe**.

📝 Note: The subscription recipients aren't listed on the dashboard subscription emails.

As you subscribe to dashboards, take note of these notes, considerations, and limitations:

- Each user can set up subscriptions for up to 5 dashboards.

- You can subscribe to filtered dashboards, but dashboard filters are never applied to emailed dashboards. When you open the email, the dashboard is displayed unfiltered.

- Dashboard subscription emails don't reflect changes made to color palette and theme.

- Dashboards that are configured to display data as **The dashboard viewer** under **View Dashboard As** settings in **Dashboard Properties** don't support subscriptions. You can't subscribe to them.

- In the email to subscribers, the 'From' address is taken from My Email Settings. If no address is specified in My Email settings, the 'From' address is taken from the User object.

- For funnel charts, the total value is not included in the subscription email.

- Each subscription supports up to 500 recipients. Each recipient is a single user, role, role and subordinates, or group. Roles, roles and subordinates, and groups can each have more than 500 users, but subscriptions send a maximum of 500 emails. If a recipient role, role and subordinates, or group has lots of users, some of them don't receive subscription emails.

  After including all users from roles, roles and subordinates, and groups, if subscriptions have more than 500 users as recipients, users are prioritized over roles, roles are prioritized over roles and subordinates, and roles and subordinates are prioritized over groups. Each time the subscription sends an email, the role and group users who receive the email are chosen again and can be different each time a subscription email sends.

  For example, say that a subscription has 100 recipients: 98 users, 1 role which includes 500 users, and one group which includes 400 users. The total number of users associated with the subscription is 998, so when the subscription email sends, 98 users from the role don't receive subscription emails and none of the 400 users in the group receive emails.

- Each Salesforce organization can schedule up to 500 dashboard subscriptions and 500 report subscriptions on a given hour of a given day, such as Monday at 9:00am.

Schedule a Dashboard Refresh in Salesforce Classic

Schedule dashboards to refresh them on a schedule that you set (daily, weekly, or monthly), and receive refreshed dashboards results by email.

> **Note:**
> - Users with the "View Setup and Configuration" permission can view all the dashboards scheduled to refresh for your organization on the All Scheduled Jobs page. To view the All Scheduled Jobs page, from Setup, enter `Scheduled Jobs` in the `Quick Find` box, then select **Scheduled Jobs**. Users with "Modify all Data" permission can click **Del** next to a specific scheduled dashboard refresh to permanently delete all instances of the scheduled refresh.

You can also set up Salesforce to send an email with an HTML version of the dashboard when the refresh completes. For email applications that don't support HTML, the email includes text and a link to the dashboard.

1. On the Dashboards tab, select a dashboard using the `View Dashboard` field.

2. Click **Refresh** and choose **Schedule Refresh**

   Filters aren't applied when you schedule or email a dashboard.

   Each dashboard has a *running user*, whose security settings determine which data to display in a dashboard.

   > **Note:** If the running user becomes inactive, the dashboard doesn't run.

3. Select notification settings.
   - Click `To me` to send an email to your user's address.
   - Click `To others...` to send an email to additional Salesforce users.

     > **Note:** Portal users receive report and dashboard refresh email notifications when the `Allow Reports and Dashboards to Be Sent to Portal Users` option is enabled.

   - Dashboard refresh notifications may not display properly in Outlook 2007.
   - In HTML-formatted dashboard refresh notifications, users can click the name of the dashboard to log in to Salesforce and view the dashboard.
   - To send a dashboard refresh notification to other users, store the dashboard in a public folder with access granted to others. Other users can't access dashboards in your personal folders. To add a dashboard to a public folder, edit the dashboard properties.
   - Users can click components in a dashboard refresh notification to view the source report in Salesforce.
   - Dashboard components that include Visualforce pages and s-controls may not display in dashboard refresh notifications. Users must view them in Salesforce.
   - Dashboard refresh notifications can be viewed offline in email clients.
   - If a dashboard has filters, only the unfiltered version is emailed.
   - By default, Salesforce sends images in dashboard emails as `.png` (Portable Network Graphic) files, which are not supported in Lotus Notes. When you enable the `Use Images Compatible with Lotus Notes in Dashboard Emails` > option, Salesforce uses `.jpg` images, which Lotus Notes supports, when sending dashboard emails. The "Schedule Dashboard" permission is required to view this option.

EDITIONS

Available in: Salesforce Classic (not available in all orgs)

Available in: **Enterprise**, **Performance**, and **Unlimited** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

USER PERMISSIONS

To schedule and email a dashboard refresh:
- **Legacy Folder Sharing**
  Schedule Dashboards

  Enhanced Folder Sharing
  Schedule Dashboards

To delete the schedule to refresh a dashboard:
- **Legacy Folder Sharing**
  Modify All Data

  Enhanced Folder Sharing
  Modify All Data

> 📝 **Note:** Dashboard emails that contain images compatible with Lotus Notes are substantially larger and the image quality can be lower.

4. Schedule the refresh.

    a. Set the `Frequency` field.

       Click the `Daily`, `Weekly`, or `Monthly` fields to show more options.

       The total number and frequency of your scheduled dashboard refreshes depends on your Salesforce edition. Enterprise, Unlimited, and Performance Editions can have up to 200 scheduled dashboard refreshes. Unlimited and Performance Edition users can schedule up to two dashboard refreshes an hour per day. Enterprise Edition users can schedule up to one dashboard refresh an hour per day. Additional scheduled dashboards may be available for purchase. Contact your Salesforce representative for information.

    b. Specify dates in the `Start` and `End` fields.

       Dashboards refresh in the time zone of the user who scheduled the refresh. For example, if the `Time Zone` field on your user record is set to Pacific Standard Time (PST), and you schedule a dashboard to refresh every day at 2:00 PM, then the dashboard will refresh every day between 2:00 PM and 2:29 PM PST.If you view and save a schedule in a time zone different from the one in which it was previously scheduled, the time slot could potentially change.

    c. Under `Preferred Start Time`, click **Find available options...** to choose a time.

       The dashboard refresh runs within 30 minutes of your preferred start time. For example, if you select 2:00 PM, the refresh may happen any time between 2:00 PM and 2:29 PM, depending on availability.

> 📝 **Note:** Your preferred start time may not be available if other users have already selected it.

- If you schedule a dashboard to refresh on a specific day of every month, it only refreshes on months that have that specific day. For example, if you schedule a refresh for the 31st of every month, the dashboard won't refresh on 30-day months. To refresh on the last day of every month, choose "Last" from the `On day of every month` drop-down list.
- Dashboards won't refresh as scheduled if the running user doesn't have access to the dashboard folder.
- If a dashboard has filters, only the unfiltered version is refreshed.
- You can't schedule refreshes for dynamic dashboards. They must be refreshed manually.

5. Click **Save**.

To delete a scheduled dashboard refresh, click **Refresh** > **Schedule Refresh** > **Unschedule Dashboard**.

The scheduled refresh is permanently deleted, and not sent to the Recycle Bin. Deleting the scheduled refresh does not affect the dashboard itself.

SEE ALSO:

    Apply a Dashboard Filter

    Refresh Dashboard Data

Link from Dashboard Components in Lightning Experience

A sales overview dashboard is a great way for teams to collaborate on opportunities because it provides a complete picture of your pipeline. Dashboard components already link to source reports, so you can get more details, but what if you're ready to take action?

1. Link a Dashboard Component to a Website or Salesforce Record

   After giving dashboard readers insights about their business, drive them to take the next step by providing one-click access to Salesforce records or websites.

2. Visit a Link from a Dashboard Component

   Dashboard builders can link dashboard components to Salesforce records or other websites. Visit a link to start taking action on dashboard insights!

Link a Dashboard Component to a Website or Salesforce Record

After giving dashboard readers insights about their business, drive them to take the next step by providing one-click access to Salesforce records or websites.

User Permissions Needed

To link from dashboard components:	Create and Customize Dashboards
To link from dashboard components in Dynamic Dashboards:	Manage Dynamic Dashboards
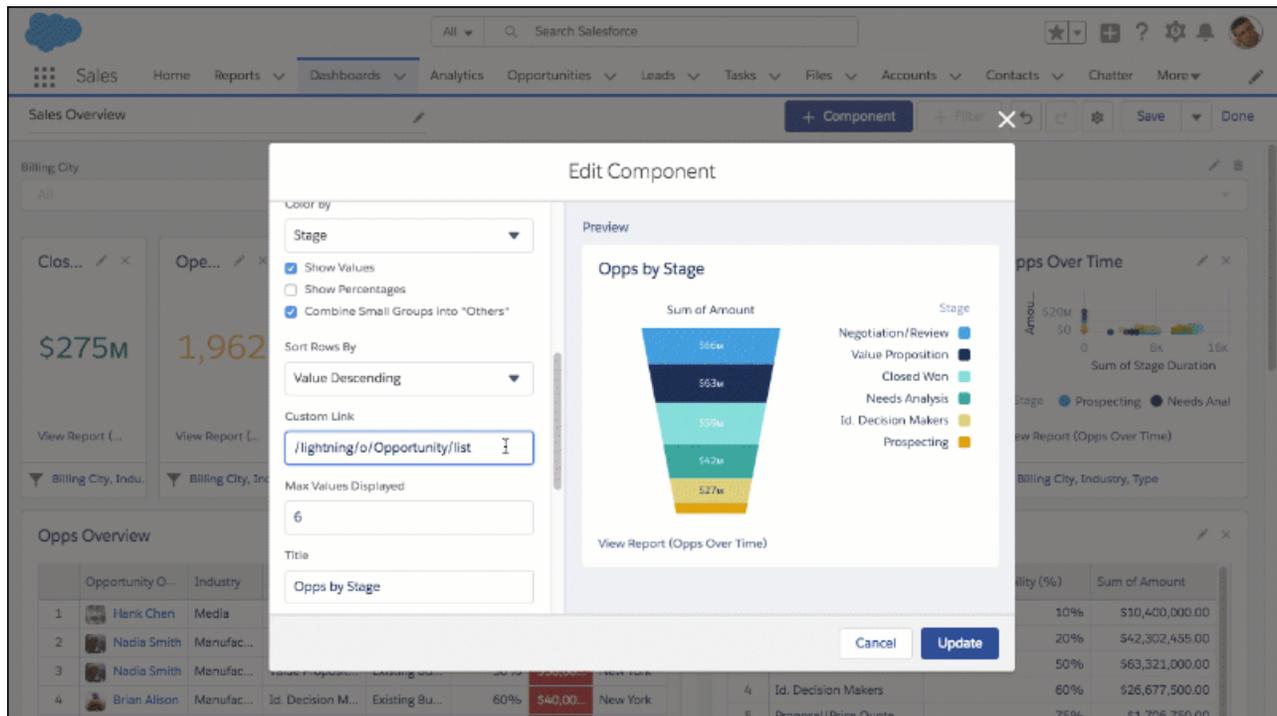
Linking from dashboard components helps readers turn insights into actions.

1. Add a new dashboard component by clicking **+ Component**, or edit an existing one by clicking the pencil icon (✐) on the component.

2. From the Add Component or Edit Component menu, in **Custom Link**, enter a link destination.

> 🛑 Important: Depending on whether you link to a destination in Salesforce or to another website, the destination you enter in **Custom Link** must follow certain conventions. Unless you append `https://` or `http://` or `www.`, the destination you enter gets pasted to the end of your Salesforce org's domain (for example, `https://`**`MyDomainName`**`.my.salesforce.com/`). If you enter `abc`, then your component links to `https://`**`MyDomainName`**`.my.salesforce.com/abc`. Appending `https://` or `http://` or `www.` lets Salesforce recognize that you intend to link to a specific website.
>
> This means that linking to `heroku.com` won't take you to the Heroku website. Instead, it'll navigate you to `https://`**`MyDomainName`**`.my.salesforce.com/heroku.com`. To navigate to the Heroku website, enter `https://www.heroku.com/`.

> **Example: Link Somewhere Inside Salesforce**
> To link to the Opportunities tab, enter `/lightning/o/Opportunity/list`.

> **Example: Link to Another Website**
> To link to Heroku dot com, as described in the prior note, enter `https://www.heroku.com/`.

3. If adding a new component, click **Add**. Or, if editing an existing component, click **Update**.

4. Click **Save**.

After dashboard viewers expand your linked component, they can visit the link by clicking 🔗.

Visit a Link from a Dashboard Component

Dashboard builders can link dashboard components to Salesforce records or other websites. Visit a link to start taking action on dashboard insights!

User Permissions Needed

To visit a dashboard component link:	Run Reports

1. Open a dashboard.

2. Expand a component. Hover over a component and click ⛶.
   A larger version of the dashboard component overlays the dashboard.

3. Visit the link. From the expanded component, click 🔗.

The linked component's destination opens.

Work with Dashboards

Click on a dashboard's name to run it. Dashboard charts are interactive, so be sure to hover and click on them to get more info! From refreshing dashboard data to sharing dashboard components, there are lots of ways to use dashboards to keep yourself and your team informed and collaborating.

📝 **Note:** Dashboards in **Group** Edition organizations are view-only.

1. Refresh Dashboard Data

   Click **Refresh** to load the latest data into the dashboard. The data is as current as the date and time displayed after `As of...` at the top right corner of the dashboard.

2. Expand Dashboard Components to See a Larger Version in Lightning Experience

   Sometimes a chart or table is just a little too small. Open it in a window so you can see all the details.

3. Share an Image of a Dashboard Component on Chatter in Lightning Experience

   Start the conversation by sharing images of dashboard components on Chatter.

4. View Filtered Source Reports in Lightning Experience

   Want to know more about the data shown in a dashboard chart? Click a chart segment, axis label, or legend to view a filtered version of the source report.

5. Resize Columns in Lightning Tables

   For those times when a Lightning table column feels a bit cramped, go ahead and open it up by resizing it.

Refresh Dashboard Data

Click **Refresh** to load the latest data into the dashboard. The data is as current as the date and time displayed after `As of...` at the top right corner of the dashboard.

1. Click **Refresh** to replenish your dashboard with the most recent data.
   When dashboard data is being refreshed, the **Refresh** button changes to **Refreshing...**You can leave the dashboard and do other things in Salesforce while the data refreshes. When the dashboard has finished refreshing, Salesforce 1 tells you that the dashboard is ready through the right-hand notification panel. Since push notifications aren't enabled for dashboard refreshes, you're notified only in the app and not in through email.

   💡 **Tip:** If your dashboard data doesn't refresh after ten minutes, the refresh automatically stops. If necessary, click **Refresh** again.

When you refresh a dashboard, the dashboard data refreshes for anyone else in your organization that has access to that dashboard. Additional refreshes submitted during a refresh and up to one minute after the completion of a refresh are ignored; users view the most current data.

When you change filters on a dashboard, the dashboard shows previously cached data, if it exists. If no data exists, the dashboard fetches the latest. Either way, the dashboard's "Last Refreshed" date shows you the timestamp for the data you're viewing. Click **Refresh** to get the most recent data.

Expand Dashboard Components to See a Larger Version in Lightning Experience

Sometimes a chart or table is just a little too small. Open it in a window so you can see all the details.

1. Open a dashboard.

2. Hover over a component and click ⛶.

A larger version of the dashboard component overlays the dashboard. Optionally, click the left and right arrows to cycle through expanded versions of the dashboard's components.

Share an Image of a Dashboard Component on Chatter in Lightning Experience

Start the conversation by sharing images of dashboard components on Chatter.

There are many reasons to share images of dashboard components. Request help with aging cases, ask a question about the pipeline, or congratulate the team for making quota!

1. View a dashboard.

2. Find the component you want to share, and expand it by clicking ⛶.

3. From the expanded component, click 📷 .

   The Chatter menu opens. From the Chatter menu, you can compose your post and view the dashboard feed by scrolling down.

4. Compose your post. You can @mention people and groups, and format with rich text. Go ahead and make your post as **bold** as you like!

5. Click **Share**.

The dashboard component image posts to the Dashboard feed. To view the dashboard feed, and your post, click 📷 .

📝 Note: If you can't post images of dashboard components on Chatter, ask your Salesforce admin to turn on feed tracking for dashboards. For more information, see Customize Chatter Feed Tracking in Salesforce Help.

Downloaded and shared images of dashboard component tables have a maximum height of 3000 pixels or approximately 100 rows. Additional rows beyond the limit are clipped. To avoid clipping, filter the chart to fewer than 100 rows.

This topic is about sharing images of dashboard components in Lightning Experience. For information about sharing images of dashboard components in Salesforce Classic, see Post Snapshots of Dashboard Components to Chatter in Salesforce Help.

View Filtered Source Reports in Lightning Experience

Want to know more about the data shown in a dashboard chart? Click a chart segment, axis label, or legend to view a filtered version of the source report.
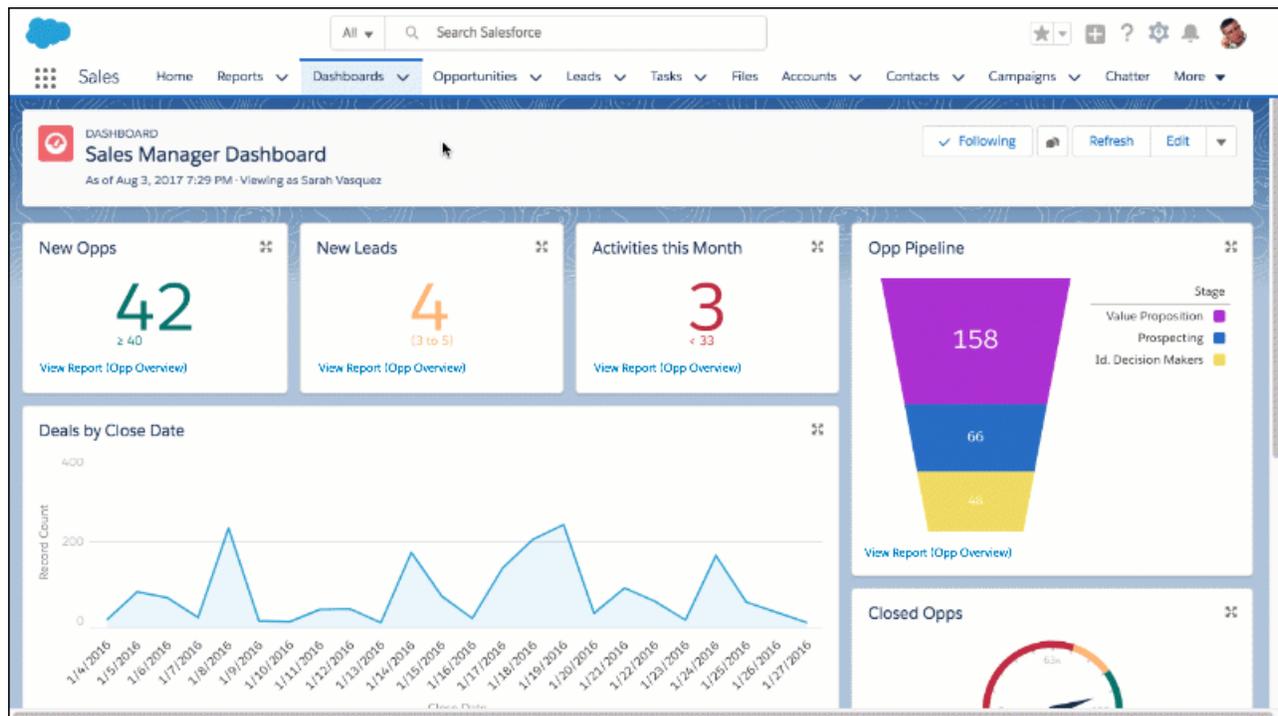
For example, click the Prospecting segment of an opportunity pipeline funnel chart to open and filter the source report by Stage equals Prospecting.

1. Open a dashboard.

2. Click a chart segment, axis label, or legend.

The source report opens. The segment you click (plus any applied dashboard filters) appear as linked filters when the report opens.

> **Note:** Linked filters don't persist after you leave the report and you can't edit them directly.

To open the filtered source report in a new tab, hold down **CTRL** on Windows® or **Command** on MacOS® and click a chart segment.

If a chart displays one of these types of data, then the source report won't filter:

- Encrypted strings
- Dates fields with month-in-year or day-in-month granularity

Resize Columns in Lightning Tables

USER PERMISSIONS		EDITIONS
To create dashboards:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	Available in: Lightning Experience
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards	Available in: **Essentials**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions
To edit and delete dashboards you created:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	Available in: Enhanced Folder Sharing
	Enhanced Folder Sharing Run Reports AND Create and Customize Dashboards	
To edit and delete dashboards you created in public folders:	**Legacy Folder Sharing** Run Reports AND Manage Dashboards	
	Enhanced Folder Sharing Edit My Dashboards	
To edit and delete dashboards you didn't create in public folders:	**Legacy Folder Sharing** Run Reports, Manage Dashboards AND View All Data	
	Enhanced Folder Sharing Manage Dashboards in Public Folders	
To create, edit, and delete dynamic dashboards:	**Legacy Folder Sharing** Manage Dynamic Dashboards	
	Enhanced Folder Sharing Manage Dynamic Dashboards	

For those times when a Lightning table column feels a bit cramped, go ahead and open it up by resizing it.

Resize a Lightning table column from the Edit Component preview or while reading a dashboard.

1. View the table you want to resize.

2. Click-and-drag the edge of the column you'd like to
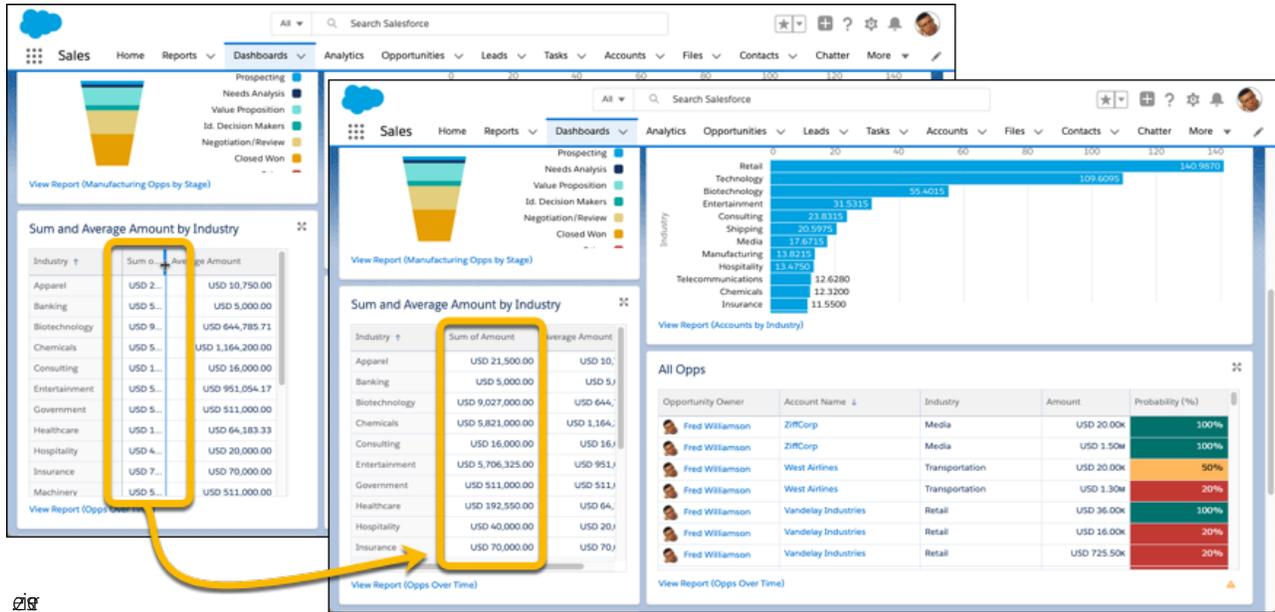
You can resize columns, but not rows. When you resize a column, you don't resize it for other dashboard viewers. Columns don't retain their sizes, so next time you open the dashboard you'll have to resize the column again.

Organize Dashboards

Keep your dashboards at your fingertips by embedding them around Salesforce, printing them, or deleting dashboards that you don't need anymore.

1. Embed Dashboards on the Home Tab and in Lightning Apps

   Embed dashboards to surface insights right where people work.

2. Print a Dashboard

   Print dashboards using your browser's print option.

3. Delete a Dashboard

   It's a good idea to delete dashboards that you no longer need.

4. Install the CRM Sample Dashboards from AppExchange

   Use these dashboards from the AppExchange as a starting point for building dashboards that meet your business needs.

Embed Dashboards on the Home Tab and in Lightning Apps

Embed dashboards to surface insights right where people work.

For example, add the Sales Pipeline dashboard to the Home tab to give your sales team the information they need when they log in to Salesforce.

Embedded dashboards are fully interactive. Refresh them, apply filters, and click chart segments to drill into filtered reports.

Dashboards need space to display charts and tables. If an embedded dashboard is squished into too small a space, then a collapsed version displays instead of the full dashboard. The collapsed version links back to the full dashboard.
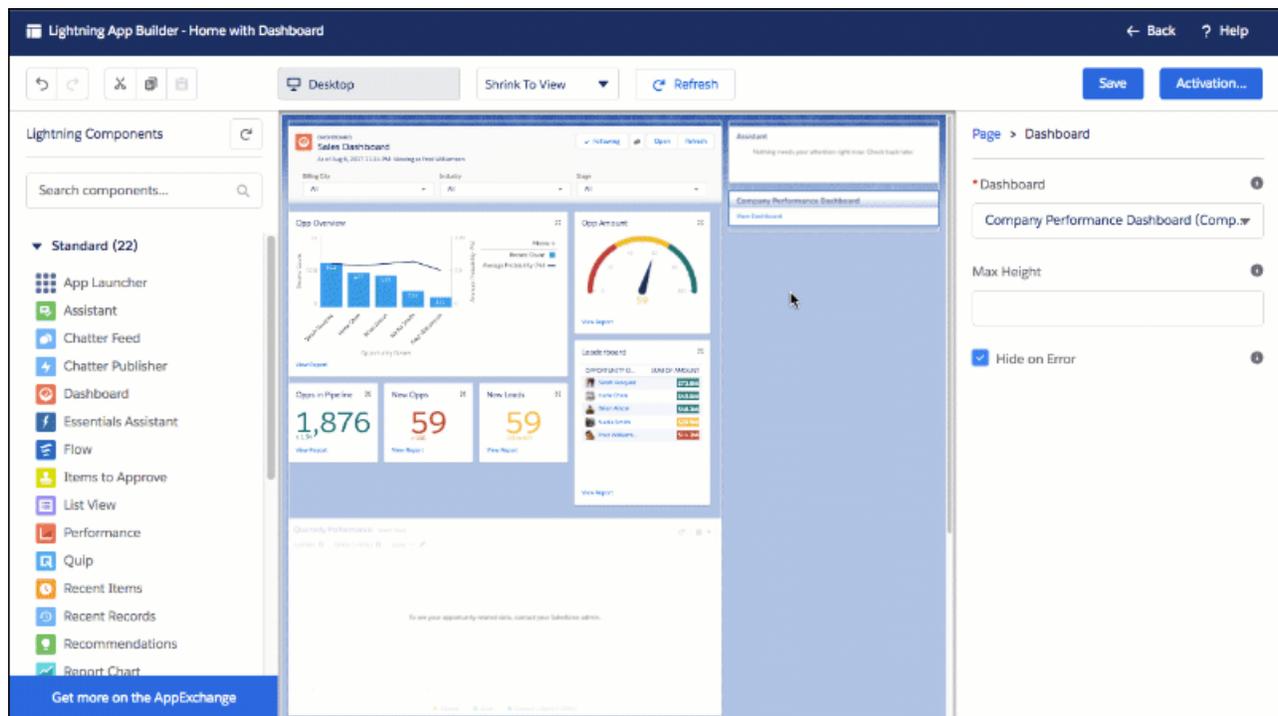
1. From Setup, enter `App Builder` in the `Quick Find` box, then select **Lightning App Builder**.

2. Click **New**.

3. Choose where to embed the dashboard. (Record pages don't support embedded dashboards.)

- To embed a dashboard on an app page, select **App Page**.
- To embed a dashboard on the Home tab, select **Home Page**.

Click **Next**

4. Give your app page or home tab layout a label. Then, click **Next**.

5. Choose a layout. Then, click **Finish**.

6. Drag and drop the **Dashboard** standard component into place.

7. From the Dashboard drop-down list, choose a dashboard to embed.

8. Optionally, specify a maximum height and choose to show or hide the dashboard if an error prevents it from loading.

9. Click **Save.**

10. Click **Activate**.

Go visit the newly updated Home tab, or your Lightning app, and marvel at the embedded dashboard!

Print a Dashboard

Print dashboards using your browser's print option.

- Set the paper orientation to print in landscape format so that it is wide enough for all three columns of dashboard components.
- If necessary, resize your columns.

EDITIONS

Available in: both Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Essentials**, **Group** (View Only), **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: both Legacy Folder Sharing and Enhanced Folder Sharing

USER PERMISSIONS

To print dashboards:
- Run Reports AND access to dashboard folder

Delete a Dashboard

It's a good idea to delete dashboards that you no longer need.

Deleting a dashboard also deletes the components within it. It doesn't delete the custom reports used by the components.

Deleted dashboards are moved to the Recycle Bin.

To delete a dashboard in Lightning Experience, open the dashboard and then click **Delete**.

To delete a dashboard in Salesforce Classic:

1. Click the Dashboards tab.

2. Click **Go To Dashboards List**.

3. Choose the folder where the dashboard is stored.

4. Click **Del** next to the name of the dashboard.

Install the CRM Sample Dashboards from AppExchange

Use these dashboards from the AppExchange as a starting point for building dashboards that meet your business needs.

The CRM sample dashboards package from AppExchange offers best-practice dashboards with underlying reports that are based on your organization's standard objects and fields. You can use these sales, marketing, service, and support dashboards to track business processes and key performance metrics for yourself, your team, and your company. Dashboards in this package include:

- **Sales & Marketing Dashboards**

    - Marketing Executive Dashboard

    - Sales Executive Dashboard

    - Sales Manager Dashboard

    - Salesperson Dashboard

- **Customer Service Dashboards**

    - Agent Supervisor Overview Dashboard

    - Service Executive Overview Dashboard

    - Service KPIs Dashboard

1. In AppExchange, search for "Salesforce CRM Dashboards" and click **Get It Now**.

2. To install the sample dashboards, install the package.

Why Doesn't My Dashboard Display the Data I Expect?

If you're not seeing data you expect, refresh for latest data, check that you have the right running user, and verify your dashboard data sources.

If your dashboard data doesn't look as you expect, check the following:

- **Running user.** Remember that you're viewing the dashboard from the perspective of the running user. What you see is based on that user's access rights. For dynamic dashboards, you can only see what you yourself can access.

- **Refresh Dashboard.** If you are viewing old data, click **Refresh** to refresh your dashboard.

- **Data sources.** Verify that your data sources (reports, s-controls, or Visualforce pages) contain the information you want displayed in the dashboard components.

For further information regarding dashboard running user and dashboard refresh, review the articles Choose a Dashboard Running User on page 404 and Refresh Dashboard Data on page 415.

Improve Dashboard Performance: Best Practices

If a dashboard is running slowly, it likely because of inefficient source reports or because many people are refreshing it at once. By optimizing source reports and planning dashboard refreshes, a slow dashboard can be sped up. Follow the tips in this guide to speed up sluggish dashboards!

Use efficient source reports

When you refresh a dashboard, all the source reports run. If source reports take a long time to run, then dashboards based on them take a long time to refresh.

Here's how you can optimize source report performance to make dashboards refresh fast.

Create source reports that run fast

The best way to speed up a slow dashboard is to speed up its source reports. The faster source reports run, the faster the dashboard refreshes!

There are lots of ways to speed up a slow source report.

- Filter report data efficiently
- Remove unnecessary columns
- Hide detail rows
- Write efficient formulas

For a complete guide to improving report performance, see Improve Report Performance: Best Practices on page 284.

SEE ALSO

- Improve Report Performance: Best Practices

Base your dashboard on as few source reports as possible

Ideally, base your dashboard on a single source report.

Each dashboard component gets data from one source report, but multiple dashboard components can get data from the same source report. Dashboards based on fewer source reports refresh faster than dashboards based on many. The fewer source reports, the fewer reports have to run for the dashboard to gather its data. The fastest refreshing dashboards are based on only one source report.

As a best practice, create one source report that returns lots of data, and then divvy up that data among the dashboard components.

For example, say you're creating a sales overview dashboard that shows details on open opportunities and closed opportunities. You could create two source reports, one that returns open opportunities and one that returns closed opportunities. But then your dashboard would have to run two source reports when refreshed. Instead, create a single source report that returns all opportunities and group by status. Now you can create the dashboard using a single source report, which refreshes more quickly than a dashboard with two source reports.

> **Note:** Source reports refresh asynchronously. That means that when you refresh a dashboard, Salesforce enqueues each source report to run and then runs them alongside any number of other processes.
>
> Depending on its place in the queue, sometimes a dashboard with only one source report loads a little slower than a dashboard with two source reports...Salesforce may have to update a case status, create an account, and send a batch of emails before it can run your dashboard's source report.
>
> And some reports take longer to run than others. A slow source report may take longer to run than two fast source reports, all other things being equal.
>
> Sometimes refreshing a dashboard that has only one source report can result in multiple runs of the same source report. Including dashboard components with different groupings, applying dashboard filters, and changing a dashboard's running

user can all cause the same source report to run multiple times. In these scenarios, having fewer source reports isn't necessarily faster.

But in general, dashboards with fewer source reports refresh more quickly than dashboards with lots of source reports.

SEE ALSO

- Build a Dashboard
- Edit and Customize Lightning Experience Dashboard Components

Refresh dashboards sparingly

Refreshing a dashboard enqueues several processes that Salesforce runs resulting in up-to-date data. Each source report runs, filters are applied, charts are drawn, and so forth. Salesforce limits the number of times each user can refresh a dashboard to once per minute. This limit ensures that Salesforce doesn't enqueue the same report runs repeatedly, which also lets other important processes run, like updating cases or creating accounts.

But what happens if many different people refresh the same dashboard all at the same time?

Salesforce limits the number of dashboard refreshes to 200 per hour per org. Even with this limit slow-running dashboards can cause busier-than-necessary background process queues, forcing other people to wait. What if 200 people refresh a dashboard and the org limit is reached? Because 200 people refreshed the same dashboard, the other team now needs to wait up to an hour to refresh theirs.

Salesforce remembers dashboard filter settings for each user. To apply user-specific filter settings, if someone refreshes a filtered dashboard then the dashboard refreshes anew when anyone who has applied a dashboard filter opens it. Filtered dashboards are therefor especially susceptible to reaching the 200 dashboard refresh per hour org limit.

Imagine a scenario in which 10,000 people refresh a dashboard that enqueues 10 source report runs. Theoretically, 10,000 x 10 = 1,000,000 reports run in the background, but because of the 200 refresh per hour per org limit, 200 x 10 = 2,000 background report runs. Even with quick-running reports, that could take a while!

Instead of having lots of people refresh dashboards at the same time, subscribe to a dashboard. When you subscribe to a dashboard, you set a schedule and set recipients. Salesforce refreshes the dashboard when you schedule it to and then sends the results to the recipients by email. Salesforce caches the data. When people open the dashboard, they see up-to-date data as of the last scheduled refresh and don't need to refresh it again.

1. From the **Dashboards** tab, click 🔽 > **Subscribe**.

2. From the Edit Subscription menu, set a refresh schedule.

3. Under Send To, you are automatically selected as a recipient. To add others or to remove yourself, click **Edit Recipients**. Only people who have permission to access the dashboard are shown.

4. Select recipients and click **Add**.

5. Click **Save**.

The dashboard subscription starts and the dashboard refreshes according to the schedule you set.

For example, suppose that you refresh your opportunity overview dashboard each Monday morning. Instead of manually refreshing the dashboard, subscribe to it and have it emailed to you and the entire sales team each Monday at 8:00 AM (1). By the time you sit down at your desk at 9:00 AM, a recently refreshed dashboard is waiting for you.

Remove unnecessary dashboard components

Just as efficient source reports run faster than inefficient ones, dashboards with fewer components tend to run faster than dashboards with lots.

Combine multiple components into one

Sometimes you can show the data in two or more components in only one. Merging multiple components into one means fewer source reports run, which means the dashboard refreshes more quickly.

For example, say that you have a dashboard that has two bar charts. One shows sum of opportunity amount by type, the other shows average opportunity amount by type. Instead of separating these related metrics in two bar charts, consolidate them on a single chart. The dashboard still graphs both sum and average opportunity amount by type, but because it only has to render one component it loads faster.

Split a dashboard in two

If you have a dashboard with many components that takes a long time to refresh, consider splitting the dashboard in two.

Say that you have one dashboard summarizing both opportunities and cases. Splitting the dashboard in to, one for opportunities and one for cases means faster refreshing dashboards. Using a custom link, you can even link back and forth between dashboards for easy navigation.

Here's how to link from one dashboard to another.

1. From the dashboard builder, edit a component by clicking ✎ .

2. From the Edit Component menu, enter `/lightning/r/Dashboard/dashboard_id/view`, where `dashboard_id` is the id of the destination dashboard.

    To get the destination dashboard id, open the dashboard and copy the id from the URL. The entry for a destination dashboard with the id `01ZRM0000005KO62AM` is `/lightning/r/Dashboard/01ZRM0000005KO62AM/view`.

3. Click **Update**.

4. Click **Save**.

Now dashboard viewers can navigate from one dashboard to another with direct links.

SEE ALSO

- Link from Dashboard Components in Lightning Experience
- Visit a Link from a Dashboard Component

Improve Dashboard Performance

Dashboard refresh times can vary based on available resources. Since Salesforce is built on a multi-tenant structure, dashboard refreshes are processed in a queue. The refresh tend to complete faster off peak hours, when more resources are available. Other factors like caching can also affect dashboard refresh times.

Dashboard performance depends mainly on the performance of its source reports. If a dashboard is refreshing slowly, each report needs to be examined to find the slow running reports.

A performance degradation can also affect the Dashboard performance. These are instance specific and can be checked at status.salesforce.com/status.

Dashboard components that include Visualforce pages can cause performance issues. To identify components that use Visualforce pages, edit them. Components using Visualforce pages say "Visualforce Page" as the chart type.

Report and Dashboard Folders

Use report and dashboard folders to organize your reports and dashboards. To share reports and dashboards, you share the folder, not the report or dashboard itself.

▶ Watch a video: ◉ Manage Report and Dashboard Folders (Lightning Experience)

For several releases we've been phasing out Legacy Folder Sharing in favor of Enhanced Folder Sharing, which has an improved UI and more fine-grained access control. Legacy Folder Sharing is currently available only for orgs created prior to Summer '13. It is due for retirement in Summer '20. For instructions on transitioning to Enhanced Folder Sharing, see Transition Your Org to Enhanced Folder Sharing on page 432.

Notes about folder sharing:

- When you save a report or dashboard, it goes by default into your private reports or dashboards folder. To save to a public public/custom folder, select the folder in the Save dialog box.
- To open a report or dashboard, you need:

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

- – Access to the folder in which the report or dashboard is saved
- – The necessary user permission
- For example, to run a report you need access to the folder in which it's saved and the "Run Reports" user permission.
- The concept of shared/not shared folders is different from the concept of public and private folders:
  - – The reports or dashboards in a private folder are always invisible to everyone except the person who created the folder.
  - – The reports or dashboards in any folder that is not private are visible to everyone who has permission to view the reports or dashboards. The following folder names are different in Salesforce Classic and Lightning Experience, but the folders are the same.
- The following folder names are different in Salesforce Classic and Lightning Experience, but the folders are the same.

Salesforce Classic	Lightning Experience
Unfiled Public Reports	Public Reports
My Personal Custom Reports	Private Reports
My Personal Dashboards	Private Dashboards

Access Levels for Report and Dashboard Folders

Each user, group, or role can have its own level of access to a report and dashboard folder. Viewers can see the data. Editors can determine what data is shown. Managers can control access.

> 💡 Tip: If you're not ready to share a report or dashboard, keep it in a personal folder that only you can access. The My Personal Custom Reports folder and the My Personal Dashboards folder are already set up for you. Create more folders if you need them. When you create a folder, it is accessible only to you and to users with administrative permissions, until you share it.

1. Viewer Access to Report and Dashboard Folders

   Viewer access allows you to view the reports or dashboards in a folder, but does not determine access to folder contents. Permissions set for the reports and dashboards themselves determine your access to the contents. For example, if you have Viewer access to a reports folder and Create and Customize Reports permission to a report in the folder, you can rename or delete the report.

2. Editor Access to Report and Dashboard Folders

   Editor access allows you to view and save the reports or dashboards in a folder, but does not determine access to folder contents. Permissions set for the reports and dashboards themselves determine your access to the folder contents. For example, if you have Editor access to a folder and Run Reports permission to a report in the folder, you can run the report but not make changes.

3. Manager Access to Report and Dashboard Folders

   Manager access allows you to view, share, save, rename, and delete the reports or dashboards in a folder, but does not determine access to folder contents. Permissions set for the reports and dashboards themselves determine your access to the folder contents. For example, if you have Manager access to a folder and Run Reports permission to a report in the folder, you can run the report, but not edit or delete it.

4. Compare Access Levels for Report and Dashboard Folders

   Use this chart for a quick view of what Viewer, Editor, and Manager access enables users to do with enhanced folder sharing for report and dashboard folders.

5. Enhanced Folder Sharing for Reports and Dashboards

   Enhanced folder sharing introduces new user permissions and changes each user's access to existing reports and dashboards.

6. Access to Dashboard Folders

   Permissions control folder access. Users must have certain permissions to access public, hidden, or shared dashboard folders.

7. Access to Report Folders

   Folder access is controlled by permissions. Users must have certain permissions to access public, hidden, or shared report folders.

8. User Permissions for Sharing Reports and Dashboards

   Each level of access to a report or dashboard folder consists of a combination of user permissions. As an administrator, you can further fine-tune users' access to dashboards and reports by assigning or removing one or more permissions.

Viewer Access to Report and Dashboard Folders

Viewer access allows you to view the reports or dashboards in a folder, but does not determine access to folder contents. Permissions set for the reports and dashboards themselves determine your access to the contents. For example, if you have Viewer access to a reports folder and Create and Customize Reports permission to a report in the folder, you can rename or delete the report.

All users have at least Viewer access to report and dashboard folders that have been shared with them. Some users have administrative user permissions that give them greater access. Guest users and Community Portal users can't access report and dashboard folders.

View access is useful in a case such as the following. Samir is a sales rep who likes to start his day by checking his position on the sales leader board, which appears on the Master Sales dashboard. He has to refresh the dashboard to get the latest standings, so he needs to view the data in the underlying reports. But he doesn't want to edit the reports or the dashboard. All he needs is Viewer access to the folder that contains the Master Sales dashboard.

EDITIONS

Available in: Salesforce Classic (not available in all orgs) and Lightning Experience

Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing and Legacy Folder Sharing

Editor Access to Report and Dashboard Folders

Editor access allows you to view and save the reports or dashboards in a folder, but does not determine access to folder contents. Permissions set for the reports and dashboards themselves determine your access to the folder contents. For example, if you have Editor access to a folder and Run Reports permission to a report in the folder, you can run the report but not make changes.

Editor access is useful in a case such as the following. Allison, a sales manager, wants to provide a different sales dashboard for each of the three regional teams. Editor access to the folder that contains the Master Sales dashboard allows her to move the underlying reports into the correct folders and then modify them to show the appropriate data.

📝 **Note:** You can't give Editor access to standard report folders. By default, all users get Viewer access to these folders.

Manager Access to Report and Dashboard Folders

Manager access allows you to view, share, save, rename, and delete the reports or dashboards in a folder, but does not determine access to folder contents. Permissions set for the reports and dashboards themselves determine your access to the folder contents. For example, if you have Manager access to a folder and Run Reports permission to a report in the folder, you can run the report, but not edit or delete it.

Manager access is useful in a case such as the following. Alan is a sales administrator who manages too many reports to pay attention to them all individually. He creates a report folder called Regional Reports. As the creator, he has Manage rights to the folder. He gives Sales Reps, a public group, Viewer access. And he makes Allison, the sales manager, another Manager on the folder.

📝 **Note:** You can't give Manager access to standard report folders. By default, all users get Viewer access to these folders.

Compare Access Levels for Report and Dashboard Folders

Use this chart for a quick view of what Viewer, Editor, and Manager access enables users to do with enhanced folder sharing for report and dashboard folders.

	View	Edit	Manage
View reports or dashboards in the folder	✅	✅	✅
See who has what level of access to the folder	✅	✅	✅
Save a report or dashboard in the folder		✅	✅
Rename a report or dashboard in the folder		✅	✅
Delete a report or dashboard from the folder		✅	✅
Share a folder			✅
Change the folder's name			✅
Change the folder's sharing settings			✅
Delete a folder			✅

Working with reports and dashboards requires user permissions in addition to folder permissions. For information about required user permissions, see User Permissions for Sharing Reports and Dashboards .

SEE ALSO:

Enhanced Folder Sharing for Reports and Dashboards

Enhanced Folder Sharing for Reports and Dashboards

Enhanced folder sharing introduces new user permissions and changes each user's access to existing reports and dashboards.

What's Different in Enhanced Folder Sharing?

Enhanced folder sharing is different from legacy folder sharing in the way you define folders and how access to reports and dashboards is determined. And some of the permissions for report and dashboard folders change in enhanced folder sharing.

Creating a Folder

It's simpler to create a folder in enhanced folder sharing.

Legacy Folder Sharing	Enhanced Folder Sharing
Enhanced folder sharing When creating a folder, you specify: • Folder label: name • Public folder access: read-only or read-write • Folder visibility: accessible by all users, hidden from all users, or accessible to specified groups and roles.	When creating a folder, you specify only the name. The folder doesn't carry any special properties for visibility or type of access.

Accessing the Reports and Dashboards in a Folder

Three access levels in enhanced folder sharing determine how you can interact with the reports and dashboards in a shared folder.

Legacy Folder Sharing	Enhanced Folder Sharing
When creating a folder, you specify the folder as read-only or read-write and assign it one of these access types: • Accessible by all users • Hidden from all users • Accessible only to specified public groups, roles, and roles and subordinates	When sharing a folder, on page 445 you specify View, Edit, or Manage access for the selected users, groups, or roles, or a combination. Depending on user permissions, some users can do more than is indicated by the access level that they're granted when a folder is shared. See Permissions for Reports and Dashboards on page 433. The following table lists the capabilities for each access level.

Folder Access Capabilities in Enhanced Folder Sharing

	View	Edit	Manage
View reports or dashboards in the folder	✅	✅	✅
See who has what level of access to the folder	✅	✅	✅
Save a report or dashboard in the folder		✅	✅
Rename a report or dashboard in the folder		✅	✅
Delete a report or dashboard from the folder		✅	✅
Share a folder			✅
Change the folder's name			✅
Change the folder's sharing settings			✅
Delete a folder			✅

Permissions for Reports and Dashboards

Some user permissions are different in enhanced folder sharing. When enhanced folder sharing is enabled, Salesforce updates the permissions as the following table shows. For a full list of permissions for reports and dashboards, see User Permissions for Sharing Reports and Dashboards on page 439.

Legacy Folder Sharing	Enhanced Folder Sharing	What you can do with this permission in enhanced folder sharing
Manage Public Reports AND View All Data	Manage Reports In Public Folders	Perform all Manage actions in Folder Access Capabilities in Enhanced Folder Sharing on page 433 for all folders except others' private folders.

		Manage Reports in Public Folders also includes these permissions: • Create and Customize Reports • Create Report Folders • Edit My Reports • View Reports in Public Folders
Manage Public Reports	Create Report Folder AND Edit My Reports	Create report folders. Perform all Manage actions in Folder Access Capabilities in Enhanced Folder Sharing on page 433 for all folders that you create. Edit, move, save, and delete reports that you create. Save your reports to a shared folder, even if you only have View access to the folder.
Manage Dashboards AND View All Data	Manage Dashboards In Public Folders	Perform all Manage actions in Folder Access Capabilities in Enhanced Folder Sharing on page 433 for all folders except others' private folders. Manage Dashboards in Public Folders also includes these permissions: • Create and Customize Dashboards • Create Dashboards Folders • Edit My Dashboards • View Dashboards in Public Folders
Manage Dashboards	Edit My Dashboards AND Create and Customize Dashboard	Edit, move, save, and delete dashboards that you create. Save your dashboards to a shared folder, even if you only have View access to the folder.
View All Data	View Reports In Public Folders AND View Dashboards In Public Folders	View the reports and dashboards in all folders except the private folders of other users.

Special Folders

The following folders are equivalent in legacy folder sharing and enhanced folder sharing. Only the names are different. Users who have access to the legacy folders continue to have the same access after the upgrade to enhanced folder sharing.

Folder Name in Legacy Folder Sharing	Folder Name in Enhanced Folder Sharing
Unfiled Public Reports	Public Reports
My Personal Custom Reports	Private Reports

My Personal Dashboards	Private Dashboards
Company Dashboards	Company Dashboards

What Happens with Enhanced Folder Sharing Enabled?

When enhanced folder sharing is enabled,

- All users get View access to the report and dashboard folders that are shared with them.

- Users with Manage Reports in Public Folders or Manage Dashboards in Public Folders can manage all dashboard or report folders (except for others' private folders) even if no folder is explicitly shared with them or they have only View access.

- Users who don't get Edit My Reports or Edit My Dashboards permission as part of the upgrade can no longer edit the reports and dashboards in folders that are shared with them. To restore edit access, a company admin who has Manage Reports in Public Folders or Manage Dashboards in Public Folders permission can assign Manage access to users for particular folders. Those users can then assign View, Edit, or Manage access to those folders to other users.

Changes in Folder Access

The following table summarizes the changes in folder access when enhanced folder sharing is enabled.

For a legacy folder that has read-only or read-write public access and is...	The following occurs after upgrading to enhanced folder sharing
Accessible by all users	The folder is shared with the public group All Internal Users. The group has View access to the folder.
	If the Customer Portal is enabled in your org, then All Customer Portal Users and All Partner Portal Users also get View access to the folder.
Hidden from all users	The folder isn't shared with anyone. Users with View Reports in Public Folders or View Dashboards in Public Folders permission can view the folder contents. Users with Manage Reports in Public Folders or Manage Dashboards in Public Folders permission can edit and manage the folder.
Accessible only to specified Public Groups, Roles, and Roles and Subordinates	The folder is shared with each group and role. The groups and roles receive View access.

Examples

These examples show how enhanced folder sharing works relative to legacy folder sharing.

Restoring Edit Access to a Report or Dashboard

In legacy folder sharing, Sally had the user permissions Manage Dashboards and View All Data, but Tim had only View All Data. Sally and Tim built a dashboard that's saved in a Read-Write folder. With enhanced folder sharing, Sally automatically has the user permissions Manage Dashboards in Public Folders and Create Dashboard Folders. Tim only has the user permission View Dashboards in Public Folders. Sally can view and edit the dashboard that she and Tim created, but Tim can only view it. Because Sally has Manage access to the folder, she's able to grant Edit access to him. The two can now continue to collaborate on the dashboard.

Uninterrupted Edit Access

Tara had the user permission Manage Dashboards in legacy folder sharing. She created and saved a dashboard in a read-write folder. In enhanced folder sharing, Tara has the user permission Edit My Dashboards. She's still able to edit her dashboard in the folder even though

she has only View access. She can also save any new dashboards that she creates in the folder. But she can't edit anyone else's dashboards that are saved in the folder.

SEE ALSO:

Compare Access Levels for Report and Dashboard Folders

Access Levels for Report and Dashboard Folders

User Permissions for Sharing Reports and Dashboards

Access to Dashboard Folders

Permissions control folder access. Users must have certain permissions to access public, hidden, or shared dashboard folders.

Public Folders

The following permissions apply to folders with these visibility settings:

- `This folder is accessible by all users, including portal users`
- `This folder is accessible by all users, except for portal users`

Access Level	Permissions Needed to Access Read-Only Folders	Permissions Needed to Access Read/Write Folders
Read	Run Reports	Run Reports
Write New	All of the following: • Run Reports • Manage Dashboards • View All Data	Both of the following: • Run Reports • Manage Dashboards
Modify/Delete	All of the following: • Run Reports • Manage Dashboards • View All Data	All of the following: • Run Reports • Manage Dashboards • View All Data

Hidden Folders

The following permissions apply to folders that have this visibility setting:

- `This folder is hidden from all users`

Access Level	Permissions Needed to Access Read-Only Folders	Permissions Needed to Access Read/Write Folders
Read	Both of the following: • Run Reports • View All Data	Both of the following: • Run Reports • View All Data

Access Level	Permissions Needed to Access Read-Only Folders	Permissions Needed to Access Read/Write Folders
Write New	All of the following: • Run Reports • Manage Dashboards • View All Data	All of the following: • Run Reports • Manage Dashboards • View All Data
Modify/Delete	All of the following: • Run Reports • Manage Dashboards • View All Data	All of the following: • Run Reports • Manage Dashboards • View All Data

Shared Folders

Access Level	Permissions Needed to Access Read-Only Folders	Permissions Needed to Access Read/Write Folders
Read	Run Reports	Run Reports
Write New	All of the following: • Run Reports • Manage Dashboards • View All Data	Both of the following: • Run Reports • Manage Dashboards
Modify/Delete	All of the following: • Run Reports • Manage Dashboards • View All Data	All of the following: • Run Reports • Manage Dashboards • View All Data

Access to Report Folders

Folder access is controlled by permissions. Users must have certain permissions to access public, hidden, or shared report folders.

The following tables show the permissions that users must have to access the three different types of report folders: public, hidden, and shared.

> **Note:** The "View Reports in Public Folders" permission is available only in organizations with enhanced sharing for reports and dashboards. Organizations without enhanced sharing for reports and dashboards use "View All Data."

Public Folders

The following permissions apply to folders with these visibility settings:

• `This folder is accessible by all users, including portal users`

- This folder is accessible by all users, except for portal users

Access Level	Permissions to Access Read-Only Folders	Permissions to Access Read/Write Folders
Read	*Any* of the following: • "View Reports in Public Folders" • "Run Reports" • "Manage Public Reports" • "View All Data"	*Any* of the following: • "Run Reports" • "Manage Public Reports" • "View All Data"
Write New	"Manage Public Reports"	"Create and Customize Reports"
Modify/Delete	"Manage Public Reports"	"Manage Public Reports"

Hidden Folders

The following permissions apply to folders that have this visibility setting:

- This folder is hidden from all users

Access Level	Permissions to Access Read-Only Folders	Permissions to Access Read/Write Folders
Read	*Any* of the following: • "View Reports in Public Folders" • "View All Data"	"View All Data"
Write New	"Manage Public Reports"	"Manage Public Reports"
Modify/Delete	"Manage Public Reports"	"Manage Public Reports"

> 📝 **Note:** The Hidden Folders setting is available only in organizations without enhanced sharing for reports and dashboards. The equivalent for organizations with enhanced sharing for reports and dashboards is a folder with no sharing settings set, that is, a folder that's not explicitly shared with anyone.

Shared Folders

Access Level	Permissions to Access Read-Only Folders	Permissions to Access Read/Write Folders
Read	*Any* of the following: • "View Reports in Public Folders" • "Run Reports" (for shared users) • "Manage Public Reports" • "View All Data"	*Any* of the following: • "Run Reports" (for shared users) • "Manage Public Reports" (for shared users) • "View All Data"
Write New	"Manage Public Reports"	"Create and Customize Reports" (for shared users)

Access Level	Permissions to Access Read-Only Folders	Permissions to Access Read/Write Folders
Modify/Delete	"Manage Public Reports"	"Manage Public Reports"

SEE ALSO:

[Enhanced Folder Sharing for Reports and Dashboards](#)

User Permissions for Sharing Reports and Dashboards

Each level of access to a report or dashboard folder consists of a combination of user permissions. As an administrator, you can further fine-tune users' access to dashboards and reports by assigning or removing one or more permissions.

When enhanced folder sharing is enabled, all users get Viewer access to report and dashboard folders, except users with higher administrative permissions. To give users broader privileges, assign Editor or Manager folder access and give report and dashboard user permissions as needed.

User Permission	Description
Create and Customize Dashboards	Create, edit, and delete dashboards in the My Personal Dashboards folder. Create dashboards and save to any shared folder if sharing rights allow.
Create and Customize Reports	Create, edit, and delete reports in the My Personal Custom Reports folder. Create reports and save to any shared folder if sharing rights allow.
Create Dashboard Folders	Create dashboard folders and manage them if sharing rights allow.
Create Report Folders	Create report folders and manage them if sharing rights allow.
Edit My Dashboards	Edit, move, save, and delete dashboards that you created in shared folders.
Edit My Reports	Edit, move, save, and delete reports that you created in shared folders.
Manage All Private Reports and Dashboards	Allow user to delete reports and dashboards from personal or private folders.
Manage Dashboards in Public Folders	Create, edit, and delete dashboards,[1] and manage their sharing in all public dashboard folders. This permission does not extend to others' personal folders. This permission allows users to edit and share dashboards in all folders, including hidden folders. They also get these permissions: • Create and Customize Dashboards • Create Dashboard Folders • Edit My Dashboards

EDITIONS

Available in: Salesforce Classic ([not available in all orgs](#)) and Lightning Experience

Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing

- View Dashboards in Public Folders

([1] To edit a dynamic dashboard, users also need Manage Dynamic Dashboards and View My Team's Dashboards permissions.)

Manage Reports in Public Folders	Create, edit, and delete reports, and manage their sharing in all public report folders. This permission does not extend to others' personal folders. This permission allows users to edit and share reports in all folders, including hidden folders. They also get these permissions: - Create and Customize Reports - Create Report Folders - Edit My Reports - View Reports in Public Folders
View Dashboards in Public Folders	View dashboards in public dashboard folders. This permission does not extend to others' personal folders.
View Reports in Public Folders	View reports in public report folders. This permission does not extend to others' personal folders.

SEE ALSO:

Share a Report or Dashboard Folder in Salesforce Classic

Compare Access Levels for Report and Dashboard Folders

Create a Report or Dashboard Folder in Lightning Experience

You can create report and dashboard folders in Lightning Experience.

1. On the Reports or Dashboards tab, click **New Folder**.

2. Name the folder. The folder name must be unique. You cannot have more than one report or dashboard folder with the same name as another report or dashboard folder.

3. Click **Save**.

The folder is now listed on the Reports or Dashboards menu under Created by Me. You can select the folder when saving a report or dashboard.

EDITIONS

Available in: Lightning Experience

Available in: **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing

USER PERMISSIONS

To create report folders:
- Create Report Folders

To create dashboard folders:
- Create Dashboard Folders

Create a Report or Dashboard Subfolder in Lightning Experience

You can create report or dashboard folders within other report or dashboard folders in Lightning Experience.

1. On the Reports or Dashboards tab, open the folder you're working with.

2. Click **New Folder**.

3. Name the subfolder. The name must be unique. You cannot have more than one report or dashboard folder with the same name as another report or dashboard folder.

4. Click **Save**.

The subfolder is created and added to the list of items in the folder you started from. You can select the subfolder when saving a report or dashboard.

Global Search for Folders

You can search for report or dashboard folders using global search. Global search for folders is not supported in the mobile version of Salesforce.

Select **Folders** from the **ALL SEARCHABLE ITEMS** list in the global search area.
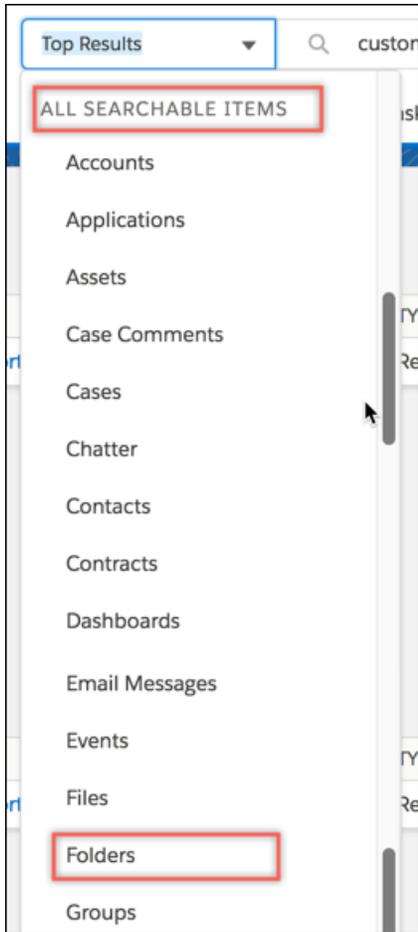
Start typing the folder name in the Global Search box to see a list of matching names to select from, or press Return to see the matching results.

If the folder you are looking for doesn't appear in the search results, click **Show More** on the left side of the page, and select **Folders** to display the matching folders.
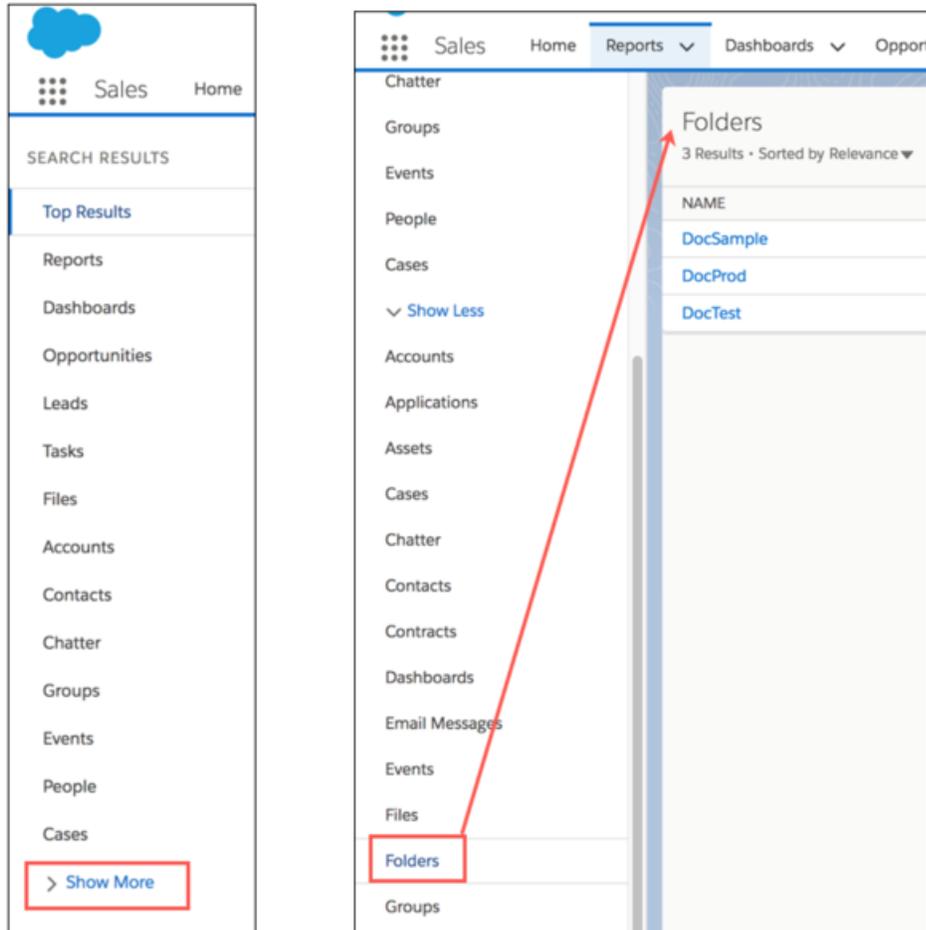
If the Folders object is already selected on the left side, the search options include searching only within Folders.

See Search for Records in Lightning Experience for details on all the global search options.

Quick Search for Folders

Quick search allows you to find folders within the selected report, dashboard or folder context.

Use global search when you want to find folders across your entire org. Use quick search when you want to find folders within the select context.

You'll find the quick search box at the top of each report, dashboard, and folder page. Select a scope on the side menu [1] and the search context instantly changes to that scope.[2]

Start entering text to see the list of matches within the selected scope.

Here are some things to remember about quick search:

- Quick search shows only the first 20 results. A message indicates if there are more results that aren't showing.
- The context is always what is selected on the side menu.
- Quick search on folders finds all folders and subfolders, but not the reports and dashboards in the folders.
- The options to customize columns and create new folders are disabled during quick search.
- Quick search keeps the current custom column selections and sorting.

Add a Folder as a Favorite

Add a report or dashboard folder as a favorite to quickly access the folders you're interested in.

You can add a folder as a favorite in any of the following ways:

- On the **Reports** or **Dashboards** page, find the report or dashboard you want to add as a favorite and select the **Favorite** row level action. (If you have customized columns on the page, you might have to scroll to the right to see the row-level action arrow ▼ .)
- Navigate to the report or dashboard folder, click ▼ next to the **New Folder** button, and then select **Favorite**.
- Open the folder to display its contents, and then click the star ★ in the Lightning Experience header.

When you favorite a folder:

- It is added as a global favorite.
- It is also added to the All Favorites list in the Reports or Dashboards menu.

For some things to keep in mind when working with favorites, see Favorites Considerations.

📝 **Note:** To display a favorite folder, click the arrow to the right of the star in the header of Lightning Experience and select the folder, or select the folder from the Favorites items on the Reports or Dashboards menu.

Share a Report or Dashboard Folder in Lightning Experience

Use enhanced folder sharing to share a report or dashboard folder with users, groups, roles, or territories. Enhanced folder sharing is the default option for all orgs created after 2013.

You can share a report or dashboard folder with up to 25 users, groups, roles, or territories from the UI. To share a folder with up to 500 users, groups, roles, or territories, use the folder sharing REST API.

1. On the **Reports** or **Dashboards** page, find the report or dashboard you want to share and select the **Share** row level action. (If you have customized columns on the page, you might have to scroll to the right to see the row-level action arrow ▼ .) Alternatively, navigate to the report or dashboard folder, click ▼ next to the **New Folder** button, and select **Share**.

2. From the **Share With** dropdown, select who you want to share with.
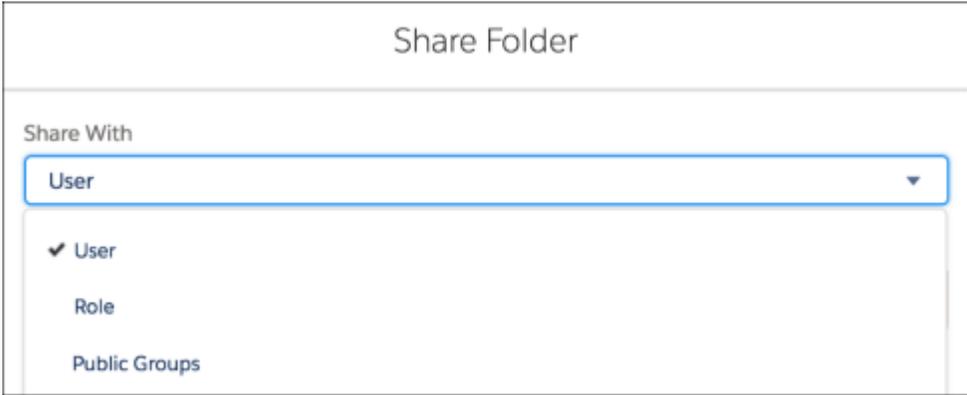
3. For **Name**, enter the name you want to match. The name must match the category (such as user, group, or role). For example, if you select **User** as the category, specify the name of a user.

4. Select the folder access level for the user, role, group, or territory.

5. Click **Share**. The sharing setting is applied to the Who Can Access list. Continue to add entries as needed. You can share with users, groups, roles, territories, or a combination. If a user is identified in more than one entity, the greatest permission is granted. For example, a group is added with view only, but a user in the group is also added with edit or manage permissions. In this case, the edit or manage permissions apply to that user. To delete a Who Can Access entry, click **X** to the right of the entry. To modify access level for an entry, select a new access level.

6. When you are done, click **X** in the upper corner of the dialog box to close it and return to the page you were on.

The folder is now listed on the Reports or Dashboards menu. You can select the folder when saving a report or dashboard.

Rename a Report or Dashboard Folder in Lightning Experience

You can rename report and dashboard folders and subfolders in Lightning Experience.

1. On the **Reports** or **Dashboards** page, find the report or dashboard you want to rename and select the **Rename** row level action. (If you have customized columns on the page, you might have to scroll to the right to see the row-level action arrow ▼ .) Alternatively, navigate to the report or dashboard folder, click ▼ next to the **New Folder** button, and select **Rename**.

2. Enter the new name.

3. Click **Save**.

The folder name is updated.

Move Dashboards Between Folders in Lightning Experience

Move dashboards between folders to organize the dashboards and control access.

1. Do either of the following on the Dashboards page:

   - Click the dashboard you want to move. Click **Edit**. Click ⏷ and then select **Save As**. In the dialog box, click **Select Folder**.

   - Find the report you want to move to and select the **Move** row level action. (If you have customized columns on the page, you might have to scroll to the right to see the row-level action arrow ⏷ .)

2. Select the target folder or subfolder.

   - Search for folders by name, or select a folder on the left and scroll to find the target folder on the right.

   - If you are looking for a subfolder, continue to navigate to the right as needed to find the subfolder.

   - If you would like create a new folder for the dashboard, click **New Folder**. Enter a name for the folder and click **Save** to create the folder and select it.

   - You can use the breadcrumb links above the search box to navigate back to parent folders.

   - When you select a folder, the **Select** button changes to include the folder name.

3. After selecting the target folder, click the **Select** button.

The dashboard is saved to the specified folder.

Move Reports Between Folders in Lightning Experience

Move reports between folders to organize the reports and control access.

1. Do either of the following on the Reports page:

   - Click the report you want to move. Select ⏷ and then select **Save As**. In the dialog box, click **Select Folder**.

   - Find the report you want to move and select the **Move** row level action. (If you have customized columns on the page, you might have to scroll to the right to see the row-level action arrow ⏷ .)

2. Select the target folder or subfolder.

   - Search for folders by name, or select a folder on the left and scroll to find the target folder on the right.

   - If you are looking for a subfolder, continue to navigate to the right as needed to find the subfolder.

   - If you would like create a new folder for the report, click **New Folder**. Enter a name for the folder and click **Save** to create the folder and select it.

   - You can use the breadcrumb links above the search box to navigate back to parent folders.

   - When you select a folder, the **Select** button changes to include the folder name.

3. After selecting the target folder, click the **Select** button.

The report is moved to the new folder.

Share a Report or Dashboard Folder in Salesforce Classic

Enhanced folder sharing is the default option for all orgs created after the Summer '13 Salesforce release. If you have orgs created before Summer '13 and don't want to reassign permissions for the legacy reports and dashboards, you can use legacy folder sharing in Salesforce Classic.

> 📝 **Note:** To give access to a folder, you must have either Manager access to that folder, the Manage Reports in Public Folders permission (for report folders), or the Manage Dashboards in Public Folders permission (for dashboard folders).

When you create a folder, you're its manager. Only you and others with administrative permissions can see it.

If a folder does not have Manager access, it's public, and users with the View Reports in Public Folders permission can view it. Depending on their object access, these users can also run the report.

You can share a report or dashboard folder with up to 25 users, groups, roles, or territories at one time. You can share a folder with up to 100 users, groups, roles, or territories using the folder sharing REST API.
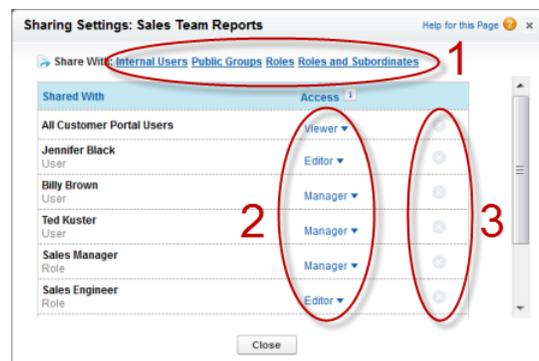
Share your folder by user, by user group, role, or by territory (1). Choose the access level you want each user, group, role, or territory to have (2). Stop sharing the folder with the user, group, role, or territory (3).

SEE ALSO:

Access Levels for Report and Dashboard Folders

User Permissions for Sharing Reports and Dashboards

Share a Report or Dashboard Folder with an Individual User in Salesforce Classic

Enhanced folder sharing is the default option for all orgs created beginning with the Summer '13 Salesforce release. Legacy folder sharing, as described in this topic, is currently supported only for orgs created prior to the Summer '13 release. It is due for retirement in Summer '20. At that time all orgs will be converted to Enhanced Folder Sharing.

1. On the Reports tab, hover over a report folder in the left pane, click ![icon], and then select **Share**.

2. Select **Internal Users**.

    ![note icon] Note: Internal users doesn't include customer portal or partner portal users.

3. Find the user you want, click **Share**, and choose an access level.

    To search, start typing a name.
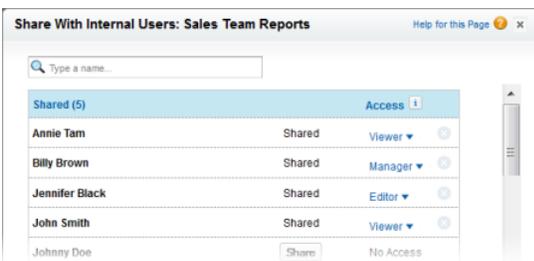
4. Click **Done**, review your changes, and click **Close**.

Share a Report or Dashboard with a Group in Salesforce Classic

Enhanced folder sharing is the default option for all orgs created beginning with the Summer '13 Salesforce release. Legacy folder sharing, as described in this topic, is currently supported only for orgs created prior to the Summer '13 release. It is due for retirement in Summer '20. At that time all orgs will be converted to Enhanced Folder Sharing.

1. On the Reports tab, hover over a report folder in the left pane, click ⚓, and then select **Share**.

2. Select **Public Groups**.

3. Find the group you want, and click **Share**.

To search, start typing a name.
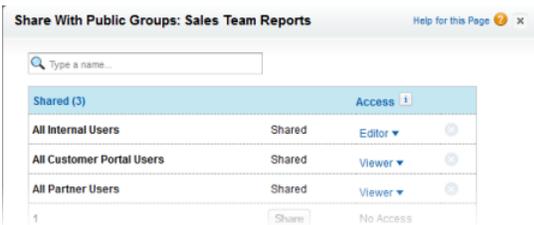
4. Choose the sharing level you want to give this group.

> 📝 **Note:** Portal users can only have Viewer access to reports, and they can't use dashboards.

5. Click **Done**, review your changes, and click **Close**.

Share a Report or Dashboard by Role in Salesforce Classic

Enhanced folder sharing is the default option for all orgs created beginning with the Summer '13 Salesforce release. Legacy folder sharing, as described in this topic, is currently supported only for orgs created prior to the Summer '13 release. It is due for retirement in Summer '20. At that time all orgs will be converted to Enhanced Folder Sharing.

You can give report or dashboard folder access to users in a role, or to those users plus users in roles subordinate to that role.

For example, suppose the VP of Sales role and its subordinates have Viewer access to a dashboard folder, while the role itself (VP of Sales) has Manager access. In this case, a user in the VP of Sales role has greater control than someone with a role that's lower in the role hierarchy. If the VP of Sales leaves the company, whoever next assumes that role can manage dashboards in the folder.

1. On the Reports tab, hover over a report folder in the left pane, click 📌, and then select **Share**.

2. Select **Roles** or **Roles and Subordinates**.

   - To give access to all users who have the role, select **Roles**.

   - To give access to those users plus everyone with a role below them in the role hierarchy, select **Roles and Subordinates**.

3. Find the role you want, click **Share**, and choose a level of access.

   To search, start typing a name.
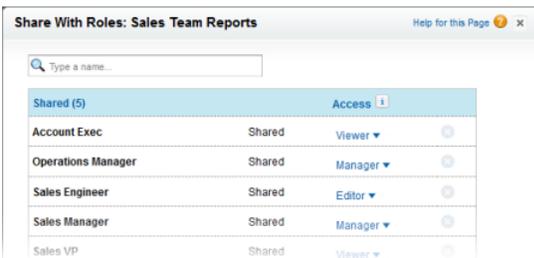
4. Click **Done**, review your changes, and click **Close**.

Move a Report or Dashboard Between Folders in Salesforce Classic

It's a good practice to keep reports and dashboards organized in folders that reflect their function and audience. You can drag and drop reports and dashboards from one folder to another.

> **Note:** You need edit access to folders before moving items between them. Moving items using drag-and-drop isn't supported in accessibility mode.

Move a report or dashboard between folders by dragging them from the list view to a report or dashboard folder on the Folders pane.

1. On the Reports tab list view, click and hold an item.

2. Drag the item to its destination folder in the Folders pane.
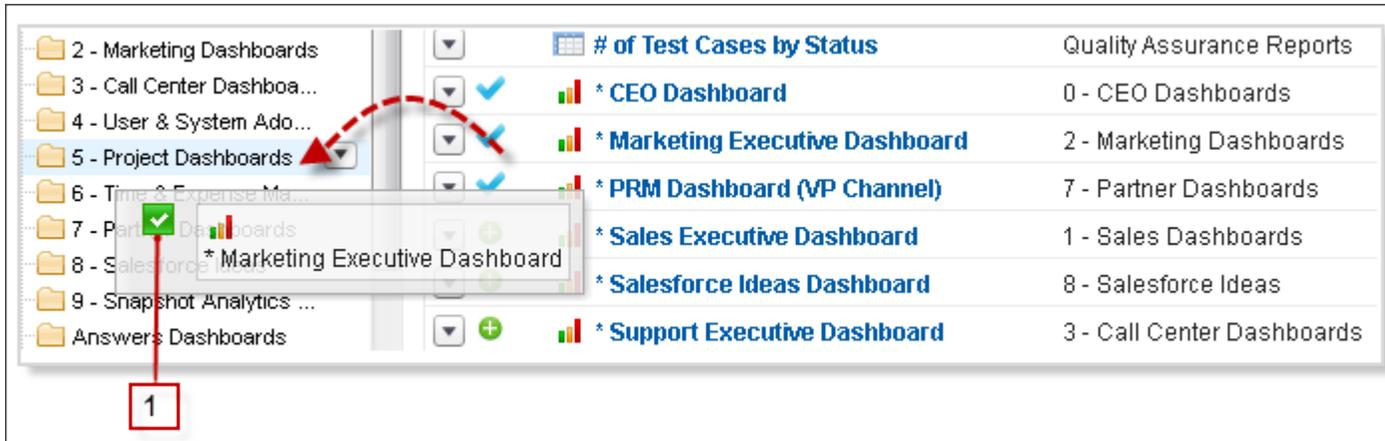
As you drag an item across a folder, a green check mark (1) indicates that the item can be moved into the selected folder. Conversely, a red icon (🚫) means that the item can't be moved into the selected folder.

Keep the following in mind when moving items.

- Drag one item at a time.
- You can't move items from installed AppExchange packages or standard report folders into other folders.
- Move reports into report folders and dashboards into dashboard folders.

Deleting Report or Dashboard Folders

Some rules apply to deletion of report and dashboard folders.

- You can delete an empty leaf folder or empty folder tree in Lightning Experience. An empty leaf folder is a folder that doesn't contain any reports or dashboards and doesn't have any subfolders. An empty folder tree is one with no reports or dashboards in the root folder or in any of the subfolders.
- If you want to delete a non-empty folder, either move the reports or dashboards in the folder to another folder, or delete the reports or dashboards. Then permanently delete the reports and dashboards from the Recycle Bin.
- You can't delete empty folders from managed packages. To delete the folder, uninstall the managed package or contact the package provider for help.

1. On the **Reports** or **Dashboards** page, find the report or dashboard you want to delete and select the **Delete** row level action. (If you have customized columns on the page, you might have to scroll to the right to see the row-level action arrow ▼ .) Alternatively, navigate to the report or dashboard folder, click ▼ next to the **New Folder** button, and select **Delete**.
2. Click **Delete** to confirm.

The folder is deleted.

EDITIONS

Available in: Lightning Experience

Available in: **Essentials**, **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Available in: Enhanced Folder Sharing

USER PERMISSIONS

To delete report folders:
- Manage access for the specific folder

To delete dashboard folders:
- Manage access for the specific folder

Report and Dashboard Limits, Limitations, Allocations, and Technical Requirements

As you report on your Salesforce data and build dashboards, be aware of these limits, limitations, and allocations. If you experience slow load times or sluggish interactions with reports and dashboards, verify that your network and device meet the minimum recommended technical requirements for Lightning reports and dashboards.

Report Limits, Limitations, and Allocations

As you report on your data, be aware of these limits, limitations, and allocations.

Reports and Dashboards: What's Different or Not Available in the Salesforce Mobile App

Lightning Reports and Dashboards Technical Requirements

Review these recommended technical requirements to get the best performance out of Salesforce reports and dashboards.

Report Limits, Limitations, and Allocations

As you report on your data, be aware of these limits, limitations, and allocations.

Salesforce Reports and Dashboards Allocations

Feature	Personal Edition	Contact Manager	Group Edition	Professional Edition	Enterprise Edition	Unlimited and Performance Edition	Developer Edition
Custom report types (Limits apply to all custom report types regardless of development status.)	N/A			50	200	2,000	400
Dashboard filters				3 per dashboard			
Dynamic dashboards per org	N/A				Up to 5	Up to 10	Up to 3
Field filters per report[1]	20						
Formulas per report	5						
Reporting snapshots	N/A			1[234]	1[2]	2[2]	1[234]

Feature	Personal Edition	Contact Manager	Group Edition	Professional Edition	Enterprise Edition	Unlimited and Performance Edition	Developer Edition
Scheduled dashboard refreshes	N/A				1^2	2^2	N/A
Scheduled reports per hour (Emailed reports can be up to 10 MB.)	N/A			1^{235}	1^2	2^2	1^{235}

[1] These allocations apply to the report builder. If you're using the report wizard, the allocation is 10.

[2] Up to 200 total.

[3] Off-peak hours (between 6 PM and 3 AM local time) only.

[4] Limited to one preferred start time per day.

[5] Limited to three preferred start times per day.

Salesforce retains historical data for the previous three months, plus the current month.

The following Salesforce Reports and Dashboards limits, limitations, and allocations apply to all supported editions.

Report Limits, Limitations, and Allocations

- The report builder preview shows a maximum of 20 rows for summary reports (grouped by rows) and matrix reports (grouped by columns), and 50 rows for tabular reports (no groupings).

- In Salesforce Classic, you can't have more than 250 groups or 4,000 values in a chart. If you see an error message saying that your chart has too many groups or values to plot, adjust the report filters to reduce the number. In combination charts, all groups and values count against the total.

- In Lightning Experience, a report chart can have at most 2000 groups. If a report has more than 2000 groups, the action 'Combine Small Groups into Others' applies only to the small groups within the 2000 that are included in the report chart. Any additional small groups are ignored.

- Reports display a maximum of 2,000 rows. To view more the rows, export the report to Excel or use the printable view for tabular and summary reports. For joined reports, printable view displays a maximum of 20,000 rows. Printable view is only available in Salesforce Classic.

  - Summary reports (grouped by rows) and matrix reports (grouped by columns) display the first 2,000 groupings when Show Details is disabled.

  - Matrix reports display a maximum of 400,000 summarized values.

  - Matrix reports display a maximum of 2,000 groupings in the vertical axis when Show Details is disabled. If there are more than 400,000 summarized values, rows are removed until the 2,000 groupings limit is met. Then columns are removed until the number of summarized values moves below 400,000.

  - Matrix reports that return more than 2,000 rows don't show details. If you click **Show Details**, nothing happens. You can only view the report with details hidden.

  - Because a matrix report includes multiple groupings, the maximum of 2000 values is typically reported in fewer than 2000 groups.

- Up to five metrics display in the Lightning Experience report header. Metrics such as summarized fields appear in the order that they appear in the report, left to right. The grand total, when shown, always displays.

- When reports that have groupings are viewed in the Salesforce mobile app, they're converted to tabular reports.

- The Salesforce mobile app supports a maximum of 25 report columns.

- By default, reports time out after 10 minutes.

- In a joined report, each block can have up to 100 columns. A joined report can have up to 5 blocks.

- When you add a block to a joined report and the block has multiple entities in common with the report, only the first entity (in alphabetical order) is shown. Only the fields from the first entity are shown in the common fields area.

- You can add up to 10 custom summary formulas to each block in a joined report. A joined report can have a total of 50 custom summary formulas.

- Each joined report can have up to 10 cross-block custom summary formulas.

- In a non-joined report, if you click a bar in a report chart, the report results are filtered according to the selected bar. In a joined report, clicking a bar doesn't apply the filter.

- If you filter on standard long text area fields, such as Description or Solution Details, only the first 1000 characters of the field are searched for matches in reports.

- Some filters (such as date range) are constructed using multiple custom filters, each of which counts towards the total of 20.

- Field-to-field filtering isn't available on currency fields for orgs that have multi-currency enabled.

- The first 999 characters in a standard rich text area or a long text area are displayed in a report. For custom fields, only the first 255 characters are displayed.

- Summary fields on tabular, summary, and matrix reports can display up to 21-digits.

- Reports can't be filtered on custom long text area fields.

- Joined reports require that the new user interface theme is enabled. Users without the new theme are unable to create, edit, or run joined reports.

- Forecast reports include only opportunities that are set to close within the forecast period, except those assigned to the Omitted forecast category.

- Internet Explorer 6 isn't supported for joined reports.

- Acceptable range for values: The maximum value allowed is 999999999999999. The minimum value allowed is -99999999999999.

- Each person in your org can subscribe to up to 5 reports.

- Up to 500 individual recipients can be added. A recipient is a user, role, or group.

- If a role or group contains more than 500 users, users sometimes don't receive the updated report.

- The State/Province picklist filter converts the selected state or province to a two-digit code (example: MO for Missouri). If a state or province in another country covered by the report has the same code (example: MO for Morales, Mexico), filtering on one of the states or provinces can return data for the other.

- In Lightning Experience, embedded report charts display the source report table's groupings, not the report chart's. In Salesforce Classic, embedded report charts display the source report chart's grouping, not the report table's groupings.

Dashboard Limits, Limitations, and Allocations

- A dashboard filter can have up to 50 values.

- Each dashboard can have up to 20 components.

- It's not possible to filter on bucket fields. However, it's possible to use a report filtered on a bucket field on the dashboard page.

- Filtering is restricted in some dashboards that contain multiple components based on different report types:

  - If a dashboard has a component based on Cases or Leads and another component based on a different report type, you can't filter the dashboard on the Case Owner or Lead Owner field. In addition, filtering on other Owner fields doesn't display Case Owner or Lead Owner as equivalent fields.

- – If a dashboard has a component based on the Tasks and Events, Activities with Accounts, or Activities with Contacts report type and another component based on a different report type, you can't filter the dashboard on the Assigned field.

- A dashboard table or chart can display up to 20 photos.

- Wait at least one minute between dashboard refreshes.

- File attachments for report subscriptions are limited to 15,000 rows, 30 columns, and 3 MB file size. Extra data is clipped or not sent.

- Each person in your org can subscribe to up to 5 dashboards.

- Up to 500 individual recipients can be added. A recipient is a user, role, or group.

- If a role or group contains more than 500 users, some users don't receive the updated dashboard.

- Downloaded and shared images of dashboard component tables have a maximum height of 3000 pixels or approximately 100 rows. Extra rows beyond the limit are clipped. To avoid clipping, filter the chart to fewer than 100 rows.

- For funnel charts, the total value isn't included in subscription emails.

Report Type Limits, Limitations, and Allocations

- A custom report type can contain up to 60 object references. For example, if you select the maximum limit of four object relationships for a report type, then you could select fields via lookup from an extra 56 objects. However, users receive an error message if they run a report from a custom report type and the report contains columns from more than 20 different objects.

- You can add up to 1000 fields to each custom report type.

Reporting Snapshot Limits, Limitations, and Allocations

- The maximum number of rows you can insert into a custom object is 2,000.

- The maximum number of runs you can store is 200.

- The maximum number of source report columns you can map to target fields is 100.

Filtering Limits, Limitations, and Allocations

- Only the first 255 characters in a custom text field count for filtering purposes.

- You can enter up to 1,333 characters for filter criteria, including commas used as OR operators.

Embedded Report Charts Limits, Limitations, and Allocations

- You can have two report charts per page.

- You can only add report charts from the enhanced page layout editor. The mini console and the original page layout editor aren't supported.

- For historical trend reports in Lightning Experience, you must set snapshot date as the primary row grouping.

- In Lightning Experience, embedded report charts display the source report table's groupings, not the report chart's. In Salesforce Classic, embedded report charts display the source report chart's grouping, not the report table's groupings.

- On detail pages, users can refresh up to 100 report charts every 60 minutes.

- Your org can refresh up to 3,000 report charts every 60 minutes.

List View Limits, Limitations, and Allocations

- Only the first 255 characters are shown for custom long text area fields in list views.

Bucket and Bucket Field Limits, Limitations, and Allocations

- Each report can include up to five bucket fields.

- Each bucket field can contain up to 20 buckets.

- Each bucket can contain up to 20 values.

- Bucket fields are available for use only in the report where they're generated. To use a bucket in multiple reports, create the field for each report, or create a separate formula field for the object that's dependent on the bucket.

  📝 **Note:** These limits don't apply to the use of Other as permitted within the bucket field's setup.

- Buckets and bucket fields aren't available for reports that include external objects.
- If a bucket field's source column has a custom index, and you filter by the bucket field, then the performance gains from the custom index are lost.

Historical Trend Report Limits, Limitations, and Allocations

- Salesforce retains historical data for the previous three months, plus the current month.
- Up to 5 million rows of historical trending data can be stored for each object. Historical data capture stops when the limit is exceeded. The admin is alerted by email when any object reaches 70 percent of the limit, and again if the limit is exceeded.
- Each historical trend report can contain up to 100 fields. In Opportunities reports, the fields include standard preselected fields, which can't be disabled.
- Formula fields aren't supported.
- Row limit filters aren't supported.
- The summary report format isn't supported.
- You can specify up to five historical snapshot dates in each historical trend report.
- You can use up to four historical filters on each historical trend report.
- These field types are supported: Number, Currency, Date, Picklist, Lookup.
- Dynamic exchange rates aren't supported. When you run a historical trend report, it uses a static exchange rate, which could be outdated.
- Internet Explorer 6 isn't supported.
- You can't subscribe to historical trend reports.
- The Report Wizard isn't supported. Historical trend reports can only be created with the Report Builder.
- Historical trend reporting with charts is supported in Lightning Experience, but tabular views of historical trend reports aren't available.

External Object Report Limits, Limitations, and Allocations

If your report includes an external object, the results probably don't reflect the full data set. External objects behave similarly to custom objects, except that they map to data that's stored outside your Salesforce org. A report that includes an external object fetches up to 20,000 records for the primary object and can encounter callout limits while fetching external object data. If the report results in few or no rows, try customizing the report to obtain more relevant external object rows.

Cross Filters

- Each report can have up to three cross filters.
- Each cross filter can have up to five subfilters.
- Filter logic applies only to field filters, not cross filters.

Lightning Experience Report Subscriptions

- Each user can set up subscriptions for up to 5 reports.
- Subscription recipients aren't listed on the report subscription emails.
- Each subscription supports up to 500 recipients. Each recipient is a single user, role, role and subordinates, or group. Roles, roles and subordinates, and groups can each have more than 500 users, but subscriptions send a maximum of 500 emails. If a recipient role, role and subordinates, or group has lots of users, some of them don't receive subscription emails.

After including all users from roles, roles and subordinates, and groups, if subscriptions have more than 500 users as recipients, users are prioritized over roles, roles are prioritized over roles and subordinates, and roles and subordinates are prioritized over groups. Each time the subscription sends an email, the role and group users who receive the email are chosen again and can be different each time a subscription email sends.

For example, say that a subscription has 100 recipients: 98 users, 1 role that includes 500 users, and one group that includes 400 users. The total number of users associated with the subscription is 998. When the subscription email sends, 98 users from the role don't receive subscription emails and none of the 400 users in the group receive emails.

- Each Salesforce organization can schedule up to 500 dashboard subscriptions and 500 report subscriptions on a given hour of a given day, such as Monday at 9:00am.

- Lightning Experience report subscriptions don't support these features:

  - Historical tracking reports

  - Joined reports

  - Conditional highlighting (You can subscribe to reports with conditional highlighting, but conditional highlighting doesn't appear in the subscription email.)

Lightning Experience Dashboard Subscriptions

- Each user can set up subscriptions for up to 5 dashboards.

- You can subscribe to filtered dashboards, but dashboard filters are never applied to emailed dashboards. When you open the email, the dashboard is displayed unfiltered.

- Dashboard subscription emails don't reflect changes made to color palette and theme.

- Dashboards that are configured to display data as **The dashboard viewer** under **View Dashboard As** settings in **Dashboard Properties** don't support subscriptions. You can't subscribe to them.

- In the email to subscribers, the 'From' address is taken from My Email Settings. If no address is specified in My Email settings, the 'From' address is taken from the User object.

- For funnel charts, the total value isn't included in the subscription email.

- Each subscription supports up to 500 recipients. Each recipient is a single user, role, role and subordinates, or group. Roles, roles and subordinates, and groups can each have more than 500 users, but subscriptions send a maximum of 500 emails. If a recipient role, role and subordinates, or group has lots of users, some of them don't receive subscription emails.

  After including all users from roles, roles and subordinates, and groups, if subscriptions have more than 500 users as recipients, users are prioritized over roles, roles are prioritized over roles and subordinates, and roles and subordinates are prioritized over groups. Each time the subscription sends an email, the role and group users who receive the email are chosen again and can be different each time a subscription email sends.

  For example, say that a subscription has 100 recipients: 98 users, 1 role that includes 500 users, and one group that includes 400 users. The total number of users associated with the subscription is 998. When the subscription email sends, 98 users from the role don't receive subscription emails and none of the 400 users in the group receive emails.

- Each Salesforce organization can schedule up to 500 dashboard subscriptions and 500 report subscriptions on a given hour of a given day, such as Monday at 9:00am.

Lightning Experience on Apple® iPad® Safari®

When you build or edit reports with Lightning Experience on Apple iPad Safari, you can't drag fields and columns in the Fields and Overview panes. As a workaround, add columns using the **Add column…** lookup. To reorder columns, remove them from the report, then add them back in the order you prefer.

When you build or edit dashboards with Lightning Experience on Apple iPad Safari, you can't drag components to reposition or resize them. We recommend that you use Lightning Experience on a desktop to reposition or resize dashboard components.

Reports and Dashboards API Limits, Limitations, and Allocations

The following limits, limitations, and allocations apply to both the Reports and Dashboards REST API and Reports and Dashboards API via Apex.

- Cross filters, standard report filters, and filtering by row limit are unavailable when filtering data.
- Historical tracking reports are only supported for matrix reports.
- Subscriptions aren't supported for historical tracking reports.
- The API can process only reports that contain up to 100 fields selected as columns.
- A list of up to 200 recently viewed reports can be returned.
- Your org can request up to 500 synchronous report runs per hour.
- The API supports up to 20 synchronous report run requests at a time.
- A list of up to 2,000 instances of a report that was run asynchronously can be returned.
- The API supports up to 200 requests at a time to get results of asynchronous report runs.
- Your organization can request up to 1,200 asynchronous requests per hour.
- Asynchronous report run results are available within a 24-hour rolling period.
- The API returns up to the first 2,000 report rows. You can narrow results using filters.
- You can add up to 20 custom field filters when you run a report.
- Your org can request up to 200 dashboard refreshes per hour.
- Your org can request results for up to 5,000 dashboards per hour.

SEE ALSO:

Reports and Dashboards: What's Different or Not Available in Lightning Experience

Reports and Dashboards: What's Different or Not Available in the Salesforce Mobile App

Reports

Considerations When Using Reports in the Salesforce Mobile App

Feature	Notes about Salesforce Mobile App Availability
Number of Rows Displayed	Reports display a maximum of 2,000 rows, same as on the desktop Salesforce site.
Groupings	When you view a report with groupings, the groupings are displayed as columns at the end of the report.
Report Formats	Reports that are grouped by rows (summary) or rows and columns (matrix) are displayed without the groupings. Joined reports aren't available.
Conditional Highlighting	You can't view reports that show conditional highlighting.

Feature	Notes about Salesforce Mobile App Availability
Filters	When you open a report from the Reports tab, you can't filter the report. When you tap a dashboard component to open the source report, you can filter the report by tapping a value on the chart. If the source report is a tabular or joined report, then you can't filter it.

Report Features Not Available

- Create, edit, or delete reports
- Export
- Print
- Feed
- Schedule report refreshes
- Subscribe
- Joined reports
- Historical trend reports
- Add to campaign
- Role hierarchy
- Custom summary formula fields
- Folders
- Hide details
- Summary information (grand totals, subtotals, summarized fields, record counts, etc.)

Other Notes about Using Reports

- You can't drill into reports that have more than three checkbox fields.
- When you view a report with more than 25 summary fields, you receive an error message.
- The Salesforce mobile app can't render reports via URLs that use dynamic parameter values. If you modify a URL to pass parameters into reports, the app shows a blank screen (a report record with no returned results).

Dashboards

Considerations When Using Dashboards

Feature	Notes about Salesforce Mobile App Availability
View As	As in the desktop Salesforce site, you can only run dashboards as a user in your role hierarchy. However, in the Salesforce mobile app you can choose from all users in your organization. If you select a user outside your role hierarchy, you get an error.
Dashboard Layout	Dashboards display in a single-column layout on phones, and up to a two-column layout on tablets.

Dashboards Features Not Available

- Create, edit, or delete dashboards
- Feed
- Schedule
- Link from a dashboard component to a website or email address
- Visualforce components on dashboards
- Folders

Other Notes about Using Dashboards

In some situations, data displayed in a dashboard component can get out of sync with data in the report that's displayed on the same page. When a dashboard component's data doesn't match the report, one of these things is happening:

- The dashboard is being refreshed as the configured user or the running user, while a report is always run as the current user.
- The report was refreshed more recently than the dashboard. A report is refreshed every time you look at it (assuming you aren't working offline). But a dashboard component is refreshed only when the dashboard it belongs to is refreshed.

The same temporary mismatch can occur in the desktop site, but there you see reports and dashboard charts on separate pages. You see the report and the dashboard chart on the same page.

Charts

Other Notes about Using Charts

- Report Charts are only available after drilling into a dashboard component's report. Report charts aren't available from the Reports tab.
- Embedded report charts don't link to the source report.

Lightning Reports and Dashboards Technical Requirements

Review these recommended technical requirements to get the best performance out of Salesforce reports and dashboards.

These technical requirements are provided to help you predict whether your hardware and network can provide an acceptable and productive user experience. Salesforce strongly recommends testing the actual end-user experience with a configuration identical to what you expect to use in production. Test using the same geographic location, hardware, browser, network settings, and the expected concurrent users for shared hardware like virtual desktops. In Lightning Experience, page load times can be captured using Lightning Component Debug Mode, or by appending `?eptVisible=1` to your URL.

```
https://myDomainName.lightning.force.com/one/one.app?eptVisible=1
```

Technical Requirements

For the fastest and most stable experience, we recommend:

- An Octane score of 30,000
- Network latency of 150 ms or lower
- Download speed of 3 Mbps or higher
- At least 8 GB of RAM, with 3 GB available for Salesforce browser tabs

Minimum requirements are:

- An Octane score of 20,000
- Network latency of 200 ms or lower

- Download speed of 1 Mbps
- At least 5 GB of RAM, with 2 GB available for Salesforce browser tabs

Based on our lab tests, the minimum requirements result in 50% slower page load times and login load times versus the recommended specifications. Users that use over 1,000 records a day are more likely to have their browser tab crash when using the minimum requirements due to memory limits.

You can find your Octane score, latency, and download speed by appending "speedtest.jsp" to your org's domain.

```
https://MyDomainName.lightning.force.com/speedtest.jsp
```

We recommend running this test on the same hardware, network, physical location, and browser as your users. For virtual environments, such as VDI, run all tests from within that virtual environment.

Improving Page Load Times

The most important factors in predicting page load times are Octane score, network latency, download speed, and the amount of customization on a given page. See this knowledge article for more information on how to improve performance.

Improving Octane Scores

Octane is a benchmark developed by Google that measures JavaScript performance. A higher Octane score correlates to faster page load times. Octane factors in your computer hardware and browser choice.

- Using newer-generation hardware with faster CPUs generates higher Octane scores.
- Using the latest version of Salesforce-supported browsers generates higher Octane scores.
  - IE11 results in low Octane scores and much slower page load speeds.

Improving Tab Longevity

The amount of RAM available has a significant impact on browser tab longevity. Having enough RAM available helps prevent crashes. Insufficient RAM can also negatively impact page load times with older browsers.

Improving Virtual Desktop Environment Performance

Virtual desktop environments often have older processors and are shared by many users, resulting in slower page load times. To predict how a virtual desktop performs with Lightning Experience, run performance tests from within the virtual environment. Use as many concurrent users as expected in production, and take their usage patterns into account. Run `speedtest.jsp` to determine the Octane score during concurrent use, and test page load times during concurrent use using Lightning Component Debug Mode or the `eptVisible=1` URL parameter.

If your Octane score is below 20,000, or you have slow page load times, Salesforce recommends upgrading your hardware, reducing the number of users per environment, or using dedicated desktops.

Explore Data and Take Action with Tableau CRM

Salesforce Tableau CRM, formerly known as Einstein Analytics and Wave, is a cloud-based platform for connecting data from multiple sources, creating interactive views of that data, and sharing those views in apps. Tableau CRM is a better way to distribute insight to business users so they can understand and act on changing information.

You've landed in the right place for Help pages on every aspect of Tableau CRM. We also recommend these top-level learning resources:

- Quickly locate content by task: Tableau CRM Learning Map
- Find all the learning opportunities: Tableau CRM Learning Resources Overview
- View expert-led webinars: Tableau CRM Best Practices Training

Salesforce Help	Other Resources

Get to Know Tableau CRM	
Get Started Using Tableau CRM	• Trailhead: Analytics Basics • Video: ▶ Get Quick Answers from Your Data with Conversational Exploration • Trailhead: Tableau CRM Dashboard Navigation • Video: ▶ Personalize, Collaborate, and Take Action from the Analytics Tab • Video: ▶ Build Unique Tableau CRM Apps in the Tableau CRM Studio
Access Insights from the Tableau CRM Mobile App	• Tableau CRM for iOS Mobile Help • Tableau CRM for Android Mobile Help • Trailhead: Mobile Analytics Exploration

Enable Tableau CRM and Integrate Data	
Set Up the Tableau CRM Platform	PDF	• Trailhead: Analytics Administration Basics • Tableau CRM Security Implementation Guide PDF
Load Data into Tableau CRM Datasets	• Video: ▶ Introducing Data Prep • Trailhead: Analytics Data Integration Basics • Tableau CRM External Data API Developer Guide • Tableau CRM External Data Format Reference

Build, Customize, and Develop Tableau CRM Assets	
Explore and Visualize Your Data in Tableau CRM	• Tableau CRM Learning Adventure App • Video: ▶ Introduction to Explorer • Trailhead: Desktop Analytics Exploration • Trailhead: Create a Custom Map for Analytics Charts • Trailhead: Build a Gauge Chart That Visually Identifies Regional Data
Build Tableau CRM Dashboards	• Video: ▶ Build Interactive Tableau CRM Dashboards • Trailhead: Tableau CRM Dashboard Building Basics • Trailhead: Create a Product Pipeline Dashboard with Analytics Charts • Analytics Interactions Developer Guide for Dashboards

Build, Customize, and Develop Tableau CRM Assets	
Embed and Customize Analytics	• Dashboard JSON Reference • Extended Metadata (XMD) Reference
Manage and Share Analytics in Apps	• Tableau CRM Templates Developer Guide
Develop on the Tableau CRM Platform	• Tableau CRM REST API Developer's Guide • Tableau CRM SAQL Reference • Tableau CRM SDK Developer Guide • Tableau CRM Dashboard Component Developer Guide • Salesforce Analytics CLI Plugin Command Reference

Accelerate with Prebuilt Analytics Apps and Templates	
Create Apps from Tableau CRM Templates: Start Here	
Sales Analytics	• Trailhead: Sales Analytics App
Service Analytics	• Trailhead: Service Analytics App
Field Service Analytics	
Event Monitoring Analytics App	• Trailhead: Event Monitoring
B2B Marketing Analytics App	

Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

SEE ALSO:

Explain, Predict, and Take Action with Einstein Discovery

Get Started Using Tableau CRM

With just a few clicks, you can run Tableau CRM apps, dashboards, and lenses.

Learn What You Can Do with Tableau CRM
Introduce yourself to the Tableau CRM way of data exploration.

Open Tableau CRM
Access your Tableau CRM home page from a tab within Salesforce. Alternatively, go to the Tableau CRM environment by opening the app menu or App Launcher in the Salesforce header and then finding and opening **Tableau CRM Studio**.

Get Oriented on Your Tableau CRM Home

With Tableau CRM, you can take either a business user path or a builder path, depending on what you want to do, and you can switch paths at any time.

View, Collaborate, and Take Action from Einstein Tableau CRM

View conversational explorations of your data, and personalize your Tableau CRM experience with saved dashboard views. Collaborate with features such as notifications, annotations, presentation mode, and downloading filtered data from Tableau CRM. Use custom menus in lenses and dashboards to take action in Salesforce directly from Tableau CRM.

Learning Resources

In addition to these Help pages, Tableau CRM has a variety of in-app assistance, and there's more educational content on Trailhead and other websites.

Tableau CRM Glossary

Familiarize yourself with common Tableau CRM terminology.

Learn What You Can Do with Tableau CRM

Introduce yourself to the Tableau CRM way of data exploration.

Exploring and Visualizing Your Data Interactively

Data exploration is an iterative process. It typically involves these steps: view, explore, refine, save, and share.

Collections of Data in Analytics Cloud

Tableau CRM collects and organizes your data in datasets, lenses, dashboards, and apps. These collections, also known as "Tableau CRM assets," represent levels of data refinement—from raw data that's uploaded from your source systems to highly curated, packaged views of your data.

Data Exploration Concepts

As you learn how to explore and visualize your data, it's helpful to review key concepts such as visualization, measure, and dimension.

What Is a Dashboard?

A Tableau CRM dashboard is a collection of widgets that work together to tell a data story from multiple angles. Depending on what you want the dashboard to show or how to behave, you can add different widgets, such as key performance indicators, charts, tables, filters, and images.

Key Elements of a Tableau CRM Dashboard

A dashboard isn't just for reading. It's for carrying on a conversation based on one or more datasets that matter to your company. Don't just look at the pictures—use the tools on the dashboard to dive deeper into the data that you care about. Go ahead, click around, and explore!

Query Your Data to Know Your Business

It's great to have a gut feeling about how well your business is doing. However, before you take action, test the feeling against your data. Ask your data questions.

SEE ALSO:

Converse with Your Data

Learn to Use Explorer

Build Tableau CRM Dashboards

Create and Share Tableau CRM Apps to Give Business Users a Big Data Picture

Exploring and Visualizing Your Data Interactively

Data exploration is an iterative process. It typically involves these steps: view, explore, refine, save, and share.
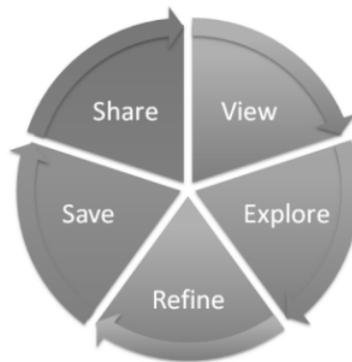
- View—View the data set.
- Explore—Get to know the boundaries and shape of your data. Play with different views, zoom in and zoom out, and see what you get with different charting options.
- Refine—Narrow your view to the most important data. Decide the level of detail and categorization that's easiest for your colleagues to understand. Choose the most appropriate chart visualization.
- Save—Save your work.
- Share—Share your view with others. For a more packaged presentation, use your visualization to build a dashboard.

Collections of Data in Analytics Cloud

Tableau CRM collects and organizes your data in datasets, lenses, dashboards, and apps. These collections, also known as "Tableau CRM assets," represent levels of data refinement—from raw data that's uploaded from your source systems to highly curated, packaged views of your data.

- A *dataset* contains a set of source data, specially formatted and optimized for interactive exploration.
- A *lens* is a particular view into a dataset's data. It's where you do exploratory analysis and visualization.
- A *dashboard* is a curated set of charts, metrics, and tables based on the data in one or more lenses.
- An *app* is a purpose-built set of analytics and answers about a specific area of your business. With apps, you can provide pathways through your data, plus powerful tools for spontaneous, deep explorations. After creating dashboards, lenses, and datasets, organize apps to present dashboards in relevant order, and then share apps with appropriate groups.

Data Exploration Concepts

As you learn how to explore and visualize your data, it's helpful to review key concepts such as visualization, measure, and dimension.

A *visualization* is commonly a chart or graph, such as a bar chart, donut chart, timeline, or heat map. It can also be data in tabular form, such as a comparison table or pivot table. Every visualization has an underlying query, which is how Tableau CRM retrieves information from the source data.

A *measure* is a quantitative value that contains numerical data like revenue and exchange rate. You can do math on measures, such as calculating the total revenue and minimum exchange rate. Measures have names (revenue) and values ($1,000,000). When you're viewing a chart visualization in Analytics Cloud, it's important to remember:

- The chart either shows a slice of your data based on the number of or amount of something, or it shows tabular data.

470

- A measure is typically *aggregated* in some way, which means that it's displayed with some math already applied to it. For example, when you first view a dataset, you often see a simple aggregation such as the count of the number of rows. You typically aggregate by a different method—sum, average, maximum, and so on—as you explore and change or add measures, but you always specify how you want to aggregate at the time when you select the measure.

- You can identify measures by their position (the far left items in the top left corner of a lens) and by the text that indicates the aggregation method (such as Sum of Revenue).

⚠ **Warning:** If you perform a query that aggregates measures—like sum or group by—and the resulting value exceeds the maximum for a numeric field (36,028,797,018,963,967), the value overflows and Analytics Cloud returns an incorrect result.

A *dimension* is a qualitative value that usually contains categorical data, such as Product Category, Lead Status, and Case Subject. Dimensions are handy for grouping and filtering your data. Unlike measures, you can't perform math on dimensions. Like measures, dimensions also have names (region) and values (northeast). Time is usually considered a dimension rather than a measure.

What Is a Dashboard?

A Tableau CRM dashboard is a collection of widgets that work together to tell a data story from multiple angles. Depending on what you want the dashboard to show or how to behave, you can add different widgets, such as key performance indicators, charts, tables, filters, and images.

When you share Analytics Cloud data via a dashboard, you're setting up your audience members to do some exploring of their own, without turning them loose on the whole data set.

One use of a dashboard is to provide a focus for your audience. A dataset can contain vast amounts of data. Some of that data is relevant to specific questions that your audience has; some of that data is irrelevant noise. Data that's critical to one set of questions is sometimes noise in relation to another set of questions.

Keep in mind that even though we call it a dashboard, we're actually doing much more with this interface than we do with a traditional dashboard in a car. This dashboard doesn't just tell us what's changed; it invites us to dig deeper into the underlying data. It's like using the speedometer in a car to tell us not just how fast we're going but how our speed has changed at various points in our trip, how traffic conditions ahead of us are changing, and how those changes affect our arrival time.

SEE ALSO:
  Build Tableau CRM Dashboards

Key Elements of a Tableau CRM Dashboard

A dashboard isn't just for reading. It's for carrying on a conversation based on one or more datasets that matter to your company. Don't just look at the pictures—use the tools on the dashboard to dive deeper into the data that you care about. Go ahead, click around, and explore!
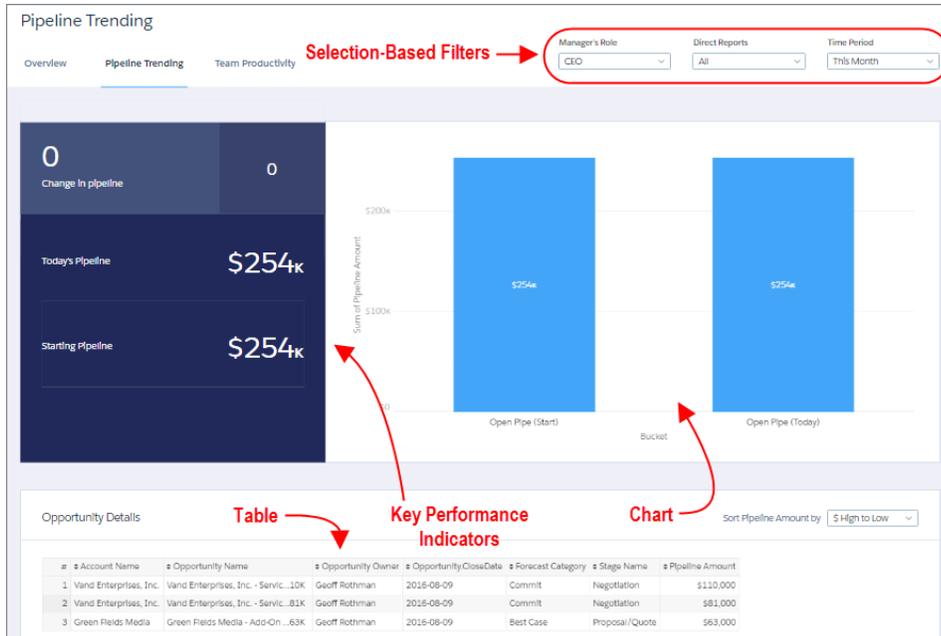
You read a Salesforce Tableau CRM dashboard interactively by clicking its displayed elements. Every element is a picture of a live, filtered query.

Selection-Based Filters

A selection-based filter lets you apply whatever filters you wish while viewing the dashboard. For example, set the minimum and maximum values in a range widget to make the dashboard show the amounts that fall within that range. These types of filters are often tied to each other so that you can progressively drill deeper. For example, use one filter to look at sales of a product. Then use the next to filter sales by region. Finally, use another to filter sales by account. The dashboard shows the results based on all three filters. Dashboard selections automatically reset each time you change a query grouping. There are different types of selection-based filter widgets: date, list, range, toggle, and charts.

Key Performance Indicators

Some information is best measured by a single number, also known a key performance indicator. Some number elements can be filtered based on selections in charts and filters. For example, in a bar chart that shows individual salespeople's pipelines, click a bar to focus on a particular person's pipeline. Notice that the number changes from showing the total value for all salespeople to show the value for the selected person.

Charts

A rich variety of charts shows you data from multiple angles. Many parts of a chart are interactive. For example, click a bar in a bar chart and watch what happens. Clicking a bar filters the query that the bar chart represents. Your revision changes other values in this chart, values in other charts, and number displays.

Tables

A table provides the record-level details. For example, a chart can show you a performance summary for each region, but the table can show you the opportunity-specific details. You can sort tables to show the top and bottom records. You can also create calculated columns in tables to compute values based on existing data.

Query Your Data to Know Your Business

It's great to have a gut feeling about how well your business is doing. However, before you take action, test the feeling against your data. Ask your data questions.

The easiest method for asking questions about data in datasets is to use the explorer. You ask questions in the form of queries. Use the left panel (1) of the explorer to create your query. You can perform calculations (referred to as measures), like sum of amount or average

case duration. To break down the measure across different areas, group it by descriptive fields (known as dimensions), like region or owner. To find the top or bottom performers, sort the measure in descending or ascending order. Then, if needed, apply filters to focus on a subset of the results.

The middle explorer panel (2) shows a visual representation of the query results. You can display the results as a chart or table. When in chart mode (📊), you can choose which chart type to use—Tableau CRM offers many. In table mode (▦), you can display the dataset records as a values table or create calculated fields and display them in a compare table. And, yes, pivot tables are also possible.

The right explorer panel (3) shows the visualization properties—configure them to modify the appearance of the chart or table. For instance, you can remove the legend, change the axis scale, or add a title. To view this panel, click ⚙.

If you're more advanced, write your own custom queries using Salesforce Analytics Query Language (SAQL) notation. In SAQL mode (>_), you have more flexibility to write robust queries, like ones that span multiple datasets or have modified query limits. For example, if the results are sorted in descending order, you set the limit to 5, and click **Run Query**, the query returns only the top five accounts.

Monitor Your Business Regularly with Tableau CRM Dashboards

Tableau CRM dashboards are populated with your data based on queries that you define. To continually get answers to common questions, save and organize your queries in interactive Tableau CRM dashboards. Each time you open the dashboard, the queries get the latest results based on the available data.

Answer Ad Hoc Questions with Tableau CRM Lenses

Dashboards are great for answering common questions, but sometimes you don't have a question in mind. You just want to poke around your data and ask questions as you notice things. To explore your data, open a dataset and create ad hoc queries. To revisit the query later, save it as a lens. The lens includes the query and the visualization—how the results appear. Like dashboards, lenses can be shared.

Monitor Your Business Regularly with Tableau CRM Dashboards

Tableau CRM dashboards are populated with your data based on queries that you define. To continually get answers to common questions, save and organize your queries in interactive Tableau CRM dashboards. Each time you open the dashboard, the queries get the latest results based on the available data.

As the sales executive, the start of your day might include a cup of coffee and the following sales leader dashboard. Each component is based on a query. For example, the timeline chart component shows the cumulative opportunity amount for each week of this quarter. The leaderboard in the left column ranks reps by sales (closed won opportunities).
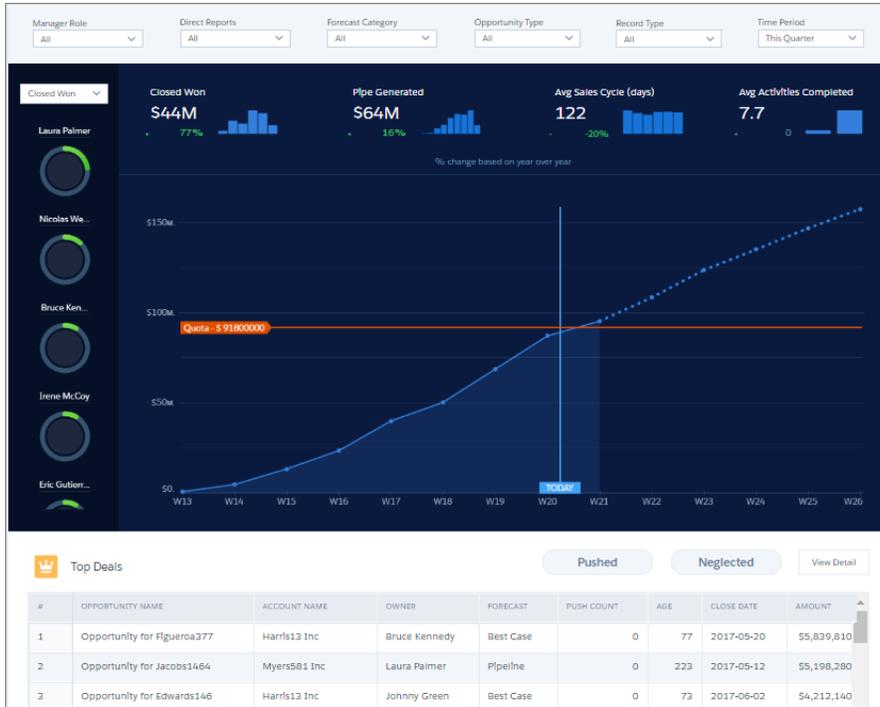
Because of built-in dashboard interactivity, you don't have to add more queries to ask follow-up questions. For example, to focus on an individual's performance, select the rep's donut chart in the leader panel on the left. Tableau CRM filters the dashboard results based on the selected rep. Behind the scenes, Tableau CRM modifies the underlying queries, filtering the results to get the specific rep's opportunities.

If the dashboard can't be used to answer a question, hover over a dashboard component and click **Explore** to create a new query based on the same dataset.
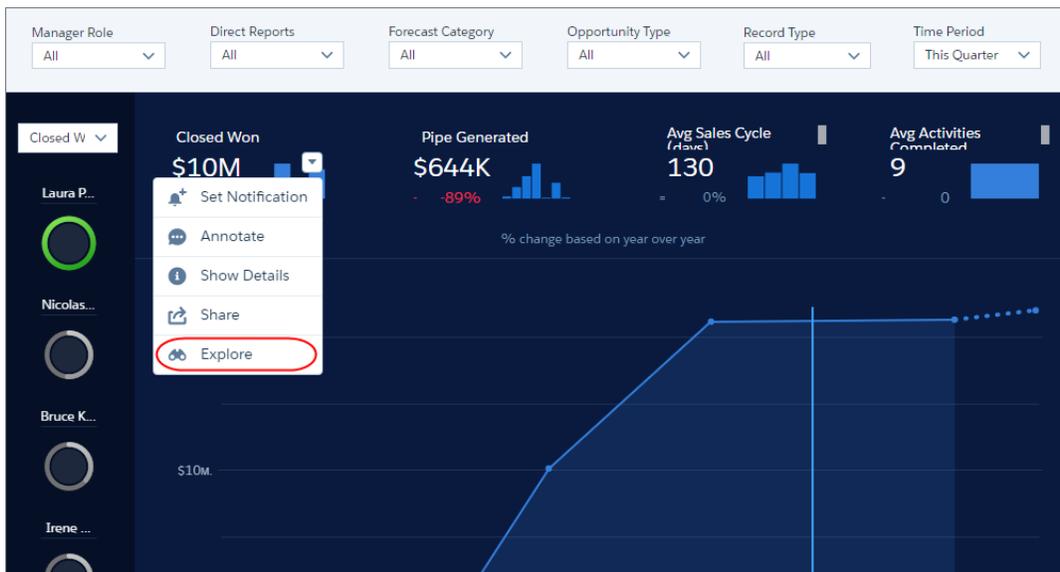
Answer Ad Hoc Questions with Tableau CRM Lenses

Dashboards are great for answering common questions, but sometimes you don't have a question in mind. You just want to poke around your data and ask questions as you notice things. To explore your data, open a dataset and create ad hoc queries. To revisit the query later, save it as a lens. The lens includes the query and the visualization—how the results appear. Like dashboards, lenses can be shared.

When you open a dataset, the explorer shows the number of dataset rows as a bar chart, by default.
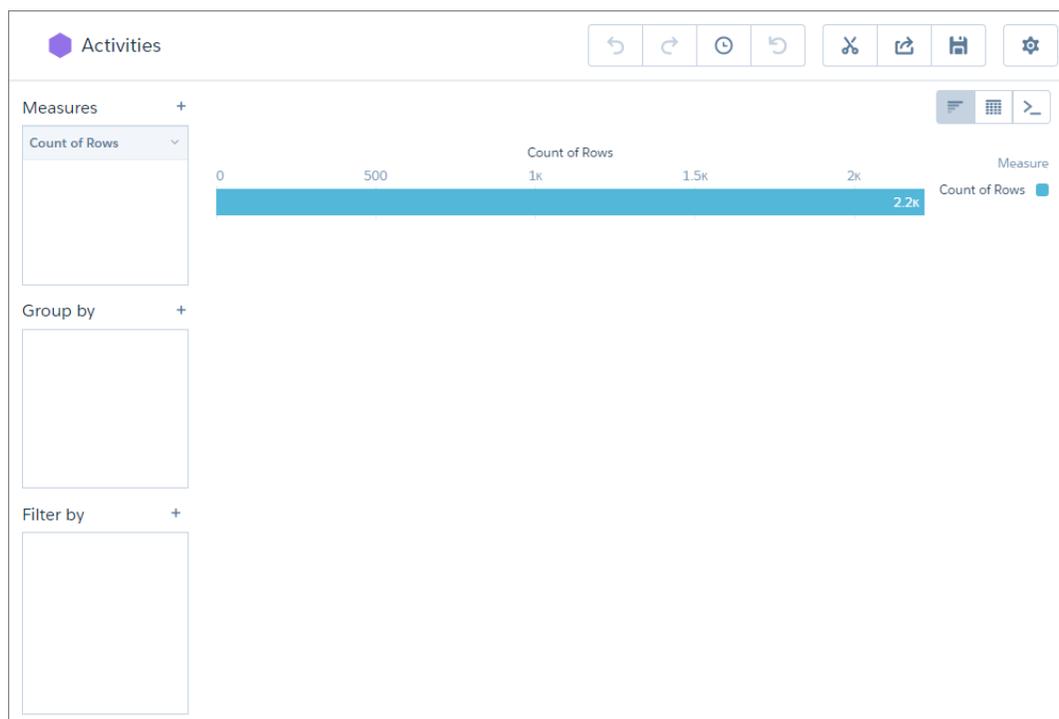
To change the query, add measures, groupings, and filters to the Measures, Group by, and Filter by boxes. To represent the results as a different chart type, click ▥. To represent the results as a table or to create calculated fields based on existing fields, click ▦. If you're more advanced and want a more robust way to create SAQL queries, click ⟩_ to manually enter the query.

To add a query to a dashboard, click ✂ to clip it.

SEE ALSO:

[Explore and Visualize Your Data in Tableau CRM](#)

Open Tableau CRM

Access your Tableau CRM home page from a tab within Salesforce. Alternatively, go to the Tableau CRM environment by opening the app menu or App Launcher in the Salesforce header and then finding and opening **Tableau CRM Studio**.

To start in the Tableau CRM Studio, from the App Launcher, find and open **Tableau CRM Studio**. To start in the Analytics tab, find and open the **Analytics** item. Each environment allows easy movement to the other. When you opened the App Launcher, if you clicked **View All**, you can click the **Tableau CRM Studio** tile (1) or the **Analytics** item (2).
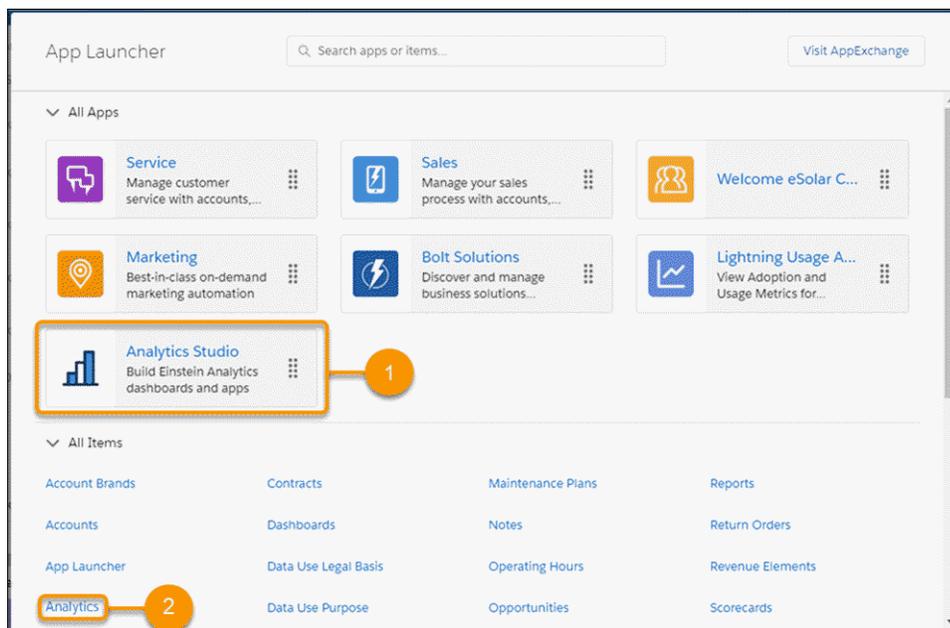
If you don't see the **Analytics** tab, ask your admin to set it up like any new tab.
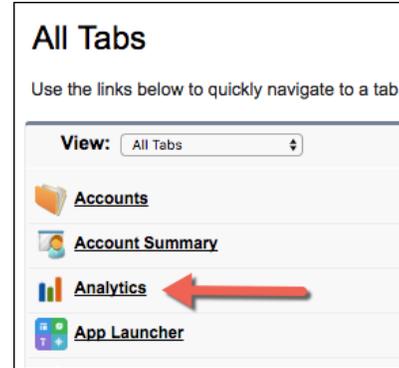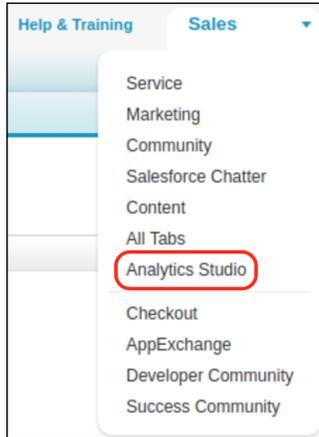
Get Oriented on Your Tableau CRM Home

With Tableau CRM, you can take either a business user path or a builder path, depending on what you want to do, and you can switch paths at any time.
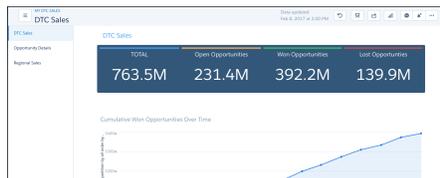
Salesforce Tab	**Two Paths**	**Tableau CRM Studio**
Stay in Salesforce		Go to Analytics > ⠿ >
Run and explore apps		Browse and build assets
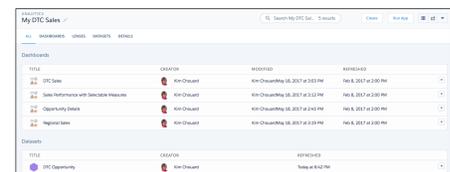
The business user path, generally for those with Use Analytics permission, leads to ready-to-explore dashboards and lenses in curated apps. For this path, work in the Tableau CRM tab to run apps and collaborate with coworkers over dashboards.

The builder path, generally for those with permission to create Tableau CRM assets, leads to ready-to-explore datasets and ready-to-edit apps, dashboards, and lenses. For this path, work in the Tableau CRM Studio to browse your assets and build new ones.

This table summarizes the differences between the Tableau CRM tab in Salesforce and the Tableau CRM Studio.

Action	Salesforce Tab (Run and Collaborate)	Tableau CRM Studio (Browse and Build)
Create		✅

Action	Salesforce Tab (Run and Collaborate)	Tableau CRM Studio (Browse and Build)
Annotate	✅	✅
Set Notification	✅	✅
Favorite	✅	✅
Edit		✅
Full Screen	✅	✅
Clip to Designer		✅
Clone in New Tab		✅
Save		✅
Share	✅	✅
Delete		✅

This table details the differences in behavior and functionality between the Tableau CRM tab in Salesforce and the Tableau CRM Studio.

Tableau CRM Asset	Salesforce Tab	Tableau CRM Studio
Apps		
What happens when you click an app name or icon on the home page	Runs app	Browses app
App actions on home page tile or list entry	Browse; Share; Open in Tableau CRM Studio	Run App; Share
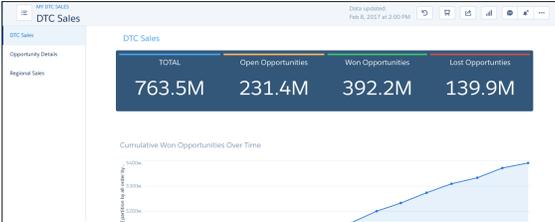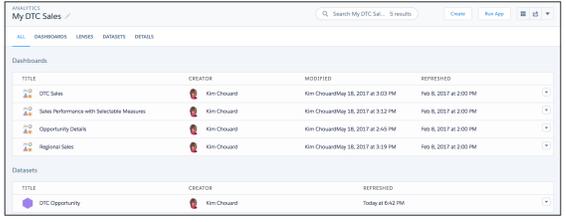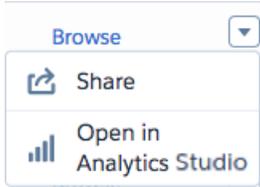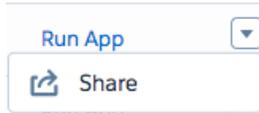
Tableau CRM Asset	Salesforce Tab	Tableau CRM Studio
Actions on the Browse App page	Run App; Tile/List View; Share	Create; Run App; Tile/List View; Share; Delete
Dashboards		
Dashboard actions on home page tile or list entry	Share; Open in Tableau CRM Studio	Edit; Share; Delete
Lenses		
Lens actions on home page tile or list entry	Share; Open in Tableau CRM Studio	Share; Delete
Datasets		
Dataset actions on home page tile or list entry	N/A	Edit
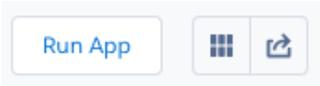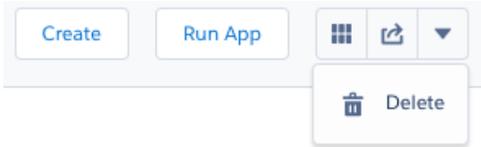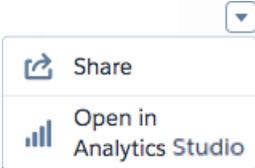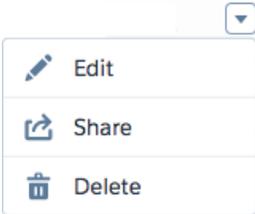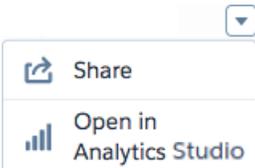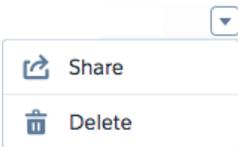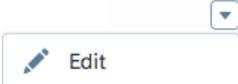
Note: Actions are available depending upon user permissions and app sharing access.

Find Your Tableau CRM Assets

Your Tableau CRM home page provides easy access to apps, dashboards, and lenses, and a variety of filter and search options help you find those Tableau CRM assets fast.

Run Apps

Tableau CRM apps are where you run dashboards and lenses.

Favorite Assets in Tableau CRM Home

Tableau CRM assets can be favorited for quick access. Favorite a Tableau CRM asset by clicking the star in the global header or use the row level action drop down menu.

Monitor Important Metrics with the Tableau CRM Watchlist

The Tableau CRM Watchlist gives users the advantage of capturing and tracking up to 20 KPIs across different dashboards, plus historical trending. No more switching between dashboards to see what has changed, and track your metrics all in one place.

Track Notifications

Quickly monitor important changes in your data by tracking the status of your notifications.

Find Your Tableau CRM Assets

Your Tableau CRM home page provides easy access to apps, dashboards, and lenses, and a variety of filter and search options help you find those Tableau CRM assets fast.

Lists of Tableau CRM assets are arranged in order of relevance. Click the list view icon (⊟) to view assets with their descriptions and additional information. Click the tile view icon (▦) to view tiles with thumbnail images of dashboards and lenses.
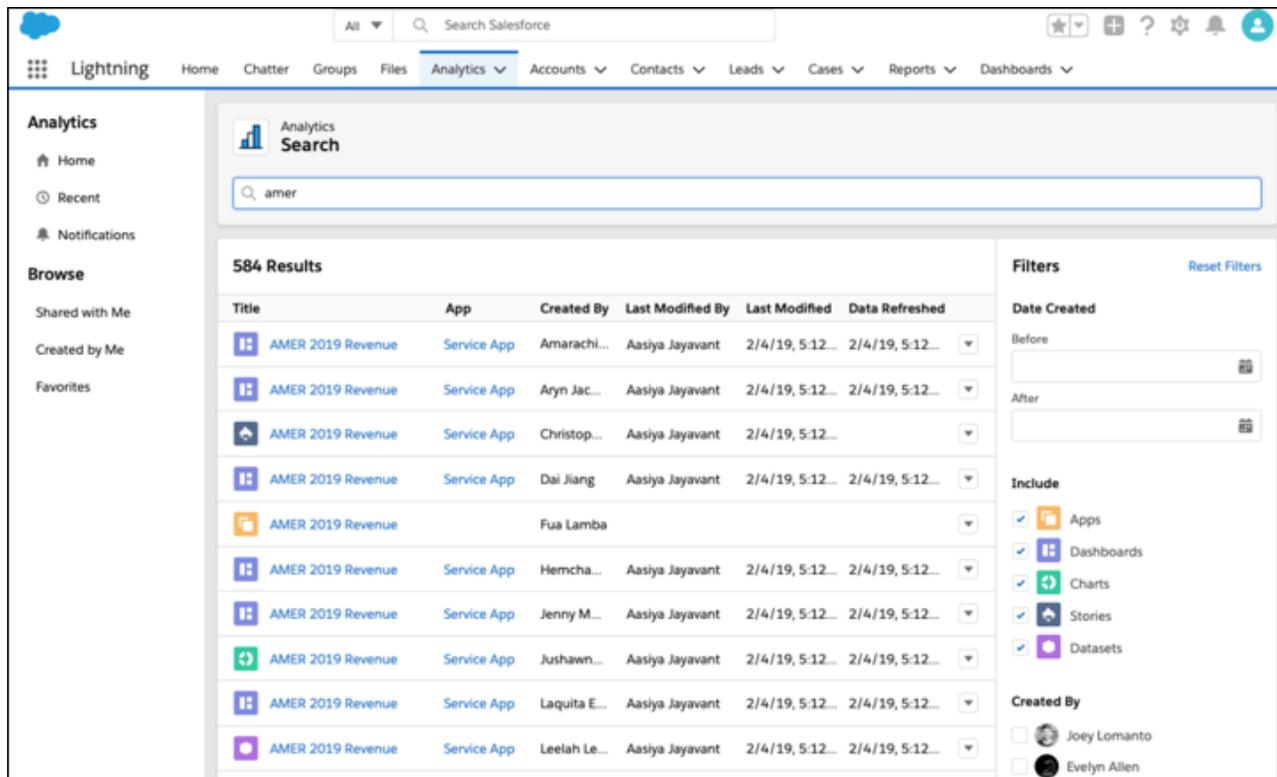
All views, whether tiles or list, include one-click access to both run and browse actions, as well as other actions if appropriate for the asset. Relevant timestamps are provided, depending on the asset. For lenses and dashboards, Data Refreshed is the last time the asset's dataset was updated. If a dashboard contains components that reference more than one dataset, the Data Refreshed timestamp will reflect the most recently updated dataset.

Filter using **Created by Me** and **Shared with Me**, as well as the **Apps**, **Dashboards**, and **Lenses** subtabs (and the **Datasets** subtab in Tableau CRM Studio). The count of items reflects the filters you selected (1), and the search box searches only filtered items (2).
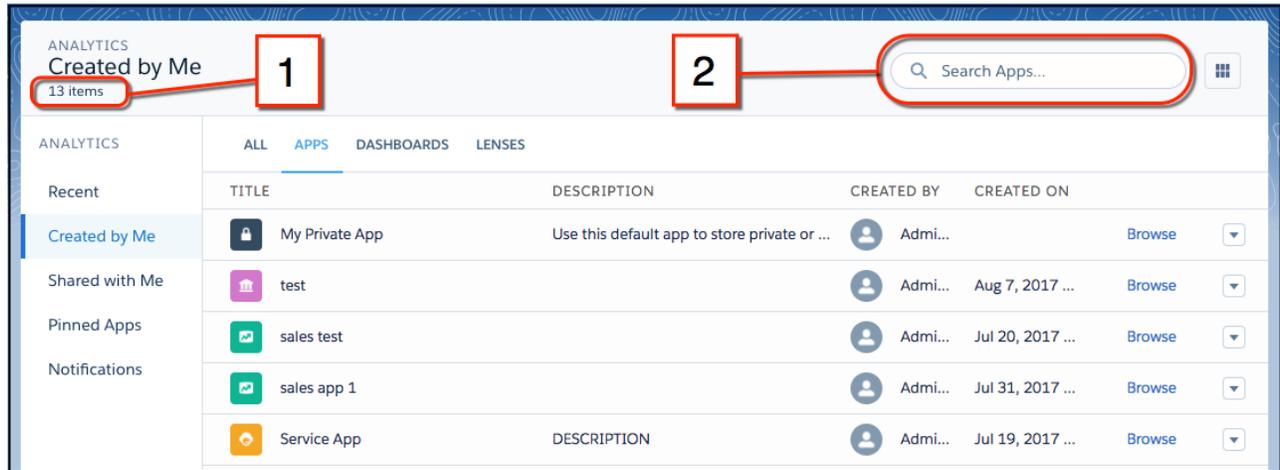
If you're working in Lightning Experience, you can use the global search box to quickly find and view lenses and dashboards. The search box dropdown list includes Tableau CRM assets among its suggestions. Selecting an asset from the list runs it in the Tableau CRM tab.
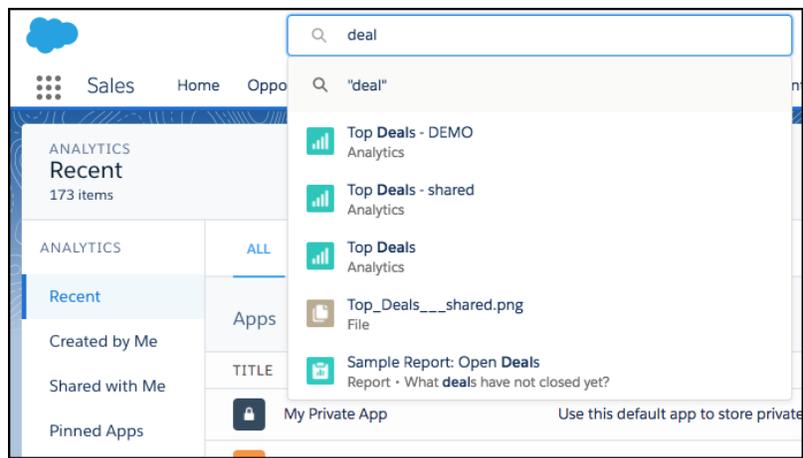
Or start typing and press Enter to see a page of found resources. To narrow the search results to only Tableau CRM lenses and dashboards, click **Tableau CRM** in the left panel. From the results, you can click to run either the asset or its app.

Run Apps

Tableau CRM apps are where you run dashboards and lenses.

In the Tableau CRM tab, click an app to run it. In Tableau CRM Studio, click **Run App**. Running an app opens it in presentation mode, displaying the first dashboard or lens on your navigation list.

To view other assets in the app, click the arrow ▼ next to the name.

When the dropdown menu is open, click any asset in the list. Dashboards and lenses let you explore live, interactive views of your data.

When you want to return the asset to its initial state, click the Reset icon (↺).

In addition, from the running app you can:

- Present full screen
- Share
- Set and manage notifications
- Annotate dashboard widgets
- View exploration history
- Open dashboards in the dashboard designer in order to edit them on page 1198 (if you have permission and access)

Favorite Assets in Tableau CRM Home

Tableau CRM assets can be favorited for quick access. Favorite a Tableau CRM asset by clicking the star in the global header or use the row level action drop down menu.
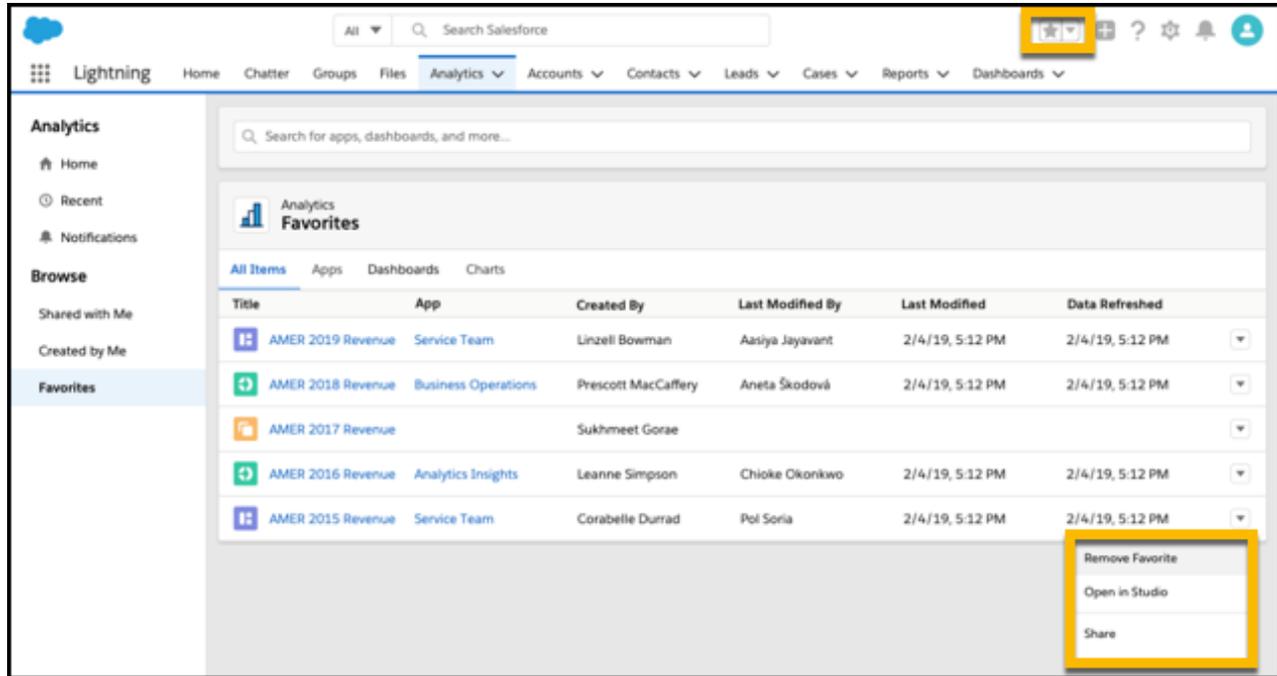
Favorites now include all your pinned apps. When you first click **Favorites** on Tableau CRM home, Tableau CRM imports your pinned apps for you.

Monitor Important Metrics with the Tableau CRM Watchlist

The Tableau CRM Watchlist gives users the advantage of capturing and tracking up to 20 KPIs across different dashboards, plus historical trending. No more switching between dashboards to see what has changed, and track your metrics all in one place.

Note: Enable Dashboard Views on page 561 must be on to use Watchlists. The feature is enabled by default. Review the referenced document for more information about the feature.

1. Access the Watchlist. The Watchlist is found in the My Analytics section of the Home page and the Watchlist tab. The Watchlist within My Analytics contains a snapshot of your metrics including current value, trending, and most recent update. The Watchlist tab (1) on the left of the Home page takes you to your full, personalized list of metrics. Click a metric to view its values (2), and timeline chart (3).
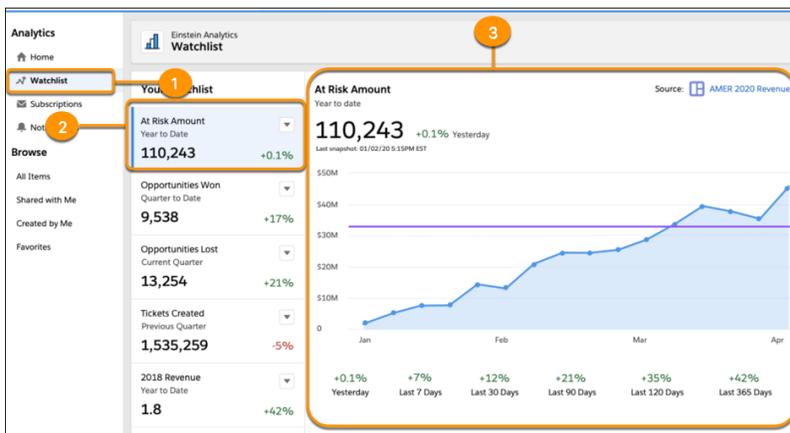
EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

2. Add a dashboard item to the Watchlist. Each user can save up to 20 metrics to their Watchlist.

   a. Open the dashboard that contains the item to watch.

   b. Locate the item's dashboard widget.

   c. Click the widget dropdown menu and click **Add to Watchlist**.

   d. Type in name for the Watchlist metric.

   e. Under **Change Format**, click the radio button to define how changes are tracked.
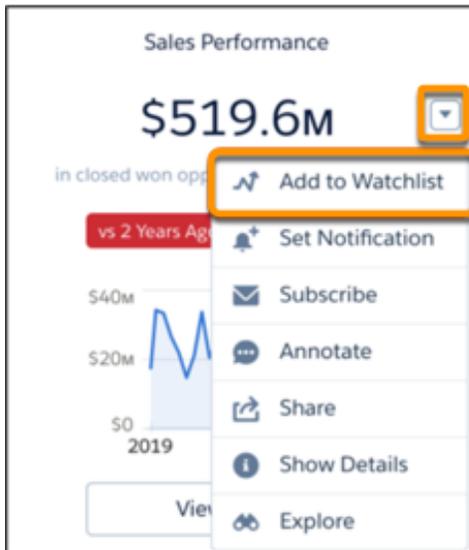
   f. Click **Save**.

3. Delete a Watchlist metric. You have the control to remove metrics from the Watchlist when they're no longer needed. Metrics can be deleted using the Watchlist tab or snapshot view in My Analytics.

   a. On the Watchlist, click the metric dropdown menu.

   b. Click **Delete** from the menu.

   c. Click **Delete** on the confirmation screen.
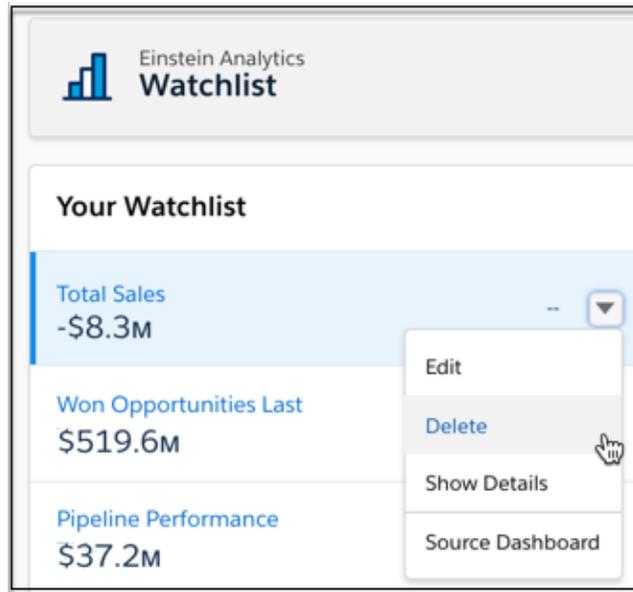
4. Edit a Watchlist metric. Watchlist metric properties can be modified at any time.

   a. On the Watchlist, click the metric dropdown menu.

   b. Click **Edit** from the menu.

   c. Make changes to the metric.

   d. Click **Save**.

5. Get item information from a Watchlist. Watchlist metrics also include last snapshot date, snapshot frequency, date metric was added, and source dashboard.

   a. On the Watchlist, click the metric dropdown menu.

   b. Click **Show Details** from the menu.

   c. Click the **X** to close the Show Details window when finished.

6. Changing metric order. You can organize Watchlist metrics in any desired order. This is especially useful when viewing metrics from My Analytics.

   a. Click the Watchlist tab from Tableau CRM home.

   b. Click **Change Order** at the bottom of the metrics listing.

   c. Click and hold a metric, and drag it to its new position.

   d. Release the metric.

   e. Click **Save**.

Manage and Track Your Metrics: Tips, Limits, and Limitations

Add and monitor dashboard items on the Watchlist while keeping these tips and limitations in mind.

- Watchlist metrics can only come from dashboards you have access to.

- The Watchlist can only display an item's metric value starting from the time it was added.

- If access to a dashboard is lost, then restored, your metric remains on the Watchlist but it doesn't fill in any gaps.

Track Notifications

Quickly monitor important changes in your data by tracking the status of your notifications.

Click the **Notifications** filter on the left of your Tableau CRM home.

If you don't have any notifications yet, navigate to a dashboard with a number or chart you'd like to follow. Dashboard widgets are where you set notifications. Hover over the upper right corner of the widget, locate the dropdown action list, and select **Set Notification**. For more information, see Set and Send Smart Notifications.

You can set active notifications to appear on your Tableau CRM home page. The notifications tracking gauges and bar charts are designed to show the progress toward the conditions set in your notification.
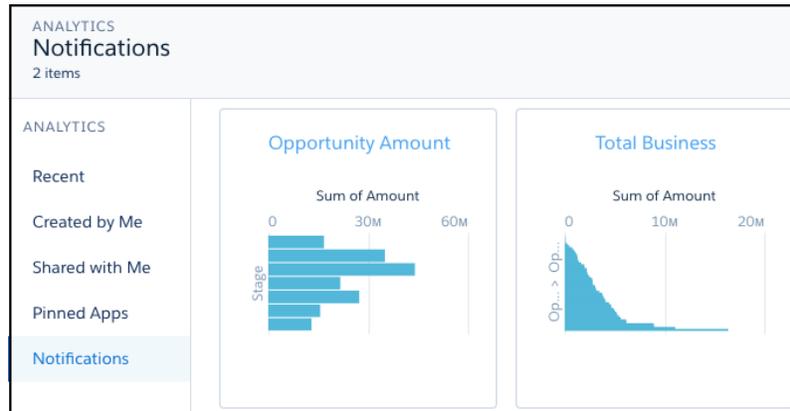
View, Collaborate, and Take Action from Einstein Tableau CRM

View conversational explorations of your data, and personalize your Tableau CRM experience with saved dashboard views. Collaborate with features such as notifications, annotations, presentation mode, and downloading filtered data from Tableau CRM. Use custom menus in lenses and dashboards to take action in Salesforce directly from Tableau CRM.

Converse with Your Data

Ask data questions using non-technical language, and view answers in automatically configured charts. Start with a common word or two, and Tableau CRM provides relevant suggestions that you can accept or edit.

Save Your Filtered View of a Dashboard

No need to reselect the filters every time you open a dashboard because you don't have permission to save the selections permanently. Change the filters and save your own view. Only you can see your views—other users still see the original dashboards.

Set and Send Smart Notifications

Set up Tableau CRM to notify you about changes to your important business metrics, when and where you choose. Attach notification criteria to widgets in dashboards, and select when to run the queries. You can see notifications in Tableau CRM, Lightning Experience, Tableau CRM for iOS, and email. Add recipients from your org to the email's distribution list to notify colleagues at the same time as yourself.

Get Scheduled Updates with Email Subscriptions

Subscribe to lenses and dashboard widgets and receive a daily or weekly email with updates on your most important charts and metrics, and table data in `.csv` attachments.

Collaborate with Dashboard Annotations

Annotate dashboard widgets with comments posted in the dashboard and in Chatter. With annotations, you can hold conversations about your data and how it's visualized, with the dashboard right there for reference.

Present Live, Interactive Dashboards

Conduct meetings directly from Tableau CRM by presenting dashboards in running apps. With dashboards instead of slides with static images, you have real-time access to your data, and you can showcase dynamic visualizations.

Share Dashboards, Widgets, and Lenses

Share a dashboard, widget, or lens with your colleagues by posting it to Chatter or copying its unique URL. A Chatter post provides an image and a link to the asset in Tableau CRM. Colleagues with the link and access to the asset can drill in and explore the information that's presented. You can also export snapshot images of Tableau CRM assets to Quip. An exported image is published to a new Quip slide deck.

Download Tableau CRM Images and Export Filtered Data

Download filtered data from lens explorations and dashboard widgets. Download formats include image (`.png`), Microsoft® Excel® (`.xls`), and comma-separated values (`.csv`) files. This feature downloads the results of a displayed query (or step).

Create PDF Files and Print Your Dashboards

Dashboards are printer-friendly, and you can print or save PDF files of dashboards from your browser.

Perform Salesforce Actions Directly from Tableau CRM

Take action on your insights directly from Tableau CRM. Perform custom and mass quick actions from charts and tables in lenses and dashboards.

Converse with Your Data

Ask data questions using non-technical language, and view answers in automatically configured charts. Start with a common word or two, and Tableau CRM provides relevant suggestions that you can accept or edit.

To start a conversation:

1. For a dashboard, click ☀ to open the conversations panel. (For a lens, the conversations text entry box is always open.)

2. Click in the text entry box.

3. Select a suggestion, or type your own question.

4. (Optional) When you have the chart that answers your question, click an action icon or the dropdown menu in the upper right corner. Available actions, such as clip, explore, save, and share, depend on the context.
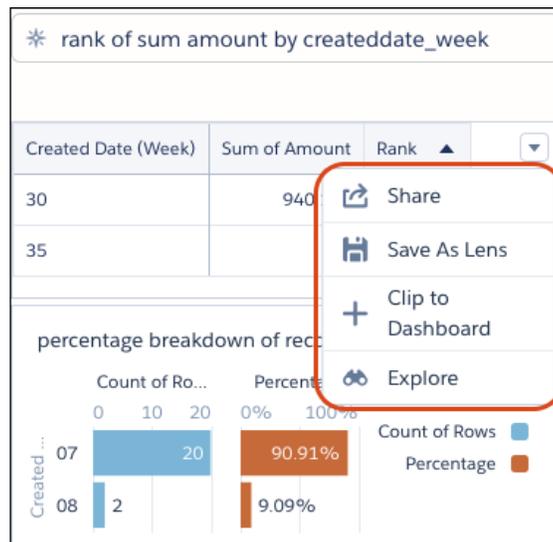
How does it work? The conversational auto-suggestions are based on fields in the available datasets. To see which datasets are open for conversational exploration, look for the purple hexagons (⬡). For a lens, click **Fields** to view the available fields to ask about. For a dashboard, if the suggestions don't include the words you expect, click **Add Dataset** to expand your exploration.

What kinds of questions can you ask? Question are built with measures and dimensions from your data combined with keywords, such as "show me" and "by." Typically, measures are aggregated and dimensions are grouped to show meaningful information. Filters are applied to narrow the focus of the question.

Example Conversational Queries

Aggregations and Limits	Measures	Dimensions	Filters
top 3		account owner	
average	CSAT		this week (Created Date)
bottom 5	amount	by account owner and stage	for closed equals (true)
	unit price and shipping cost	by product family	between 10000 and 100000
		show me industry	
		agent and case origin	
percentage change		by segment	previous cy (Close Date)
	revenue, csat	by agent, state, product	
percentage breakdown of	amount	by opportunity name	
rank	number of records	by industry	for industry equals (Agriculture, Biotechnology, Communications)
max	number		for account type contains (cust) and amount > 10000

What if my question doesn't include a measure? When no measure is specified, the response is aggregated by count of rows.

What if I don't get a chart? Sometimes your question isn't phrased properly and you see a red error message. In that case, try using the auto-suggestions to help rephrase your question. Sometimes you've asked a valid question but no results are returned. Try a wider filter or a different group. For more about what to type, see the following tables.

Elements of Conversational Exploration

Question Elements	Keywords and Symbols
Measures (numerical or aggregated)	count, number, sum, avg, max, min, median, first, last, unique, top, bottom, stddev, stddevp, var, varp, percentage breakdown, percentage change, rank
Groups	by, and
Filters	(By **dimension** values) for, in/equals, not in/not equals, contains, top, bottom, without, filter out, exclude (for multiple values, use commas and parentheses, for example "(v1, v2, v3)") (By **measure** ranges) with, equal, =, greater than, >, less than, <, between, top, bottom

Question Elements	Keywords and Symbols
	(By relative **date** ranges) today, tomorrow, yesterday, current/next/previous/last, year, quarter, month, week, day (for full list, see Relative Date Filters)
Question words	show me, show the, what is, what are
Shortcuts that depend on your dataset	deal, lead conversion, time to close, win rate (for more information, see Shortcuts)

Relative Date Filters

Relative date units	Keywords
Fiscal Year	Current FY, Previous FY, Previous 2 FY, 2 FY Ago, Next FY, Current and Previous FY, Current and Previous 2 FY, Current and Next FY
Fiscal Quarter	Current FQ, Current and Next FQ, Current and Previous FQ, Next FQ, Previous FQ, Current and Next 3 FQ
Calendar Year	Current CY, Previous CY, Previous 2 CY, 2 CY Ago, Next CY, Current and Previous CY, Current and Previous 2 CY, Current and Next CY
Calendar Quarter	Current CQ, Current and Next CQ, Current and Previous CQ, Next CQ, Previous CQ, Current and Next 3 CQ
Calendar Month	Last Month, This Month, Next Month, Current and Previous Month, Current and Next Month
Calendar Week	Last Week, This Week, Next Week
Day	Yesterday, Today, Tomorrow, Last 7 Days, Last 30 Days, Last 60 Days, Last 90 Days, Last 120 Days, Next 7 Days, Next 30 Days, Next 60 Days, Next 90 Days, Next 120 Days

Shortcuts

Keywords	Functions	Requirements	Examples
deals	sum Amount by opportunity Name	Opportunities dataset that includes "name" dimension and "amount" measure	"top 10 deals last month" or "show me deals that have moved out this quarter (createdDate)"
lead conversion rate	total leads divided by converted lead ("isConverted"=true)	Leads dataset that includes "IsConverted" dimension	"lead conversion rate by rep last month (Created Date)"
closed, won, lost (filters)	"isClosed"=true; "isWon"=true; "isWon"=false	Opportunities dataset that includes "isClosed" and "isWon"	"amount won last month (Close Date)"

> **Tip:** If you're curious about how a question is translated into a query, open the chart in the explorer and click SAQL Mode (>_).

Take the following into consideration.

* Conversational exploration requires the Tableau CRM Platform permission set license. It isn't available in embedded dashboards, mobile apps, Experience Cloud sites, or standalone Tableau CRM apps, such as the Sales Analytics app.

* Admin setup is required.

* Conversational exploration is available only in the dashboard designer, the explorer, and running apps.

* Some advanced functions aren't supported.

* The time zone feature isn't supported.

* If you have more than one dataset open in explorer, conversational exploration is disabled.

SEE ALSO:

> Enable Conversational Exploration
>
> Secure Image Sharing and Downloading
>
> Explore and Visualize Your Data in Tableau CRM

Save Your Filtered View of a Dashboard

No need to reselect the filters every time you open a dashboard because you don't have permission to save the selections permanently. Change the filters and save your own view. Only you can see your views—other users still see the original dashboards.

You can create up to 10 views per dashboard and choose one as the default that shows when you open the dashboard. All your saved views are listed in the view menu.
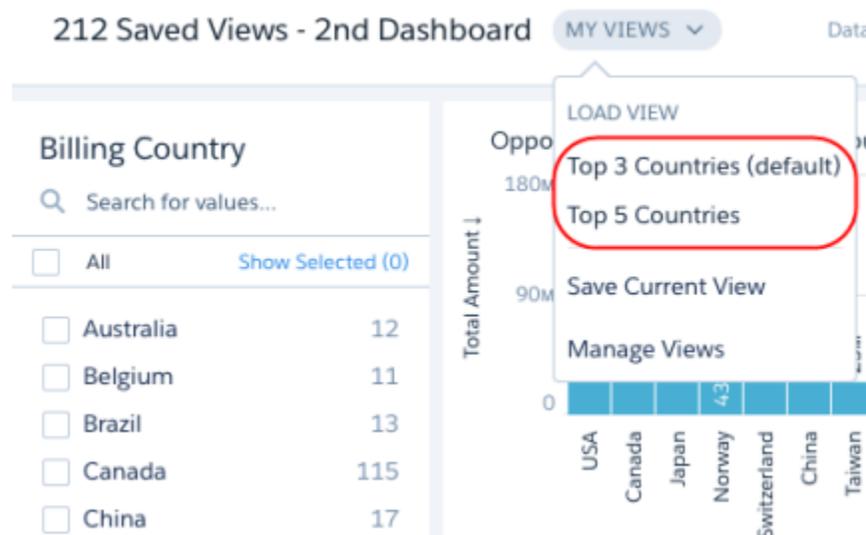
1. To create a view, open the dashboard in view mode, make your selections, and click **Save View**.

   The saved view includes all selections that you made to the current page in list, toggle, range, date, chart, compare table, and global filter widgets.

   > **Note:** In the Tableau CRM mobile app, you can access saved dashboard views, toggle between saved views, and share links to different views, but managing and saving views aren't supported.

2.
To remove your changes to a dashboard or view, including selections, click ↺ .

When you open a view and change a selection, an asterisk appears next to the view name to indicate that it's been modified. When you change a selection while viewing a dashboard without a view, Modified appears. In either case, you can save the current selections as a new view or overwrite an existing one.

3. To remove the view applied to a dashboard so you can view the original dashboard, click **Clear View** in the view menu.

4. To set your default view or delete a view, click **Manage Views**.

5. To share your view with others, open the view and select **Share** > **Get URL**.
When users open the link, they see your view of the dashboard. If they want, they can save it as one of their own views.

Considerations When Opening a Dashboard View
There are special cases when a view doesn't open or isn't applied completely.

Considerations When Opening a Dashboard View

There are special cases when a view doesn't open or isn't applied completely.

If you open a dashboard that has external selections or filters, the default view isn't applied. If you open a view, the external selections and filters are removed. If steps or widgets in your dashboard have errors, views might not be applied properly.

A dashboard maintains existing selections and filters when you use a link or navigation widget to switch between pages of a multi-page dashboard. If you use a saved view to switch between pages, any previous selections and filters are removed and the state from the saved view is applied.

If you open a view and Tableau CRM can't display all the criteria defined in a view, it displays what it can and shows a warning next to the view name.
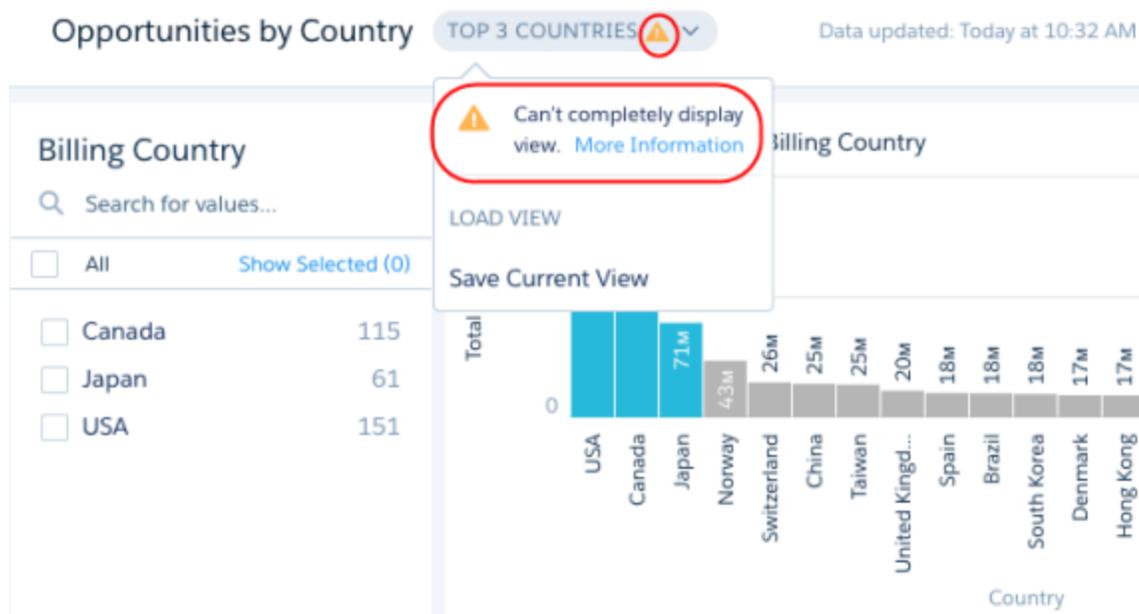
Tableau CRM can't display all view criteria for the following cases.

- Tableau CRM ignores selections and filters that no longer apply to the dashboard. For example, Tableau CRM ignores references to datasets, fields, field values, and steps that no longer exist. These issues arise when the dashboard or data changes but the view isn't updated.

- If a view includes a selection on a widget that doesn't exist in a layout, Tableau CRM ignores the selection. A view applies to all layouts defined in the dashboard.

- If a view is created based on a dashboard page that has since been deleted, Tableau CRM shows the view based on the first page defined for the dashboard.

Set and Send Smart Notifications

Set up Tableau CRM to notify you about changes to your important business metrics, when and where you choose. Attach notification criteria to widgets in dashboards, and select when to run the queries. You can see notifications in Tableau CRM, Lightning Experience, Tableau CRM for iOS, and email. Add recipients from your org to the email's distribution list to notify colleagues at the same time as yourself.

To set up a Tableau CRM notification, follow these steps.

1. Locate a dashboard widget that shows information you want to be notified about. You can set notifications on any chart except timeline or values table. If needed, you can use the dashboard's selectors to create a more personalized query of your data.

2. Select **Set Notification** from the widget's dropdown list.
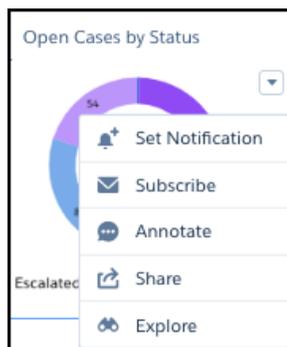
3. In the Notifications panel, enter a name for your notification, if needed.

4. Set the value that you want to know about as soon as your widget query returns it. Depending on the chart, the panel provides relevant options to select for creating a notification. You can include up to four groupings. If the chart has multiple measures, select the one you want from the list of options.

   For example, if you wanted to know when your team has reached 75% of quota, you'd click **Set Notification** on the Quota Attainment number widget. In the Notifications panel, you'd select **Equals or is greater than** from the dropdown list and enter 75 for the threshold value.

   Notifications read roughly like a sentence. For example, "Notify me when any channel's sum of open cases is less than 20." Or, "Notify me when any product or reason count of cases is greater than 30."

5. Select the frequency and time when you want the query to run.

6. If the dashboard is in a shared app, then you can add people to your email distribution list. Click **Edit Recipients** to add users who have access to the dashboard.

7. Test your notification on the spot with **Save and Run**, or simply click **Save**.

A tracking tile appears in the Notifications section of your Tableau CRM home page.
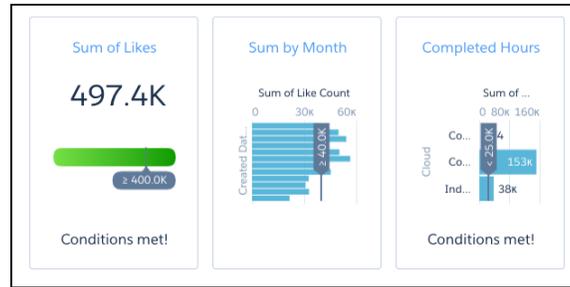
When the conditions you've set for the notification are met, you get the message on the tile. You're also notified via email, and in Tableau CRM and Lightning Experience, the bell icon (🔔) alerts you with a number in red.

In Tableau CRM, the dropdown list from the bell includes the option to open the Manage Tableau CRM Notifications dialog box. You can also manage notifications on the dashboard. The bell icon with the plus (🔔⁺) opens the All Notifications panel and shows numbers in blue (1) on widgets that have notifications. In the All Notifications panel, each notification has a dropdown action list for editing and deleting the notification.
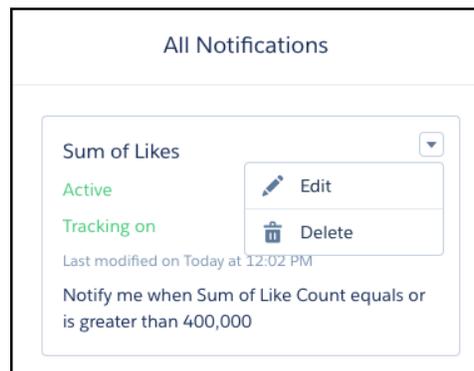
Clicking the link in a notification email, the notification manager, or a notification tile on your Tableau CRM home page, takes you to the dashboard view relevant for the notification. You immediately see the dashboard page with your filters and selections, and with the notification set up in the side panel. If you want to make any changes, click the **Edit** button. In edit mode, you can also activate or deactivate the notification, and select whether to show the tracking tile on the Tableau CRM home page.

Take note of these considerations for Tableau CRM notifications.

- Each user can set up to 10 notifications.
- If a notification is created in the Analytics Studio, the link in the notification email opens the dashboard in Analytics Studio. If a notification is created on an embedded dashboard or in Analytics Tab, the link in notification email navigates the user to the Analytics Tab.
- Notification tracking is available in the Tableau CRM app for iOS, but it isn't available in the Tableau CRM app for Android or in the Salesforce mobile app.
- Notification tracking tiles use flat gauge or bar chart visualizations to show progress toward the conditions set in the notifications. The tiles are separate from the dashboard and don't use the dashboard's chart types and widget properties.
- Notifications are available on all widgets except timeline charts and values tables.
- For queries with groupings, all results are evaluated for the notification criteria, not just the first row.
- Notifications for widgets with queries based on results bindings aren't supported, including dynamic widget titles.
- Notifications for widgets based on SOQL, Apex, Salesforce Direct, and Tableau CRM Direct queries aren't supported.

- If a notification is on a widget based on a SAQL query, the filters aren't shown in the Notifications panel.

- Adding recipients doesn't create personal notifications for them. The added recipients don't get in-app notification via the bell icon, and they don't see the notification on their home page. Recipients get the notification email sent from the email address of the user who set up the notification.

- Once added to the notification email distribution list, recipients can't remove themselves and must ask the email sender to be removed.

Get Scheduled Updates with Email Subscriptions

Subscribe to lenses and dashboard widgets and receive a daily or weekly email with updates on your most important charts and metrics, and table data in `.csv` attachments.

To set up an email subscription, follow these steps.

1. Locate a dashboard widget or a lens with information you want delivered to you with regular email updates.

    - For a widget, you can use the dashboard's filters and selectors to create a more personalized view of your data.

    - For a lens, be sure to save before subscribing.

2. Select **Subscribe** from the menu on a dashboard widget, or from a lens, click the **Show Subscriptions** icon (✉).

3. The first time you subscribe, you're asked to set the schedule for your subscription email. Select the frequency and timing, and click **Save**.

4. Enter a title for the subscription, and click **Subscribe**. The subscription appears in the Subscriptions panel.

    That's it! All subscriptions arrive together in one email.

Here's more you can do with subscriptions.

- When viewing a dashboard, click the envelope (✉) to open the Subscriptions panel and show numbers in blue (**1**) on widgets that have subscriptions.

- To view the filters on a widget subscription, click the subscription tile in the dashboard Subscriptions panel.
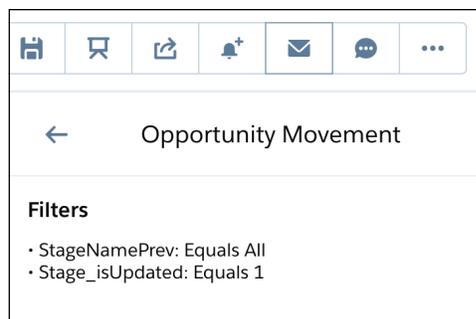
> **Note:** Filters in SAQL queries aren't listed in the Subscriptions panel but are referenced as "custom."

- To see a preview of your subscription, go to the Tableau CRM home page and click **Subscriptions**. In the Subscription Preview, click the title of a subscription to go to its dashboard or lens.

- In the Subscription Preview, you can arrange the order of your subscriptions. Newly subscribed widgets and lenses are added last by default. To reorder subscriptions, use the arrow buttons and menu actions. You see the new order in the Subscription Panel immediately and in your subscription email when it next arrives in your inbox.

- To take action from the Subscription Preview, open the menu on the chart or table.
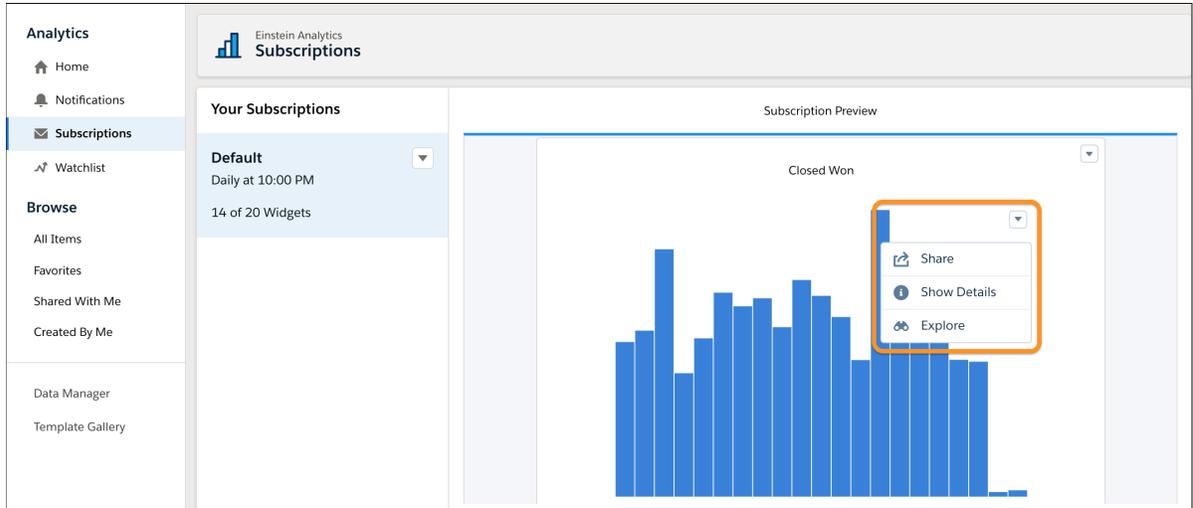
> **Note:** When you subscribe to a table, the Subscription Preview shows the first rows of the table and you can scroll to see succeeding rows. The subscription email body contains a placeholder for the table and a `.csv` file of the table data is attached.

- To change the frequency or timing of subscription emails, go to Subscriptions on the Tableau CRM home page, open the menu on Default, and select **Edit**.
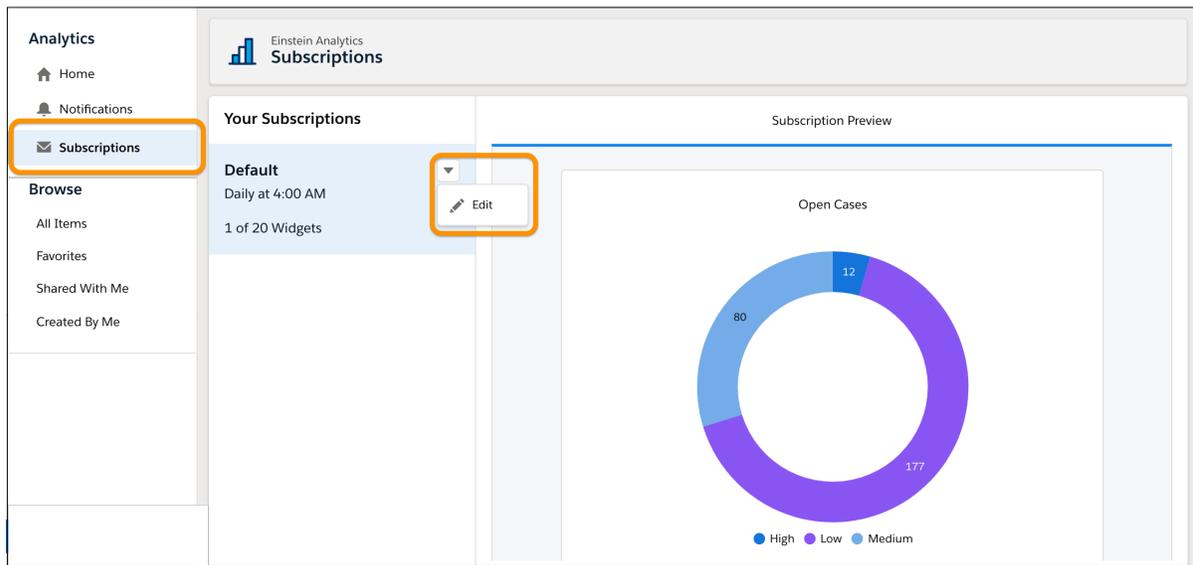
> **Note:** To receive emails, **Enabled** must be selected.

- To delete a subscription, select **Delete** from the menu on the subscription in the Subscriptions panel of the dashboard or lens. You can also remove a widget from your subscription with the menu next to the widget title in the Subscription Preview.

Take note of these considerations for Tableau CRM subscriptions.

- Set up subscriptions in the Tableau CRM Studio, the Analytics tab, and in embedded Tableau CRM dashboards.
- You can schedule one email with up to 20 widget subscriptions.
- Subscriptions are available on all widgets and lenses except maps.
- A lens subscription is dynamic, which means that any change saved to a subscribed lens is automatically reflected in the subscription email and preview. Because a lens subscription is dynamically in sync with its lens, you can't delete a lens while there's a subscription to it.
- You can subscribe to any lens that is saved in SAQL format in the explorer. If you must convert a lens to SAQL, we let you know when you try to subscribe. To convert to SAQL, open Query Mode, click **Run Query**, and then save the lens.
- A widget subscription is based on a snapshot of the query at the time you subscribed, and the subscription isn't automatically updated when the dashboard is edited. After a subscription is set, any change to the dashboard or update to the query or widget properties doesn't change the subscription. Conditional formatting, measure formats, and column names in compare tables are captured when a subscription is created and aren't automatically updated.
- If a widget's query includes results from another query, the outside query is run only at the time that the subscription is created. Queries with dependencies can cause subscription emails to become inaccurate over time. Before subscribing, consider revising a widget's query to remove its dependencies.
- You can subscribe to the same widget multiple times to get updates with different filtered views. Each subscription captures the state of the widget with any dashboard filters or selections you've applied.
- Subscriptions to widgets based on SOQL, Apex, Salesforce Direct, and Tableau CRM Direct queries aren't supported.
- Table data `.csv` attachments can be up to 3 MB each, with total attachments per email up to 15 MB.

SEE ALSO:

   Enable Downloading Data from Tableau CRM

   Secure Image Sharing and Downloading

Collaborate with Dashboard Annotations

Annotate dashboard widgets with comments posted in the dashboard and in Chatter. With annotations, you can hold conversations about your data and how it's visualized, with the dashboard right there for reference.

To create an annotation, select **Annotate** from the dropdown menu on any dashboard widget.

📝 Note: Annotations on dashboards in the Private App aren't available unless you have the View All Data user permission. If you don't see the Annotate option, try moving the dashboard to another app.

To open the Annotations panel, click the speech bubble icon (💬) in the dashboard control bar. When the panel is open, widgets with annotations show a blue square (1) with the number of open annotations. Tableau CRM saves both open and resolved annotations. And if you delete a widget, its open annotations are saved as detached annotations. You can access detached annotations from the blue circle at the bottom of the dashboard.

Click a blue square or circle to open an annotation for comments.
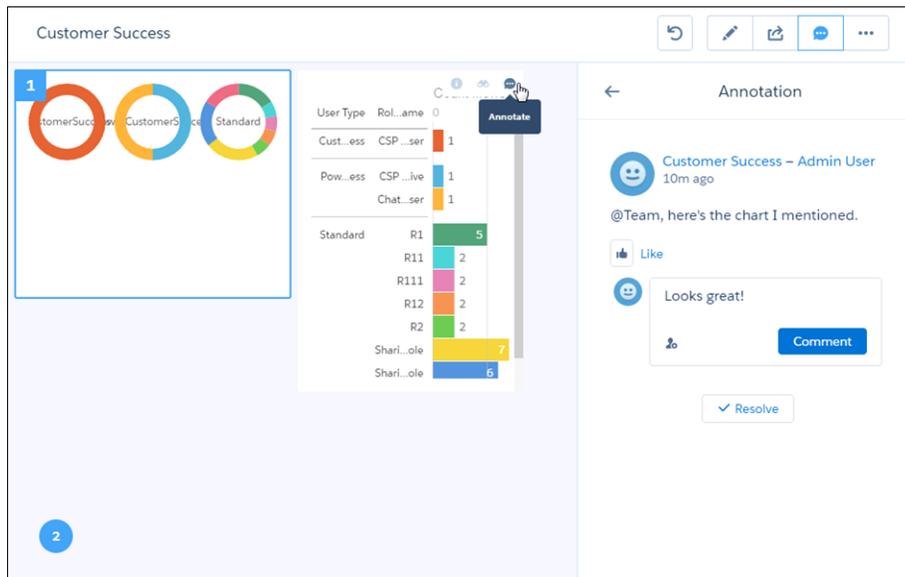
⛔ Important: Although you can @mention anyone in your org, your security setup is what controls access to the dashboard. In contrast, if you include a screenshot, an image of the dashboard appears in the annotation's Chatter post and could be visible to users without access the dashboard. ☑ Attach current screenshot

Tableau CRM dashboard annotations are natively integrated with Chatter, and comments on dashboards also appear as Chatter posts. Clicking the link included in a Chatter post opens the dashboard view just as it was set up at the time you posted. In addition, the Annotations panel opens with the relevant Chatter feed. When colleagues click your annotation, they see the dashboard page with all of your filters and selections applied, and they can easily respond in the Annotations panel.

On an open annotation, you can refresh the feed or close the annotation and go back to the list of all annotations in the dashboard.

← Widget Annotations ↻

In the dashboard panel, each annotation has a dropdown menu for resolving or deleting the annotation.
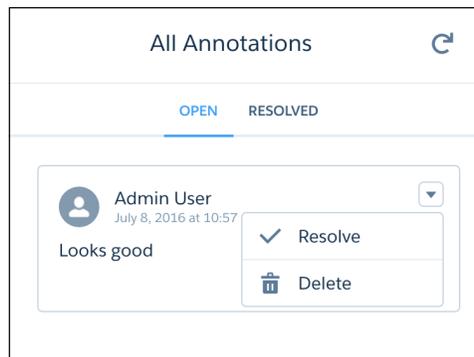
To receive all posts on a dashboard, click its ⊞ Follow button.

Present Live, Interactive Dashboards

Conduct meetings directly from Tableau CRM by presenting dashboards in running apps. With dashboards instead of slides with static images, you have real-time access to your data, and you can showcase dynamic visualizations.

Start a presentation by clicking the Full Screen icon (🖳) in the control bar of a running app or dashboard.

During the presentation, you can use all the Tableau CRM functionality, such as filtering, faceting, and exploring lenses. In addition, the presentation has a control bar.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To present from Analytics Cloud:
- Use Analytics

Service Analytics Overview ▾

You can share, set notifications, and collaborate in a dashboard's feed with annotations.

To move to another dashboard or lens, click ▼ to open the navigation list and select the next asset to present.

Share Dashboards, Widgets, and Lenses

Share a dashboard, widget, or lens with your colleagues by posting it to Chatter or copying its unique URL. A Chatter post provides an image and a link to the asset in Tableau CRM. Colleagues with the link and access to the asset can drill in and explore the information that's presented. You can also export snapshot images of Tableau CRM assets to Quip. An exported image is published to a new Quip slide deck.

📝 **Note:** **Connect to Quip** is available in Tableau CRM tabs and embedded Tableau CRM dashboards. The feature isn't available in the Tableau CRM Studio.

1. Click **Share**.

🔗

2. Click the tab for the sharing method that you want. **Post to Feed** posts an image and link in Chatter. **Export to Quip** publishes an image in a new slide deck with a cover slide and an image slide. **Get URL** provides a unique URL to the asset.

   a. If you're posting to Chatter, select **User** or **Group** feed, enter the name, and then type your comment. You can also remove the image from your post by hovering over it and clicking the **X**. **Note**: Posted images could be visible to users without access the lens or dashboard.

   b. If you're exporting to Quip, and you're not already connected, click **Connect to Quip** and then click **Publish**. **Note**: After publishing a Tableau CRM snapshot, other Quip users in your org can view the data in the image, including confidential data.

   c. If you're getting the unique URL, select the link to the Analytics tab in Lightning Experience or the link to the Tableau CRM Studio. Copy the link and then paste it wherever you want to share it.

   💡 **Tip:** Dashboard links shared from Tableau CRM Studio can include views with selections and filters applied. To generate a URL with page and view specifications, either use **Get URL** or remove the image from **Post to Feed** by hovering over the image and clicking the **X**.

3. If applicable, click **Give Access** to set the level of sharing access.

4. Click **Done**.

   📝 **Note:** For colleagues to use the link to a dashboard or lens, you must give them access to the app that contains it. Lenses, datasets, and dashboards within the default Shared App are accessible to all Tableau CRM users, unless administrators have restricted access. All other apps are private unless someone with Manager access to the app has shared it with a specific user, group, or role.

SEE ALSO:

Download Tableau CRM Images and Export Filtered Data

Secure Image Sharing and Downloading

Download Tableau CRM Images and Export Filtered Data

Download filtered data from lens explorations and dashboard widgets. Download formats include image (`.png`), Microsoft® Excel® (`.xls`), and comma-separated values (`.csv`) files. This feature downloads the results of a displayed query (or step).

Select **Share** from the menu on a dashboard widget, or from a lens, click the **Share** icon. In the Share dialog, select **Download**.

When downloading filtered data, the query limit determines the number of rows in the Excel or .csv file.

Take note of these limitations on downloading data to an `.xls` or `.csv` file.

- For table data downloads, hidden columns, grouped column order, reordered columns, and merged cells aren't supported.

- For pivot tables, queries customized with SAQL union statements download as raw data instead of in pivot table format.

SEE ALSO:

Enable Downloading Data from Tableau CRM

Secure Image Sharing and Downloading

Create PDF Files and Print Your Dashboards

Dashboards are printer-friendly, and you can print or save PDF files of dashboards from your browser.

Printing is available in the dashboard preview mode in Tableau CRM Studio.

1. Click ••• to open the menu, and select **Print Preview**.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To print Tableau CRM dashboards:
- Use Analytics
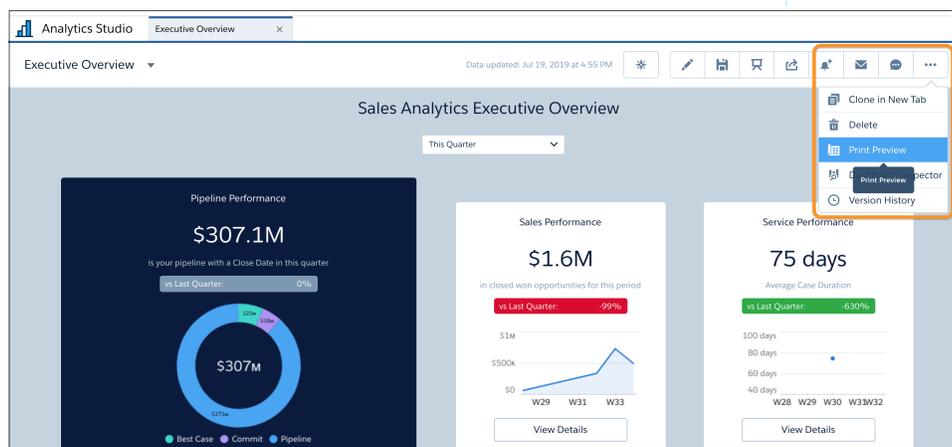
A snapshot of your dashboard opens.

2. Click **Print** to open your browser's print dialog, where you can send the image to your printer or save it as a PDF file. To include background color or images, click **More settings** and select **Background graphics**.

The **Cancel** button closes the print preview and returns you to the running dashboard.

When printing a dashboard, consider the following.

- Printing is supported in Chrome browsers only.
- The Print action is not included with embedded dashboards.
- The print is a snapshot of the dashboard with your current filters and selections. If a widget is scrollable, only the current view of the widget prints. If the dashboard has multiple pages, only the currently open page prints.
- Colors and page orientation are based on your print settings.

Perform Salesforce Actions Directly from Tableau CRM

Take action on your insights directly from Tableau CRM. Perform custom and mass quick actions from charts and tables in lenses and dashboards.

Perform object-specific actions and global actions that are defined on Salesforce objects directly from Tableau CRM. You can invoke these actions to create, update, and interact with Salesforce records. For example, you can create cases, update accounts, and post to Chatter. You can also use defined List View actions to perform mass quick actions on up to 100 Salesforce records at once.

Take Control of Your Salesforce Experience with Tableau CRM Custom Actions Menus

Tableau CRM custom actions menus in charts and tables let you take advantage of Salesforce actions. With action menus, you can post to Chatter, create an event or opportunity, update a record, or open a Salesforce record directly.

Take Action on Multiple Records with Mass Action Links

Mass action links in Tableau CRM let you take advantage of Salesforce mass quick actions by performing an action on a list of up to 100 Salesforce records.

Take Control of Your Salesforce Experience with Tableau CRM Custom Actions Menus

Tableau CRM custom actions menus in charts and tables let you take advantage of Salesforce actions. With action menus, you can post to Chatter, create an event or opportunity, update a record, or open a Salesforce record directly.

With custom actions menus, you can use quick actions that were added to Salesforce objects—directly from Tableau CRM dashboards and lenses. You can also open records from Salesforce or elsewhere on the Web through the menus.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To use custom actions menus:
- "Use Analytics"

You can access actions menus from these parts of charts and tables in Tableau CRM:

- Value labels on charts (such as Opportunity for Pearson in the preceding image).
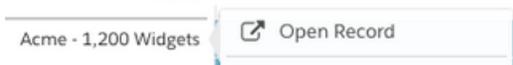
- Legend value labels in table cells.

- Values in table cells.
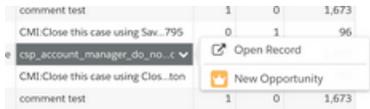
Hover over the labels to see if an actions menu is available. Click the value, cell, or legend to see the menu's contents, and then select an action or open a record. Records open in new Lightning Experience tabs unless your admin sets them to open in new browser tabs.

For dimensions in lenses and dashboards, you can use the Actions menu in the explorer to set how one-click actions behave on a table's dataset fields. For more efficiency, you can select the record action you want to invoke for all records in a dimension column. Or, for greater flexibility, you can choose to view all available one-click actions when you click a column's records.

For example, imagine that you want to update opportunity records when you click the Opportunity Name field in the Opportunities dataset. After your Salesforce admin has configured actions for this field, open the Opportunities dataset.

1. With Opportunity Name selected in the table mode, in the Column Properties panel, scroll to the Actions section.

2. Select **Set up one-click actions**.

3. Enable **Select an Action**. Alternatively, to view all available actions when you click an opportunity name, select **Open Actions Menu**

4. From the Salesforce Actions list, select **Update Opportunity**.

Optionally set a hyperlink color for the Opportunity Name field.

To learn more, see Enable Actions for Tableau CRM Lenses and Dashboards

Take Action on Multiple Records with Mass Action Links

Mass action links in Tableau CRM let you take advantage of Salesforce mass quick actions by performing an action on a list of up to 100 Salesforce records.

With mass quick action links, you can use quick actions that were added to Salesforce objects List View layouts, directly from Analytics dashboards. You can perform a mass action on up to 100 records at once, using Analytics filtering to target the list of records you want to perform the action on.

Access mass quick actions by clicking the mass action link on the dashboard.

1. Filter your records.

2. View the list of records in the table. In this example, there are 29 accounts of the industry type banking.

3. To perform the mass quick action on all 29 account records, click **Mass Update**.

Each mass action has a custom window that opens when the link is clicked. The window is the same as you see when you take the mass action in the record list view of Lightning Experience. As you fill in the fields, the Fields to update section updates. When you have the action completed, click **Save**.

The mass action confirmation dialog shows how many records the action is performed on and which fields are updated. If more than 100 records are in the list, the dialog informs you that only 100 records are updated. Click **OK** to perform the action.

📝 Note: The update to the records is immediately visible in the Salesforce records, but the Analytics dataset must be updated to reflect the record updates. Your dashboard gets the updated record data the next time the dataflow or recipe runs.

SEE ALSO:

Configure Mass Quick Actions on Multiple Salesforce Records from Tableau CRM Dashboards

Learning Resources

In addition to these Help pages, Tableau CRM has a variety of in-app assistance, and there's more educational content on Trailhead and other websites.

Get Started, Educate Yourself, and Stay Informed in the Learning Center

The Learning Center is the home of assistance resources in the Tableau CRM Analytics Studio. With getting started experiences and continuing educational content, there's learning for all users, anywhere along their analytics journey.

Access Help Resources When and Where You Need Them

In-app assistance includes the welcome mat, help menus, point-in-time videos, and widget-specific learning resources.

Learn Tableau CRM Through Video

Sometimes it's easier to understand how things work when you see them in action. Check out these video playlists for all areas of Tableau CRM. Video playlists are available in Help Menus, and most videos are also available from in-app video icons in context throughout Tableau CRM.

Find the Right Learning Resource for Each Step of the Tableau CRM Journey

The Einstein Analytics Learning Map gives you a list of key resources for each action in every stage of your Analytics journey. The map is a fun, one-stop shop that helps you grasp the places you can go with Tableau CRM. Access it from the Learning Center and help menus, or go directly: **http://www.einsteinanalyticslearningmap.com/**

Learn Tableau CRM with In-App Examples

The Tableau CRM Learning Adventure app walks you through best practice examples for designing just the right visualizations and for building powerful dynamic apps. Hands-on examples are presented in dashboards that show you how to build dashboards. Because it's an app, you can look under the hood and see the JSON used to build each dashboard.

Learn Tableau CRM with Trailhead

Trailhead is the fun and free way to learn Salesforce. Now you can get acquainted with Tableau CRM through Trailhead, using a special Tableau CRM-enabled Developer Edition org.

Get Started, Educate Yourself, and Stay Informed in the Learning Center

The Learning Center is the home of assistance resources in the Tableau CRM Analytics Studio. With getting started experiences and continuing educational content, there's learning for all users, anywhere along their analytics journey.

Travel along guided learning journeys in the Start Your Analytics Adventure section. Click a tile to open a fast-paced journey through key product areas. Take an entire journey at once, or stop anywhere along the way and later pick up where you left off. You never lose your place because we're tracking each step. The available journeys depend on your permissions.
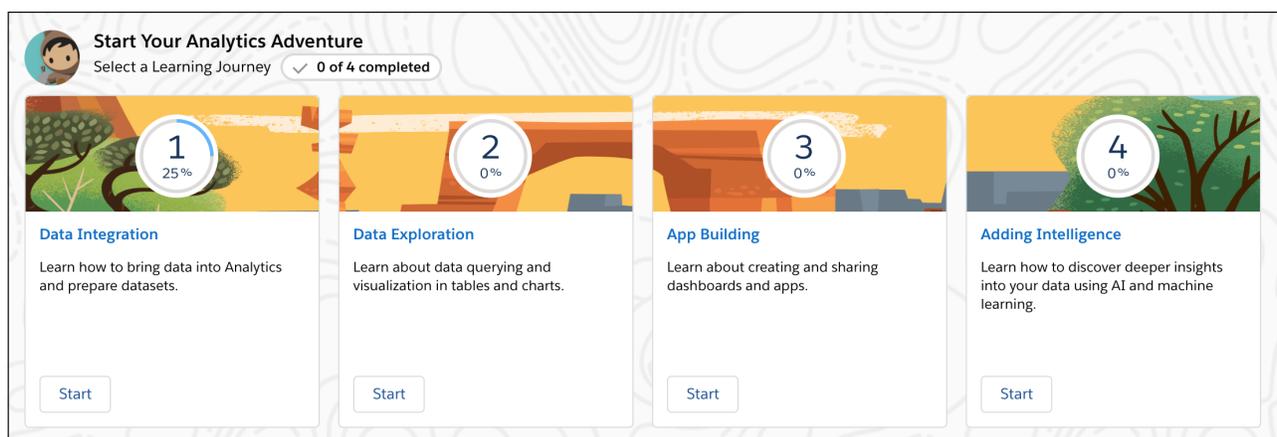
In-app badges reward completion of the learning journeys. Four in-app badges are available, depending on your permissions. The badges appear in your Trailhead profile.

Data Integration in Tableau CRM	Data Exploration in Tableau CRM	App Creation in Tableau CRM	Einstein Discovery Intelligence

Refer to the Extended Learning section for links to important user assistance resources, such as the Einstein Analytics Learning Map.

Extended Learning
Visit these resources to elevate your skills

Learning Map
Find the resources you need at every stage of your Analytics adventure.

Learn More

Einstein Analytics Help
Consult the complete, source-of-truth Help pages.

Learn More

Trailblazer Community
Ask questions and look for answers with the Customer Success Ohana.

Learn More

Einstein Analytics Trainings
Join hands-on training sessions taught by experts.

Learn More

Raise your Tableau CRM education to the next level with the new Popular Topics section in the Learning Center. Links to learning resources are arranged by function and you can click **Explore More** to go to the Best Practices page of the Einstein Analytics Learning Map.

Popular Topics
Browse learning content based on the topic that interests you

Explore More

Data Preparation
Integrate Your Data in Einstein Analytics
Data Manager video series
Einstein Analytics Data Integration Basics

Data Visualization
Explore and Visualize Your Data in Einstein Analytics
Explorer video series
Einstein Analytics Desktop Exploration

App Building
Build Einstein Analytics Dashboard
Dashboard Designer video series
Einstein Analytics Dashboard Building Basics

Adding Intelligence
Explain, Predict, and Recommend with Einstein Discovery
Einstein Discovery video series
Gain Insight with Einstein Discovery

Learn about top features and general updates each release in the What's New section.

What's New in Summer '19

Connect to Your Data in Oracle Eloqua and NetSuite (Generally Available)

Connect to even more enterprise application data with the Oracle Eloqua and NetSuite connectors.

Learn More

Sync More Data Through Amazon Redshift Connections

For each object, you can now load up to 100 million rows or 50 GB, depending on which limit is reached first.

Learn More

Set Connection Mode for Connected Salesforce Objects in One Place

Allow Analytics to clean up incremental sync conflicts. We also added Full Sync as a connection mode option.

Learn More

Financial Services Analytics Templates Have New Names and Licensing

Use new templates to build an Einstein Analytics solution based on your Financial Services Cloud data.

Learn More

Access Help Resources When and Where You Need Them

In-app assistance includes the welcome mat, help menus, point-in-time videos, and widget-specific learning resources.

Once you're up and running with Tableau CRM, in-app resources support you along the way. First thing you'll notice after login to Tableau CRM Studio is the welcome mat. We've packed it with videos, trails, and documentation to get you started. If you click out of it, you can always get it back from the help menu.
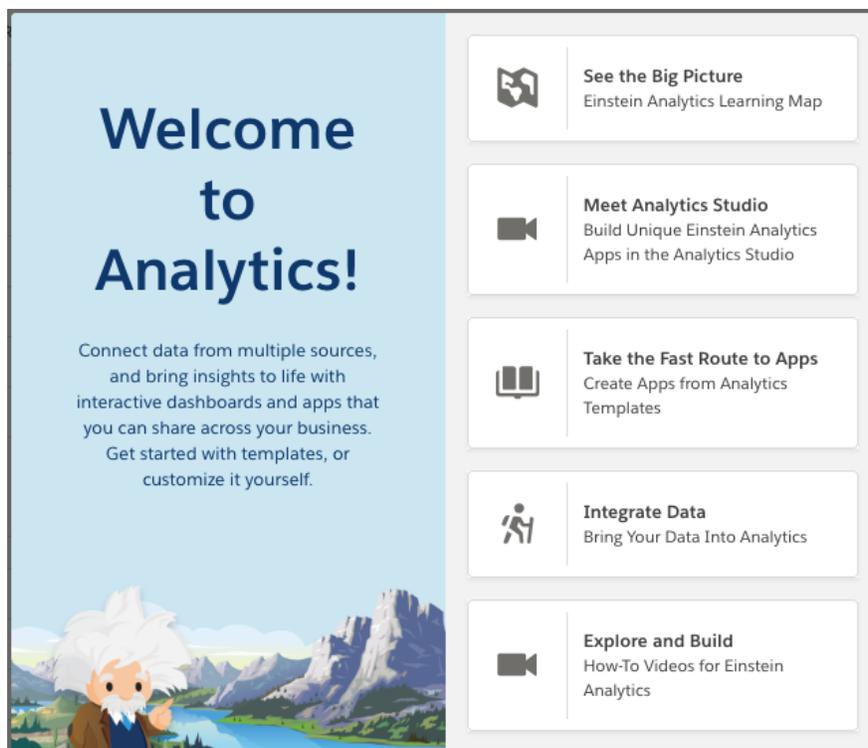
That little question mark (?) in the upper right corner opens context-aware, persistent help menus that provide relevant help topics targeted for the page you're on. You can preview the Help page right in the menu, and even pin it to the bottom of the page to reference while working.

In-app video icons (▶) throughout Tableau CRM launch a video or video series that teaches about the feature you're working with. Videos play in a draggable, resizable frame. Put the frame wherever you want so the video never blocks the product screen while you watch. You can even interact with the product as you follow along step-by-step tutorials.

If your dashboard is configured with the **Learn** option, select it to open the draggable, resizable frame loaded with customized, widget-specific content.
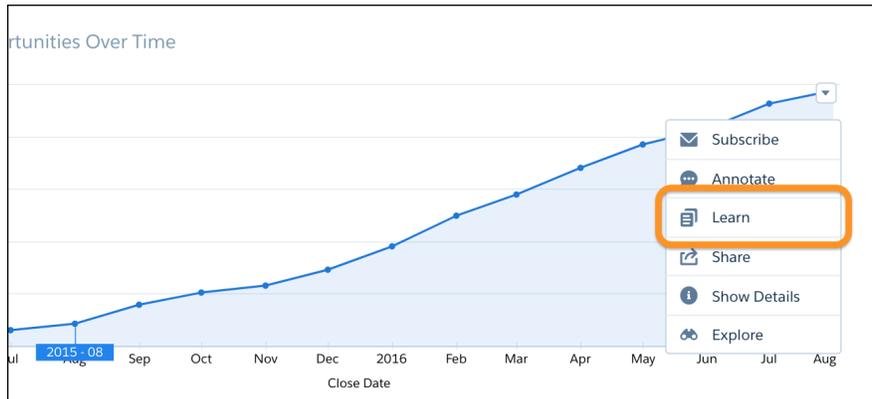
When you have at least one widget with the **Learn** option, the Learning Resources icon appears. Click the icon to open and close the Learning Resources panel. When the panel is open, hover on a learning resource to highlight the chart that goes with it. Each learning resource is numbered and the numbers appear on the charts for easy reference.
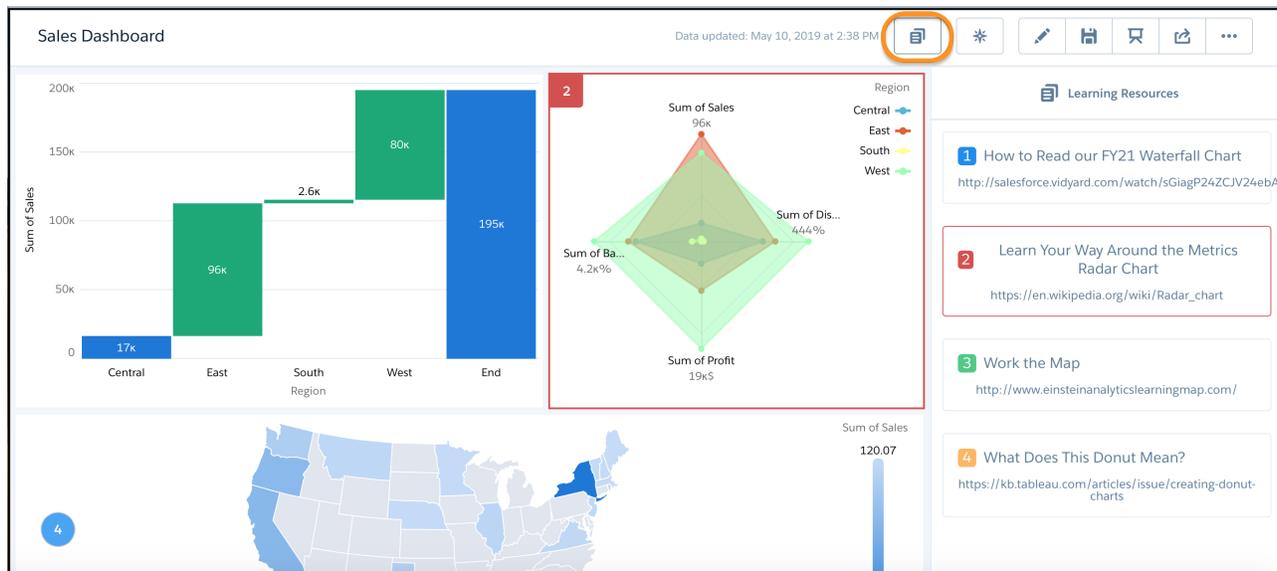
Learn Tableau CRM Through Video

Sometimes it's easier to understand how things work when you see them in action. Check out these video playlists for all areas of Tableau CRM. Video playlists are available in Help Menus, and most videos are also available from in-app video icons in context throughout Tableau CRM.

All videos are in English.

- **Tableau CRM Studio Home Page Series**

  Learn your way around the Tableau CRM Studio home page, and get to know basic dashboard operations, favorites, sharing, and more.

- **Navigate Tableau CRM Dashboards Series**

  Learn your way around Tableau CRM dashboards, and get to know conversational exploration, notifications, annotations, and more.

- **Einstein Analytics Explorer Series**

  Learn to uncover data insights in Tableau CRM using conversational queries, compare tables, SAQL editor, and more.

- **Build Interactive Tableau CRM Dashboards Video Series**

  Learn the basics of building Tableau CRM designer dashboards, including information about widget types, building steps, creating dashboards from templates, generating custom layouts for different devices, and making dashboards interactive with faceting and bindings.

- **Einstein Analytics Data Manager Series**

  Gain access to data inside and outside Salesforce with connections. If needed, prepare the data before loading it into Tableau CRM datasets. Data preparation is the process of transforming your data into a form that's meaningful and valuable to the people consuming it.

- **Einstein Analytics Recipe Editor Series**

  Recipes are a great way to prepare data in Tableau CRM, but where do you start? Join us for a quick tour of where the magic happens: the recipe editor, also known as "Data Prep." Learn how to set up your data preview, transform fields with smart suggestions, and get intelligent insights about your data with the column profile.

- **Einstein Discovery Video Series**

  Einstein Discovery augments your data analysis with AI and machine learning. Create stories and investigate insights to learn what happened in your data, why it happened, what could happen, and ways to improve predicted outcomes.

- **Getting Started series**

  - Getting Started: Bring in Data
  - Getting Started: Explore Data
  - Getting Started: Build an App
  - Getting Started: Create a Story

Find the Right Learning Resource for Each Step of the Tableau CRM Journey

The Einstein Analytics Learning Map gives you a list of key resources for each action in every stage of your Analytics journey. The map is a fun, one-stop shop that helps you grasp the places you can go with Tableau CRM. Access it from the Learning Center and help menus, or go directly: **http://www.einsteinanalyticslearningmap.com/**

With so many resources out there—trails, documentation, videos, webinars—how do you know where to look? We created the Einstein Analytics Learning Map to address just that. We scoured all of our educational resources, pulled together the best content, and organized it by area of product engagement to make it easier to find the right help at the right time.

Click each point on the trail to see lists of resources for each stage of the Tableau CRM journey. The lists aren't exhaustive, but if you don't find what you need right away, they provide entry points into documentation that makes finding more advanced topics easy.
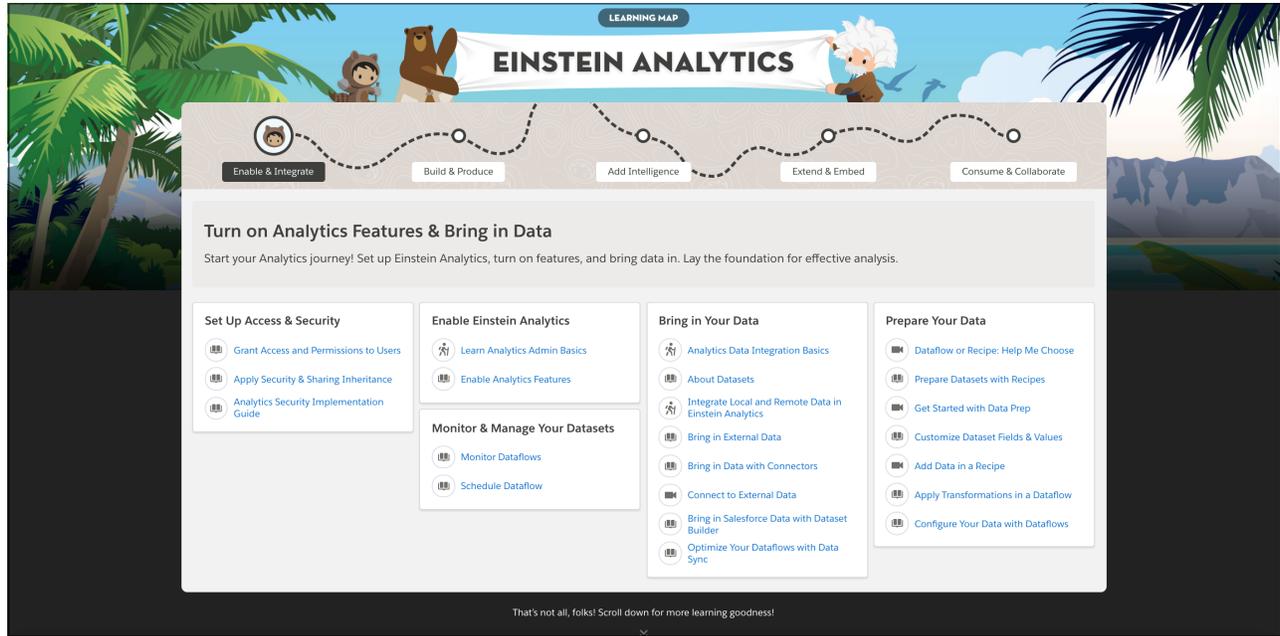
The lower part of the Learning Map provides links to more Learning Map pages and other resources. Don't miss Best Practices and the Trailblazer Content Center.
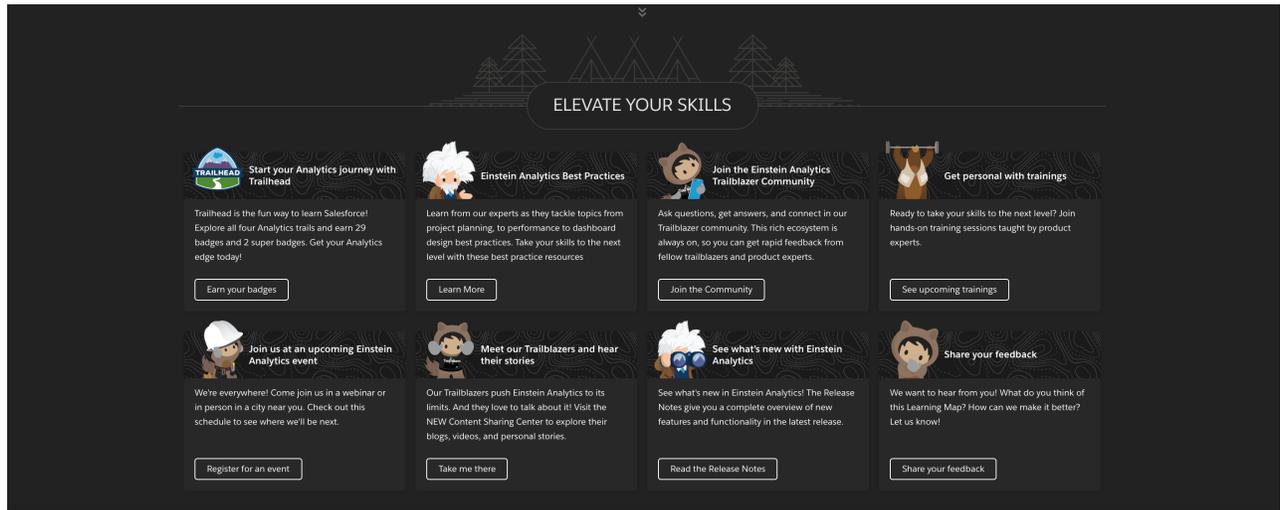
Learn Tableau CRM with In-App Examples

The Tableau CRM Learning Adventure app walks you through best practice examples for designing just the right visualizations and for building powerful dynamic apps. Hands-on examples are presented in dashboards that show you how to build dashboards. Because it's an app, you can look under the hood and see the JSON used to build each dashboard.

With the Tableau CRM Learning Adventure app, you learn as you play with example charts and dynamic visualizations. Some things you can expect to get out of this Learning Adventure:

- Learn how to choose the right visualization for your data, and your audience.
- Explore real examples of comparison, trending, relationship, composition, distribution, metrics, and location charts – see what's possible!
- Understand the power of tables and learn how to take advantage of the rich features across table types.
- Write customized queries with help from the SAQL examples.
- View the code behind dynamic visualizations (bindings) – explained in plain English! – and apply it yourself with step-by-step instructions.

To add the Learning Adventure App to Tableau CRM Studio, click the **Create** button and select **App**. Or click **Template Gallery** in the left panel of the Tableau CRM Studio home page. Then in the gallery, select **Learning Adventure**.

EDITIONS

Available in: Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in: **Developer Edition**.

USER PERMISSIONS

To install the Learning Adventure App in Tableau CRM Studio:
- Manage Analytics Templated Apps

To access the Learning Adventure App after it's installed:
- Use Analytics Templated Apps
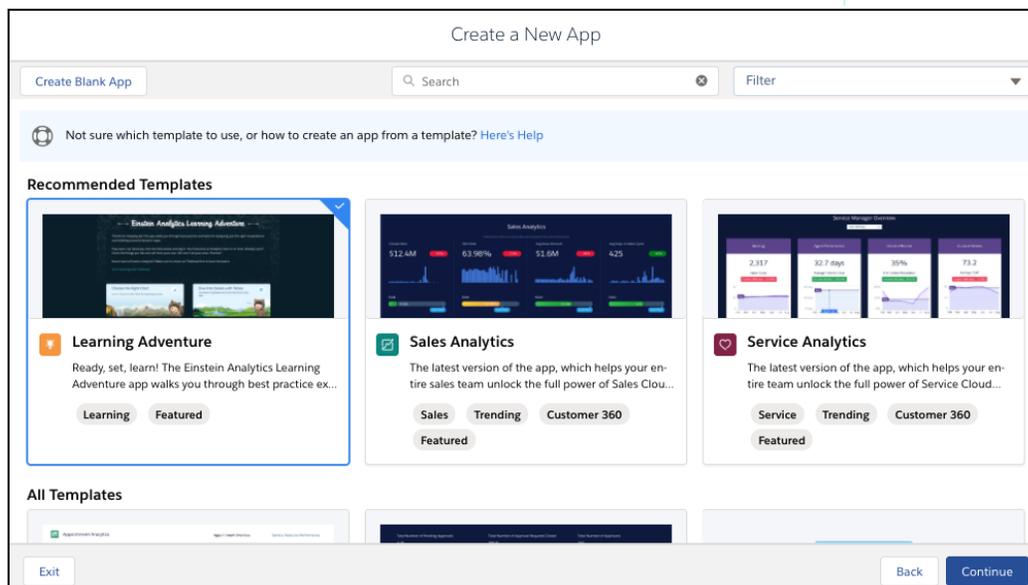
Learn Tableau CRM with Trailhead

Trailhead is the fun and free way to learn Salesforce. Now you can get acquainted with Tableau CRM through Trailhead, using a special Tableau CRM-enabled Developer Edition org.

For the Tableau CRM trails, you can't use an old Developer Edition org. You must sign up for the special Developer Edition that comes with a limited Tableau CRM Growth license and contains sample data required for the Tableau CRM trails.

You'll find the sign up for the special Developer Edition org in these trails:

- Explore with Analytics: http://trailhead.salesforce.com/trail/wave_analytics_explorer
- Build and Administer Analytics: https://trailhead.salesforce.com/trails/wave_analytics_enable_and_produce
- Accelerate Analytics with Apps: http://trailhead.salesforce.com/trail/wave_analytics_apps
- Gain Insight with Einstein Discovery: https://trailhead.salesforce.com/trails/wave_analytics_einstein_discovery

And learn more about Analytics with projects, superbadges, and trailmixes:

- Quick Start: Tableau CRM: https://trailhead.salesforce.com/content/learn/projects/quick-start-einstein-analytics
- Integrate Local and Remote Data in Tableau CRM: https://trailhead.salesforce.com/content/learn/projects/integrate-local-and-remote-data-in-einstein-analytics
- Build a Gauge Chart That Visually Identifies Regional Data: https://trailhead.salesforce.com/projects/analytics_charts_icons
- Create a Custom Map for Tableau CRM Charts: https://trailhead.salesforce.com/projects/wave_charts_custom_map
- Create a Product Pipeline Dashboard with Tableau CRM Charts: https://trailhead.salesforce.com/projects/wave_charts_prod_pipeline
- Build Advanced Tableau CRM Dashboards: https://trailhead.salesforce.com/content/learn/projects/advanced-dashboard-techniques
- Embed a Tableau CRM Dashboard in Lightning Experience: https://trailhead.salesforce.com/content/learn/projects/embed-an-einstein-analytics-dashboard-in-lightning-experience
- Einstein Analytics Data Preparation Specialist: https://trailhead.salesforce.com/en/superbadges/superbadge_analytics_integration_specialist
- Einstein Analytics and Discovery Insights Specialist: Einstein Analytics and Discovery Insights Specialist
- Einstein Analytics and Discovery Consultant: https://trailhead.salesforce.com/credentials/einsteinanalyticsanddiscoveryconsultant
- Architect Journey: Analytics Architecture: https://trailhead.salesforce.com/users/00550000006yDdKAAU/trailmixes/architect-analytics-architecture

SEE ALSO:
  Tableau CRM Limits
  Tableau CRM Limits
  Tableau CRM Limits

Tableau CRM Glossary

Familiarize yourself with common Tableau CRM terminology.

App

An *app* contains dashboards, lenses, and datasets in any combination that makes sense for sharing your data analyses with colleagues. Apps are like folders. They allow users to organize their data projects—private and shared—and to control sharing.

App template

An *app template* is a ready-made dashboard package that is automatically populated with Salesforce data.

Binding

A *binding* is an expression to reference information in a query via string replace. Bindings allow for setting dynamic variables for chart and query configurations.

Column Chart

A *column chart* is a name for a vertical bar chart.

Co-group

A *co-group* is used to combine data from two or more data streams. The data streams can be joined using common fields.

Connected data

Connected data is an object or file from a data source that is synced. This data isn't queryable until it's converted into a dataset.

Dashboard

A *dashboard* is a curated set of charts, metrics, and tables based on the data in one or more lenses.

Dashboard JSON file

A *dashboard JSON file* defines the components that a dashboard contains and describes how they're connected.

Data Monitor

The *data monitor* tool allows you to monitor dataflow and system jobs. You can also use it to start, stop, reschedule, download, and upload dataflows.

Dataflow

A *dataflow* is a set of instructions that specifies what data to extract from Salesforce objects or datasets, how to transform the datasets, and which datasets to make available for querying.

Dataflow definition file

A *dataflow definition file* is a JSON file that contains transformations that represent the dataflow logic.

Dataflow job

A *dataflow job* processes the logic in a dataflow.

Dataset

A *dataset* is a collection of related data that is stored in a denormalized, yet highly compressed form that is optimized for interactive exploration.

Dataset builder

The *dataset builder* is a point-and-click user interface that allows you to easily select and extract data from related Salesforce objects to create a single dataset.

Data sync

A *data sync* is the method used to refresh connected data in Tableau CRM.

Date

A *date* can be represented as a day, month, year, and, optionally, time. You can group, filter, and perform math on dates.

Designer

Designer is the interface where you create dashboards.

Derived dimensions

A *derived dimension* is a qualitative value that is not part of a dataset, but is instead calculated in a query.

Derived measures

A *derived measure* is a quantitative value that is not part of a dataset, but is instead calculated in a query.

Dimension

A *dimension* is a qualitative value that usually contains categorical data, such as Product Category, Lead Status, and Case Subject. Dimensions are handy for grouping and filtering your data. Unlike measures, you can't perform math on dimensions.

Explorer

Explorer is the tool used to interactively explore and visualize your data.

Digest

The *digest* transformation extracts data from connected objects in a dataflow. Use it to extract synced data from your local Salesforce org or remote connections to include in a dataset.

Extended metadata (XMD)

Extended metadata (XMD) allows you to customize the formatting of many dashboard elements in Tableau CRM.

External data

External data is data that resides outside of Salesforce, such as data from outside applications and spreadsheets.

External Data API

Use the *External Data API* to upload external data files to Tableau CRM to create datasets.

Lens

A *lens* is a view into a dataset used in an exploratory mode or to get insight to a specific business question. The lens can be saved and shared independently. It can also be clipped to a dashboard.

Measure

A *measure* is a quantitative value that contains numerical data like revenue and exchange rate. You can do math on measures, such as calculating the total revenue and minimum exchange rate.

Metadata file

A *metadata file* is a JSON file that describes the structure of an external data file.

Predicate

See *security predicate*.

Query

A *query* retrieves data from one or more data sources, including datasets, Salesforce objects, user-defined data, or external data. The query results can be displayed in different formats (such as a table or chart) or used as input to another query.

> **Note:** Prior to the Winter '20 release, *queries* were called *steps*.

Recipe

A *recipe* is a user-interface tool that lets you take data from your existing datasets and connected objects, apply transformations, and the output the results to a new dataset.

Register

The *sfdcRegister* transformation saves the transformed data as a dataset, which can then be queried in Tableau CRM. Users can't view or run queries against unregistered datasets.

Salesforce Analytics Query Language (SAQL)

Use *SAQL* to access and analyze data in Tableau CRM datasets.

Security Predicate

A *security predicate* is a filter condition that defines row-level access to records in a dataset.

Step

A *step* retrieves data from one or more data sources, including datasets, Salesforce objects, user-defined data, or external data. The step results can be displayed in different formats (such as a table or chart) or used as input to another step.

> **Note:** As of the Winter '20 release, *steps* are called *queries*.

Story

A *story* is the output of Einstein Discovery's comprehensive statistical analysis of your Tableau CRM dataset. A story represents a collection of insights around a metric (outcome) that highlights any of the following: important trends, explanations on what may have influenced those trends, comparisons between factors, predictions on future outcomes, and suggested actions that may improve outcomes.

Transformation

A *transformation* refers to the manipulation of data. You can add transformations to a dataflow to extract data from Salesforce objects or datasets, transform datasets that contain Salesforce or external data, and register datasets.

Trellis

A *trellis* is the name for a grid (horizontal, vertical, or both) of visualizations. A trellis is also known as *small multiples* and *lattice*.

Unit symbol (for measures)

A *unit symbol* represents the unit of a shortened number that appears in a chart or number widget. Tableau CRM uses the following symbols.

Symbol	Unit	Example
K	Thousand (10^3)	4K = 4,000
M	Million (10^6)	4M = 4,000,000
B	Billion (10^9)	4B = 4,000,000,000
T	Trillion (10^12)	4T = 4,000,000,000,000
P	Quadrillion (10^15)	4P = 4,000,000,000,000,000
X	Quintillion (10^18)	4X= 4,000,000,000,000,000,000
Z	Sextillion (10^21)	4Z = 4,000,000,000,000,000,000,000
Y	Septillion (10^24)	4Y = 4,000,000,000,000,000,000,000,000
m	Thousandth (10^-3)	4m = 0.004
μ	Millionth (10^-6)	4μ = 0.000004
n	Billionth (10^-9)	4n = 0.000000004
p	Trillionth (10^-12)	4p = 0.000000000004
f	Quadrillionth (10^-15)	4f = 0.000000000000004
a	Quintillionth (10^-18)	4a = 0.000000000000000004
z	Sextillionth (10^-21)	4z = 0.000000000000000000004
y	Septillionth (10^-24)	4y = 0.000000000000000000000004

Visualization

A *visualization* is commonly a chart or graph, such as a bar chart, donut chart, timeline, or heat map. It can also be data in tabular form, such as a comparison table or pivot table. Every visualization has an underlying query, which is how Tableau CRM retrieves information from the source data.

Access Insights from the Tableau CRM Mobile App

Tableau CRM on mobile gives you a fast, fluid way to discover the compelling insights within your data and to share the right visuals wherever you are. This page has resources for Tableau CRM in the Salesforce app and Tableau CRM for iOS and Android.

Which Analytics Mobile Experience Do I Use?

Tableau CRM gives you multiple ways to get insights on the go: Tableau CRM in the Salesforce app and Tableau CRM for iOS and Android. Find out which mobile experience is right for you.

Which Analytics Mobile Experience Do I Use?

Tableau CRM gives you multiple ways to get insights on the go: Tableau CRM in the Salesforce app and Tableau CRM for iOS and Android. Find out which mobile experience is right for you.

The right mobile analytics experience depends on what you want to do.

- For the best mobile analytics experience in the CRM workflow, use Tableau CRM in the Salesforce app, which features embedded dashboards and record actions.
- For the most focused analytics on-the-go experience, use Tableau CRM for iOS or Android and its advanced features like explorer, notifications, and offline data access.

Check out this breakdown of key Tableau CRM features based on business needs.

Business Need	Tableau CRM for iOS	Tableau CRM for Android	Salesforce app (iOS and Android)
Want to add mobile-optimized insights to a record page? Put relevant insights where they matter most: directly in a Salesforce record. To optimize CRM workflow with embedded insights, use Tableau CRM in the Salesforce app.			✔
Want to take action on insights directly from Tableau CRM? To create, update, and interact with Salesforce records from charts and tables in dashboards, we recommend Tableau CRM in the Salesforce app. While Tableau CRM for iOS and Android support some record actions, Tableau CRM in the Salesforce app is optimized for turning your insights into actions, all in one place. Learn more about enabling actions for analytics users on page 1351.			✔

Business Need	Tableau CRM for iOS	Tableau CRM for Android	Salesforce app (iOS and Android)
Want to launch analytics dashboards from a Salesforce record page? If you want to fast-track how mobile users access dashboards from records, use Tableau CRM in the Salesforce app. For example, create a custom quick action on an account record so account owners can launch a sales metrics dashboard with relevant filters applied directly from the record.			✔
Want to explore data and uncover key business insights on your mobile device? If you want to use a lens to dive into the details in your data and build an on-the-go query, use Tableau CRM for iOS or Android. They're optimized for exploration beyond visualizations like charts and tables.	✔	✔	
Want to launch analytics assets directly from your own custom mobile app? To open a specific dashboard from your custom mobile app or apply filters to a dashboard when it's launched, use Tableau CRM for iOS or Android. They support deep linking, which lets you launch a dashboard from a third-party or web application. Learn more about deep linking on page 541.	✔	✔	
Want to view and analyze data when internet access isn't available? If you need your data any time and any place, use Tableau CRM for iOS. It lets you access important insights even when you're offline. Learn more about offline access on page 533.	✔		
Want to track important metrics in your Watchlist? A Watchlist is your one-stop hub for tracking trends across multiple dashboards. If you love your watchlist in Tableau CRM desktop, you can access it in Tableau CRM for iOS.	✔		
Want to get notified when a metric reaches a goal or dips below a specified amount? If on-the-go alerts and notifications are important to your business needs, use Tableau CRM for iOS. Learn more about push notifications on page 540.	✔		

SEE ALSO:

    Download Tableau CRM for Mobile

Tableau CRM Mobile Features

Tableau CRM is available as a downloadable app on iOS and Android devices. This table details supported Tableau CRM features in the mobile apps.

Feature	Tableau CRM for iOS	Tableau CRM for Android
General		
Local playground	✔	
Filter list views (by asset)	✔	✔
Search	✔	✔
App navigation list	✔	
Access to app sharing controls		
View favorite dashboards, lenses, datasets, and apps	✔	✔
Add dashboards, lenses, datasets, and apps to favorites	✔	
Push notifications	✔	
Track notifications	✔	
Custom bulk actions (via Visualforce)		
Tableau CRM REST API	✔	✔
Support for Experience Cloud sites		
Explore Salesforce objects from the dashboard designer with Salesforce Direct		
Connect external data to Tableau CRM with Salesforce Direct for Snowflake Data		
Actions		
View asset information	✔	
View, filter, and facet dashboards and lenses	✔	✔
Drill in and get details from lenses and widgets	✔	✔
Explore datasets, lenses, and widgets	✔	✔
Create and edit dashboards		
Save and delete lenses and dashboards	✔	
Create apps		
Import data (from CSV, TSV, XLSX files)	✔	
Delete datasets and apps		
Post images or links to Chatter	✔	✔
Copy link to dashboard or lens	✔	✔
Open Salesforce records directly from dashboards	✔	✔
Use external links to other sources	✔	✔

Feature	Tableau CRM for iOS	Tableau CRM for Android
Open to asset from link in external app	✔	✔
Annotate dashboard widgets		
Do Salesforce actions from custom action menus		
Access and share saved dashboard views	✔	✔
Add to or edit saved dashboard views		
Exploration and Queries (Steps)		
Apply aggregation functions on datasets	✔	✔
Apply groupings on datasets	✔	✔
Filter by measures, dimensions, and absolute or relative dates	✔	✔
View lenses using SAQL or SOQL step types	✔	✔
View lenses using Apex step types (beta)	✔	
Support for query results limits	✔	✔
Charts and Tables		
Bar	✔	✔
Column	✔	✔
Stacked Bar	✔	✔
Stacked Column	✔	✔
Scatter Plot	✔	✔
Donut	✔	✔
Heat Map	✔	✔
Line	✔	✔
Horizontal/Vertical Dot Plot	✔	✔
Timeline	✔	✔
Time Bar	✔	✔
Time Combo	✔	✔
Waterfall	✔	✔
Stacked Waterfall	✔	✔
Combo (Bar and Line)	✔	✔
Funnel	✔	✔

Feature	Tableau CRM for iOS	Tableau CRM for Android
Origami	✔	✔
Sankey	✔	✔
Metrics Radar	✔	✔
Map	✔	✔
Bubble Map	✔	✔
Custom Map	✔	✔
Geo Map	✔	✔
Pyramid	✔	✔
Stacked Pyramid	✔	✔
Gauge	✔	✔
Polar Gauge	✔	✔
Flat Gauge	✔	✔
Rating	✔	✔
Matrix	✔	✔
Tree Map	✔	✔
Calendar Heat Map	✔	✔
Parallel Coordinates	✔	✔
Values Table	✔	✔
Compare Table	✔	✔
Pivot Table		
Totals (supported in tables)	✔	
Subtotals (supported in tables)	✔	
Chart and Table Properties		
Title, subtitle customization	✔	✔
Border customization	✔	✔
Text customization	✔	✔
Number customization	✔	✔
Conditional formatting/styling options	✔	✔
Images in charts	✔	✔

Feature	Tableau CRM for iOS	Tableau CRM for Android
Table column and cell customization	✔	✔
Images in tables	✔	
Themes in tables	✔	✔
Trellis	✔	✔
Reference lines	✔	✔
Selection of multiple chart elements	✔	✔
Focus on multiple sections		
Dashboard Widgets		
Global filters	✔	✔
Dashboard Navigation	✔	✔
Container	✔	✔
Date Range selector	✔	✔
Link	✔	✔
Image (except .svg files)	✔	✔
List selector	✔	✔
Number (view only)	✔	✔
Measure Range selector	✔	✔
Text	✔	✔
Text Tooltip		✔
Toggle	✔	✔
Open as lens	✔	✔
Dashboard Properties		
Custom background images	✔	✔
Themes	✔	✔
Gutter color options		
Dynamic start values	✔	✔
User tokens in queries	✔	✔
Static step, selection, and result bindings	✔	✔
Nested bindings	✔	✔

Feature	Tableau CRM for iOS	Tableau CRM for Android
Bindings using the visualizationType widget property (dynamic chart types)	✔	✔
Connected data sources	✔	✔
Customized display values	✔	✔
Image tooltips	✔	✔
Access and Security		
Configure connections to multiple instances from the login page	✔	✔
Switch between multiple Salesforce accounts without logging out	✔	
Control app access with a security passcode	✔	✔
Forget all trusted domains	✔	
Use Connected App security settings to meet industry compliance regulations	✔	✔
Get enhanced security with Mobile Device Management (MDM)	✔	✔
Miscellaneous		
Developer mode	✔	✔
Spotlight search	✔	

Download Tableau CRM for Mobile

Click the link to download Tableau CRM in the Salesforce app, or Tableau CRM for iOS and Android.

For Tableau CRM in the Salesforce app, go to Download and Log In to the Salesforce Mobile app.

For Tableau CRM for iOS, go to iTunes.

For Tableau CRM for Android, go to Google Play.

SEE ALSO:

> Which Analytics Mobile Experience Do I Use?

Learn How to Use Tableau CRM for iOS

Access your data anywhere with the Tableau CRM for iOS mobile app. For the complete online help, go to Tableau CRM for iOS. Get up and running with these online help pages.

> Log In from Tableau CRM for iOS
> Log in to Salesforce to explore data from your org.

> Learn Your Way Around Analytics Cloud for iOS
> Explore your data using the control bar at the bottom of the screen and a few simple gestures.

Log In from Tableau CRM for iOS

Log in to Salesforce to explore data from your org.

1. Tap the icon in the upper left corner to open the Select Account screen.

2. Tap ➕ **Add New Account**.

3. (Optional) To connect to a sandbox instance or to add a new connection, tap ⚙ .

4. Enter your user name and password, and then tap **Log In**.

5. Tap **Allow** to grant permission to Tableau CRM on mobile.
   Tableau CRM reloads on your device, and you'll see your pinned apps on the home screen!

Learn Your Way Around Analytics Cloud for iOS

Explore your data using the control bar at the bottom of the screen and a few simple gestures.

Tableau CRM Screen

When viewing the Tableau CRM screen, tap the header to view Recents, Alphabetical, and Pinned Apps. On the bottom of the Tableau CRM screen, choose Analytics, Tracking, or Notifications.

Notice that each asset has ⋯ . Tap it to open a panel where you can see information such as the asset's dataset or which app it's in. From the panel, you can also mark as "Favorite," Open, Get Link, or Delete, depending upon the asset and your access to it.

Tap the tile to open the app's browse screen and access the app's assets. Or, tap ⋯ on the tile to open the info/action panel. App actions can include Browse (🖼), Run (▶), Pin(📌), and Get Link (🔗).

Customize your Tableau CRM screen by pinning your favorite and often used apps. Whether you pin apps on mobile or in Tableau CRM on desktop, you'll see the same pinned apps on your Analytics Cloud screen.

If you've marked assets as "Favorite," they have a blue star (⭐), and a **Favorites** filter appears.

Viewing Dashboards

When viewing a dashboard, tap the down arrow (⌄) in the header next to the dashboard's name to open a dropdown list and select another dashboard in the app. To refresh a dashboard and return to its default state, tap ⋯ and then tap the curved arrow (↺).

Tracking

Open the notifications tracking screen by tapping 👁 . When you set notifications in Tableau CRM, the tiles for tracking progress appear here. Tap a tile to go directly to the dashboard that contains the notification.

Notifications

Open the push notifications screen by tapping 🔔 . When you set notifications in Tableau CRM, the notifications will appear here. Tap a tile to go directly to the dashboard that contains the notification.

> 📝 Note: To enable push notifications, open the device Settings, tap Notifications, and allow notifications for Tableau CRM.

Search

Tap the 🔍 in the upper right corner of the screen. Search Favorites, Apps, Dashboards, Lenses, or Datasets to find an asset.

Settings

Open Settings by tapping the icon in the upper left corner of the screen. Tap ⚙ in the upper right corner of the screen. The Settings screen is where you can:

- Switch on Presentation Mode.

- Add data to Analytics Cloud.
- Change settings, send feedback, get help, and share the Analytics Cloud App.
- Log out of your org.

A Few Simple Gestures

These gestures help you quickly navigate through Analytics Cloud.

Gesture	Result
Swipe up	Scroll down lists and dashboards.
Swipe down	- Scroll up lists and dashboards. - Refresh lists when at the top of the list. - When creating a filter in Explorer, pull down to open the text entry field for the "Contains" option.
Swipe left	- When in a paginated dashboard, navigate between screens. - When on the home screen, navigate between asset filters. - If you have permission, swiping left on an app, lens, dashboard, or dataset reveals buttons to run, pin, share, favorite, or delete, depending on the asset. - While adding measures, filters, or groupings to a lens, swipe left over an applied condition to reveal a delete button.
Swipe right	- When in a paginated dashboard, navigate between screens. - When on the home screen, navigate between asset filters.
Press (for devices with 3D Touch)	- On the device's Home screen, press the Tableau CRM app icon to open the shortcuts menu. The menu lets you share, search within Tableau CRM, or go directly to the two most recently viewed assets. - On a dashboard or lens, press lightly to peek at the asset, and press deeper to pop it open.

Learn How to Use Tableau CRM for Android

Access your data anywhere with the Tableau CRM for Android mobile app. For the complete online help, go to Tableau CRM for Android. Get up and running with these online help pages.

Log In from Tableau CRM for Android

Log in to Salesforce to explore data from your org.

Learn Your Way Around Analytics Cloud for Android

Use Explorer on Android, the new toolbar, and a few simple gestures to create meaningful visualizations, and explore and analyze your data in new ways.

Log In from Tableau CRM for Android

Log in to Salesforce to explore data from your org.

1. Tableau CRM opens to the log in screen. If you are already logged in and want to change to another org, tap ≣ to open the navigation menu. Then tap **Log out**.

2. Enter your user name and password, and then tap **Log In**.

3. Tap **Allow** to grant permission to Tableau CRM on mobile.
   Tableau CRM reloads on your device, and you'll see your pinned apps on the home screen!

Learn Your Way Around Analytics Cloud for Android

Use Explorer on Android, the new toolbar, and a few simple gestures to create meaningful visualizations, and explore and analyze your data in new ways.

Home Screen

The Home Screen is the first thing that you see when you open Analytics Cloud on your mobile device. The Home Screen consists of the header and a list of Analytics Cloud assets.

In the header, you can access:

- The lists of your apps, dashboards, lenses, and datasets —in most recently used order.
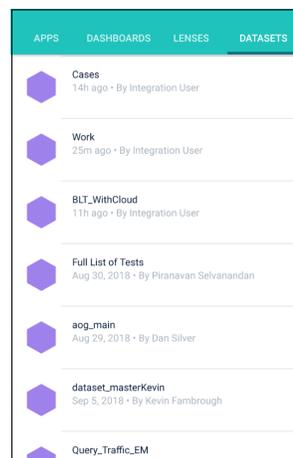
- Navigation (≣) to change settings, get help, and log out.
- Search (🔍).

Navigation Menu

Open the navigation menu from the home screen by tapping ≣. The navigation menu is where you can:

- Change settings.
- Send feedback.
- Get help.
- Log out of your org.

Lens/Dataset Exploration

At the bottom of the lens screen, you can access the toolbar to explore and filter data in lenses with the new build, filter, view, and actions tools for data insights.

A Few Simple Gestures

These gestures help you quickly navigate through Analytics Cloud.

Gesture	Result
Swipe up	Scroll down lists and dashboards.
Swipe down	• Scroll up lists and dashboards. • Refresh lists when at the top of the list.
Swipe left	• When in a paginated dashboard, flip between screens. • When on the home screen, flip between asset filters.
Swipe right	• When in a paginated dashboard, flip between screens. • When on the home screen, flip between asset filters.
Scroll	Use one finger to scroll a dashboard or a scrolling widget; use two fingers to scroll the dashboard when directly over a scrolling widget.

Set Up and Manage Tableau CRM on Mobile

Optimize dashboard layouts for mobile devices, and administer Tableau CRM for mobile.

Access Data When You're Offline (Beta)

View and analyze data when internet access isn't available.

Generate Unique Tableau CRM Dashboard Layouts for Different Devices

After you add widgets to the dashboard, optimize the layout for each device on which the dashboards can be viewed. For example, you can remove widgets from a mobile phone layout to reduce the dashboard size for the smaller screen. You can also move widgets around in one layout and it doesn't affect the other layouts.

Connected App for Tableau CRM on Mobile

Install the Tableau CRM App Package to allow your mobile clients to easily connect to your org. This package contains the Connected App component that gives you control over who's logging in and how your mobile clients share images and links.

Connected App for Wave Web

Use the Wave Web App Package when you want to manage how web clients like mobile browsers and apps display Tableau CRM content. This package contains the Wave Web Connected App component, which gives you access to and control of the Tableau CRM REST API endpoints that the app uses.

Enable Tableau CRM App Push Notifications

Push notifications let your users know when their data conditions have been met. For example, push notifications let users know when a number goal was reached or when it dipped below a specified amount.

Customize Tableau CRM for iOS using Mobile SDK

Set custom login servers and manage access to settings on Tableau CRM for iOS.

Launch Tableau CRM Assets with URLs

The Tableau CRM mobile app supports deep linking URL schemes, which allow a user to launch the app from a third-party or web application.

Access Data When You're Offline (Beta)

View and analyze data when internet access isn't available.

> **Note:** As a beta feature, Mobile Offline is a preview and isn't part of the "Services" under your master subscription agreement with Salesforce. Use this feature at your sole discretion, and make your purchase decisions only on the basis of generally available products and features. Salesforce doesn't guarantee general availability of this feature within any particular time frame or at all, and we can discontinue it at any time. This feature is for evaluation purposes only, not for production use. It's offered as is and isn't supported, and Salesforce has no liability for any harm or damage arising out of or in connection with it. All restrictions, Salesforce reservation of rights, obligations concerning the Services, and terms for related Non-Salesforce Applications and Content apply equally to your use of this feature. You can provide feedback and suggestions for Mobile Offline in the Trailblazer Community.

Contact your Salesforce account representative to enable Mobile Offline.

When enabled, users can download assets and access them without an internet connection. Users can also remove offline access for individual assets or all assets, and they can browse all assets available for offline access.

Keep these considerations in mind when enabling and using Mobile Offline.

- Mobile Offline is available in Tableau CRM for iOS using iOS version 13.3 and higher.
- Download assets for offline access when you have an internet connection. Tap the three dots next to the name of a dashboard, lens, or dataset, then tap **Make Available Offline**.
- Tableau CRM uses live data whenever internet access is available. Also, offline assets aren't automatically updated when internet access is available. Users can refresh offline data by tapping **Refresh Offline Data** when internet access is available.
- By default, offline data is downloaded using Wi-Fi only. Users can choose to download using both Wi-Fi and cellular networks. In **Settings**, tap **Offline**, and then turn off **Wi-Fi Only**.
- If assets exceed 250 MB or the storage capacity of the device, they aren't available for offline access.
- Offline settings apply to individual mobile devices and aren't synced across multiple devices.
- When users remove all offline data, all data used by an asset is removed from the device unless another offline asset uses the same data.

- All offline data is removed from the device when users log out or uninstall the app.

- Offline-enabled assets are stored on the device using Salesforce encryption standards for mobile devices. Unencrypted data is never stored on a device at any time.

- Control access to Mobile Offline in connected apps by setting the `OFFLINE_BETA` flag to false in the connected app. To learn more, refer to Connected App for Tableau CRM on Mobile.

- If an app is terminated while an asset is being processed for offline access, the process won't resume automatically the next time the app is launched. Users must manually restart the process.

- Some features aren't supported.

  - Image widgets, background images of dashboards or container widgets, and images in tables don't appear in offline assets. Also, leaderboards or charts with icons don't show their icons. Otherwise, dashboards, charts, and tables appear normally.

  - Dashboards and lenses with custom maps can be marked for offline use, but custom maps don't show in offline assets.

  - Navigating to a dashboard or lens that's linked in the current offline dashboard doesn't work unless the other dashboard or lens is also marked for offline access.

  - Saved personal views of dashboards aren't accessible. When viewing offline, the dashboard shows in its original view.

  - Saving lenses in offline assets isn't supported.

  - Sharing snapshots or links by posting to Chatter or sharing via email isn't available for offline assets.

  - Assets that use APEX or SOQL queries can't be marked for offline.

Generate Unique Tableau CRM Dashboard Layouts for Different Devices

After you add widgets to the dashboard, optimize the layout for each device on which the dashboards can be viewed. For example, you can remove widgets from a mobile phone layout to reduce the dashboard size for the smaller screen. You can also move widgets around in one layout and it doesn't affect the other layouts.

Watch a Demo: ▶ Optimize Wave Dashboards for Devices with Layouts (English Only)

When you generate a new layout, Tableau CRM adds the widgets from the currently selected layout to the new layout. If the current layout contains multiple pages, Tableau CRM also copies the pages to the new layout.

📝 Note: Although you can add all chart types to a mobile layout, some aren't supported on mobile devices. For more information, see Tableau CRM mobile limitations.

1. Open in the dashboard designer dashboard.

2. Because each layout can contain different widgets, open the layout that contains the widgets that you want to include in your new layout.

3. From the Layouts menu, select **Manage Layouts**.

4. Select the layout template, and enter a unique layout name.

Each layout template comes with predefined layout properties that you can change.

5. Click **Generate**.

The layout properties appear for the new layout. If needed, click ⚙ to show the Layout panel.

6. In the Layout panel, expand the following sections and change the default properties, if needed.

General
Set the layout name, designer grid settings, and background color.

Device
Specify information about the devices that can use this layout. For more information about how Tableau CRM uses these properties to choose the right layout, see Rules for Choosing a Layout for a Device.

Background Image
To apply a background image to the entire dashboard when this layout is used, enter the details about the background image. You don't have to include a background image.

The designer previews the dashboard and layout based on the layout property settings. The designer updates the preview in real time so that you can see how your changes affect the display. If you shrink the maximum dashboard width, the designer shows you how the dashboard fits when using this layout. If needed, the designer rearranges the widgets to fit the new size.

7. Rearrange the widgets, if needed.

8. To hide a widget from the layout, select the widget, and click .
   If you hide a widget, it appears in the Unused Widgets section of the layout properties.

9. To add an unused widget to the layout, drag the widget from the Unused Widgets section to the canvas.

10. To create a widget, drag the widget from the widget toolbar to the canvas.

> 📝 **Note:** Tableau CRM adds the widget only to the current layout. To add the widget to another layout, open the other layout and drag the widget from the "Unused Widgets" section of the layout properties.

11. To save your layout changes, save the dashboard.

12. To delete a layout, in the Layouts menu, click **Manage Layouts**, and then click ✕ next to the layout.

> 📝 **Note:** You can't delete the layout that's currently open. To delete the current layout, switch to another one first.

Layout Properties for Tableau CRM Dashboards

Layout properties specify the layout name, designer grid settings, background settings, and requirements for devices that can use this layout. The properties apply only to dashboards created in the dashboard designer.

Rules for Choosing a Layout for a Device

If multiple layouts are defined for a dashboard, Tableau CRM chooses the optimal layout when displaying the dashboard on a device. To determine the optimal layout, Tableau CRM uses the device properties specified for each layout.

Connected App for Tableau CRM on Mobile

Install the Tableau CRM App Package to allow your mobile clients to easily connect to your org. This package contains the Connected App component that gives you control over who's logging in and how your mobile clients share images and links.

1. While logged in as an admin, navigate to:

   Production

   https://login.salesforce.com/packaging/installPackage.apexp?p0=04tB0000000cHCH

   Sandbox

   https://test.salesforce.com/packaging/installPackage.apexp?p0=04tB0000000cHCH

2. Select **Install for All Users**.

   If you install the Connected App only for certain roles, then only those clients are subject to your policies.

3. Click **Done**, then click the Salesforce Analytics App Package and then **View Components**.

4. Select iOS or Android. To configure policies and other settings, click **Edit**. When you're finished, save your changes.

   For more information about Connected App settings, see Edit a Connected App.

5. If you want to disable sharing options for mobile users, click **New** in the Custom Attributes section.

   All sharing options are enabled by default. To disable an option, add its attribute key and enter "`false`" for the attribute value. See the table for the supported attribute keys.

Attribute Key	Mobile clients can	Mobile Device
CACHE_QUERY_RESULTS	Store query cache results to disk	iOS
OFFLINE_BETA	Store data for offline access	iOS
SHARE_IMAGE_AIRDROP	Share images via AirDrop	iOS
SHARE_IMAGE_CAMERAROLL	Share images to Camera Roll	iOS
SHARE_IMAGE_OPEN	Share images to other apps and activities	iOS
SHARE_IMAGE_S1	Share images to Salesforce	iOS
SHARE_IMAGE_EMAIL	Share images via email	iOS

Attribute Key	Mobile clients can	Mobile Device
SHARE_LINK_NFC	Share links via Near Field Communication (NFC)	Android
SHARE_LINK_BLUETOOTH	Share links via Bluetooth	Android
SHARE_LINK_AIRDROP	Share links via AirDrop	iOS
SHARE_LINK_CLIPBOARD	Share links to Clipboard	iOS and Android
SHARE_LINK_S1	Share links to Salesforce	iOS and Android
SHARE_LINK_EMAIL	Share links via email	iOS and Android
SPOTLIGHT_SEARCH	See Tableau CRM assets in Spotlight search results	iOS

Note: To apply custom attributes to connected users, you can revoke them and ask them to reconnect.

6. To view and control how mobile clients connect to your Tableau CRM-enabled org, from Setup, in the `Quick Find` box, enter `Connected Apps`, and then select **Connect Apps OAuth Usage**.

You can block user sessions, revoke individual users, and drill in to user details. For more information, see Monitoring Usage for a Connected App.

Note: For certificate-based authentication, users must set their desktop browsers to allow cookies.

For extra security, use Mobile Device Management (MDM) with the option to show only authorized hosts. The "OnlyShowAuthorizedHosts" key allows you to hide the "Add New Account" plus sign in the mobile app. For more information, see Using MDM with Salesforce Mobile SDK Apps.

Connected App for Wave Web

Use the Wave Web App Package when you want to manage how web clients like mobile browsers and apps display Tableau CRM content. This package contains the Wave Web Connected App component, which gives you access to and control of the Tableau CRM REST API endpoints that the app uses.

1. The Wave Web Connected App is installed automatically when Tableau CRM is enabled. To install it if you don't have it, while logged in as an admin, navigate to:

   Production
   https://login.salesforce.com/packaging/installPackage.apexp?p0=04tj00000000YSf

   Sandbox
   https://test.salesforce.com/packaging/installPackage.apexp?p0=04tj00000000YSf

2. Select **Install for All Users**.

   If you install the Connected App only for certain roles, then only those clients are subject to your policies.

3. Click **Done**, then click the Wave Web App Package and then **View Components**. The app package contains the Wave Web Connected App.

4. Use the App Manager to change the Wave Web Connected App configuration.

   For more information about Connected App settings, see Edit a Connected App.

5. To view and control how clients connect to your Tableau CRM-enabled org, from Setup, in the `Quick Find` box, enter `Connected Apps`, and then select **Connect Apps OAuth Usage**.

   You can block user sessions, revoke individual users, and drill in to user details. For more information, see Monitoring Usage for a Connected App.

   📝 **Note:** For certificate-based authentication, users must set their desktop browsers to allow cookies.

For issues with loading Tableau CRM dashboards in the Salesforce Mobile app when continuous IP restrictions are enabled, you can relax the IP restrictions for the Wave Web Connected App to allow users to log in to the specified app regardless of IP restrictions set for the user's profile. For more information, see Connected App IP Relaxation and Continuous IP Enforcement.

Enable Tableau CRM App Push Notifications

Push notifications let your users know when their data conditions have been met. For example, push notifications let users know when a number goal was reached or when it dipped below a specified amount.

To enable full content for notifications, follow these steps:

1. Install the latest Connected Apps package for Tableau CRM Mobile. For more information, see Connected App for Tableau CRM on Mobile.

   📝 **Note:** Even if you have a previous version of the Connected Apps package installed, you must install the latest version to get Push Notifications.

2. From Setup, enter `Connected Apps` in the Quick Find box, then select **Manage Connected Apps**.

3. Choose the Connected App you want to edit. If your org uses multiple Connected Apps, complete the steps for each one.

4. If you're authorized to do so for your company, select **Display full content push notifications**.

5. Click **Save**.

> 📝 **Note:** Advise users to enable push notifications by opening the device Settings, tapping Notifications, and allowing notifications for Analytics.

These types of notifications can appear to Tableau CRM app users.

- Push Notifications: Alerts that appear on a mobile device when a user has installed Tableau CRM and Tableau CRM for iOS but isn't using it. These alerts can consist of text, icons, and sounds, depending on the device type. If an administrator enables push notifications for your organization, users can choose individually whether to receive push notifications on their devices.

- In-app Notifications: Alerts that keep users aware of relevant activity while they're using the Tableau CRM app. By tapping 🔵, a user can view the 20 most recent notifications received within the last 90 days.

- Tracking Screen: Screen where you can see progress toward goals. When setting notifications on dashboard widgets, enable tracking to view charts on this screen.

> 📝 **Note:** Notifications may include text about a certain metric and its current and threshold values. To ensure that sensitive information isn't distributed through a third-party service without proper authorization, push notifications include minimal content unless you enable full content in push notifications.
>
> For example, suppose an in-app notification reads: "'Quota Remaining' met conditions: Amount is 265K and is greater than 250K" By default, the equivalent push notification would be "Notification conditions met." However, if you enabled full content in push notifications, this push notification would include the same (full) content as the in-app notification.

Customize Tableau CRM for iOS using Mobile SDK

Set custom login servers and manage access to settings on Tableau CRM for iOS.

The following set up options are available for Tableau CRM for iOS.

Set Custom Login Servers

Manage Access to Settings

Launch Tableau CRM Assets with URLs

The Tableau CRM mobile app supports deep linking URL schemes, which allow a user to launch the app from a third-party or web application.

An administrator or developer can customize a URL scheme to perform one of the following actions when the app launches:

- View a particular asset type in the Tableau CRM app (dashboard, lens, or application)

- Apply a runtime state (selections and filters)

For example, you can create a deep link to a dashboard in the Tableau CRM app. Then when users receive the link in an email, they can click it to go directly to the dashboard.

> 📝 **Note:** Some deep link scheme features, such as passing filters and selections, apply only to dashboards. Support is only for dashboards that are built using the dashboard designer.

URL Scheme Format

Use the following format to make a request to a Tableau CRM app.

```
<scheme_name>://<assetType>/<assetID>?orgId=<orgId>&loginHost=<loginHost>&dashboardState=<url-encoded json>
```

The following parameters are supported.

Variable	Type	Description	Required
scheme_name	String	The value must be `sfanalytics`.	Yes

assetType	String	Supported values:
		
		`dashboard`
		
		`lens`
		
		`application`
			Yes

assetID	String	Unique asset identifier. The identifier is the 18-character code beginning with 0FK found in the asset URL.	Yes
orgID	String	ID of the organization that contains the content. If users are logged in to the same org ID, they are directed to the content. If users are not logged in to the same org ID, they are redirected to the login screen.	Yes
loginHost	String	URL that displays the login screen when the user is not logged in to the application.	Yes
dashboardState	String	Available only when `assetType` is `dashboard`. State of the dashboard and all its datasets. The state depends on user selection, start values, global filters, and external or initial filters that are applicable to the dataset.	No

Example (not including `dashboardState`):

`sfanalytics://dashboard/0FKB00000006TFVOA2?orgId=00DB00000000XXX&loginHost=cs4.salesforce.com`

Refer to the information in the following tables when constructing the JSON string for `dashboardState`.

dashboardState

Property Name	Type	Description
datasets	Map<String, state[]>	The list of datasets for which a specific state is applied. The key string is the API name of the dataset.

state

Property Name	Type	Description
fields	String[]	The list of dimension or measure fields for which selection or filtering is applied.
filter	dashboardFilter	The filter that is applied to the dashboard.
selection	String[]	The list of values that is selected.

dashboardFilter

Property Name	Type	Description
operator	String	Supported operators for dimensions: "in", "not in", "matches"
		Supported operators for measures: "==", "!=", ">=", ">", "<=", "<"
values	String[] or Double[]	The list of dimension or measure values corresponding to the fields.

JSON Examples

The following syntax causes `value1` and `value2` to be selected on `field1`. It also filters on `field2` based on `operator1` with `values3` and `values4`.

```
{'datasets' : {'dataset1': [ {'fields': ['field1'], 'selection': ['$value1', '$value2']},
 {'fields': ['field2'], 'filter': { 'operator': 'operator1', 'values': ['$value3',
'$value4']}}]}}
```

Example of the `dashboardState` JSON.

```
{
    "datasets": {
        "Opportunities": [
            {
                "fields": ["OpptyName"],
                "selection": ["RelateIQ", "BeyondCore"],
                "filter": {
                    "operator": "not in",
                    "values": ["Salesforce"]
                }
            },
            {
                "fields": ["State", "Country"],
                "selection": [["TX", "US"]],
                "filter": {
                    "operator": "in",
                    "values": [["TX, US"], ["AL", "DE"]]
                }
            }
        ],
        "Accounts": [
            {
                "fields": ["Revenue"],
                "selection": [],
                "filter": {
                    "operator": ">=",
                    "values": [[1000000]]
                }
            }
        ]
    }
}
```

The JSON must be rewritten with no whitespace characters.

{"datasets":{"Opportunities":[{"fields":["OpptyName"],"selection":["RelateIQ","BeyondCore"],"filter":{"operator":"not

in","values":["Salesforce"]}},{"fields":["State","Country"],"selection":[["TX","US"]],"filter":{"operator":"in","values":[["TX,

US"],["AL","DE"]]}}],"Accounts":[{"fields":["Revenue"],"selection":[],"filter":{"operator":">=","values":[[1000000]]}}]}}

Finally, the JSON must be rewritten with URL encoding.

(unreadable URL-encoded JSON string)

With this URL encoded JSON, the example earlier in this topic is modified as follows.

```
sfanalytics://dashboard/0FKB00000006TFVOA2?orgId=00DB00000000XXX
```

(unreadable URL-encoded JSON string)

Limitations of Analytics Cloud on iPhone® and iPad®

Be aware of these mobile-specific limitations.

General Limitations

Feature	Limitations
Layout	Landscape mode is supported only for iPad. Portrait mode is supported for both iPad and iPhone. Split screen isn't supported.
Share	You can't create, edit, or delete sharing permissions for apps.
Delete	• Deleting a dashboard or lens in Analytics Cloud on your mobile device also deletes it in your Salesforce org on the Web. • Once deleted, a dashboard or lens can't be recovered. The only exception is the playground sample data, which can be restored from the Settings screen. • Datasets can't be deleted from your mobile device.
Playground	Imported or created data, lenses, and dashboards aren't backed up in a device backup. Playground sample data can be restored from the Settings screen.
Favorites	Pinned apps are shared across all your logged in devices. Favorites of dashboards, lenses, and datasets are stored locally on your mobile device and aren't synced to other devices.
Other Features	• Creating apps and dashboards isn't supported. • Custom bulk actions via Visualforce aren't supported. • Offline mode isn't available.

Data Exploration and Visualization Limitations

Feature	Limitations
Charts and Tables	• Charts don't support multiple selection actions such as copy and focus. • In Compare Tables, summary rows aren't shown.
Widgets	**Image** The image widget doesn't support .svg files. **Number** Exploring a number widget isn't supported.
Queries	• If more measures or groupings are added to a chart than the chart type supports, then the chart becomes a Compare Table. • The maximum results returned by a query vary based on its type. For example, the default query limit for an aggregate query with multiple groupings is 500, so just the first 500 records are displayed by default. The default can be changed in the dashboard designer in Tableau CRM Studio. • Apex steps (beta) aren't supported on mobile.
Dashboards designed in the Tableau CRM Studio	• Analytics Cloud dashboards designed in your Salesforce org on the Web can be viewed in Analytics Cloud for iOS on your local device. Some widgets appear different on mobile. • To optimize dashboards for mobile, generate a layout for phone or tablet in the dashboard designer. • Gutter color options aren't supported.

Data Import Limitations

Feature	Limitations
Tableau CRM Data Connector	• Only CSV, TSV, and XLSX files, and ZIP files containing these types, can be imported as datasets. • Imported datasets aren't available in Analytics Cloud on the Web. They are only available on your mobile device in the playground. • Imported datasets aren't encrypted.
Date Formats	Dates can be imported as any of these formats: • YYYY • YYYY-MM • YY-MM • YYYY-MM-DD • YY-MM-DD • YYYYMMDD • YYMMDD • YYDDMM • MM/DD/YY

Feature	Limitations
	• DD/MM/YY
	• YY/MM/DD
	• YY-MM-DD HH:MM:SS
	• YYYYMMDDHHMMSS
	• YYMMDDHHMMSS
	• YY-MM-DDTHH:MM:SS
	• YYYY-MM-DDTHH:MM:SSZ
	• YY-MM-DDTHH:MM:SSZ
	• YYYY-MM-DD HH:MM:SSZ
	• YY-MM-DD HH:MM:SSZ
	• YYYYMMDDTHH:MM:SSZ
	• YYMMDDTHH:MM:SSZ
	• YYYYMMDD HH:MM:SSZ
	• YYYYDDMM
	• YYYY/MM/DD
	• YYYY/DD/MM
	• DD/MM/YYYY
	• MM/DD/YYYY
	• YYYY-DD-MM
	• DD-MM-YYYY
	• MM-DD-YYYY
	• DD.MM.YYYY
	• DD.MM.YYYY HH:MM:SS
	• DD.MM.YY HH:MM:SS
	• MM.DD.YYYY
	• DD-Mon-YY
	• DD-Mon-YYYY
	• DD-Mon-YYYY HH:mm
	• YYYYMMDD HH:MM:SS
	• YYYY/MM/DD HH:MM:SS
	• MM/DD/YYYY HH:MM:SS
	• MM/DD/YYYY HH:MM:SS AM/PM
	• MM/DD/YY HH:MM:SS AM/PM
	• DD/MM/YYYY HH:MM:SS
	• YYYY/MM/DD HH:MM
	• MM/DD/YYYY HH:MM
	• DD/MM/YYYY HH:MM
	• DD/MM/YY HH:MM:SS

Feature	Limitations
	• DD/MM/YYYY HH:MM:SS AM/PM
	• DD/MM/YY HH:MM:SS AM/PM
	• YYYY-MM-DD HH:MM:SS
	• YY-MM-DD HH:MM:SS
	• DD-MM-YYYY HH:MM:SS
	• DD-MM-YY HH:MM:SS
	• DD-MM-YYYY HH:MM:SS AM/PM
	• DD-MM-YY HH:MM:SS AM/PM
	• MM-DD-YYYY HH:MM:SS
	• MM-DD-YYYY HH:MM:SS AM/PM
	• MM-DD-YY HH:MM:SS AM/PM
	• YYYY-MM-DD HH:MM
	• DD-MM-YYYY HH:MM
	• MM-DD-YYYY HH:MM
	• YYYY-MM-DDTHH:MM:SS
	• DD-MM-YYYYTHH:MM:SS
	• MM-DD-YYYYTHH:MM:SS
	• YYYY-MM-DDTHH:MM
	• DD-MM-YYYYTHH:MM
	• MM-DD-YYYYTHH:MM
	• YYYY-MM-DDTHH:MM:SS-HH:MM
	• YYYY-MM-DDTHH:MM:SS+HH:MM
	• YYYY-MM-DDTHH:MM:SSTZD
	• YYYYMMDDTHH:MM:SS
	• HH:MM:SS DD/MM/YYYY
	• HH:MM:SS DD/MM/YY
	• DD-MM-YYYYTHH

Limitations of Analytics Cloud on Android Devices

Be aware of these mobile-specific limitations.

General Limitations

Feature	Limitations
Layout	Landscape mode is supported only for tablets. Portrait mode is supported for both tablets and phones. Split screen isn't supported.
Share	You can't create, edit, or delete sharing permissions for apps.

Feature	Limitations
Save and clone	You can't save or clone lenses.
Delete	You can't delete datasets or lenses from your mobile device.
Import	You can't import data to your mobile device.
Playground	Local playground data isn't supported.
Favorites	Pinned apps are shared across all your logged in devices. Favorites of dashboards, lenses, and datasets are stored locally on your mobile device and aren't synced to other devices.
Links	You can link to external sites from Tableau CRM assets but not from external sites.
Other Features	- Creating apps and dashboards isn't supported. - Custom bulk actions via Visualforce aren't supported. - Offline mode isn't available. - To switch orgs, you need to log out first. - You can't view item metadata.

Data Exploration and Visualization Limitations

Feature	Limitations
Charts and Tables	- Charts don't support multiple selection actions such as copy and focus. - In Compare Tables, summary rows aren't shown. - Pivot tables aren't supported. - Sankey charts aren't supported. - Rating charts aren't supported.
Widgets	**Image** The image widget doesn't support .svg files. **Number** Exploring a number widget isn't supported.
Queries	- If more measures or groupings are added to a chart than the chart type supports, then the chart becomes a Compare Table. - The maximum results returned by a query vary based on its type. For example, the default query limit for an aggregate query with multiple groupings is 500, so just the first 500 records are displayed by default. The default can be changed in the dashboard designer in Tableau CRM Studio. - Apex steps (beta) aren't supported on mobile.
Dashboards designed in the Tableau CRM Studio	- Analytics Cloud dashboards designed in your Salesforce org on the Web can be viewed in Analytics Cloud for Android on your local device. Some widgets appear different on mobile. - To optimize dashboards for mobile, generate a layout for phone or tablet in the dashboard designer.

Feature	Limitations
	• Gutter color options aren't supported.

Set Up the Tableau CRM Platform

Set up your organization to use Tableau CRM and assign basic permission sets or create and assign your own custom permission sets.

The information here describes how to set up your organization to use Tableau CRM and Tableau CRM templated apps and Apps.

1. Learn About Tableau CRM Platform Licenses and Permission Sets

   Introduce yourself to Tableau CRM setup fundamentals, which apply to both basic and advanced setup procedures.

2. Learn About Internal Analytics Users

   Analytics uses internal users, Integration User and Security User, to access Salesforce data, preview data in Data Prep recipes, and enforce row-level security on datasets.

3. Basic Tableau CRM Platform Setup

   Follow these few steps to give users in your organization access to Tableau CRM platform, templates, and apps quickly. Basic setup uses permission sets included with the Tableau CRM Plus license, which cover the majority of most organizations' analytics needs.

4. Advanced Tableau CRM Platform Setup

   Follow these steps to learn the details of Tableau CRM permissions and to create and assign your own custom permission sets.

5. Complete Setting up the Tableau CRM Platform

   After assigning permission sets with either basic or advanced setup, follow these optional procedures to enhance the Tableau CRM experience and fine-tune Tableau CRM access to Salesforce data.

6. Tableau CRM Requirements

   This section provides requirements for using Tableau CRM.

7. Tableau CRM Limits

   This section describes Tableau CRM limits.

8. Tableau CRM Limitations

   Tableau CRM differs from other Salesforce features in some ways.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer Edition**.

SEE ALSO:

Set Up the Tableau CRM Platform with Licenses Purchased Before October 20, 2015

Migrating From Tableau CRM Licenses Purchased Before 10/20/2015 to New Tableau CRM Platform Licenses

Tableau CRM Requirements

Tableau CRM Limits

Tableau CRM Limitations

Learn About Tableau CRM Platform Licenses and Permission Sets

Introduce yourself to Tableau CRM setup fundamentals, which apply to both basic and advanced setup procedures.

Each Tableau CRM Growth and Tableau CRM Plus license is a single-user license that provides access to Tableau CRM. The license limits your instance of the Tableau CRM to 1 billion rows of data. If you require more data, you can purchase Tableau CRM - Additional Data Rows, which entitles you to 100 million rows more.

🛑 **Important:**

- Tableau CRM license data storage limits are contractual, not technical. Licensee agrees to strictly monitor its total number of data rows.
- If you purchased an Tableau CRM Growth license before October 20, 2015 with Analytics Cloud Builder or Analytics Cloud Explorer permission set licenses: Read Set up the Tableau CRM Platform With Licenses Purchased Before October 20, 2015. If you're migrating users from Builder or Explorer licenses to the new Tableau CRM Growth license: See Migrating From Tableau CRM Licenses Purchased Before 10/20/2015 to New Tableau CRM Platform Licenses before you start the setup process for those users.
- If you disable Tableau CRM, user permissions are removed from each defined permission set. If you re-enable Tableau CRM later, you must define the permission sets again.

Tableau CRM Plus License Permission Sets

The Tableau CRM Growth license includes two prebuilt permission sets:

- **Tableau CRM Plus Admin** enables all permissions required to administer the Tableau CRM platform, including permissions to enable creating Tableau CRM templated apps and Apps.
- **Tableau CRM Plus User** enables all permissions required to use the Tableau CRM platform and Tableau CRM templated apps and Apps

The Tableau CRM Plus license includes two prebuilt permission sets:

- **Tableau CRM Plus Admin** enables all permissions required to administer the Tableau CRM platform and Einstein Discovery, including permissions to enable creating Tableau CRM templated apps and Apps.
- **Tableau CRM Plus User** enables all permissions required to use the Tableau CRM platform, Einstein Discovery, and Tableau CRM templated apps and Apps

Basic setup involves choosing and assigning the prebuilt permission sets. To create your own custom Tableau CRM permission sets, use advanced setup.

When you assign any Tableau CRM permission set to users in your org, Salesforce *auto-assigns* the Tableau CRM Growth permission set license to that user.

📝 **Note:** For efficiency, you can assign a permission set to groups of users. You can also assign multiple permission sets to a user.

You can assign a Tableau CRM permission set license along with any of the following Salesforce user licenses:

- Lightning Platform (app subscription)
- Lightning Platform (one app)
- Full CRM
- Salesforce Platform

- Salesforce Platform One

SEE ALSO:

Tableau CRM Limitations

Tableau CRM Limitations

Learn About Internal Analytics Users

Analytics uses internal users, Integration User and Security User, to access Salesforce data, preview data in Data Prep recipes, and enforce row-level security on datasets.

Analytics uses the permissions of the Integration User to extract data from Salesforce objects and fields when a dataflow or recipe job runs. Because the Integration User has View All Data access, consider restricting access to particular objects and fields that contain sensitive data. If the dataflow or recipe is configured to extract data from an object or field on which the Integration User does not have permission, the job fails. The Integration User permissions restrict the data extracted from Salesforce only—they don't affect access to the data in datasets. To restrict user access to data in datasets, set up row-level security.

To enable the interactive preview in recipes, Data Prep uses the Security User. When a user previews the results of a recipe, Data Prep shows only the results that the logged-in user has permission to access. The permissions of the Security User don't affect the data shown in the preview.

When you query a dataset that has row-level security based on the Salesforce User object, Analytics uses the permissions of the Security User to access the User object and its fields. The Security User must have at least read permission on each User object field included in a predicate. (A predicate is a filter condition that defines row-level security for a dataset.) By default, the Security User has read permission on all standard fields of the User object. If the predicate is based on a custom field, then grant the Security User read access on the field. If the Security User does not have read access on all User object fields included in a predicate expression, an error appears when you try to query the dataset using that predicate.

> ⊘ **Important:** Because Analytics requires the Integration User and Security User to access Salesforce data and preview recipe changes to data, do not delete either of these users.

SEE ALSO:

Control Access to Salesforce Objects and Fields

Set Up Dataset Security to Control Access to Rows

Basic Tableau CRM Platform Setup

Follow these few steps to give users in your organization access to Tableau CRM platform, templates, and apps quickly. Basic setup uses permission sets included with the Tableau CRM Plus license, which cover the majority of most organizations' analytics needs.

Enable Analytics Cloud

1. Go to Salesforce Setup and enter *Analytics* in the **Quick Find / Search** field.

2. Select **Getting Started**.

3. Click **Enable Einstein Analytics**.

Select and Assign Permission Sets

1. In Salesforce Setup, select **Users**.

2. Select **Permission Sets**.

3. Select one of the following prebuilt permission sets.

   - If you and users of your Salesforce org have Tableau CRM Plus Licenses:

     – **Tableau CRM Plus Admin** enables all permissions required to administer the Tableau CRM platform, including permissions to enable creating Tableau CRM templated apps and Apps.

     – **Tableau CRM Plus User** enables all permissions required to use the Tableau CRM platform and Tableau CRM templated apps and Apps

   - If you and users of your Salesforce org have Tableau CRM Plus Licenses:

     – **Tableau CRM Plus Admin** enables all permissions required to administer the Tableau CRM platform and Einstein Discovery, including permissions to enable creating Tableau CRM templated apps and Apps.

     – **Tableau CRM Plus User** enables all permissions required to use the Tableau CRM platform, Einstein Discovery, and Tableau CRM templated apps and Apps

   That opens the page for the selected permission set.

4. Click **Manage Assignments**. The next page shows the users already assigned the selected permission set.

5. Click **Add Assignments** to see all the users in your org.

6. Select one or more users by checking the box next to their name in the left-hand column.

7. Click **Assign**.

You see a page that tells you if the permission set has been assigned. If permission set didn't succeed for one or more users, Salesforce tells you why. You can fix the problem and try again.

Advanced Tableau CRM Platform Setup

Follow these steps to learn the details of Tableau CRM permissions and to create and assign your own custom permission sets.

💡 Tip: For best results, follow the steps in the order shown.

1. Learn About Tableau CRM Permission Set Licenses and User Permissions

   When you create your own Tableau CRM permission sets, you start by selecting the Tableau CRM Growth permission set license instead of using the prebuilt permission sets. The permission set license includes the user permissions required to explore data with and manage the Tableau CRM platform.

2. Identify Analytics Cloud Platform User Types

   Identifying types of Tableau CRM platform users helps assure that custom permission sets meet your team's analytics needs.

3. Enable Analytics Cloud and Create Permission Sets

   After identifying user types, be sure to enable Analytics Cloud in your org. Then select a permission set license and create your permission sets.

4. Assign Analytics Cloud Permission Sets to Users

   Assign custom permission sets to one or more users—either one at a time, or in bulk—to give them access to Tableau CRM functionality.

Learn About Tableau CRM Permission Set Licenses and User Permissions

When you create your own Tableau CRM permission sets, you start by selecting the Tableau CRM Growth permission set license instead of using the prebuilt permission sets. The permission set license includes the user permissions required to explore data with and manage the Tableau CRM platform.

- Each Tableau CRM platform user needs an Tableau CRM Growth permission set license to use the Tableau CRM platform. Creating a permission set and assigning it to a user *auto-assigns* the Tableau CRM platform permission set license to that user. You can also manually assign permission set licenses to users.

- Experience Cloud site users need an Tableau CRM for Communities permission set license to view Tableau CRM apps shared via dashboards embedded in sites or Visualforce pages.

The Tableau CRM Growth permission set license enables the following permissions, except Manage Analytics Private Assets.

User Permission	What It Enables
Access Analytics SAQL Editor (Pilot Feature)	This user permission is no longer used and will be removed in a future release.
Add Analytics Remote Connections	Add connections to access data from external data sources.
Adoption Analytics Templates and Apps	Create and use apps based on the Adoption Analytics template.
Create Analytics Apps	Create, edit, delete, and share Analytics Cloud applications.
Create and Edit Analytics Dashboards	Create, edit, and delete Analytics Cloud dashboards.
Download Analytics Data	Download screenshots and data in tabular format through the Analytics Cloud user interface.
Edit Analytics Dataflows	Edit, delete, and use remote connections; add and remove connected objects; run and schedule data sync; create, edit, delete, run, schedule, and monitordataflows and recipes. Use discretion when assigning this user permission because it enables access to all Salesforce object data to which the Integration User has access. See Salesforce Data Access in Tableau CRM.
Edit Dataset Recipes	Create, edit, and run recipes to create datasets. Monitor dataflow and system jobs in the monitor. Doesn't enable editing security predicates in existing recipes, or running and scheduling recipes based on datasets that have security predicates.
Manage Analytics Private Assets	Delete datasets from another user's My Private App. See Delete a Dataset.
	📝 Note: This user permission isn't included in any of the packaged permission sets. You have to manually assign it to a permission set. To add this user permission to a permission set in a Sandbox org, the permission set can't have a license assigned to it.
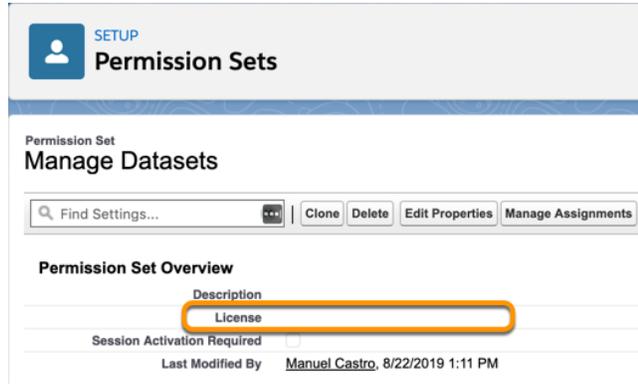
If a license is assigned to the permission set, the Manage Analytics Private Assets user permission doesn't appear when adding user permissions to permission set.

Manage Analytics	Access all Analytics Cloud features. Provides Tableau CRM administrator-level capabilities.
Manage Analytics Custom Maps	Add, edit, and delete maps, and upload geoJSON files.
Manage Analytics Templated Apps	Create and manage apps based on Tableau CRM Templates. Edit Tableau CRM dataflows. Edit recipes. Monitor dataflow and system jobs in the monitor. Some templates require extra permissions. See the help for each template.
Trend Report Data in Analytics	Trend the data of a report with a Tableau CRM dataflow.
Upload External Data to Analytics	Upload external data to Analytics Cloud to create a dataset. Monitor dataflow and system jobs in the monitor.
Use Analytics	Open Tableau CRM; run Analytics Cloud apps to which you have access; view their datasets, lenses, and dashboards; and create and edit lenses.
Use Analytics Templated Apps	Use apps based on Tableau CRM Templates. Can also create, edit, and delete Analytics Cloud dashboards. Some templates require extra permissions. See the help for each template.

You create your own permission sets based on these permissions.

📝 Note: When you create permission sets for Tableau CRM users, selecting any Analytics Cloud permission automatically enables the "Use Analytics Cloud" permission as well. Also, the type of access granted on an app controls the actions that can be performed on its datasets, lenses, and dashboards. For example, if a user has the "Use Analytics" permission, the user must also have Viewer access on an app to view its datasets, lenses, and dashboards.

The Tableau CRM for Communities permission set license enables the "View Analytics on Communities pages" permission. That permission enables external users to view Tableau CRM dashboards embedded in their Experience Cloud sites. See Enable Tableau CRM for Experience Cloud Sites on page 559.

Identify Analytics Cloud Platform User Types

Identifying types of Tableau CRM platform users helps assure that custom permission sets meet your team's analytics needs.

The Tableau CRM Growth license includes the permissions that enable users in your organization to import, work with, and explore data using the Tableau CRM platform. The license also includes permissions that let administrators manage Tableau CRM. To set up Tableau CRM in the most useful way, think through the different ways people access Tableau CRM features and categorize them into user types. Here are a few examples.

Most people on your team fall into two basic categories:

- Users, who view Tableau CRM dashboards, lenses, and datasets and occasionally import data to Tableau CRM.
- Administrators or managers, who access to the full range of Tableau CRM functionality to create apps, dashboards, lenses, and datasets, edit data, and otherwise customize the Tableau CRM experience.

Accordingly, when you set up the Tableau CRM platform, creating two basic permission sets can meet the needs of most of your organization:

- A "view" permission set that contains the permissions needed by most basic-level users when they access Tableau CRM.
- A "manage" permission set that's reserved for a select few administrators and managers and contains permissions to use all Tableau CRM features

You can set up Tableau CRM however you want, defining other user types and permissions sets based on the needs of users in your organization. For example, you can create a "superuser" permission set for users who create and share apps and dashboards. Or, a "data wizard" permission set would enable other users to work closely with datasets.

We focus on two main types of people on your team -- users and administrators -- in the example permission sets we describe in our detailed setup instructions.

Enable Analytics Cloud and Create Permission Sets

After identifying user types, be sure to enable Analytics Cloud in your org. Then select a permission set license and create your permission sets.

Enable the Tableau CRM Platform

To use the Tableau CRM platform, first enable it for your organization.

1. In the Salesforce Setup menu, under Administer, click **Analytics | Getting Started**.

2. Click **Enable Einstein Analytics**.

Create Permission Sets

The next step in the setup process is to create permission sets made up of Tableau CRM user permissions. We show you how to create two permission sets:

- *Manage Tableau CRM* permission set, for users who create and administer Tableau CRM.
- *View Tableau CRM* permission set, for users who explore app dashboards and datasets.

📝 Note:

- Customers who purchased Tableau CRM before October 20, 2015: See Set up the Analytics Cloud Platform With Licenses Purchased Before October 20, 2015 before following the instructions here.

- The Manage Analytics Cloud permission set enables the equivalent functionality provided by the previous Analytics Builder license (purchased before October 20, 2015). If you have a Builder license or you're migrating to the current Analytics Cloud platform license, follow these steps to create a permission set with the same functionality.
- The View Analytics Cloud permission set enables the equivalent functionality provided by the previous Analytics Cloud Explorer license (purchased before October 20, 2015). If you have an Explorer license or you're migrating to the current Analytics Cloud platform license, follow these steps to create a permission set with the same functionality.

1. In the Setup menu, under administer, click **Manage Users | Permission Sets** and then click **New**.

2. Enter `Manage Tableau CRM` or `View Tableau CRM` in the Label field, as appropriate. This automatically creates the API name as well. You can change it if you like, but it can't include spaces, end with an underscore, or have two consecutive underscores.

3. For License, select the license to associate with this permission set. In this case, select the Tableau CRM Growth permission set license.

When you select a specific permission set license, any user assigned to the permission set is *auto-assigned* the permission set license. If you select --None--, you must *manually* assign the permission set license to users before you can add them to the new permission set.

4. Click **Save**. The Permission Set Overview page shows the new permission set. Now add user permissions to the set.

5. Click **System Permissions** to open the list of user permissions enabled by the Tableau CRM Growth license. Then click **Edit**.

6. Do one of the following.

- If you're creating the Manage Tableau CRM permission set, select "Manage Analytics" user permissions and click **Save**. You've successfully created a permission set that enables access to all Tableau CRM features. You don't have to select any of the other individual permissions. Assign this permission set sparingly because it provides access to all Tableau CRM features, many of which are inappropriate for most analytics users.
- If you're creating the View Tableau CRM permission set, select "Use Analytics" and "Upload External Data to Analytics" user permissions and click **Save**. You've successfully created a permission set that enables a user to view the Tableau CRM datasets, lenses, and dashboards that they have permission to view, and to upload data files from outside Salesforce.

7. You can now assign your new permission set to users, which is covered in Assign Permission Sets to Users.

Repeat these steps for every user in your organization who requires access to Tableau CRM.

Assign Analytics Cloud Permission Sets to Users

Assign custom permission sets to one or more users—either one at a time, or in bulk—to give them access to Tableau CRM functionality.

**1.** In the Setup menu, under administer, click **Manage Users | Permission Sets**.

2. Click the permission set you want to assign to users. If you've followed the instructions in Enable Analytics Cloud and Create Service Analytics Permission Sets on page 555, select either the Manage Analytics Cloud or View Analytics Cloud permission sets.

3. Click **Manage Assignments** and click **Add Assignments**.

4. Select the users who need access to Tableau CRM, and click **Done**. When you assign the permission set to users, Salesforce *auto-assigns* the Tableau CRM Cloud - Analytics Cloud Platform *permission set license* to those users.

Assign the Manage Analytics Cloud permission set sparingly, since it provides access to all Tableau CRM features. Assign it only to users who administer or manage the Tableau CRM platform.

USER PERMISSIONS
To assign a permission set to users:
• Assign Permission Sets

Complete Setting up the Tableau CRM Platform

After assigning permission sets with either basic or advanced setup, follow these optional procedures to enhance the Tableau CRM experience and fine-tune Tableau CRM access to Salesforce data.

Enable Preview Thumbnails for All Lenses and Dashboards

Make it easier for users to find their lenses and dashboards on the home page and on app pages. Replace the generic icons with preview thumbnails of the contents of all lenses and dashboards, including those assets that have row-level security restrictions.

Enable Tableau CRM for Experience Cloud Sites

Securely share Tableau CRM apps with Experience Cloud site partner and customer users.

Enable Conversational Exploration

Quickly query and visualize data in guided, natural language conversations.

Enable Direct Data in Tableau CRM

Run queries directly against external data sources with Direct Data.

Enable the Tableau CRM REST API

You can access Tableau CRM datasets and lenses programmatically using the Tableau CRM REST API. Selecting this option overrides the "APIEnabled" permission for individual users and gives all Tableau CRM users (including Experience Cloud sites users) access to the API.

Enable Dashboard Views

Enable this feature to allow each Tableau CRM user to create their own unique views of a dashboard. By default, this feature is enabled for your org. If the feature is disabled and you re-enable it, all previously created views are available.

Enable Annotations on Dashboard Widgets

Annotate dashboard widgets with comments posted in the dashboard and in Chatter.

Enable Tableau CRM Watchlist

Enable this feature to capture and track up to 20 KPIs across different dashboards, plus historical trending. No more switching between dashboards to see what has changed, and track your metrics all in one place.

Enable Downloading Data from Tableau CRM

Enable users to download the results from lens explorations and dashboard widgets as Microsoft® Excel® (`.xls`) or comma-separated values (`.csv`) files. Enable users to subscribe to tables and receive email with the table data in attached `.csv` files.

Secure Image Sharing and Downloading

Control who shares dashboard and lens images to prevent users or profiles from sharing sensitive images. By default, all users can share images of lenses, dashboard widgets, and whole dashboards that contain your org's data.

Enable Null Measure Handling

Null measure handling lets you specify null as the default value for numeric columns in your recipes, dataflows, and CSV uploads. When no default value is specified and null measure handling is enabled, Tableau CRM replaces blanks with nulls in numeric columns in your dataset.

Open Salesforce Records in New Browser Tabs

By default, selecting a Salesforce record action in a Tableau CRM dashboard opens the record in a new Lightning Experience tab. To change the default behavior, so that selecting an action opens the record in a new browser tab, enable this setting. The setting applies to dashboards in the Tableau CRM tab and dashboards embedded in Salesforce pages.

Enable Custom Fiscal Year Support

To inherit custom fiscal years in Tableau CRM, first enable the setup option for your org.

Enable Single Custom Time Zone Support (Beta)

Time zone support lets you view time-specific data on dashboards in a time zone that you specify for your org. After you enable time zone support, you can't disable it, wo we recommend that you first enable the feature in a sandbox org containing a recent copy of your production data. You can then perform end-to-end testing before you enable the feature in your production org. Existing dashboards continue to work.

Control Which Analytics Tab Users Access the Analytics Studio

Specify who can create apps and edit and delete datasets and dashboards by making Analytics Studio invisible to Analytics Tab users on a per profile basis. When you remove Analytics Studio visibility from a profile, Analytics tab users with the profile don't see Analytics Studio in the App Launcher. They also don't have the Open in Analytics Studio action in asset menus and dashboard headers.

Connected App for Tableau CRM on Mobile

Install the Tableau CRM App Package to allow your mobile clients to easily connect to your org. This package contains the Connected App component that gives you control over who's logging in and how your mobile clients share images and links.

Tableau CRM Security Implementation Guide

Analytics Cloud has different levels of security that your organization can implement to ensure that the right user has access to the right data.

Salesforce Data Access in Tableau CRM

Tableau CRM requires access to Salesforce data when extracting the data and also when the data is used as part of row-level security. Tableau CRM gains access to Salesforce data based on permissions of two internal Tableau CRM users: Integration User and Security User.

Set Up an Allowlist of Trusted Sites for Embedded Tableau CRM Dashboards

Specify your trusted sites in the Tableau CRM allowlist, you can include embedded Tableau CRM dashboards in websites and apps outside of Salesforce servers.

Tableau CRM Encryption

The Tableau CRM Encryption solution enhances security for Salesforce customers by extending encryption capabilities to the data at rest that is stored on the Salesforce file system.

Enable Preview Thumbnails for All Lenses and Dashboards

Make it easier for users to find their lenses and dashboards on the home page and on app pages. Replace the generic icons with preview thumbnails of the contents of all lenses and dashboards, including those assets that have row-level security restrictions.

If no security predicate is defined for a dataset, then preview thumbnails always appear. If a security predicate is defined, then preview thumbnails don't show unless this setting is enabled. With the setting enabled, preview thumbnails appear for all assets.

🚫 **Important:** Even if row-level security applies to the dataset used by a lens or dashboard, the preview thumbnail could expose data from restricted rows. Row-level security restrictions take effect only after you've clicked through to the lens or dashboard.

1. From Setup, enter `Analytics` in the `Quick Find` box, then select **Settings**.

2. Select `Show preview thumbnails for lenses and dashboards with row level-security enabled`.

3. Click **Save**.

Enable Tableau CRM for Experience Cloud Sites

Securely share Tableau CRM apps with Experience Cloud site partner and customer users.

📝 **Note:** Only users with a Customer Community Plus, Partner Community, or Lightning External Apps Plus license can use this feature. This feature is supported in sites but not in portals.

To enable Tableau CRM for sharing in Experience Cloud sites, complete the following steps.

1. From Setup, enter `Analytics` in the `Quick Find` box, then select **Settings**. Select `Share Analytics with Communities`.

2. In Setup, assign your community members the Tableau CRM for Communities permission set license.

3. In Setup, create a permission set that includes the View Analytics on Communities pages permission and assign it to your community members.

4. Continue setting up Tableau CRM for Communities with Step 2 in Share Tableau CRM in Experience Cloud Sites.

Enable Conversational Exploration

Quickly query and visualize data in guided, natural language conversations.

To set up Tableau CRM to explore data using conversations, complete the following steps.

1. From Setup, enter *Analytics* in the `Quick Find` box, then select **Settings**.

2. Select `Enable Conversational Exploration`.

3. Click **Save**.

SEE ALSO:

Converse with Your Data

Enable Direct Data in Tableau CRM

Run queries directly against external data sources with Direct Data.

To set up running queries against external data sources with Direct Data, first enable the Explore External Data Directly org preference for your Salesforce org. Then complete these steps.

1. From Setup, enter *Analytics* in the `Quick Find` box, then select **Settings**.

2. Select `Enable Einstein Analytics Direct Data for external sources`.

3. Click **Save**.

SEE ALSO:

Gain Insights on Data Directly in Snowflake

Enable the Tableau CRM REST API

You can access Tableau CRM datasets and lenses programmatically using the Tableau CRM REST API. Selecting this option overrides the "APIEnabled" permission for individual users and gives all Tableau CRM users (including Experience Cloud sites users) access to the API.

1. From Setup, enter `Analytics` in the `Quick Find` box, then select **Settings**.

2. Select `Grant all users access to Wave API`.

3. Click **Save**.

For complete information about the API, see the *Tableau CRM REST API Developer Guide*.

Enable Dashboard Views

Enable this feature to allow each Tableau CRM user to create their own unique views of a dashboard. By default, this feature is enabled for your org. If the feature is disabled and you re-enable it, all previously created views are available.

1. From Setup, enter `Analytics` in the `Quick Find` box, then select **Settings**.

2. Select **Enable Wave dashboard saved views**, and then click **Save**.

Enable Annotations on Dashboard Widgets

Annotate dashboard widgets with comments posted in the dashboard and in Chatter.

To enable annotations, complete the following steps.

1. From Setup, enter `Feed Tracking` in the `Quick Find` box, and then select **Feed Tracking** under Chatter.

2. Scroll down and select **Analytics Asset**.

3. Select `Enable Feed Tracking`.

4. Click **Save**.

SEE ALSO:

[Collaborate with Dashboard Annotations](#)

Enable Tableau CRM Watchlist

Enable this feature to capture and track up to 20 KPIs across different dashboards, plus historical trending. No more switching between dashboards to see what has changed, and track your metrics all in one place.

1. From Setup, enter `Analytics` in the `Quick Find` box, then select **Settings**.

2. Select **Enable Watchlist**, and then click **Save**.

Enable Downloading Data from Tableau CRM

Enable users to download the results from lens explorations and dashboard widgets as Microsoft®
Excel® (`.xls`) or comma-separated values (`.csv`) files. Enable users to subscribe to tables and
receive email with the table data in attached `.csv` files.

To set up users to download data from Tableau CRM, complete the following steps.

1. From Setup, enter *Users* in the `Quick Find` box, and then select **Permission Sets**.

2. Select the Tableau CRM permission set where you want to add the permission, and click **Edit**.

3. Select **System Permissions**.

4. Select `Download Analytics Data`.

5. Click **Save**.

> 📝 **Note:** To subscribe to tables, users also need the Use Subscription Emails user permission.

> 📝 **Note:** Complete dashboards can be downloaded only as images.

SEE ALSO:
> Download Tableau CRM Images and Export Filtered Data
> Get Scheduled Updates with Email Subscriptions
> Enable Tableau CRM Watchlist

Secure Image Sharing and Downloading

Control who shares dashboard and lens images to prevent users or profiles from sharing sensitive
images. By default, all users can share images of lenses, dashboard widgets, and whole dashboards
that contain your org's data.

To block image downloads and shares for everyone in your org, complete the following steps.

1. From Setup, enter *Analytics* in the `Quick Find` box, then select **Settings**.

2. Select `Secure Image Sharing and Downloading`.

3. Click **Save**.

    To grant image sharing to select users or profiles, complete the following steps.

    a. From Setup, enter *Users* in the `Quick Find` box, and then select **Permission Sets**.

    b. Select the Tableau CRM permission set where you want to add the permission, and click **Edit**.

    c. Select **System Permissions**.

    d. Select `Share Analytics Images`.

    e. Click **Save**.

With the Share Analytics Images user permission enabled, users can do the following actions with
images of lenses, dashboard widgets, and whole dashboards.

- Download .png images
- Export images to Quip
- Post images to Chatter

- Include images in annotations
- Share images from conversational explorations and subscription previews

Enable Null Measure Handling

Null measure handling lets you specify null as the default value for numeric columns in your recipes, dataflows, and CSV uploads. When no default value is specified and null measure handling is enabled, Tableau CRM replaces blanks with nulls in numeric columns in your dataset.

Null measure handling is enabled in orgs created after the Spring '17 release and can't be disabled. To enable this feature in other orgs, first contact Salesforce Customer Support to evaluate your org and determine what extra steps to take. Other steps may be necessary because null measure handling can't retroactively change zeros to nulls, if those zeros have previously replaced blank values. This behavior can result in datasets containing a mixture of zeros coming explicitly from the source data, zeros replacing blank values, and nulls. Aggregate calculations in such datasets are incorrect.

> **USER PERMISSIONS**
>
> To enable null measure handling:
> - Customize Application

When Salesforce makes null measure handling available in your org, you can enable it from Setup. Enter `Analytics` in the Quick Find box, then click **Settings**. Select **Enable null measure handling in Analytics**, and then click **Save**.

Customer Support may recommend that you take one or more of these steps after enabling null measure handling.

1. Update your existing dataflow definition files to use null instead of 0 in `defaultValue` attributes for measure fields in these transformations.

computeExpression	defaultValue: "null"
computeRelative	defaultValue: "null"
dim2mea	measureDefault: "null"
sfdcDigest	defaultValue: "null"

> 📝 Note: When defining the `defaultValue` as null, use the string "`null`" (with quotes) for measures and `null` (without quotes) for dimensions.

2. Create new instances of long-lived datasets used as a source for dataflows, using a `defaultValue` of null for measure fields. Long-lived datasets are typically reference datasets that are not updated through a dataflow and used to augment data in other datasets. For example, you might use a geolocation dataset to augment lead information based on ZIP codes.

3. If you have incrementally built datasets in which the source data is no longer available, use a custom dataflow to manually convert 0 values to null.
   Incrementally built datasets are datasets that are created over time by appending rows, such as logs.

4. Reimplement `delta` transformations in your dataflows, using `computeRelative` and `computeExpression` transformations. The `delta` transformation is not supported when null measure handling is enabled and dataflows containing delta transformations fail.

   In this example, the `delta` transformation on the left calculates the difference between an opportunity amount and its previous amount. The `computeRelative` and `computeExpression` transformations on the right respectively calculate the previous amount and its difference to the current amount.

delta **Transformation**	computeRelative **and** computeExpression **Transformations**

<pre>"Calculate_Delta": {
      "action": "delta",
      "parameters": {
         "dimension": "OpportunityId",
        "epoch": "CreatedDate_day_epoch",
 	
         "inputMeasure": "Amount",
         "outputMeasure": "DeltaAmount",
         "source":
"Extract_Opportunity_History"
         }
    },</pre>
	<pre>"Compute_Previous": {
	    "action": "computeRelative",
	    "parameters": {
	      "partitionBy": [
	        "OpportunityId"
	      ],
	      "orderBy": [
	        {
	          "name": "CreatedDate",
	          "direction": "asc"
	        }
	      ],
	      "computedFields": [
	        {
	          "name": "PrevAmount",
	          "expression": {
	            "sourceField": "Amount",
	            "offset": "previous()",
	            "default": "current()"
	          }
	        }
	      ],
	      "source":
	"Extract_Opportunity_History"
	    }
	  },
	"Compute_Delta": {
	    "action": "computeExpression",
	    "parameters": {
	      "computedFields": [
	        {
	          "name": "DeltaAmount",
	          "type": "Numeric",
	          "precision": 18,
	          "defaultValue": 0,
	          "scale": 2,
	          "saqlExpression": "Amount -
	PrevAmount"
	        }
	      ],
	      "source": "Compute_Previous"</pre>

delta **Transformation**	computeRelative **and** computeExpression **Transformations**

	``
	    }
	  },
	``

SEE ALSO:

computeRelative Transformation

computeExpression Transformation

Open Salesforce Records in New Browser Tabs

By default, selecting a Salesforce record action in a Tableau CRM dashboard opens the record in a new Lightning Experience tab. To change the default behavior, so that selecting an action opens the record in a new browser tab, enable this setting. The setting applies to dashboards in the Tableau CRM tab and dashboards embedded in Salesforce pages.

To set up Tableau CRM dashboards to open records in new browser tabs, complete the following steps.

1. From Setup, enter *Analytics* in the Quick Find box, then select **Settings**.

2. Select Open Salesforce records in new browser tabs.

3. Click **Save**.

Enable Custom Fiscal Year Support

To inherit custom fiscal years in Tableau CRM, first enable the setup option for your org.

1. From Setup, enter *Analytics* in the Quick Find box, then select **Settings**.

2. Select **Inherit the custom fiscal year defined in Salesforce Setup**, and then click **Save**. When selected, Tableau CRM displays fiscal date values and filters based on the custom fiscal year defined in Salesforce Setup. When unselected, Tableau CRM bases date values and filters on a standard fiscal year.

3. Choose which date to base fiscal year on.

    a. **End Date**—If your fiscal year starts in April 2020 and ends in March 2021, your fiscal year is 2021.

    b. **Start Date**—If your fiscal year starts in April 2020 and ends in March 2021, your fiscal year is 2020.

4. Click **Save**.

Tableau CRM inherits the custom fiscal years defined in Salesforce. Custom fiscal year data and functionality, including fields suffixed `_Fiscal`, becomes available on a per-dataset basis after each dataset's dataflow runs. After inheriting custom fiscal years, users can work with fiscal data in SAQL queries and new fiscal filter and group options appear in Dashboards, like "Fiscal Year" and "Fiscal Quarter".

Enable Single Custom Time Zone Support (Beta)

Time zone support lets you view time-specific data on dashboards in a time zone that you specify for your org. After you enable time zone support, you can't disable it, wo we recommend that you first enable the feature in a sandbox org containing a recent copy of your production data. You can then perform end-to-end testing before you enable the feature in your production org. Existing dashboards continue to work.

Enable Time Zone Support

1. From Setup, enter `Analytics` in the `Quick Find` box, then select **Settings**.

2. Select `Enable Einstein Analytics Time Zones`.

   If the Enable Einstein Analytics Time Zone option isn't available, contact your Customer Success Manager or your Salesforce Account Executive.

3. From the **Org Supported Timezone** picklist, select the time zone for Tableau CRM. Tableau CRM converts date-time values in your datasets from GMT to the time zone that you select.

   The Internal Time Zone, GMT, is the assumed time zone for date-time values before conversion.

4. From the **Default Dashboard Timezone** picklist, select the time zone to use for date-time values in time zone enabled dashboards. You can choose between the Analytics supported time zone or GMT.

5. Select `GMT` if you don't want to convert date-time values in time zone enabled dashboards.

6. Click **Save**.

Sync Connected Objects and Refresh Datasets

1. Run a full Data Sync of your connected objects after you enable time zone support. This ensures that datasets created from these objects in dataflows and recipes are time zone enabled. See Run Data Sync Manually.

2. Run your recipes to refresh the datasets created from them. See Run a Recipe Manually.

3. Run your dataflows to refresh the datasets created from them. See Run a Dataflow Manually.

4. Replace the data in datasets created from .csv files. See Edit a Dataset.

   Date values in .csv data is assumed to be GMT unless you select a different source time zone when you upload the file. See Set a Source Time Zone for .CSV Data.

What Happens When You Enable Time Zone Support

This table summarizes what happens when you enable time zone support, the actions you need to take to update assets or stop using the feature.

Feature	Time Zone Support Off	Time Zone Support Enabled	Actions to Upgrade Assets	Front-End Revert
Data Sync	Salesforce Date and Date Time fields generated as type **Date**	Existing connected objects continue to work. When data sync runs, Salesforce Date and Date Time fields are auto mapped to type **DateOnly** or **DateTime**. Input date values assumed to be GMT.	Run data sync to update connected Salesforce objects.	No changes. Time zone fields are still generated.
Remote Connections	All date fields generated as type **Date**.	When sync runs, **Date** fields auto mapped to type **DateOnly** or **DateTime**. Input date values assumed to be GMT.	Run data sync to update connected external objects.	No changes. Time zone fields are still generated.
Data Prep Recipes and User Dataflows	Salesforce Date and Date Time fields generated as type **Date**	Existing dataflows and Data Prep Recipes continue to work, and generate time zone enabled datasets. Data Sync on: Date fields inherit type **DateTime** or **DateOnly** from connected objects. Data Sync off: Salesforce Date and Date Time fields auto mapped to type **DateOnly** or **DateTime**. Registered datasets with legacy **Date** type fields fetched in "edgemart" nodes are converted to datasets with **DateTime** type at runtime.	Run dataflows and Data Prep Recipes to update datasets. Warnings prompt optional action. Some nodes have new parameter **dateSyntax**, which determines what syntax is used in their SAQL expressions.	No changes. Time zone fields are still generated.

Feature	Time Zone Support Off	Time Zone Support Enabled	Actions to Upgrade Assets	Front-End Revert
File Uploads	All date fields generated as type **Date**.	Automated uploads continue to work, and generate time zone enabled datasets. UI uploads generate time zone enabled datasets. **Date** field type auto detected and mapped to type **DateOnly** or **DateTime**. If auto detect not possible, **Date** field type mapped to **DateTime**. Input date values assumed to be GMT. In UI uploads, user can change auto-detected date field types and source timezone (if not GMT).	No required action. Warnings prompt optional action.	No changes. Time zone fields are still generated.
Dashboards	All date fields generated as type **Date**.	Existing dashboards continue to work and are backwards-compatible with time zone enabled datasets.	Users can choose to build new time zone enabled dashboards, which use the Default Dashboard Timezone set in Analytics settings. New SAQL syntax must be used with time zone enabled Dashboards. If a step references a non time zone enabled dataset, it fails at runtime, but the rest of the dashboard loads.	Users must revert dashboards to use previous syntax and delete time zone enabled dashboards. Time zone enabled dashboards continue to work. No option to create time zone enabled dashboard or lenses from the Spring '18 release.

Time Zone Support Limitations

Consider these limitations when working with time zone support.

- Once activated, this feature can't be deactivated.
- You can't view converted date-time values in a lens.
- Time zone enabled dashboards don't support these features:
  - Packaging
  - Clipping a lens from explorer to the dashboard\
  - Exploring a dashboard widget as a lens
  - Notifications
  - Natural language queries
  - Data source connections between a static step and a dataset date field
  - Text-based search on date fields in list widgets
  - Displaying month names, such as "Jan" and "Feb" (months display as numbers only, such as "01" and "02")

Control Which Analytics Tab Users Access the Analytics Studio

Specify who can create apps and edit and delete datasets and dashboards by making Analytics Studio invisible to Analytics Tab users on a per profile basis. When you remove Analytics Studio visibility from a profile, Analytics tab users with the profile don't see Analytics Studio in the App Launcher. They also don't have the Open in Analytics Studio action in asset menus and dashboard headers.

To restrict access to the Analytics Studio within the Analytics tab, complete the following steps.

1. From Setup, in the `Quick Find` box, enter `App Manager`.

2. In the App Manager, open the Analytics Studio menu and select **Edit**.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To modify settings:
- Manage Analytics

3. To restrict access to the Analytics Studio for a profile, clear its check from the **Visible** column in the Assign to Profiles section.

⊘ **Important:** Although disabling visibility removes UI links to Analytics Studio within the Analytics tab, any URL to the Studio version of an asset always opens it in Studio. The Studio version URL is available in the Share dialog and is provided in notification and subscription emails. However, for notifications created on embedded dashboards and in Analytics Tab, links in notification emails navigate the user to the Analytics Tab.

✎ **Note:** The Visible setting applies to the Analytics tab only. Access to Analytics Studio from embedded dashboards is controlled through the component's attributes.

Connected App for Tableau CRM on Mobile

Install the Tableau CRM App Package to allow your mobile clients to easily connect to your org. This package contains the Connected App component that gives you control over who's logging in and how your mobile clients share images and links.

1. While logged in as an admin, navigate to:

   Production

   https://login.salesforce.com/packaging/installPackage.apexp?p0=04tB0000000cHCH

   Sandbox

   https://test.salesforce.com/packaging/installPackage.apexp?p0=04tB0000000cHCH

2. Select **Install for All Users**.

   If you install the Connected App only for certain roles, then only those clients are subject to your policies.

3. Click **Done**, then click the Salesforce Analytics App Package and then **View Components**.

4. Select iOS or Android. To configure policies and other settings, click **Edit**. When you're finished, save your changes.

   For more information about Connected App settings, see Edit a Connected App.

5. If you want to disable sharing options for mobile users, click **New** in the Custom Attributes section.

   All sharing options are enabled by default. To disable an option, add its attribute key and enter "false" for the attribute value. See the table for the supported attribute keys.

Attribute Key	Mobile clients can	Mobile Device
CACHE_QUERY_RESULTS	Store query cache results to disk	iOS
OFFLINE_BETA	Store data for offline access	iOS
SHARE_IMAGE_AIRDROP	Share images via AirDrop	iOS
SHARE_IMAGE_CAMERAROLL	Share images to Camera Roll	iOS
SHARE_IMAGE_OPEN	Share images to other apps and activities	iOS
SHARE_IMAGE_S1	Share images to Salesforce	iOS
SHARE_IMAGE_EMAIL	Share images via email	iOS
SHARE_LINK_NFC	Share links via Near Field Communication (NFC)	Android
SHARE_LINK_BLUETOOTH	Share links via Bluetooth	Android
SHARE_LINK_AIRDROP	Share links via AirDrop	iOS
SHARE_LINK_CLIPBOARD	Share links to Clipboard	iOS and Android
SHARE_LINK_S1	Share links to Salesforce	iOS and Android
SHARE_LINK_EMAIL	Share links via email	iOS and Android

Attribute Key	Mobile clients can	Mobile Device
SPOTLIGHT_SEARCH	See Tableau CRM assets in Spotlight search results	iOS

> **Note:** To apply custom attributes to connected users, you can revoke them and ask them to reconnect.

6. To view and control how mobile clients connect to your Tableau CRM-enabled org, from Setup, in the `Quick Find` box, enter `Connected Apps`, and then select **Connect Apps OAuth Usage**.

You can block user sessions, revoke individual users, and drill in to user details. For more information, see Monitoring Usage for a Connected App.

> **Note:** For certificate-based authentication, users must set their desktop browsers to allow cookies.

For extra security, use Mobile Device Management (MDM) with the option to show only authorized hosts. The "OnlyShowAuthorizedHosts" key allows you to hide the "Add New Account" plus sign in the mobile app. For more information, see Using MDM with Salesforce Mobile SDK Apps.

Tableau CRM Security Implementation Guide

Analytics Cloud has different levels of security that your organization can implement to ensure that the right user has access to the right data.
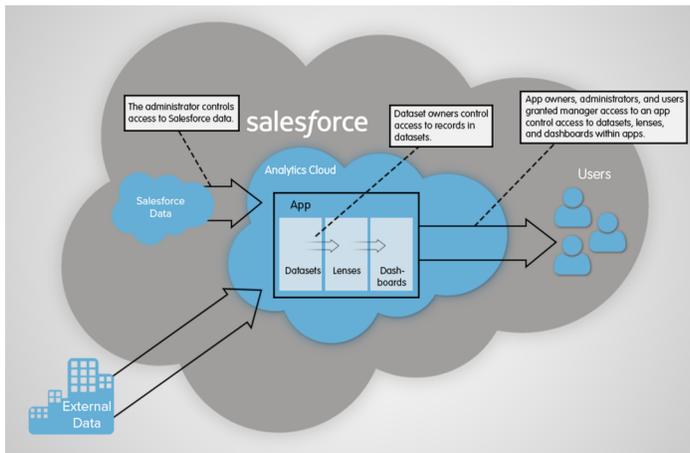
- The administrator can implement object-level and field-level security to control access to Salesforce data. For example, the administrator can restrict access to prevent the dataflow from loading sensitive Salesforce data into datasets. This document describes how Analytics Cloud uses object-level and field-level security on Salesforce data and how to configure permissions on Salesforce objects and fields.

- Dataset owners can implement row-level security on each dataset that they create to restrict access to it's records. If a dataset does not have row-level security, users who have access to the dataset can view all records. This document describes how to configure row-level security on datasets and provides some sample implementations based on datasets created from Salesforce data and external data.

- App owners, administrators, and users granted manager access to an app control access to datasets, lenses, and dashboards within apps. This document describes the different levels of access for apps and how to share datasets, lenses, dashboards in an app with other users.

> **Note:** Analytics Cloud supports security predicates, a robust row-level security feature that enables you to model many different types of access controls on datasets. Also, Analytics Cloud supports sharing inheritance, to synchronize with sharing that's configured in Salesforce, subject to certain limitations. If you use sharing inheritance, you must also set a security predicate to take over in situations when sharing settings can't be honored.

For complete information about implementing Analytics Cloud security, see Tableau CRM Security Implementation Guide.

Salesforce Data Access in Tableau CRM

Tableau CRM requires access to Salesforce data when extracting the data and also when the data is used as part of row-level security. Tableau CRM gains access to Salesforce data based on permissions of two internal Tableau CRM users: Integration User and Security User.

Tableau CRM uses the permissions of the Integration User to extract data from Salesforce objects and fields when a dataflow job runs. Because the Integration User has View All Data access, consider restricting access to particular objects and fields that contain sensitive data. If the dataflow is configured to extract data from an object or field on which the Integration User does not have permission, the dataflow job fails.

When you query a dataset that has row-level security based on the User object, Tableau CRM uses the permissions of the Security User to access the User object and its fields. The Security User must have at least read permission on each User object field included in a predicate. A predicate is a filter condition that defines row-level security for a dataset. By default, the Security User has read permission on all standard fields of the User object. If the predicate is based on a custom field, then grant the Security User read access on the field. If the Security User does not have read access on all User object fields included in a predicate expression, an error appears when you try to query the dataset using that predicate.

> **Important:** Because Tableau CRM requires the Integration User and Security User to access Salesforce data, do not delete either of these users.

Control Access to Salesforce Objects and Fields

Analytics Cloud requires access to Salesforce data when extracting the data and also when the data is used as part of row-level security. Configure the permissions of the Integration User on Salesforce objects and fields to control the dataflow's access to Salesforce data. Configure the permissions of the Security User to enable row-level security based on custom fields of the User object.

SEE ALSO:

Manage Datasets in Dashboard Components

Control Access to Salesforce Objects and Fields

Analytics Cloud requires access to Salesforce data when extracting the data and also when the data is used as part of row-level security. Configure the permissions of the Integration User on Salesforce objects and fields to control the dataflow's access to Salesforce data. Configure the permissions of the Security User to enable row-level security based on custom fields of the User object.

When configuring permissions for the Integration User or Security User, make changes to a cloned version of the user profile.

1. From Setup, enter `Profiles` in the `Quick Find` box, then select **Profiles**, and then select the user profile.

   For the Integration User, select the Analytics Cloud Integration User profile. For the Security User, select the Analytics Cloud Security User profile.

2. Click **Clone** to clone the user profile.

3. Name and save the cloned user profile.

4. Click **Object Settings**.

5. Click the name of the Salesforce object.

6. Click **Edit**.

   a. To enable permission on the object, select **Read** in the Object Permissions section.

   b. To enable permission on a field of the object, select **Read** for the field in the Field Permissions section.

   > 📝 Note: You can't change the permissions on standard fields of the User object.

7. Save the object settings.

8. Assign the cloned user profile to the Integration User or Security User.

   a. From Setup, enter `Users` in the `Quick Find` box, then select **Users**.

   b. Select the user to which you want to assign the user profile.

   c. Click **Edit**.

   d. In the Profile field, select the user profile.

   e. Click **Save**.

9. Verify that the Integration User or Security User has the right permissions on fields of the objects.

Set Up an Allowlist of Trusted Sites for Embedded Tableau CRM Dashboards

Specify your trusted sites in the Tableau CRM allowlist, you can include embedded Tableau CRM dashboards in websites and apps outside of Salesforce servers.

To set up an Analytics allowlist, complete the following steps.

1. From Setup, enter `Allowlist` in the `Quick Find` box, then select **Allowlist** under Analytics.

2. On the Allowlist for Embedding Dashboards page, click **New**.

3. On the Allowed Site Edit page, enter the information for the site you want to add to the Tableau CRM allowlist.

4. Click **Save**.

Tableau CRM Encryption

The Tableau CRM Encryption solution enhances security for Salesforce customers by extending encryption capabilities to the data at rest that is stored on the Salesforce file system.

If digital data is in transit, currently being processed, or stored in memory, that data is considered to be in use. By contrast, digital data is considered at rest if it is stored physically in persistent storage but is not currently in use. Tableau CRM Encryption is for encrypting registered datasets in Tableau CRM. To encrypt data at rest and preserve functionality, Tableau CRM Encryption services are built natively into the Tableau CRM platform. The solution applies strong, probabilistic encryption on data stored at rest. Platform encryption is based on the Advanced Encryption Standard (AES) with 256-bit keys using CTR mode for every write.

All operations, including sort and group-by, function the same as without encryption (except for key management functions, as noted in this topic).

How Tableau CRM Encryption Works

As a prerequisite to Tableau CRM Encryption, you must be approved by the Tableau CRM Encryption Product Manager. Your org must have a Shield Platform Encryption tenant secret. (Tableau CRM Encryption uses PE key management, so it's not necessary to encrypt objects and fields in core Salesforce.)

When using Tableau CRM with your encryption-enabled instance, data read from and written to disk is automatically encrypted with the unique key for your account. For more information about the encryption technology, refer to the Shield Platform Encryption Architecture white paper. To verify that encryption is enabled, go to the Analytics Settings page in the Setup UI.
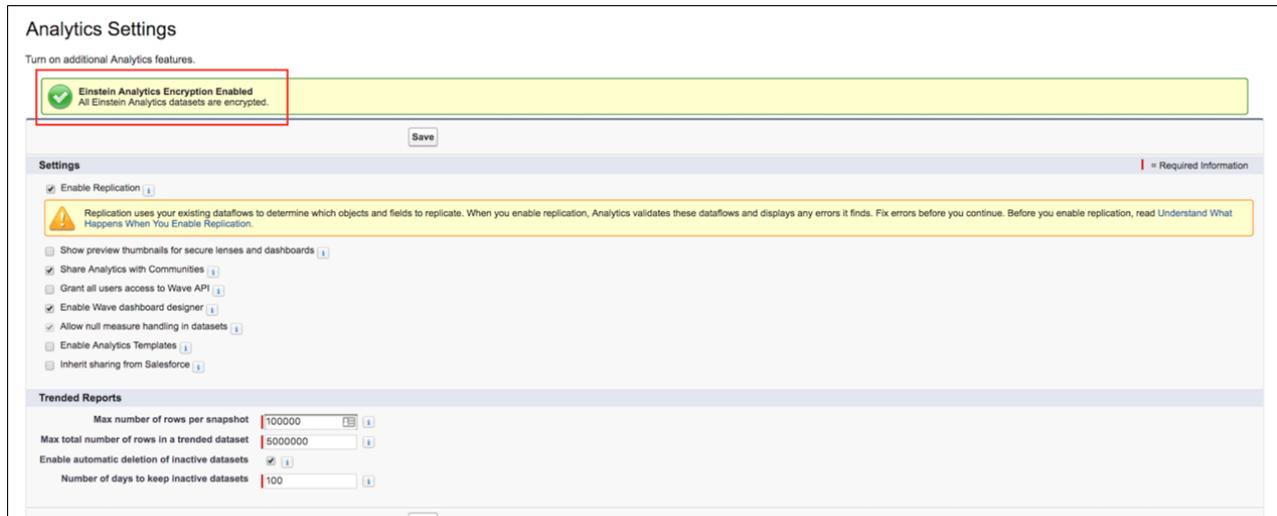
For Tableau CRM Encryption to function properly, you must define IP ranges for the Analytics Cloud Integration and Analytics Cloud Security user profiles as documented in, Best Practices: Manage Integration and Security Users in Tableau CRM (Einstein Analytics). Also, add designated Salesforce IPs to your allowlist, Salesforce IP Addresses and Domains to Allow.

Features

Key export, Bring Your Own Key, key rotation, key revocation, and key import are available.

All Tableau CRM features are supported, with the following exceptions:

- Pre-existing data is not encrypted.
- Data that was in Tableau CRM before encryption was enabled is not encrypted.
- If pre-existing data is imported from Salesforce objects through the dataflow, the data becomes encrypted on the next dataflow run.
- Other pre-existing data (such as CSV data) must be reimported to become encrypted.
- Although pre-existing data is not encrypted, it is still accessible and fully functional in its unencrypted state when encryption is enabled.

Frequently Asked Questions

Is data encrypted in transit? Yes. However, encryption in transit is different from encryption at rest. This feature covers encryption at rest.

Can Tableau CRM bring in data encrypted with Shield Platform Encryption? Yes. Tableau CRM reads platform encrypted data in the same way that any user reads platform encrypted data. The data is then written and encrypted in Tableau CRM. It is not required that data be encrypted with Shield Platform Encryption to be encrypted in Tableau CRM. It is only required that a Tenant Secret exist for the org.

Can I mask my data? No. Masking data is not currently available and it does not fall under the Tableau CRM scope.

Are the keys different for Shield Platform Encryption and Tableau CRM Encryption? Yes, although Shield and Tableau CRM use the same key management technology, they use different keys.

Do mobile dashboards still work with encryption? Yes. All Tableau CRM functionality, including support for mobile devices, works with encryption enabled. All data stored on mobile (mainly JSON code and thumbnails) is AES 256 encrypted.

Is there any impact on application performance? The Tableau CRM Encryption solution is designed to have at most a minimal performance impact on your Tableau CRM application experience.

Tableau CRM Requirements

This section provides requirements for using Tableau CRM.

Supported Browsers

Tableau CRM supports all browsers supported by Lightning Experience, with the following exceptions.

- Tableau CRM isn't supported on Apple® Safari®.
- Data Prep isn't supported on Internet Explorer 11.

Browser Zoom

Browser zoom settings other than 100% aren't supported for Tableau CRM.

Screen Resolution

The minimum screen resolution required to support all Salesforce features is 1024 x 768 pixels.

Supported Mobile Devices

Tableau CRM for iOS: Apple® devices running iOS 10 or later.

Tableau CRM for Android: Android™ devices running Android 5 or later.

Technical Requirements

To optimize your implementation, review the minimum and recommended technical requirements for Tableau CRM. These technical requirements help you predict whether your hardware and network can provide an acceptable and productive user experience. We strongly recommend testing the actual end-user experience with a configuration identical to what you expect to use in production. Test using the same geographic location, hardware, browser, network settings, and the expected concurrent users for shared hardware like virtual desktops.

Load times are measured in Experienced Page Time (EPT). Dashboard EPT measures how long it takes for a dashboard to load, where the page fully renders and is responsive to user interaction.

For the fastest and most stable experience, we recommend:

- An Octane score of 30,000 or greater
- Network latency of 150 ms or less
- Download speed of 3 Mbps or greater
- At least 8 GB of RAM, with 3 GB available for Salesforce browser tabs

Minimum requirements are:

- An Octane score of 20,000 or greater

- Network latency of 200 ms or less

- Download speed of 1 Mbps or greater

- At least 5 GB of RAM, with 2 GB available for Salesforce browser tabs

Based on our lab tests, the minimum requirements result in 50% slower page load times and login load times versus the recommended specifications.

You can find your Octane score, latency, and download speed by running the Salesforce Performance test. To run the test, append `speedtest.jsp` to your org's domain.

`https://yourDomain.lightning.force.com/speedtest.jsp`

We recommend running this test on the same hardware, network, physical location, and browser as your users. For virtual environments, such as VDI, run all tests from within that virtual environment.

Octane is a benchmark developed by Google that measures JavaScript performance. A higher Octane score correlates to faster page load times. Octane factors in your computer hardware and browser choice:

- Using newer-generation hardware with faster CPUs generates higher Octane scores.

- Using the latest version of Salesforce-supported browsers generates higher Octane scores.

- IE11 results in low Octane scores and much slower page load speeds.

For information, see Improve Speed and Performance in Lightning Experience.

Tableau CRM Limits

This section describes Tableau CRM limits.

API Call Limits

These limits apply to all supported editions.

Limit	Value
Maximum concurrent Analytics Cloud API calls per org	100
Maximum Analytics Cloud API calls per user per hour	10,000

> **Note:** Tableau CRM uses the bulk API, but does **not** count towards Salesforce bulk API limits. The dataflow submits a separate bulk API call to extract data from each Salesforce object. The dataflow uses a batch size of 100,000–250,000, depending on whether the dataflow or the bulk API chunks the data. As a result, to extract 1 million rows from an object, the dataflow creates 4–10 batches.

Dataset Row Storage Allocations

Your Salesforce org's total row storage limit for all registered datasets combined depends on your org's license combination. Each license allocates a set number of rows.

> **Note:** The data limits shown in the following table apply *only* to Analytics Cloud - Analytics Platform and Analytics Cloud - Sales Analytics and Service Analytics licenses purchased on or after October 20, 2015.

License	Allocated Rows
Tableau CRM Plus license baseline row allocation	10 billion
Tableau CRM Growth license baseline row allocation	100 million

License	Allocated Rows
Sales Analytics license baseline row allocation	25 million
Service Analytics license baseline row allocation	25 million
Event Monitoring Analytics license baseline row allocation	50 million
B2B Marketing Analytics license baseline row allocation	25 million
Tableau CRM for Financial Services Cloud license baseline row allocation	25 million
Tableau CRM for Health Cloud license baseline row allocation	25 million
Extra Data Rows license	100 million

Your org's total row storage limit is a combination of your org's active licenses. For example:

Tableau CRM Plus License

- If your org has a Tableau CRM Plus license and adds the Event Monitoring license, your total row limit becomes 10.05 billion. (10 billion *plus* 50 million)

- If your org has a Tableau CRM Plus license and adds the Additional Data Rows license, your total row limit becomes 10.1 billion. (10 billion *plus* 100 million)

- If your org has a Tableau CRM Plus license and adds two more Data Rows licenses, your total row limit becomes 10.2 billion. (10 billion *plus* (2 X 100 million)

Tableau CRM Growth License

- If your org has a Tableau CRM Growth license and then obtains an extra license for Services Analytics, your limit increases to 125 million. (100 million *plus* 25 million)

- If your org has a Tableau CRM Growth license and adds the Event Monitoring license, your total row limit becomes 150 million. (100 million *plus* 50 million)

> **Note:** TheTableau CRM Plus license already includes a Sales Analytics and Service Analytics license, so your total row allocation remains 10 billion. Similarly, the Tableau CRM Growth license already includes a Sales Analytics and Service Analytics license, so your total row allocation remains 100 million. In these situations, if your org obtains another Sales Analytics or Services Analytics license, your org row limit increases by 25 million for each app obtained.

Dataset Row Limits

Each dataset supports up to 2 billion rows. If your Salesforce org has less than 2 billion allocated rows, then each dataset supports up to your org's allocated rows.

Dataset Field Limits

Limit	Value
Maximum number of fields in a dataset	5,000 (including up to 1,000 date fields)
Maximum number of decimal places for each value in a numeric field in a dataset (overflow limit)	17 decimal places When a value exceeds the maximum number of decimal places, it *overflows*. Both 100,000,000,000,000,000 and 10,000,000,000,000,000.0 overflow because they use more than 17 decimal places. A number also overflows if it's greater (or less)

Limit	Value
	than the maximum (or minimum) supported value. 36,028,797,018,963,968 overflows because its value is greater than 36,028,797,018,963,967. -36,028,797,018,963,968 overflows because it is less than -36,028,797,018,963,967.
	When a number overflows, the resulting behavior in Tableau CRM is unpredictable. Sometimes Tableau CRM throws an error. Sometimes it replaces a numeric value with a null value. And sometimes mathematical calculations, such as sums or averages, return incorrect results. Occasionally, Tableau CRM handles numbers up to 19 digits without overflowing because they are within the maximum value for a 64-bit signed integer (2^{63} - 1). But numbers of these lengths are not guaranteed to process.
	As a best practice, stick with numbers that are 17 decimal places or fewer. If numbers that would overflow are necessary, setting lower precision and scale on the dataset containing the large numbers sometimes prevents overflow.
	✎ Note: If your org hasn't enabled Handle Null Numeric Values on page 604, then the maximum number of decimal places for each value in a numeric field in a dataset is 16. All orgs created after Spring '17 have Null Measure Handling enabled.
Maximum value for each numeric field in a dataset, including decimal places	36,028,797,018,963,967 For example, if three decimal places are used, the maximum value is 36,028,797,018,963.967
Minimum value for each numeric field in a dataset, including decimal places	-36,028,797,018,963,968 For example, if five decimal places are used, the minimum value is -36,028,797,018,9.63968
Maximum number of characters in a field	32,000

Data Sync Limits
If you currently extract more than 100 objects in your dataflows, contact Salesforce Customer Support before you enable data sync.

Limit	Value
Maximum number of concurrent data sync runs	3
Maximum number of objects that can be enabled for data sync	100
Maximum amount of time each data sync can run for local objects	48 hours
Maximum amount of time each data sync can run for remote objects	12 hours

> Note: Each connector's help describes its row count and volume data sync limit. Data sync can extract remote data up to the connector's maximum row count or volume limit, or up to 12 hours, whichever comes first.

Recipe and Dataflow Limits

Limit	Value
Maximum number of recipes	1,000
Maximum number of dataflows definitions (with data sync enabled)	100
Maximum number of dataflow and recipe runs in a rolling 24-hour period	60
	> Note: Dataflow and recipe runs that take less than 2 minutes (and data sync) don't count toward this 24-hour run limit. However, if you reach the 24-hour run limit, you can't run any dataflow, recipe, or data sync job, regardless of size. This maximum can be increased by contacting Salesforce Customer Support.
Maximum number of concurrent dataflow runs	2 for production orgs with the Tableau CRM Plus platform license
	1 for production orgs with the Tableau CRM Growth platform license or sandbox orgs
Maximum number of Data Prep previews per hour per user	4,000
Timeout for data load jobs (dataflows, CSV uploads, recipes, and data sync) that have been scheduled but not executed	5 minutes
Maximum length of a dataflow definition file	1,000,000 characters

Trending Data Limits

Trended datasets count toward the overall Tableau CRM platform limits, including total number of rows.

Limit	Value
Maximum number of trended datasets per user	5
Maximum number of rows per snapshot	100,000
Maximum number of rows in the report to be trended	500,000 for admins, 100,000 for non-admins
Maximum total number of rows in a trended dataset	5,000,000
Maximum monthly number of rows for all snapshots per org	40 million

External Data Limits

Limit	Value
Maximum file size per external data upload	40 GB

Limit	Value
Maximum file size for all external data uploads in a rolling 24-hour period	50 GB
Maximum number of external data jobs per dataset that can be run in a rolling 24-hour period	50
Maximum number of characters in a field	32,000
Maximum number of fields in a record	5,000 (including up to 1,000 date fields)
Maximum number of characters for all fields in a record	400,000

> 📝 **Note:** Each connector's help describes each its row count and volume data sync limit. These limits apply when using the external data API.

Story Creation and Prediction Limits

For limits on story creation and predictions, see Einstein Discovery Limits on page 1605.

Sales Analytics and Service Analytics App Limits

Sales Analytics and Service Analytics app limits custom object support to no more than 10 custom objects and one dataflow for each org that implements the app. These limits are contractual, not technical.

Security Predicate Limit

This limit applies to all supported editions.

Limit	Value
Maximum number of characters in a security predicate	5,000

Query Limits

Limit	Value
Maximum concurrent queries per organization	50 per platform
Maximum concurrent queries per user	10
Maximum number of rows returned per query 10,000 is the default value. To set a different value, use the SAQL `limit` statement	10,000*
Query timeout	2 minutes

*The maximum results returned by a query vary based on the query's type and whether it's executed from a desktop or the mobile app.

Query Type	Desktop Default Limit	Mobile Default Limit
aggregateflex	2,000	500

Query Type	Desktop Default Limit	Mobile Default Limit
`grain` (for values tables only)	100	100
`saql`	2,000	2,000
`soql`	See *SOQL and SOSL Reference*.	See *SOQL and SOSL Reference*
`staticflex`	n/a	n/a

> **Tip:** To see the `limit` for a given query, click ⟩_ in the explorer. If it's not set, Tableau CRM returns up to 10,000 results. To change the limit, edit the SAQL or SOQL query. The higher you increase the limit, the longer the query runs. For more information about SAQL queries, see the *Analytics Cloud SAQL Reference*. For information about SOQL queries, see *Salesforce Object Query Reference*.

The limit only affects the number of records returned for display; it doesn't affect calculations across all dataset records. For instance, say that a query groups by account name and the dataset contain 1 million account names. With a limit of 20, Tableau CRM only returns 20 results for display, but the summary row adds up all 1 million records.

Tableau CRM Direct Data for Snowflake Query Limits

Limit	Value
Maximum concurrent queries per organization	25
Maximum concurrent queries per user	5
Maximum number of rows returned per query	5,000
Query timeout	2 minutes

Tableau CRM Developer Edition Limits

Tableau CRM is built on the Lightning Platform and is subject to the Lightning Platform limits. For example, the SOQL query limit (100,000 characters) controls the number of fields from which the sfdcDigest transformation can extract data. If you exceed a Lightning Platform limit, an error occurs. See Lightning Platform limits.

Limit	Value
Maximum number of data rows	250,000
Maximum number of concurrent queries	5

Developer Edition Limits

Limit	Value
Maximum number of data rows	250,000
Maximum concurrent queries	5

> **Note:** These limits apply to the Wave-enabled Developer Edition.

Lens and Dashboard Limits

These limits apply to the creation and use of lenses and dashboards.

Limit	Value
Maximum JSON file size per dashboard	4 MB
Character limit for description fields	1000
Default number of rows in a compare table	2000 (To set a different value, use the SAQL `limit` statement)
Default number of rows in a values table	100 (To set a different value, use the SAQL `limit` statement)
Maximum file size for `geojson`	10 MB
Maximum number of notifications per user	10
Maximum number of recipients per notification email	500
Maximum number of data columns that can be downloaded in Excel format from a lens or widget	16383

Tableau CRM Limitations

Tableau CRM differs from other Salesforce features in some ways.

Update Schedule

All customers receive Tableau CRM updates on the same schedule, regardless of instance.

Localization and Internationalization

Tableau CRM has been localized with the following exceptions.

- We do not support right-to-left languages, such as Arabic and Hebrew.
- A subset of error messages is available only in English.
- The mobile iOS app includes only English versions of sample datasets and dashboards.

Tableau CRM provides internationalization support with the following limitations.

- You must set both locale and language to see translated labels.
- Data within datasets is not modified for localization or internationalization.
- Each dataset can have a single locale specified in the metadata. The metadata locale is not overridden by individual user locale settings. All users see the same date, time, and number formats, and dimension names, regardless of their own locale and language settings.
- Search in filters is not case-sensitive.
- Multicurrency is not supported. When Tableau CRM extracts your organization's default currency, it uses the currency for monetary values and doesn't convert to another currency.

Field-Level Security

Field-level security isn't available for external data that is uploaded via files. Field-level security that was implemented in the original database or Salesforce object isn't preserved when the data is loaded into a Tableau CRM dataset. For more information, see the Tableau CRM Security Implementation Guide.

Set Salesforce field-level security to enable the Tableau CRM Integration User to see all fields used in your app—default fields and those selected in the configuration wizard. Integration users run the dataflow, and if they don't have proper field-level security permissions, the dataflow fails.

SAQL Query Limitations

The timeseries function requires a Tableau CRM Platform license.

Explorer Limitations

You can't filter or group by the hour, minute, or second components of a date field.

You can't filter on values containing HTMLEncoded text.

Embedded Tableau CRM Dashboards

Limitations for embedded Tableau CRM dashboards are covered in Embed Dashboards Everywhere.

Tableau CRM in Experience Cloud Sites

Limitations for Tableau CRM within Experience Cloud are covered in Share Tableau CRM in Experience Cloud Sites.

Tableau CRM on Mobile Devices

To view Tableau CRM on a mobile device, connect to your org using the native Tableau CRM mobile app for Android or iOS. For limitations of the Tableau CRM mobile apps, refer to the See Also links at the end of this page.

Outside of the native Tableau CRM mobile app, for mobile devices the only supported access to Tableau CRM is via Lightning app pages viewed in the Salesforce mobile app. Embedded Tableau CRM dashboards accessed via mobile browsers aren't supported.

Limitations for the Wave Dashboard component in a Lightning app page in the Salesforce mobile app are covered in View Embedded Dashboards on Mobile Devices.

Tableau CRM Smart Dashboard and App Templates

Use of the SAQL timeseries feature in the Time Series smart dashboard template requires a Tableau CRM Platform license.

Limitations for most Tableau CRM app templates are covered in the Help topic for each template. See below for Adoption Analytics, Sales Analytics, and Service Analytics app limitations.

Sales Analytics Limitations

These limitations apply to Sales Analytics.

Sales Analytics Data Requirements

Sales Cloud data must meet the following requirements for Sales Analytics dashboards to function correctly:

- Use standard sales objects.
- Have at least one event and one task connected with an opportunity.

- Enable history tracking for Amount, Stage, and CloseDate fields on the Opportunities object.

- Cases must be connected to Accounts if you choose to import Cases data to Sales Analytics using the configuration wizard.

- There are two requirements if you choose to import Leads data to Sales Analytics using the configuration wizard.

  1. At least one lead must be converted to an opportunity.

  2. The lead must be connected to an account.

- There are two requirements if you choose to import Campaigns data to Sales Analytics using the configuration wizard.

  1. At least one opportunity must be connected to a campaign.

  2. At least one campaign member must be connected to a campaign.

- There are two requirements if you choose to import opportunity record types to Sales Analytics using the configuration wizard.

  1. At least one opportunity record type must be defined.

  2. The opportunity record type must be connected to at least one opportunity.

Sales Analytics Support for Salesforce Objects and Fields

Sales Analytics supports all Salesforce standard and custom objects and data. To add custom objects or additional fields not included when you first create the app, you need to update the Sales Analytics dataflow. For details see Design Datasets with Dataflows and the Dataset Builder on page 871.

When you first create the app it includes only a predefined set of objects and fields. Sales Analytics creates a dataflow that exposes selected fields from the following standard Salesforce objects:

- Accounts
- Users
- Roles
- Opportunities
- Products (Opportunities line item)
- Tasks
- Events

The fields provided are determined by the answers you provide when you create the app using the built-in configuration wizard.

Sales Analytics supports standard and custom fields on **standard Sales Cloud objects**. The fields provided are determined by the answers you provide when you create the app using the built-in configuration wizard.

Sales Analytics Support for Non-Salesforce Data

Sales Analytics does not support external data sources except a CSV file that contains quota data at the user level. Importing other external data requires an extra license. See your Salesforce representative for details.

To include data about quotas in Sales Analytics, you must upload a CSV file with the details. Sales Cloud Einstein customers who use Sales Analytics must use Collaborative Forecasts to see quota data. They can't edit the quota data set. For more information, see Collaborative Forecasting and Quotas Data in Sales Analytics.

Other Contractual Sales Analytics Limitations

The Sales Analytics Apps license does not support use of Tableau CRM platform bulk actions or Apex steps functionality. This limitation is contractual, not technical. Licensee agrees to strictly monitor its use of Tableau CRM platform features.

Sales Analytics Limitations for Sales Cloud Einstein and Tableau CRM Growth or Plus (Platform) Customers

Salesforce makes Sales Analytics available through three stock-keeping units (SKUs). Consult this chart to see limitations for each.

	Standalone	**Sales Cloud Einstein**	**Tableau CRM Growth or Plus**
Data sources	Salesforce data and CSV file for quotas data (see Sales Analytics Support for Non-Salesforce Data, above)	Salesforce data	Salesforce and external data
Object support	Standard and custom objects	Standard objects	Standard and custom objects
Data volume	25 million rows	25 million rows	• Tableau CRM Plus: 10 billion rows • Tableau CRM Growth: 100 million rows
Can customize existing dashboards?	Yes	Yes	Yes
Can create dashboards?	Yes	No	Yes
Can customize existing datasets?	Yes	No	Yes
Can create datasets?	Yes (using standard Salesforce objects and up to 10 custom objects)	No	Yes
Can create custom Tableau CRM apps?	No	No	Yes
Supports Einstein Discovery and Experience Cloud integration?	No	No	Yes
Supports bulk actions and APEX steps?	No	No	Yes
Supports Sales Cloud Einstein artificial intelligence?	No	Yes	No
Supports Salesforce Inbox?	No	Yes	No

Service Analytics Limitations

The following limitations apply to Service Analytics.

Service Analytics Data Requirements

Service Cloud data must include at least one each of the following for Service Analytics dashboards to function correctly: Events, tasks, closed cases, or published articles attached to a case or an opportunity. It also requires that your org have at least one contact ID associated with a case.

Custom formula fields on the Cases object must exist or be created for the SLA missed (text) and FCR (Boolean) fields.

Service Analytics Support for Salesforce Objects and Fields

Service Analytics supports all Salesforce standard and custom objects and data. When you first create the app by default it includes only a predefined set of objects and fields. The app creates a dataflow that exposes selected fields from the following Salesforce objects:

- Account
- Cases
- Contact
- User
- UserRole
- Task

Service Analytics also uses fields from the following options depending on the answers you select in the configuration wizard:

- Customer satisfaction (CSAT)
- Knowledge
- Business hours
- Case history
- Case record types
- Queues
- Opportunities
- Opportunity record types
- Events
- Telephony
- Chat
- Omni-Channel

Service Analytics supports standard and custom fields on standard Salesforce objects. The fields provided are determined by the answers you provide when you create the app using the built-in configuration wizard.

Service Analytics limits custom objects support to no more than 10 custom objects for each org that implements Service Analytics. This limitation is contractual, not technical.

To add custom objects or extra fields not included when you first create the app, you must update the Service Analytics dataflow. For details, see Design Datasets with Dataflows and the Dataset Builder on page 871.

More Service Analytics Limitations

Service Analytics does not support external data sources or registering new datasets from the app-generated dataflow.

Including CSAT score in dashboards/dataflow is optional. The app configuration wizard lets you associate CSAT from a custom case field or Custom Object associated to the Cases object.

You must manually add any security predicates or other dataset filters to the app dataflow.

The app does not support multi-hierarchy (team/manager) when you create it. You must manually add that functionality.

The Knowledge Usage dashboard currently reports on a single Article Type, which you select when you use the configuration wizard.

Adoption Analytics Limitations

These limitations apply to Adoption Analytics.

- Datasets created for the app count against row limits for your Tableau CRM license.

- Includes only Tableau CRM logs. For any Salesforce logs, use Event Monitoring.
- The first time you use Adoption Analytics, it may take 24 hours for data to collect in event logs before it can be registered in the app's datasets and dashboard.
- Salesforce stores event log data for 30 days.
- Tableau CRM stores event log data in a dataset for up to 365 days, which counts against your license's row limit.

Client Segmentation Analytics Limitations

Use of the Client Segmentation Analytics app is covered by a restricted use license created for Financial Services Cloud customers. Usage of Client Segmentation Analytics is limited to the following:

- Salesforce data only.
- Standard and custom objects.
- 10 million rows of data.

For access to additional Tableau CRM capabilities, purchase a Tableau CRM Growth, Tableau CRM Plus, or Einstein Predictions license.

SEE ALSO:

Tradeoffs and Limitations of Shield Platform Encryption

Limitations of Analytics Cloud on iPhone® and iPad®

Limitations of Analytics Cloud on Android Devices

Set Up the Tableau CRM Platform with Licenses Purchased Before October 20, 2015

If you purchased an Analytics Cloud Builder or Analytics Cloud Explorer license before October 20, 2015, you can continue to use your licenses to set up the Tableau CRM plaform for users in your organization or you can migrate to the new single-user Tableau CRM Growth license.

Note: This information applies only to customers who purchased Tableau CRM licenses before October 20, 2015, and continue to use the Analytics Cloud Builder or Analytics Cloud Explorer licenses after that without migrating to the new Tableau CRM Growth license. If you purchased Tableau CRM on or after October 20, 2015, go to Set Up the Tableau CRM Platform.

Important: If you're migrating from the previous Analytics Cloud Builder or Analytics Cloud Explorer platform licenses to the single-user Sales Analytics Apps license, read Migrating From Analytics Licenses Purchased Before 10/20/2015 to New Analytics Platform Licenses.

The Tableau CRM Growth License

A Tableau CRM license purchased before October 20, 2015, enables you to turn on Analytics Cloud for your organization. Each previous Tableau CRM Growth license is contractually limited to a maximum of 400 users. If you need more than 400 users, you can purchase additional Analytics Cloud licenses.

Each previous Analytics Cloud license enables you to:

- Store up to 250 million rows in your final registered datasets.
- Run up to 50 concurrent queries in Analytics Cloud.

The Analytics Cloud Permission Set License

Each user needs one of two Analytics permission set licenses to use the product. The permission set license defines what user permissions can be assigned to the user. The administrator assigns permission set licenses to users.

For Analytics Cloud licenses sold before October 20, 2015, Salesforce provided the following permission set licenses:

- Analytics Cloud Builder permission set license enables users to access all Analytics Cloud features.
- Analytics Cloud Explorer permission set license enables users to upload external data to Analytics Cloud and explore data.

Here are Analytics Cloud user permissions included with each permission set license.

User Permission	Included in Explorer or Builder Permission Set License?	What It Enables
"Create and Edit Analytics Dashboards"	Builder only	Creating and editing Tableau CRM dashboards.
"Create Analytics Apps"	Builder only	Creating and sharing Tableau CRM applications.
"Edit Analytics Dataflows"	Builder only	Downloading, uploading, starting, stopping, and rescheduling the dataflow. Viewing dataflow and system jobs in the data monitor.
"Manage Analytics"	Builder only	Accessing all Tableau CRM features. Provides Tableau CRM administrator-level capabilities.
"Upload External Data to Analytics"	Builder and Explorer	Uploading external data to Tableau CRM to create a dataset. Viewing dataflow and system jobs in the data monitor.
"Use Analytics"	Builder and Explorer	Using Tableau CRM, and viewing the datasets, lenses, and dashboards that the user has permission to view. Automatically enabled when you select any other Tableau CRM permission.

📝 **Note:** When you create permission sets for Tableau CRM users, selecting any of the other Analytics Cloud permissions automatically enables the "Use Analytics" permission.

You can assign previous Analytics Cloud permission set licenses along with any of the following Salesforce user licenses:

- Lightning Platform (app subscription)
- Lightning Platform (one app)
- Full CRM
- Salesforce Platform

- Salesforce Platform One

SEE ALSO:

Migrating From Tableau CRM Licenses Purchased Before 10/20/2015 to New Tableau CRM Platform Licenses

To complete migration from previous Analytics Cloud Builder and Analytics Cloud Explorer licenses purchased before October 20, 2015, to the new Tableau CRM Growth license, you may need to set up Tableau CRM users in your organization with new permission sets.

When Salesforce provisions your organization with the new Tableau CRM Growth license, you may have to set up some users with the new license.

- **Analytics Cloud Builder license users.** Users who access Tableau CRM with the Analytics Cloud Builder license should be able to continue to use the Tableau CRM platform without any additional setup. The Builder license is simply renamed *Tableau CRM Growth*, and you'll see that permission set license in Salesforce Setup.

- **Analytics Cloud Explorer license users.** During the new license provisioning process, Salesforce may remove the Analytics Cloud Explorer license from your org. If that's the case, you'll see the new Tableau CRM Growth license instead of the Explorer license and you have to set up users with that license. If the Explorer permission set license is still visible, you do not have to repeat the setup process for Explorer users.

If users in your organization require you to set them up with the new Tableau CRM Growth license—even if they used Tableau CRM with a previous license—follow the steps in Tableau CRM Platform Setup on page 549. You need to assign them the new permission set license, create one or more new permission sets, and assign the permission sets to users.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer Edition**.

Collaborative Forecasting and Quotas Data in Sales Analytics

Sales Analytics gives you a choice for how to include your team's quotas depending on whether you use the Sales Cloud Collaborative Forecasts feature to store quotas data.

> Note: Sales Cloud Einstein customers who use Sales Analytics must use Collaborative Forecasts to see quota data. They can't edit the quota data set.

You have the following options for including quotas data in Sales Analytics.

USER PERMISSIONS

To create and manage Tableau CRM apps:
- Manage Analytics Templated Apps

- Edit Analytics Dataflows

Table 4: Sales Analytics Quotas Data Options

Option	Sales Cloud Settings	Configuration Wizard Settings	Additional Setup Requirements
1	- Sales Cloud Collaborative Forecasts enabled - Collaborative Forecasts includes forecast quota data.	Select Collaborative Forecasting in first wizard screen that lets you add objects.	None. Sales Analytics automatically adds quota data from the Forecasts object.

Option	Sales Cloud Settings	Configuration Wizard Settings	Additional Setup Requirements
2	• Sales Cloud Collaborative Forecasts enabled • Collaborative Forecasts *does not* include forecast quota data.	Select Collaborative Forecasting in first wizard screen that lets you add objects.	Create .CSV file with quotas data, upload to Tableau CRM, rerun dataflow. See instructions below.
3	Sales Cloud Collaborative Forecasts *not* enabled	No setting available. Wizard does not give you the option of adding Collaborative Forecasts data.	Create .CSV file with quotas data, upload to Tableau CRM, rerun dataflow. See instructions below.

Note: Selecting both Opportunity Splits and Collaborative Forecasting pulls in quota data for the split type you've selected in your org.
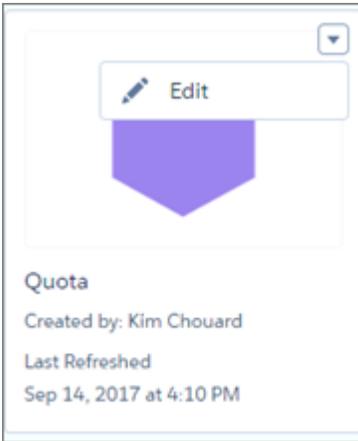
Wizard settings are optional, depending on whether you want to include Sales Cloud Collaborative Forecasts data in your app.

With Option 1, Sales Analytics automatically adds quotas data to your app and its dashboards. With Options 2 and 3, add quotas data to your app by following these instructions.

Note: When you work with .CSV files you want to import to Sales Analytics, create and open them using only a UTF-8-compliant text editor. Opening them in Microsoft Excel or other spreadsheet software reformats .CSV files and makes them unusable in Sales Analytics

Important: When you create the .CSV file, be sure it contains the following fields, in this order, with exactly these names. Field names are case-sensitive:

1. StartDate (in yyyy-mm-dd format)
2. QuotaAmount
3. OwnerName
4. Username

1. Create a .CSV file to include the fields just described, that is StartDate (in yyyy-mm-dd format), QuotaAmount, OwnerName, Username. For an example, see Sales Analytics Example .CSV File.
2. Save the file to a location that you can easily remember.
3. In Salesforce, go to the Analytics Cloud home page and find the Quota dataset.
4. Click the arrow at the upper right corner of the dataset panel and select **Edit**.

5. Salesforce displays the dataset editing screen for the Quota dataset. Look for **Replace Data** in the upper right corner and click it.

6. In the dialog box that opens, navigate to the .CSV file you created in Step 1, and double-click it.

7. Click **Next** to open the Replace Dataset Data page.

8. If your fiscal period is different than calendar period, that is if it starts on a date other than January 1, update the Quota Metadata file. If your fiscal period starts on January 1, skip to the next step.

   a. Copy the JSON from Sales Analytics Quota Dataset JSON File and paste it into a text editor of your choice.

   b. Change the value of `"fiscalMonthOffset"` from 4 to a number that represents the month your fiscal period starts. In Sales Analytics metadata, the numeral "0" stands for January, "1" stands for February, and so on up to "11," which stands for December. Save the file to your desktop.

   c. In Tableau CRM Studio, go to the Replace Dataset Data page and locate the Data Schema File area of the page. Click the arrow next to Quota .JSON file, select Replace File, find the file you saved and upload it to Tableau CRM.

9. On the Replace Dataset Data page, click **Next** to open the Edit Field Attributes page. The first column —QuotaAmount— should be selected. If not, select it. In the **Field Attributes** panel on the right, make sure **Field Type** is set to **Measure**.

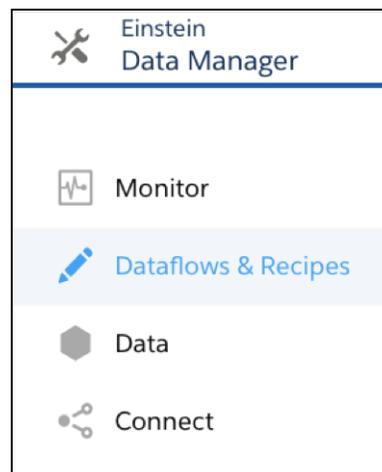**10.** Click **Upload**. You're asked to confirm that you want to replace the file. Click **Replace** to upload the file.

11. After uploading your quota data, rerun the dataflow to update the dashboards.

    a. Click the Gear menu at the upper right of the Sales Analytics screen and select Data Manager.

    b. Select Dataflow view from the menu at the top left of the Data Manager screen.

c. Find your app; you may have to scroll down the page. Open the menu on the far right of the screen next to the app icon and name, and click

it. The dataflow assures that Sales Analytics has your company's latest sales data. You can learn more about dataflows from Schedule the Sales App Daily Dataflow.

Edit the Forecast Dashboard Data Source Connection Make Sure Quotas Data Is Accurate

If Sales Analyticsquotas numbers look higher than expected in the Forecast dashboard, it may be because the app counts some quotas twice. To make sure that the dashboard accurately reflects your team's quotas, specify the dataset field the dashboard uses as its data source

Sales Analytics Example .CSV File

Here's an example of the .CSV file you create to update the Sales Analytics Quota (Target) dataset.

Sales Analytics Quota Dataset JSON File

Use this file to update the Sales Analytics Quota dataset with your fiscal year start date if it's *not* January 1.

Sales Analytics Example .CSV File

Here's an example of the .CSV file you create to update the Sales Analytics Quota (Target) dataset.

📝 **Note:** This file is for example purposes only. Create a unique .CSV file with quota data for members of your team including the following fields:

- QuotaAmount
- StartDate
- OwnerName
- Username

Save the .CSV file in UTF-8 format. Field names are case-sensitive and must appear in your file exactly as shown here.

🛑 **Important:** Do not open the .CSV file with Microsoft Excel or another spreadsheet application, which can corrupt the file format.

See Collaborative Forecasting and Quotas Data in Sales Analytics on page 1526.

👁 Example:

```
QuotaAmount,StartDate,OwnerName,Username
150000,2016-01-01,Chris Riley,trailhead9.ub20k5i9t8ou@example.com
150000,2016-02-01,Chris Riley,trailhead9.ub20k5i9t8ou@example.com
150000,2016-03-01,Chris Riley,trailhead9.ub20k5i9t8ou@example.com
150000,2016-01-01,Harold Campbell,trailhead14.jibpbwvuy67t@example.com
150000,2016-02-01,Harold Campbell,trailhead14.jibpbwvuy67t@example.com
150000,2016-03-01,Harold Campbell,trailhead14.jibpbwvuy67t@example.com
150000,2016-01-01,Jessica Nichols,trailhead19.d1fxj2goytkp@example.com
150000,2016-02-01,Jessica Nichols,trailhead19.d1fxj2goytkp@example.com
150000,2016-03-01,Jessica Nichols,trailhead19.d1fxj2goytkp@example.com
150000,2016-01-01,Catherine Brown,trailhead16.kojyepokybge@example.com
150000,2016-02-01,Catherine Brown,trailhead16.kojyepokybge@example.com
150000,2016-03-01,Catherine Brown,trailhead16.kojyepokybge@example.com
150000,2016-01-01,Kelly Frazier,trailhead7.zdcsy4ax10mr@example.com
150000,2016-02-01,Kelly Frazier,trailhead7.zdcsy4ax10mr@example.com
150000,2016-03-01,Kelly Frazier,trailhead7.zdcsy4ax10mr@example.com
150000,2016-01-01,Dennis Howard,trailhead4.wfokpckfroxp@example.com
150000,2016-02-01,Dennis Howard,trailhead4.wfokpckfroxp@example.com
150000,2016-03-01,Dennis Howard,trailhead4.wfokpckfroxp@example.com
```

Sales Analytics Quota Dataset JSON File

Use this file to update the Sales Analytics Quota dataset with your fiscal year start date if it's *not* January 1.

Copy the contents of this file into an editor of your choice and change the "fiscalMonthOffset" value (shown in **bold**) to the month your fiscal period begins. In metadata, the numeral "0" stands for January, "1" stands for February, and so on up to "11," which stands for December. In the code below, the number is set to "4", which stands for May. Use the number that represents the month your fiscal period begins. Then save the file and upload it to Sales Analytics following the instructions in Collaborative Forecasting and Quotas Data in Sales Analytics.

```
{
    "objects": [
        {
            "connector": "CSV",
            "fullyQualifiedName": "Quota_csv",
            "label": "Quota.csv",
            "name": "Quota_csv",
            "fields": [
                {
                    "fullyQualifiedName": "QuotaAmount",
                    "name": "QuotaAmount",
                    "type": "Numeric",
                    "label": "QuotaAmount",
                    "precision": 18,
                    "defaultValue": "0",
                    "scale": 0
                },
                {
                    "fullyQualifiedName": "StartDate",
                    "name": "StartDate",
                    "type": "Date",
```

```
                              "label": "StartDate",
                              "format": "yyyy-MM-dd",
            "fiscalMonthOffset": 4,
            "isYearEndFiscalYear": true

                      },
                      {
                              "fullyQualifiedName": "OwnerName",
                              "name": "OwnerName",
                              "type": "Text",
                              "label": "OwnerName"
                      },
                      {
                              "fullyQualifiedName": "Username",
                              "name": "Username",
                              "type": "Text",
                              "label": "Username"
                      }
                  ]
              }
          ]
}
```

Integrate Data into Tableau CRM Datasets

You can use Tableau CRM lenses and dashboards to analyze your data. To increase performance when processing large amounts of data, load the data into datasets first, and then build lenses and dashboards based on the datasets.

Get Started with Data Integration

In Tableau CRM, data integration involves gathering and preparing the Salesforce and external data you want to analyze. External data is data that resides outside of the Salesforce org that you use for Analytics, such as data from another Salesforce org, outside applications, spreadsheets, and databases. After integrating the data, users can explore and visualize it through Analytics lenses and dashboards.

Prepare and Load Data into Datasets with Recipes and Dataflows

Unlike other Tableau CRM tools that help you create datasets, recipes and dataflows allow you to prepare your data before loading it into datasets. Data preparation is the process of transforming your data into a form that's meaningful and valuable to the people consuming it. For example, you can define data preparation logic that combines data from two data sources and cleans up inconsistencies, such as differently formatted dates and codes.

Other Ways to Integrate Data

Dataflows and recipes can integrate most data from internal to external data sources. However, Tableau CRM provides additional ways to integrate data that are more effective for specific use cases. For example, if you don't have to transform data before loading it into a dataset, you can import data directly from CSV or Excel files. If you need to transform it, you can upload the file to create an intermediate dataset, and then use a recipe or dataflow to change that data before loading the results into another dataset. Or, to analyze Salesforce report data over time, you can create a dataset and dashboard that trends the report with just a few clicks. With trending, Tableau CRM creates a dataset based on snapshots of your Salesforce report data.

Manage Datasets

Edit a dataset to update its data, change its extended metadata, apply row-level security, or restore the dataset to a previous version.

Get Started with Data Integration

In Tableau CRM, data integration involves gathering and preparing the Salesforce and external data you want to analyze. External data is data that resides outside of the Salesforce org that you use for Analytics, such as data from another Salesforce org, outside applications, spreadsheets, and databases. After integrating the data, users can explore and visualize it through Analytics lenses and dashboards.

For faster queries when using large amounts of data involving millions or billions of rows, load the data into datasets. A dataset is a collection of related data that is stored in an indexed, denormalized, and highly compressed form. Tableau CRM offers multiple ways to load your data into datasets. Choose the approach that best meets your needs.

For example, with a recipe, you can perform data preparation tasks before loading it into a dataset. You can clean, aggregate, and transform your data. You can also create new columns based on calculations of existing data.

💡 **Tip:** With Salesforce Direct Data, you don't need to load Salesforce data into a dataset to analyze it in an Analytics dashboard. Instead, a dashboard widget can query a Salesforce object directly. Direct Data queries are especially useful when a dataset can't be refreshed fast enough, such as when you want to analyze data every 2 minutes, but the dataset refreshes every 15 minutes.

However, it's important to monitor query performance when using Direct Data on large objects. If the Salesforce object contains millions of records, sometimes queries can be faster on a dataset than direct queries on the Salesforce object.

About Datasets

A Tableau CRM *dataset* is a collection of related data that can be viewed in a tabular format. The data can come from many sources, including Salesforce objects, external data sources, and even other datasets.

Ways to Get Data from Data Sources Into Datasets

To populate datasets with data from data sources, you can import the data directly from the source. Or, you can extract the data from the source and prepare it before loading it into a dataset.

Plan for Your Data Integration Project

Before you build your data integration solution, think about the use case you want to analyze and the data you need to get you there.

Considerations Before Integrating Data into Datasets

This section covers expected behavior and limitations to consider before integrating data into datasets.

About Datasets

A Tableau CRM *dataset* is a collection of related data that can be viewed in a tabular format. The data can come from many sources, including Salesforce objects, external data sources, and even other datasets.

Columns and Rows in Datasets

A dataset is analogous to a table in a database. It organizes data by columns and rows.

Column Types in Datasets

Datasets support the following column types: dates, dimensions, and measures.

System Columns in Datasets

System columns are dataset columns that don't appear in the Tableau CRM user interface, such as Explorer or Dashboard Designer. For example, to enable dataset security based on the Salesforce role hierarchy, the Flatten transformation generates columns that store the role hierarchy details. To prevent a Tableau CRM user from seeing and querying these columns, flag them as system columns.

The data in Salesforce objects is normalized, meaning that it's stored efficiently in a collection of related tables to minimize redundancy, enhance consistency, and ensure data integrity. In contrast, data in datasets is denormalized, where it's stored in an optimal format for querying. All the work required to combine the data from multiple sources is done ahead of time. The data is centralized, compressed, and ready to be analyzed.

For each dataset that you create, you can apply security to restrict access to sensitive rows in the dataset.

Columns and Rows in Datasets

A dataset is analogous to a table in a database. It organizes data by columns and rows.

A dataset stores data in a file storage system, where data is organized by strings (dimensions) and numbers (measures). To make it easier to visualize, you can think of a dataset as a table, where the fields are columns and the values are rows.

Name	Amount	Forecast Category	Probability (%)	Closed	Close Date
salesforce.com - 5000 Widgets	500,000	Closed	100	true	2017-04-09
salesforce.com - 500 Widgets	50,000	Closed	100	true	2017-04-09
Global Media - 400 Widgets	40,000	Pipeline	60	false	2017-06-10
Acme - 1,200 Widgets	140,000	Pipeline	50	false	2017-05-12
Acme - 600 Widgets	70,000	Pipeline	20	false	2017-07-08
Acme - 200 Widgets	20,000	Pipeline	10	false	2017-09-10
salesforce.com - 1,000 Widgets	100,000	Pipeline	90	false	2017-05-12
salesforce.com - 2,000 Widgets	20,000	Pipeline	50	false	2017-07-10

- A *column* represents a category of information, such as an opportunity source or account name. Each column has a name, a data type, and other properties.
- A *row* represents an instance of data in the dataset. Rows can contain transactional data, such as individual invoices, or they can contain summary data, such as weekly invoice totals. What's important is that rows in a dataset should contain the same level of granularity, such as all invoice transactions or all weekly totals, rather than mixed levels.

Note: In a dataset, columns are analogous to fields in Salesforce objects and rows are analogous to records. However, when talking about datasets, we always use the terms column and row.

Column Types in Datasets

Datasets support the following column types: dates, dimensions, and measures.

Type	Description
Date	Can be represented as a day, month, year, and, optionally, time. You can group, filter, and perform math on dates.
Dimension	A qualitative value that usually contains categorical information (text), such as Product Category, Lead Status, and Case Subject. Dimensions are useful for grouping and filtering your data. Unlike measures, you can't perform math on dimensions. To increase query performance, Tableau CRM indexes all dimension columns in datasets.
Measure	A quantitative value that contains numerical data, such as revenue or exchange rate. You can do math on measures, such as calculating the total revenue or the minimum exchange rate.

System Columns in Datasets

System columns are dataset columns that don't appear in the Tableau CRM user interface, such as Explorer or Dashboard Designer. For example, to enable dataset security based on the Salesforce role hierarchy, the Flatten transformation generates columns that store the role hierarchy details. To prevent a Tableau CRM user from seeing and querying these columns, flag them as system columns.

Denormalized Data

The data in Salesforce objects is normalized, meaning that it's stored efficiently in a collection of related tables to minimize redundancy, enhance consistency, and ensure data integrity. In contrast, data in datasets is denormalized, where it's stored in an optimal format for querying. All the work required to combine the data from multiple sources is done ahead of time. The data is centralized, compressed, and ready to be analyzed.

Dataset Security

For each dataset that you create, you can apply security to restrict access to sensitive rows in the dataset.

SEE ALSO:

Set Up Dataset Security to Control Access to Rows

Ways to Get Data from Data Sources Into Datasets

To populate datasets with data from data sources, you can import the data directly from the source. Or, you can extract the data from the source and prepare it before loading it into a dataset.

Import Data

You can import data from Excel spreadsheets (via a connector) and CSV files directly into a dataset. This approach populates the dataset without modifying the data.
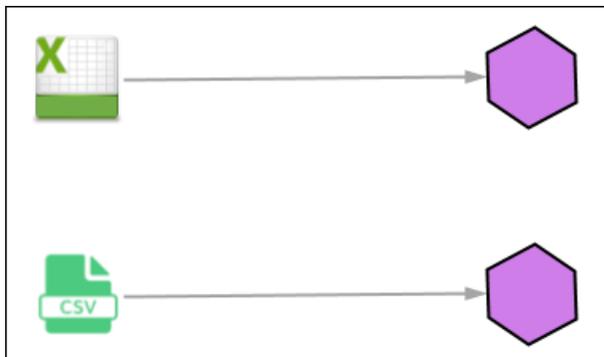
Prepare and Load Data

If you want to make any changes to the data, you can use recipes and dataflows. Recipes and dataflows are powerful features that you can use to cleanse, prepare, and optimize your data for analysis. You can use these tools to extract data from different data sources, both inside and outside of Salesforce.
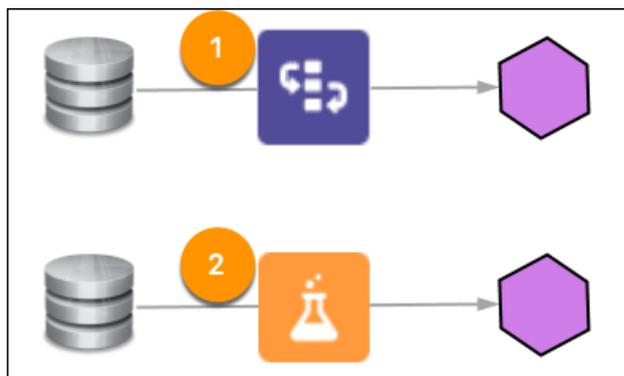
SEE ALSO:

Prepare and Load Data into Datasets with Recipes and Dataflows

Other Ways to Integrate Data

Plan for Your Data Integration Project

Before you build your data integration solution, think about the use case you want to analyze and the data you need to get you there.

Define What You Want To Analyze and Why

Begin by clarifying the purpose of your analysis. What area of your business do you want to target? What specific improvements do you want to make in this area? What KPIs can help you measure improvements? Understand what you want to learn from your data. That way, you can focus your efforts on what's most important.

Define the Data Elements Needed for Your Analysis

What objects and fields are the most relevant? Start with the obvious ones, and then consider some that aren't so obvious. Design the dataset that supports the analysis you want to conduct.

Identify the Best Data Source for Each Data Element

For each element, what is its data source? A Salesforce object? An external data source? Some data integration solutions merge data from multiple sources into a single dataset.

Investigate Details About Individual Data Elements

Drill down into these details so that you understand what data cleansing tasks, if any, are required to prepare your data for analysis. For each data element, determine how it is defined in the data source. What is its datatype? How is it formatted? How frequently do format inconsistencies occur? What is its default value, if any? Are null values allowed and, if so, how frequently do null values occur? How accurate are the values? How frequently do inaccuracies occur? Are values written once, or are they subsequently updated and, if so, how often?

Determine What Data You Need for Your Analysis

To yield credible results from your analysis, do you need all rows of data from the data source, or can you filter on a specific segment (subset)? For example, does your analysis target business activity in all countries or just certain ones? Are you focusing on a particular demographic or a range of values, such as customers with a total sales above 100,000?

Determine the Time Window for Your Data

How far back in time do you want to go for your analysis? Two years? Six months? Last week? Filter out any data that is outside your time range. And try to filter data as early as possible to minimize unnecessary data processing.

Commit to Continuous Improvement

We all learn as we go. As you test your data integration and review the results, you can discover ways to improve your outcomes. Perhaps you reconsider previous assumptions, change data sources, add or drop columns, or filter out extraneous rows to focus on just a particular segment. Tableau CRM tools can help you adapt to changes quickly so that you can make incremental improvements while keeping the solution in place.

Considerations Before Integrating Data into Datasets

This section covers expected behavior and limitations to consider before integrating data into datasets.

Handle Numeric Values
Tableau CRM internally stores numeric values in datasets as long values. For example, Tableau CRM stores the number 3,200.99 with a scale of 2 as 320099. The user interface converts the stored value back to decimal notation to display the number as 3200.99.

Handle Date Values
When Tableau CRM loads dates into a dataset, it breaks up each date into multiple columns, such as day, week, month, quarter, and year, based on the calendar year. For example, if you extract dates from a `CreateDate` column, Analytics Cloud generates columns such as `CreateDate_Day` and `CreateDate_Week`. If your fiscal year differs from the calendar year, you can enable Tableau CRM to generate fiscal date columns as well.

Handle Custom Time Zone Values (Beta)

Time zone support lets you view time-specific data on dashboards in a time zone that you specify for your org. By default, Tableau CRM datasets aren't time-zone aware, so Tableau CRM treats all date-time values as being in GMT. The data you see on your dashboards is in GMT, regardless of your local time zone. When you enable time zone support, Tableau CRM converts date-time values in your datasets to the time zone selected for Tableau CRM. You can then create time zone enabled dashboards to display these converted date-time values. Users see dashboard data in the single custom time zone you set, not their personal timezone specified in Salesforce.

Handle Text Values

Confirm that text values in a column are uniform in formatting, spelling, and language. Inconsistencies can occur within data sources and after merging data from multiple data sources.

Handle Missing Values

Gaps in your data can throw off analysis. Missing values are best fixed in the source application by, for example, making it a required field. However, if you can't do that, you can use Tableau CRM to fill in missing data.

Dataset Capacity and Limits

Before you create any datasets, review the limits. For example, each Salesforce org has a maximum number of rows for all datasets in the org. There are also limits on the number of columns in a dataset and characters in a column.

Handle Numeric Values

Tableau CRM internally stores numeric values in datasets as long values. For example, Tableau CRM stores the number 3,200.99 with a scale of 2 as 320099. The user interface converts the stored value back to decimal notation to display the number as 3200.99.

Range Limits for Numeric Values

The maximum numeric value that can be stored in a dataset is 36,028,797,018,963,967 and the minimum numeric value is -36,028,797,018,963,968.

Considerations for Numeric Values

Confirm that numeric values in a column are uniform in granularity and units. Inconsistencies can occur within a data source and after merging data from multiple data sources.

Handle Null Numeric Values

When you create or update a dataset through your dataflow, recipe, or a CSV upload, for example, Tableau CRM replaces blank numeric values with the specified default value. When no default value is specified, Tableau CRM replaces blanks in numeric columns with 0 or null based on whether you enable null measure handling.

Range Limits for Numeric Values

The maximum numeric value that can be stored in a dataset is 36,028,797,018,963,967 and the minimum numeric value is -36,028,797,018,963,968.

> ⚠️ **Warning:** If a numeric value is not within this range, you can receive unexpected results. For example, if you try to load the value 3.7E-16 with a scale of 16 into a dataset, Tableau CRM tries to store the value as 37000000000000000. However, because this value exceeds the maximum, Tableau CRM fails to load the entire row. In addition, if you perform a query that aggregates measures (such as a sum or group by) and the resulting value exceeds the maximum, the value overflows and Tableau CRM returns an incorrect result.

Considerations for Numeric Values

Confirm that numeric values in a column are uniform in granularity and units. Inconsistencies can occur within a data source and after merging data from multiple data sources.

Use the same granularity.

Don't mix aggregate totals with transaction values in the same column. For example, if you have a column that contains invoice charges, make sure that the values in that column are all at the same level of detail: either the total invoice charge or charges for individual line items, but not both. Similarly, don't mix values from different time intervals, such as weekly and monthly totals.

Standardize on a single unit of measure.

If a column contains numeric values represented by different units of measure, such as currencies, weights, or dimensions, consider creating a calculated column and converting those numbers into a common unit of measure. That way, you compare apples to apples, not apples to oranges to lemons.

Use a consistent numeric precision or scale.

If you have long numbers or numbers with many decimal places, consider creating a calculated column that simplifies the values by rounding or dividing by a multiple of 10.

Handle Null Numeric Values

When you create or update a dataset through your dataflow, recipe, or a CSV upload, for example, Tableau CRM replaces blank numeric values with the specified default value. When no default value is specified, Tableau CRM replaces blanks in numeric columns with 0 or null based on whether you enable null measure handling.

📝 **Note:** Null measure handling is enabled in orgs created after the Spring '17 release and can't be disabled. To enable this feature in other orgs, see Enable Null Measure Handling.

Tableau CRM replaces blanks with:

- 0 when null measure handling is disabled and no default value is specified
- null when null measure handling is enabled and no default value is specified

This behavior is important because Tableau CRM treats null and 0 differently. They aren't the same. Null means that no value exists. 0 represents a value. Math operations performed on null results in null. For example,

```
10 + null = null
```

💡 **Tip:** To return 0 instead of null in a math operation, you can use an expression similar to this one: `10 + coalesce(null,0)`. For example, `10 + coalesce('Amount', 0)`, where Amount is null.

Aggregate functions—such as sum, average, count, min, and max—exclude null values from calculations. To understand how 0 and null are treated differently with aggregate functions, consider the following customer satisfaction scores, where one score is missing for Customer2.

SOURCE DATA			DATASET WITH ZEROS FOR BLANKS			DATASET WITH NULLS FOR BLANKS	
Customer	**CSAT**		**Customer**	**CSAT**		**Customer**	**CSAT**
Customer1	80		Customer1	80		Customer1	80
Customer2			Customer2	0		Customer2	*null*
Customer3	90		Customer3	90		Customer3	90
Customer4	100		Customer4	100		Customer4	100
			Avg CSAT	67.5		**Avg CSAT**	90.00
			Min CSAT	0		**Min CSAT**	80

When Tableau CRM replaces the blank value with a zero, the average and minimum calculations are incorrect. When the replacement is null, the average and minimum calculations are correct.

Null measure handling allows the use of null values in these dataflow transformations.

append	Tableau CRM can append datasets with different measure columns. For example, one dataset has an Amount column that is not present in a second dataset. After the append, the new dataset has an Amount column containing nulls in rows from the second dataset. Without null measure handling, appending datasets is not possible when measure columns are different.
augment	When the left key is null or has no match on the right, Tableau CRM inserts nulls in the columns added from the right. Without null measure handling, Tableau CRM inserts zeros for measures.

computeExpression	The `defaultValue` attribute for a computed column accepts a value of null. For example:
	
	 ``json
	"SAQLNode": {
	        "action": "computeExpression",
	        "parameters": {
	          "source": "extractOpp",
	          "mergeWithSource": true,
	          "computedFields": [
	            {
	              "name": "Amount2",
	              "type": "Numeric",
	              "precision": 18,
	              "defaultValue": "null",
	              "scale": 5,
	              "saqlExpression": "Amount+Amount"
	            }
	``

computeRelative	The `defaultValue` attribute for a computed column accepts a value of null. When a value can't be calculated, such as when `previous()` or `next()` has no value, Tableau CRM uses this default. For example:
	
	 ``json
	"computeTrending": {
	        "action": "computeRelative",
	        "parameters": {
	            "partitionBy": ["OpportunityId"],
	            "orderBy": [
	                {
	                    "name":"CreatedDate",
	                    "direction":"asc"
	                }
	            ],
	            "computedFields": [
	                {
	                    "name": "AmountPrev",
	                    "defaultValue" : "null",
	                    "expression": {
	                        "sourceField": "Amount",
	                        "offset": "previous()",
	                        "default": "null"
	                    }
	                }
	``

dim2mea	The `measureDefault` parameter accepts a value of null. Tableau CRM uses this value when it can't convert a dimension to a valid numeric value. When no default is specified, Tableau CRM uses a value of null. For example:

	``
	"Create_Measure_From_Dimension": {
	        "action": "dim2mea",
	        "parameters": {
	            "dimension": "StageVal__c",
	            "measure": "StageValue",
	            "measureDefault": "null",
	            "source": "Extract_Opportunities"
	        }
	    }
	``

sfdcDigest	The `defaultValue` attribute for a column accepts a value of null. When no default is specified, Tableau CRM uses a value of null. For example:

	``
	Sample Sfdc-digest :{
	    "Extract_Opportunities": {
	        "action": "sfdcDigest",
	        "parameters": {
	            "object": "Opportunity",
	            "fields": [
	                {
	                    "name": "Amount",
	                    "defaultValue": "null",
	                    "precision":18
	                    "scale":2
	                }
	``

Update	Tableau CRM can update non-null values with nulls.

> **Note:** The `delta` transformation is not supported when null measure handling is enabled and dataflows containing delta transformations fail. To calculate changes in measure values over time, use `computeRelative` and `computeExpression` transformations instead. For an example, see Enable Null Measure Handling.

Handle Date Values

When Tableau CRM loads dates into a dataset, it breaks up each date into multiple columns, such as day, week, month, quarter, and year, based on the calendar year. For example, if you extract dates from a `CreateDate` column, Analytics Cloud generates columns such as `CreateDate_Day` and `CreateDate_Week`. If your fiscal year differs from the calendar year, you can enable Tableau CRM to generate fiscal date columns as well.

Standard Date Fields

Tableau CRM generates the following date columns.

Field Name	Field Type	Description
<date field name>_Second	Text	Number of seconds. If the date contains no seconds, value is 0.

Field Name	Field Type	Description
<date field name>_Minute	Text	Number of minutes. If the date contains no minutes, value is 0.
<date field name>_Hour	Text	Number of hours. If the date contains no hours, value is 0.
<date field name>_Day	Text	Day of the month.
<date field name>_Week	Text	Week number in calendar year.
<date field name>_Month	Text	Month number in calendar year.
<date field name>_Quarter	Text	Quarter number in calendar year.
<date field name>_Year	Text	Calendar year.
<date field name>_Week_Fiscal	Text	Week number in fiscal year.
<date field name>_Month_Fiscal	Text	Month number in fiscal year.
<date field name>_Quarter_Fiscal	Text	Quarter number in fiscal year.
<date field name>_Year_Fiscal	Text	Fiscal year.
<date field name>_sec_epoch	Numeric	Number of seconds that have elapsed since January 1, 1970 (midnight UTC).
<date field name>_day_epoch	Numeric	Number of days that have elapsed since January 1, 1970 (midnight UTC).

Inherited Custom Fiscal Years Fields

After you have defined a custom fiscal year in Salesforce and enabled the setting in Tableau CRM (see Custom Fiscal Year Support on page 610), have Tableau CRM inherit the custom fiscal year so your users can work with custom fiscal year data in SAQL queries and dashboards.

When Tableau CRM loads dates into a dataset as part of a dataflow, it generates multiple fields that describe periods of time, such as day, week, and month. For example, when Tableau CRM loads a date field called `CreatedDate`, it generates fields like `CreatedDate_Day`, `CreatedDate_Week`, and `CreatedDate_Month`.

After Tableau CRM inherits custom fiscal years from Salesforce, Tableau CRM generates more fields that describe the inherited custom fiscal years. The custom fiscal year fields are named with the suffix `_Fiscal`. Continuing with the `CreatedDate` example, Tableau CRM generates fields like `CreatedDate_Week_Fiscal`, `CreatedDate_Month_Fiscal`, and `CreatedDate_Quarter_Fiscal`.

📝 Note: If Tableau CRM custom fiscal years were previously defined using the standard fields (not inherited), then those fields are overwritten with data inherited from the custom fiscal years defined in Salesforce.

Here are the Fiscal fields Tableau CRM generates for each date field.

Field Name	Field Type	Description
<date field name>_Week_Fiscal	Text	Week number in fiscal year.
<date field name>_Month_Fiscal	Text	Month number in fiscal year.

Field Name	Field Type	Description
<date field name>_Quarter_Fiscal	Text	Quarter number in fiscal year.
<date field name>_Year_Fiscal	Numeric	Fiscal year.

Considerations for Date Columns

Confirm that date values in a column are uniform in format and time zone. Inconsistencies can occur within a data source and after merging data from multiple data sources.

Date Formats and Fiscal Dates for Source Data

You can set metadata attributes to control how dates are loaded into Tableau CRM and to enable Tableau CRM to generate fiscal date columns.

Considerations for Date Columns

Confirm that date values in a column are uniform in format and time zone. Inconsistencies can occur within a data source and after merging data from multiple data sources.

Perform the following tasks for date columns.

Ensure consistent date formats.

When combining data from different data sources, sometimes dates are stored in different formats, such as MM-dd-yyyy and yy-MM-dd'T'HH:mm:ss'Z'. Use the Format Date transformation to make the formats consistent.

Fix time zone differences in datetime columns.

Tableau CRM doesn't officially support multiple time zones. If a column contains a mix of time zones, consider creating a calculated column with the Formula transformation to add or subtract hours to datetime values to ensure a single time zone.

📝 **Note:** All date column labels in a dataset must be unique.

Date Formats and Fiscal Dates for Source Data

You can set metadata attributes to control how dates are loaded into Tableau CRM and to enable Tableau CRM to generate fiscal date columns.

You set the metadata attributes in the sfdcDigest transformation parameters for Salesforce data or in the metadata file for external CSV data.

Keep the following tips in mind when setting the metadata attributes for date columns:

- You can't configure dates that are obtained from external data via connectors.

- Before loading dates from an external data file, confirm that the date format meets the requirements specified here.

- Confirm that the column names in the external data file don't conflict with the generated date column names. For example, if you load a CSV with column `Create_Date`, Tableau CRM generates the `Create_Date_Year` column in the dataset. If the CSV also has a column named `Create_Date_Year`, Analytics Cloud throws an error.

Fiscal Periods in Analytics Cloud

If the calendar and fiscal year differ, you can enable Tableau CRM to generate the fiscal date columns in the dataset, in addition to calendar date columns. To enable Tableau CRM to generate fiscal date columns, set the `fiscalMonthOffset` attribute to a value other than 0. Set this attribute for each date column that you want to generate fiscal date columns for. If you set the offset to 0 or you don't specify a value, Tableau CRM doesn't generate any fiscal date columns.

> **Note:** Check out Custom Fiscal Year Support on page 610 to learn how Tableau CRM can apply your Salesforce org's custom fiscal calendar settings automatically.

Additionally, to configure the fiscal periods, set the following metadata attributes for each date column.

fiscalMonthOffset

In addition to enabling fiscal date columns, this attribute also determines the first month of the fiscal year. Specify the difference between the first month of the fiscal year and first month of the calendar year (January) in `fiscalMonthOffset`. For example, if your fiscal year begins in April, set `fiscalMonthOffset` to 3.

isYearEndFiscalYear

Because the fiscal year can start in one calendar year and end in another, you must specify which year to use for the fiscal year. The `isYearEndFiscalYear` attribute indicates whether the fiscal year is the year in which the fiscal year ends or begins.

For example, if `isYearEndFiscalYear` = true or you don't specify this attribute, then the fiscal year is the year in which the fiscal year ends. As shown in the following diagram, any dates between 4/1/2015 and 3/31/2016 are part of the fiscal year 2016 because the fiscal year ends in 2016.
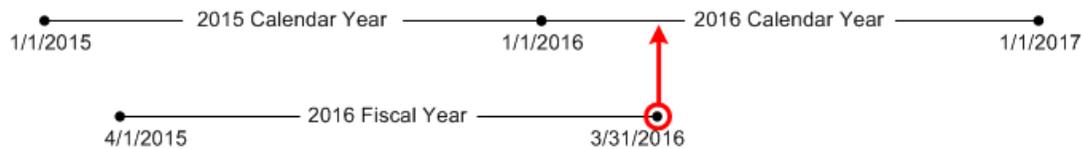
If `isYearEndFiscalYear` = false, then the fiscal year is the year in which the fiscal year begins. As shown in the following diagram, any dates between 4/1/2015 and 3/31/2016 are part of the fiscal year 2015 because the fiscal year begins in 2015.
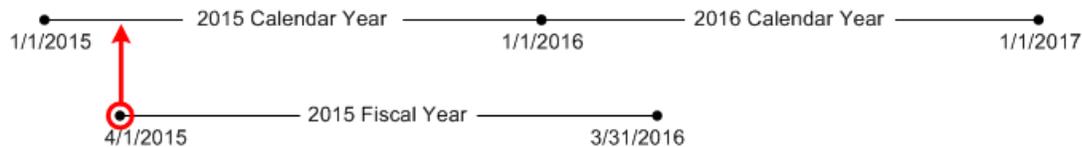
Week Numbering in Analytics Cloud

For each date loaded into a dataset, Tableau CRM generates the corresponding week number for the calendar year and, if applicable, fiscal year. Similar to the SOQL function `WEEK_IN_YEAR`, week 1 in Tableau CRM is January 1–7. (This is different from the UTC `week()` calculation.)

If needed, you can configure the week to start on a particular day of the week by setting the `firstDayOfWeek` attribute. For example, if January 1 is a Saturday and you configure the week to start on a Monday, then week 1 is January 1–2. Week 2 starts on Monday, January 3. Week 3 starts January 10, the following Monday. Notice that week 1 can be a short week to ensure that the subsequent weeks start on the specified day of the week.

Inherit Custom Fiscal Years from Salesforce

If your company uses a custom fiscal year, and has defined the custom fiscal year in Salesforce, then Tableau CRM can inherit the custom fiscal year definition from Salesforce. Once inherited, the custom fiscal year is available for use throughout Tableau CRM. For example, a dashboard filter can return opportunities from this fiscal quarter.

Inherit Custom Fiscal Years from Salesforce

If your company uses a custom fiscal year, and has defined the custom fiscal year in Salesforce, then Tableau CRM can inherit the custom fiscal year definition from Salesforce. Once inherited, the custom fiscal year is available for use throughout Tableau CRM. For example, a dashboard filter can return opportunities from this fiscal quarter.

Before inheriting a custom fiscal year from Salesforce, you must define one in Salesforce Setup. If you have already defined one or more custom fiscal years in Salesforce, then you are ready for Tableau CRM to inherit custom fiscal years.

To learn how to define a custom fiscal year in Salesforce, see Define your Fiscal Year.

To learn how to turn on custom fiscal year inheritance for Tableau CRM, see Enable Custom Fiscal Year Support on page 566.

To learn about the new fields Tableau CRM generates for each date field, see the Inherited Custom Fiscal Years Fields section of Handle Date Values on page 606.

> Custom Fiscal Years Inheritance Troubleshooting and Considerations
>
> When you inherit custom fiscal year settings from Salesforce to Tableau CRM, use these tips for troubleshooting and keep these considerations in mind.

Custom Fiscal Years Inheritance Troubleshooting and Considerations
When you inherit custom fiscal year settings from Salesforce to Tableau CRM, use these tips for troubleshooting and keep these considerations in mind.

Troubleshoot Custom Fiscal Years in Tableau CRM

Why don't my datasets have custom fiscal year support?

If your datasets don't support fiscal periods after you inherit custom fiscal years from Salesforce, run the dataset's dataflow or recipe. Support begins after the dataflow finishes running.

Why doesn't my dashboard filter have custom fiscal year options?

If your dashboard filter doesn't give you the option of filtering by fiscal periods, run the dataflow for each dataset in the dashboard. Support begins after each dataflow finishes running.

It's a new fiscal year, and suddenly Tableau CRM returns null for all dates?

If a date is out of range of a defined fiscal year inherited from Salesforce, Tableau CRM returns null. Define a custom fiscal year in Salesforce, then have Tableau CRM inherit it. Finally, run each dataset's dataflow. After the dataflows finish running, Tableau CRM once again returns data instead of null values.

Considerations and Limitations

- Custom Fiscal Years are not compatible with these SAQL Statements:
  - fill
  - timeseries

Handle Custom Time Zone Values (Beta)

Time zone support lets you view time-specific data on dashboards in a time zone that you specify for your org. By default, Tableau CRM datasets aren't time-zone aware, so Tableau CRM treats all date-time values as being in GMT. The data you see on your dashboards is in GMT, regardless of your local time zone. When you enable time zone support, Tableau CRM converts date-time values in your datasets to the time zone selected for Tableau CRM. You can then create time zone enabled dashboards to display these converted date-time values. Users see dashboard data in the single custom time zone you set, not their personal timezone specified in Salesforce.

> 📝 **Note:** As a beta feature, Time Zone Support is a preview and isn't part of the "Services" under your master subscription agreement with Salesforce. Use this feature at your sole discretion, and make your purchase decisions only on the basis of generally available

products and features. Salesforce doesn't guarantee general availability of this feature within any particular time frame or at all, and we can discontinue it at any time. This feature is for evaluation purposes only, not for production use. It's offered as is and isn't supported, and Salesforce has no liability for any harm or damage arising out of or in connection with it. All restrictions, Salesforce reservation of rights, obligations concerning the Services, and terms for related Non-Salesforce Applications and Content apply equally to your use of this feature.

Why Is Time Zone Support Important?

Time zone support lets your dashboard viewers see data in a time zone that's relevant to them. Here's an example.

👁 **Example:** The customer service operations manager at a California-based company monitors daily case creation on a Service Operations dashboard. Here's what she sees today, Wednesday August 22, before time zone support is enabled in her org.
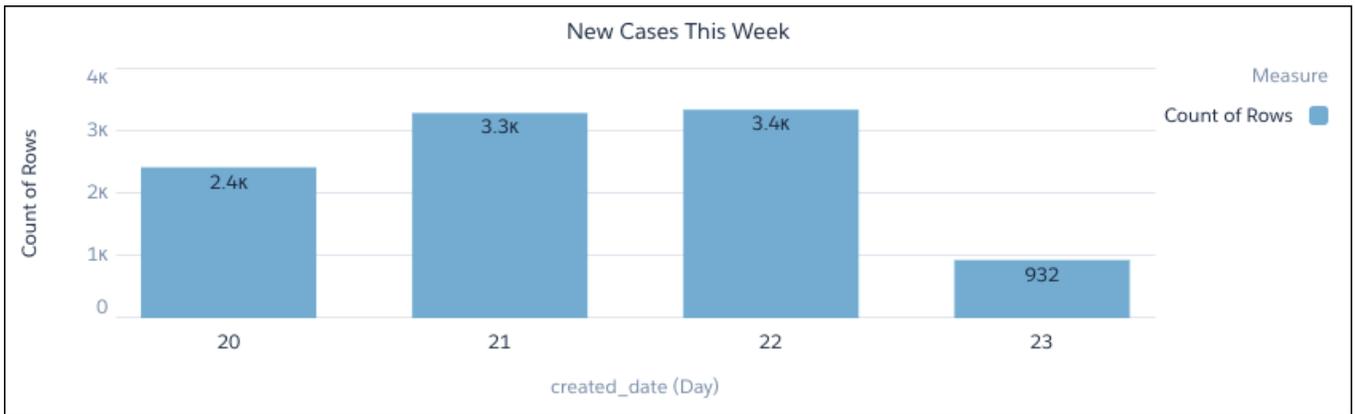
The dashboard chart shows the number of cases created so far this week. But it also shows cases created tomorrow, Thursday 23rd. The customer service operations manager is in the PDT time zone, but the dashboard shows data in GMT, which is 8 hours ahead. So, any cases created after 4:00pm in PDT appear as being created the next day when displayed in GMT.
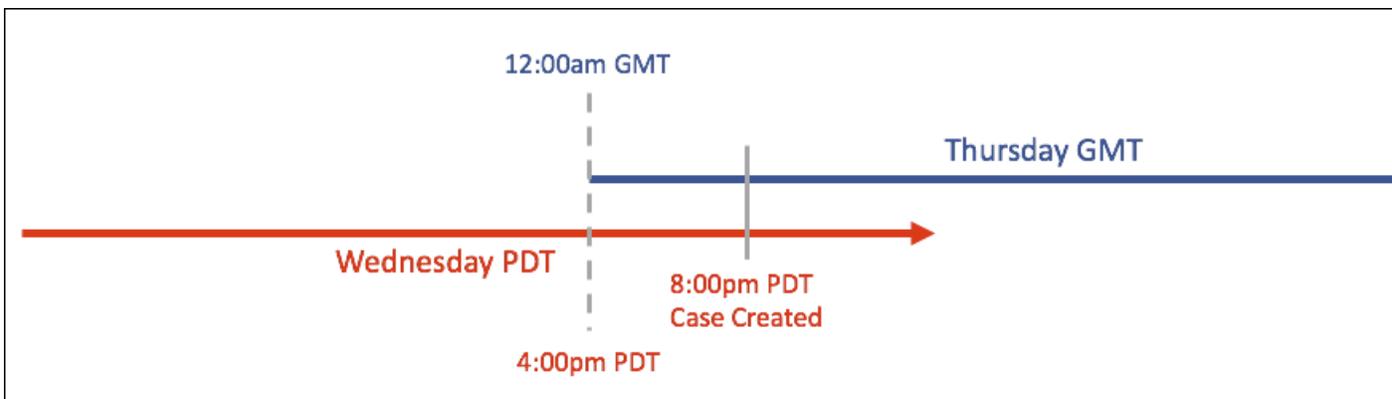
If these 932 cases were created in a GMT location, then the chart would be accurate. But the company has customer service agents only in California, so the widget isn't helpful. What's more, cases created on the last day of a reporting period can end up in the wrong month, quarter, or even year.

With time zone support enabled, and the supported time zone set to the company's local timezone of PDT, the customer service operations manager gets a much more accurate view of case creation this week.
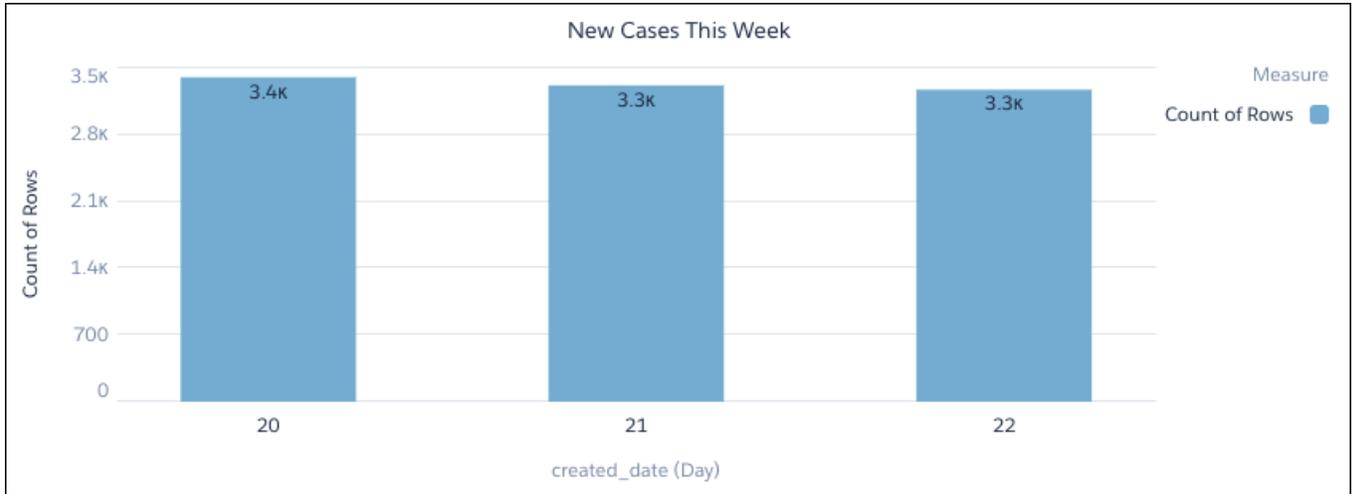
How Does Time Zone Support Work?

When you enable time zone support, you select a supported time zone for Tableau CRM. Tableau CRM converts date-time values in your datasets or connected objects from GMT to your selected time zone. When you create a dashboard, you have the option to create a time zone enabled dashboard. Time zone enabled dashboards use the converted date-time values from datasets, instead of the original values.

📝 **Note:** Connected objects are objects that sync with Salesforce or external data sources through data sync.

Let's take a closer look at how Tableau CRM converts date-time values. After you enable time zone support, you have to refresh your connected objects and datasets. Refresh connected objects by syncing them, and refresh datasets by running the related recipe or dataflow, or uploading the .csv file. During the refresh, Tableau CRM runs this process:
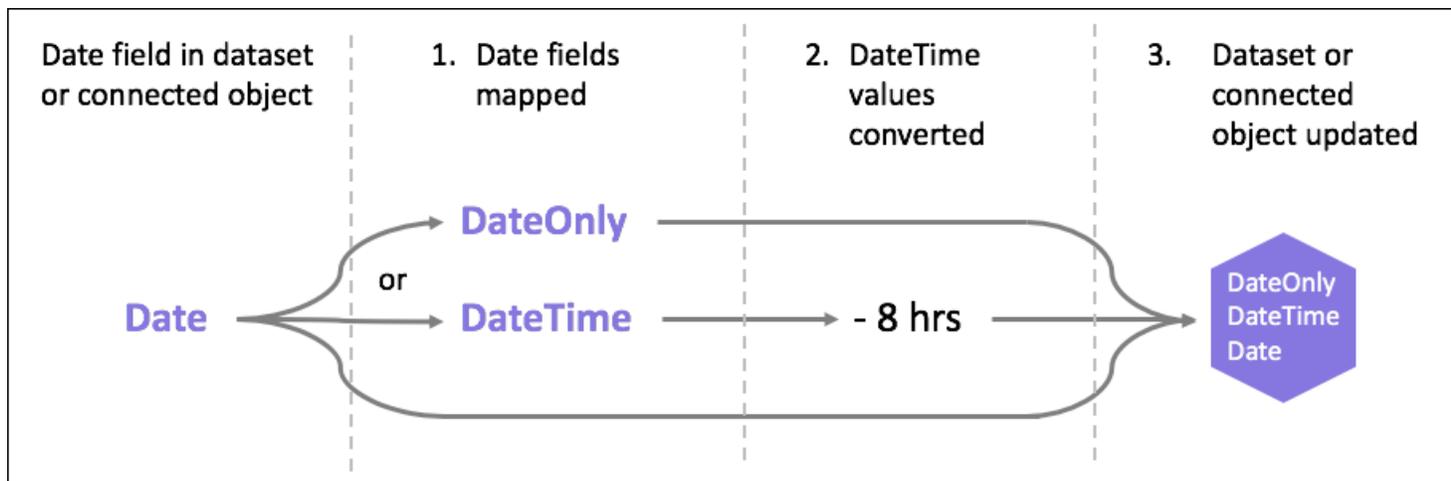
1. Tableau CRM maps each Date field in the dataset or connected object to a new DateOnly or DateTime field, depending on whether the field values have a time component.
2. Values in the new DateTime field are assumed to be GMT, and are converted to the time zone selected for Tableau CRM.

> Note: Dates from .csv-sourced datasets are assumed to be GMT unless you select a different source time zone when you upload the file. See Set a Source Time Zone for .Csv data.

3. The DateTime field generates fields in the new time zone and GMT (the original version), which are added to the updated dataset or connected object. The GMT field is included to ensure the dataset still works with your existing dashboards.

Handle Text Values

Confirm that text values in a column are uniform in formatting, spelling, and language. Inconsistencies can occur within data sources and after merging data from multiple data sources.

To ensure that your data is grouped properly during analysis, standardize the text strings used for dimensions.

- Use the Bucket transformation to replace similar values with a consistent value. For example, replace abbreviations with full text strings, or replace text strings with abbreviations.
- Fix other spelling inconsistencies, such as typographical errors, plurals, uppercase and lowercase characters, and so on.
- Use a column profile in Tableau CRM to review string length. For example, some column values require the same length, such as a 5-digit zip code or 10-digit phone number.

Handle Missing Values

Gaps in your data can throw off analysis. Missing values are best fixed in the source application by, for example, making it a required field. However, if you can't do that, you can use Tableau CRM to fill in missing data.

If your data has columns with missing values:

- Use a column profile in a recipe to determine if a column contains missing values and, if so, how many.
- Add the Predict Missing Values transformation to a recipe to fill in missing dimension column values.
- If possible, set a default value for null measures.
- Enable Null Measure Handling to properly handle null numeric values in source data.

SEE ALSO:

Handle Null Numeric Values

Profile Columns to Understand Data in a Data Prep Recipe

Predict Missing Values Transformation: Fill In Missing Values

Dataset Capacity and Limits

Before you create any datasets, review the limits. For example, each Salesforce org has a maximum number of rows for all datasets in the org. There are also limits on the number of columns in a dataset and characters in a column.

SEE ALSO:

Tableau CRM Limits

Prepare and Load Data into Datasets with Recipes and Dataflows

Unlike other Tableau CRM tools that help you create datasets, recipes and dataflows allow you to prepare your data before loading it into datasets. Data preparation is the process of transforming your data into a form that's meaningful and valuable to the people consuming it. For example, you can define data preparation logic that combines data from two data sources and cleans up inconsistencies, such as differently formatted dates and codes.

Dataflows and recipes follow the same process for preparing data. To speed up the process of getting data into datasets, a data sync pulls data from the data source in advance and stores it in connected objects inside Tableau CRM. A recipe or dataflow then uses the connected objects as sources, prepares that data, and then loads the results into one or more datasets.
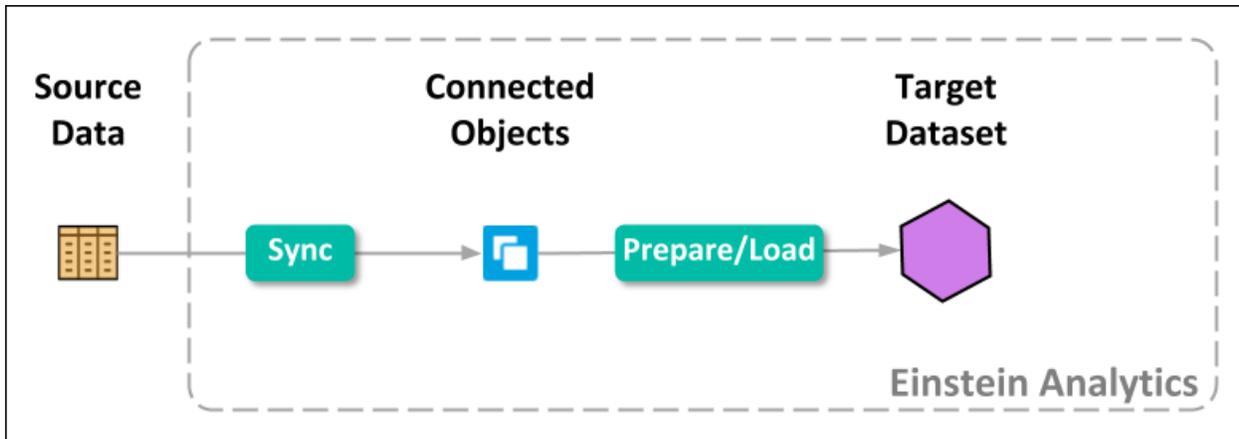
To set up access to source data, create a connection. When you create a connection, select objects and columns to pull data from. You can add a filter to the connection to extract a subset of all rows. In the connection properties, you also specify a user account that determines what data the connection can access. For example, to access data in Amazon S3, specify an Amazon S3 user account. If the user account doesn't have access to an object, the connection can't pull data from that object.

After you create a connection, run the associated data sync job to extract the data from each selected object in the data source and store it in the corresponding connected object. After you run a data sync for the first time, you can add the connected objects as sources for recipes and dataflows. In recipes and dataflows, you can add transformations to prepare the data in the connected objects and load the results into datasets.
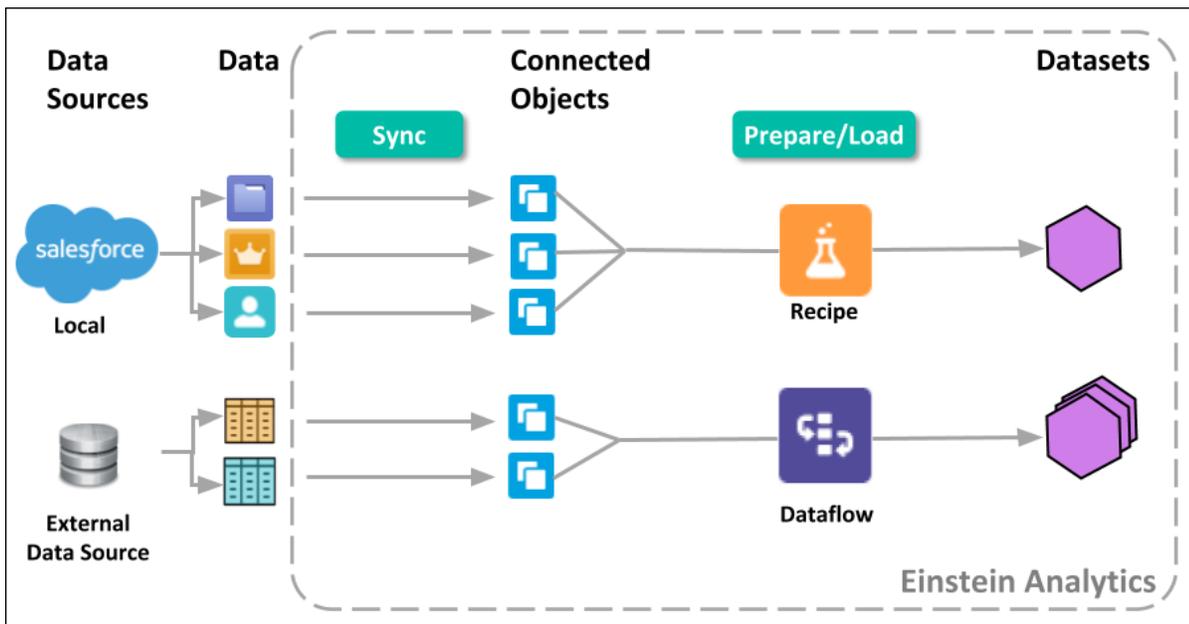
Run the recipes and dataflows to create the datasets. Continue to run them to refresh the data. You can run data syncs, dataflows, and recipes on demand. You can also schedule them to run on an ongoing basis. To ensure that your recipes and dataflows use the latest data, see to it that the data sync jobs complete before dependent recipes and dataflows run.

1. Connect and Sync Your Data to Tableau CRM

   Tableau CRM connectors give you an easy way to connect data inside and outside of Salesforce with Tableau CRM. Tableau CRM provides a prebuilt connector for data in your local org and a range of configurable connectors for remote data in external Salesforce orgs, apps, data warehouses, and database services. Connectors are available if you turned on Tableau CRM after the Winter '20 release or after enabling Data Sync for your org.

2. Design Datasets from Synced Data and Other Datasets with Recipes and Dataflows

   You can use a recipe or dataflow to design a dataset based on input data. In each recipe and dataflow, select the input data (connected objects or existing datasets), add data preparation logic to transform that data, and specify the dataset to load the results into. For example, you can use a recipe to combine data from different sources, clean the data to make it consistent, and then load the results into a new dataset.

3. Run Data Sync, Recipes, and Dataflows to Create and Refresh Datasets

   Whether you use local Salesforce data or pull data from an external source, you must set up Tableau CRM to load the data, make it available to Tableau CRM, and keep it up to date.

SEE ALSO:

Considerations Before Integrating Data into Datasets

Ways to Get Data from Data Sources Into Datasets

Should I Use a Recipe or Dataflow?

Connect and Sync Your Data to Tableau CRM

Tableau CRM connectors give you an easy way to connect data inside and outside of Salesforce with Tableau CRM. Tableau CRM provides a prebuilt connector for data in your local org and a range of configurable connectors for remote data in external Salesforce orgs, apps, data warehouses, and database services. Connectors are available if you turned on Tableau CRM after the Winter '20 release or after enabling Data Sync for your org.

Watch a Demo: ▶ Connect to External Data (English Only)

After you use a connector to create the connection to a data source, configure data sync to extract the needed source data into connected objects. You can then prepare the data in these connected objects, using a recipe or dataflow, to create your final dataset.
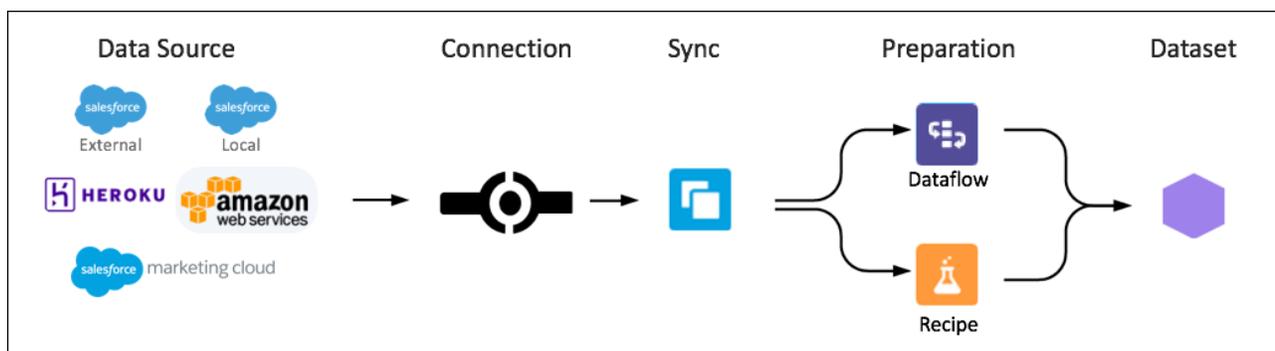
Connect to Local Salesforce Data

Use the Salesforce Connector to manage the data synced between your local Salesforce org and Tableau CRM. Filter data synced to Tableau CRM, create more connections using the Salesforce Local connector, manage objects between connections, and add or remove objects and fields from sync. Use the Salesforce External connector to sync data from a remote Salesforce org.

Add and Remove Local Salesforce Objects and Fields from Data Sync

Update data sync to keep up with your changing business by adding local Salesforce objects, removing objects, and managing which fields are synced. Synced data is stored in Tableau CRM as objects that you can use in dataflows and recipes.

📝 Note: For help with managing data sync for remote objects, see

1. In Tableau CRM, click the gear icon (⚙️), and select **Data Manager**.

2. Click the **Connect** tab. All objects configured to sync are listed under their connection name.

3. To add an object and its fields:

     a. Select **Connect to Data**.

     b. Select the source for your object. For a local Salesforce object, the default source is SFDC_LOCAL.

     c. Select the object, and click **Continue**.

     d. Select the fields that you want to sync, and click **Continue**.

     e. Optionally, filter the data on page 618 synced to Tableau CRM.

     f. Save your changes.

4. To add or remove a field from data sync for an object already configured for sync:

     a. To open the field picker, select the object's name. Fields already included for sync are selected.
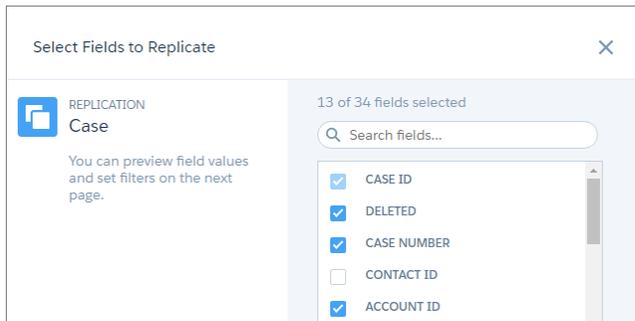
     b. To add a field, check the box next to the field. To remove a field, uncheck the box next to the field.

           Note: Consider the following when changing the field selections for a Salesforce object.

              • You can't deselect a field that's used in a dataflow or deselect an object's Id field.

              • New fields that you select here are synced separately from the dataflow through the Salesforce Local connection.

              • If you delete any sync-enabled field or make any sync-enabled field inaccessible to the Analytics Integration User profile, an object's data sync fails. Undelete the field, restore access for the Analytics Integration User profile, or deselect the field from data sync to ensure that sync runs successfully.

     c. Select **Continue**.

     d. To view a field's attributes, click the field's column header, and then click [✎]. You can review the field's API name and type, and settings such as default value and precision. Field attributes for local Salesforce objects are read only.

     e. Click **Save**.

     Note: To add an object to data sync when using dataflow, add an sfdcDigest node for the object to one of your dataflows. To add a field for a Salesforce object, add it to an existing sfdcDigest node for the object in one of your dataflows. The new object or field is added automatically to data sync when you update the dataflow.

Create Another Salesforce Local Data Connection

Create connections using the Salesforce Local connector to strategically manage when your Salesforce data syncs to Tableau CRM using individually scheduled local connections. A local connection, SFDC_LOCAL, is included when your Tableau CRM org is activated.

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click **Salesforce Connector**.

6. Enter the connection settings.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections. *SFDC_LOCAL_* is automatically added to the beginning of the connection name after save. For example, if you assigned the connection name `Hourly_Sales`, you see the connection SFDC_LOCAL_Hourly_Sales added to your list. The SFDC_LOCAL_ can be edited or removed after the first save.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.

7. After you finish entering the settings, save your changes.

With the local connection created you can add new objects on page 616 to sync, move existing objects on page 618 from another local connection, or filter which data is synced on page 618.

When you use data from multiple local connections in a recipe or dataflow, event-based scheduling will run only after all local connections are synced. For example, if one local connection is scheduled to sync hourly and another syncs daily, an event-based scheduled recipe using data from both runs one time daily.

Move Salesforce Objects Between Local Connections

You can reassign objects from one local connection to another local connection.

1. Expand the local connection where the object is assigned.

2. Click the down arrow to the right of the object's name.

3. Click **Switch Connection**. This option is available after you've created another local connection.

4. Click the connection to associate this object with.

5. Save your changes.

An object that is moved to another connection runs data sync when next used by a dataflow.

Filter Local Data Synced to Tableau CRM

Exclude unnecessary or sensitive data from syncing to Tableau CRM with data sync filters. Filters run on the source object and speed up data sync by pulling only the data you need into Tableau CRM. If you want to use excluded data in the future, use a dataflow or recipe filter to limit the data shown in a dataset instead of a data sync filter.

1. In Tableau CRM, click the gear icon ( ), and select **Data Manager**.

2. Click the **Connect** tab. All objects configured to sync are listed under their connection name.

3. Select the object to filter.

4. Select **Continue**.

5.

From the Preview Source Data page, select [filter icon]. Enter a filter.

> 📝 Note: Field attributes for local Salesforce objects are read only.

6. Save your changes.

> 👁 Example: Salesforce local object filters use the SOQL WHERE clause expression as described in the SOQL and SOSL Reference.
> For example, `"(FiscalQuarter = 2 OR FiscalQuarter = 3) AND Amount > 1000 AND Amount <= 20000"`.

Salesforce Connector for Local Salesforce Data Considerations

Keep these behaviors in mind when working with the Salesforce Connector.

- You can save up to 10 Salesforce local connections.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Connect to Remote Data Outside of Your Salesforce Org

Create a remote connection to sync external data with Tableau CRM. Connectors are available if you turned on Tableau CRM after the Winter '20 release or after enabling Data Sync for your org. For each connector, we provide information on how to create the connection, configure its properties, and track important considerations.

Salesforce Connectors

Use these connectors to sync your data in systems that are part of the Salesforce ecosystem, including external Salesforce orgs, to Tableau CRM.

Application Connectors

Use these connectors to sync data from your external applications to Tableau CRM.

Database Connectors

Use these connectors to sync data from your external database services to Tableau CRM.

Analytics Mulesoft Connectors

Pull your cloud-based data via CloudHub into Tableau CRM without code using Analytics Mulesoft connectors for Tableau CRM. After the connection to your data source via CloudHub is configured, choose the objects to sync to Tableau CRM. Once in Tableau CRM, your data can be analyzed independently and with other synced data from your Salesforce org or other remotely connected data sources.

Object Store and No SQL Connectors

Use these connectors to sync data from your object store and non-SQL services to Tableau CRM.

Pilot and Beta Connectors

You can use these connectors as a pilot or beta feature to sync external data with Tableau CRM. Before choosing a connector, review its pilot or beta instructions.

Add and Remove Remote Objects and Fields from Data Sync

Update data sync for connected external data sources to keep up with your changing business by adding objects, removing objects, and managing which fields are synced. Synced data is stored in Tableau CRM as objects that can be used in dataflows and recipes.

Salesforce Connectors

Use these connectors to sync your data in systems that are part of the Salesforce ecosystem, including external Salesforce orgs, to Tableau CRM.

Salesforce External Connection

Create a remote connection using the Salesforce External connector to sync data from another Salesforce org to Tableau CRM.

Salesforce Marketing Cloud Contacts OAuth 2.0 Connection

Create a remote connection using the Salesforce Marketing Cloud Contacts OAuth 2.0 connector to sync contacts data from Marketing Cloud to Tableau CRM.

Salesforce Marketing Cloud Contacts Connection

Create a remote connection using the Salesforce Marketing Cloud Contacts connector to sync contacts data from Marketing Cloud to Tableau CRM.

Customer 360 Global Profile Data Connection (Beta)

Use the Customer 360 Global Profile Data connector to access global profile data from connected orgs, along with other Salesforce and remote data, for in-depth analysis in Tableau CRM.

Salesforce External Connection

Create a remote connection using the Salesforce External connector to sync data from another Salesforce org to Tableau CRM.

📝 **Note:** Use Salesforce Output Connection (Beta) on page 738 to push data from Tableau CRM to a Salesforce org.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Username	User name used to connect to external Salesforce org for sync of objects and fields.
Service URL	Login URL for the external org. This is completed for you, but you can change it if the org is a sandbox or it uses a My Domain name or custom domain. Enter in the format: `<http or https>://<MyDomainName.my or CustomDomain>.salesforce.com/services/Soap/<Identifier>/<Version Number>.0` For example: `https://login.salesforce.com/services/Soap/u/34.0`
Password	Password for the user specified in *Username*. Depending on your security settings, append the user's security token to the end of the password. For more information about obtaining a security token, see Reset Your Security Token.

Filter Data Synced to Tableau CRM

Exclude unnecessary or sensitive data from syncing to Tableau CRM with data sync filters. Filters run on the source object and speed up data sync by pulling only the data you need into Tableau CRM.

> 📝 **Note:** If you might use excluded data in the future, use a dataflow or recipe filter to limit the data shown in a dataset instead of a data sync filter.

When adding or modifying an object as described in Add and Remove Remote Objects and Fields from Data Sync, click [▼]. For the Salesforce External connector, enter a filter using a SOQL WHERE clause expression as described in the SOQL and SOSL Reference. For example, `"(FiscalQuarter = 2 OR FiscalQuarter = 3) AND Amount > 1000 AND Amount <= 20000"`.

Salesforce External Connector Considerations

Keep these behaviors in mind when working with the Salesforce External connector.

- This connector can sync up to 20 million rows or 10 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- The security settings of the user that you use to connect determine the objects and fields available for sync in the external org.
- Standard objects available in API versions 34.0 to 46.0 are eligible for data sync with this connector.
- To troubleshoot connectivity issues between Salesforce orgs, review your org IP address restrictions and consider disabling the setting "Lock sessions to the IP address from which they originated" in both orgs. See this knowledge article for more information.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

Salesforce External Connector Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

Salesforce Marketing Cloud Contacts OAuth 2.0 Connection

Create a remote connection using the Salesforce Marketing Cloud Contacts OAuth 2.0 connector to sync contacts data from Marketing Cloud to Tableau CRM.

> ⛔ **Important:** If your Marketing Cloud package app was created using a legacy endpoint or without OAuth 2.0 authentication, use the Salesforce Marketing Cloud Contacts Connection instead of this connector.

Connection Details

Marketing Cloud packages using a tenant-specific endpoint and OAuth 2.0 authentication are required for this connector.

The security settings of the user that you use to connect determine the objects and fields available for sync.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
ClientSecret	The client secret of Salesforce Marketing Cloud required to generate a valid access token.
ClientId	The client ID of Salesforce Marketing Cloud required to generate a valid access token.
Salesforce Marketing Cloud Url	The URL used to connect to the Salesforce Marketing Cloud WSDL. This is already completed, but you can change it.
UTC Offset	Optional setting. Used to read data from Salesforce Marketing Cloud in a UTC offset time zone.

Salesforce Marketing Cloud Contacts OAuth 2.0 Connector Considerations

Keep these behaviors in mind when working with the Salesforce Marketing Cloud Contacts connector.

- This connector can sync up to 10 million rows or 5 GB per object, whichever limit it reaches first.

- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.

- The Salesforce Marketing Cloud Contacts OAuth 2.0 connector supports these types of data extensions:

  – System Data.Contact

  – System Contact Linked Data Extension, and Child and Grandchild System Contact Linked Data Extensions

  – Custom Contact Linked Data Extension, and Child and Grandchild Custom Contact Linked Data Extensions

- Marketing Cloud packages from a Marketing Cloud sandbox environment are not supported.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Salesforce Marketing Cloud Contacts Connection

Create a remote connection using the Salesforce Marketing Cloud Contacts connector to sync contacts data from Marketing Cloud to Tableau CRM.

> ⊘ Important: If your Marketing Cloud package app was created using a tenant-specific endpoint and OAuth 2.0 authentication, use the Salesforce Marketing Cloud Contacts OAuth 2.0 Connection instead of this connector.

Connection Details

Marketing Cloud packages created using a legacy endpoint and without OAuth 2.0 authentication are required for this connector.

The security settings of the user that you use to connect determine the objects and fields available for sync.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Username	The Salesforce Marketing Cloud username.
ClientSecret	The client secret of Salesforce Marketing Cloud required to generate a valid access token.
ClientId	The client ID of Salesforce Marketing Cloud required to generate a valid access token.
Salesforce Marketing Cloud Url	The URL used to connect to the Salesforce Marketing Cloud WSDL. This is already completed, but you can change it.
UTC Offset	Optional setting. Used to read data from Salesforce Marketing Cloud in a UTC offset time zone.
Password	The password for the Salesforce Marketing Cloud username.

Salesforce Marketing Cloud Contacts Connector Considerations

Keep these behaviors in mind when working with the Salesforce Marketing Cloud Contacts connector.

- This connector can sync up to 20 million rows or 10 GB per object, whichever limit it reaches first.
- The Salesforce Marketing Cloud Contacts connector supports these types of data extensions:
  - System Data.Contact
  - System Contact Linked Data Extension, and Child and Grandchild System Contact Linked Data Extensions
  - Custom Contact Linked Data Extension, and Child and Grandchild Custom Contact Linked Data Extensions
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

Salesforce Marketing Cloud Contacts Connector Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

Customer 360 Global Profile Data Connection (Beta)

Use the Customer 360 Global Profile Data connector to access global profile data from connected orgs, along with other Salesforce and remote data, for in-depth analysis in Tableau CRM.

> Note: As a beta feature, Customer 360 Global Profile Data Connector is a preview and isn't part of the "Services" under your master subscription agreement with Salesforce. Use this feature at your sole discretion, and make your purchase decisions only on the basis of generally available products and features. Salesforce doesn't guarantee general availability of this feature within any particular time frame or at all, and we can discontinue it at any time. This feature is for evaluation purposes only, not for production use. It's offered as is and isn't supported, and Salesforce has no liability for any harm or damage arising out of or in connection with it. All restrictions, Salesforce reservation of rights, obligations concerning the Services, and terms for related Non-Salesforce Applications and Content apply equally to your use of this feature.

Connection Details

Before you can use this connector, connect a Salesforce org with Tableau CRM as a data source in Customer 360 Data Manager. Create and load a data file from your connected data source into Customer 360 Data Manager for processing into global profiles. Then export the global profile data so that it can be pushed to Tableau CRM.

Select one or more objects that you want to sync to Tableau CRM. From this preview, you can view field attributes for all the files that you are loading from the folder.

Enable Customer 360 Global Profiles and Data

To set up Tableau CRM to include Customer 360 global profiles and data, complete the following steps.

1. From Setup, enter `Analytics` in the Quick Find box, then select **Settings**.

2. Select **Enable Data Sync and Connections**.

3. Select **Create dashboards from Customer 360 Data Manager global profile data**.

4. Click **Save**.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Master Symmetric Key	Salesforce uses this master symmetric key to enable client-side data encryption.
Secret Key	This Salesforce-managed secret access key is a read-only field.
Region Name	Region of your Salesforce service. This field is read-only.
Folder Path	Path to the folder that Salesforce connects to. This field is read-only.
AWS Access Key ID	A Salesforce-managed, read-only bucket access key ID.

Connector Considerations

Keep these behaviors in mind when working with this connector.

- This connector can sync up to 100 million rows or 50 GB per object, whichever limit it reaches first.

- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.

- Disabling the connector on the Tableau CRM Settings page removes the connection. You can disable the connector only if it isn't in use.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Application Connectors

Use these connectors to sync data from your external applications to Tableau CRM.

Google Analytics Connection

Create a remote connection using the Google Analytics connector to sync Google Analytics data to Tableau CRM.

Microsoft Dynamics CRM Connection

Create a remote connection using the Microsoft Dynamics CRM connector to sync data from your Microsoft Dynamics 365 account to Tableau CRM.

NetSuite Connection

Create a remote connection using the NetSuite connector to sync data from NetSuite to Tableau CRM.

Oracle Eloqua Connection

Create a remote connection using the Oracle Eloqua connector to sync data from Oracle Eloqua to Tableau CRM.

Google Analytics Connection

Create a remote connection using the Google Analytics connector to sync Google Analytics data to Tableau CRM.

Connection Details

Use a Google service account instead of a user account to grant access to your Google Analytics data. For more information about Google service accounts, see Understanding Service Accounts.

Generate a JSON file containing the required client email and private key from the IAM & Admin page in the Google Cloud Platform Console.

Create the Connection

1.
   In Tableau CRM, click the gear icon ( ) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.

Setting	Description
APIVersion	Version of the Google Core Reporting API used for the connection. The Google Analytics connector uses Core Reporting API v3. This value is populated for you.
	🖉 **Note:** For information about limits associated with Core Reporting API v3, see Google's Limits and Quotas on API Requests guide.
Client Email	Enter the `client_email` value from the JSON file.
Private Key	Enter the `private_key` value from the JSON file.
	🖉 **Note:** Don't include `"private_key":` `"-----BEGIN PRIVATE KEY-----\n` at the start of the key and `\n-----END PRIVATE KEY-----\n"` at the end.

Advanced Object Properties

Each time you connect to an object with a Google Analytics connection, you also provide advanced properties for the object. The Google Core Reporting API uses these properties to query the object in Google Analytics. All settings require a value.

Property	Value
StartDate	The start date of the date range that you want to sync.
	Enter in the format `YYYY-MM-DD`, or use a relative date keyword, such as `today`, `yesterday`, or `NdaysAgo`, where `N` is a positive integer.

Property	Value
ViewID	The View ID for the Google Analytics account, which determines which data is available for sync. You can find the View ID in the account's View Settings in Google Analytics.
EndDate	The end date of the date range that you want to sync.
	Enter in the format $YYYY-MM-DD$, or use a relative date keyword, such as $today$, $yesterday$, or $NdaysAgo$, where N is a positive integer.

Note: For more information about supported StartDate and EndDate values, see Google's Core Reporting API Reference Guide.

Google Analytics Connector Considerations

Keep these behaviors in mind when working with the Google Analytics connector.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

- This connector can sync up to 10 million rows or 5 GB per object, whichever limit it reaches first.

- Google Analytics limits the number of rows that you can sync for each object to 1 million. When you sync more than 1 million rows, Tableau CRM aggregates data from the extra rows and syncs it as a row marked "Others."

- Custom fields are not supported.

- Google Analytics stores your website data as a binary large object (blob). It uses SQL views to query the blob to create 32 virtual objects that you can connect to.

    Note: Knowledge of Google Analytics Reporting API is necessary when selecting which virtual objects to connect to. Google Analytics Reporting API rules for allowed object and metric interactions, described in the Dimensions & Metrics Explorer, impact connecting to Tableau CRM. When troubleshooting, use the Google Analytics Query Explorer to confirm that the fields that you intend to query are valid.

- Tableau CRM creates a GLOBAL_ACCESS_OBJECT virtual object containing fields from the other virtual objects. Use this object to prepare datasets for more complex scenarios.

- Eight of the virtual objects contain only dimensions. You can't connect to these objects.

- The Google Core Reporting API includes these limits.

    - You can select up to seven dimensions and 10 measures for each virtual object that you connect to.

    - You can sync up to one million rows for each virtual object. When you sync more than one million rows, Tableau CRM aggregates the data from the extra rows and syncs the data as a row marked "Others." To sync more than one million rows for an object, consider using the Google BigQuery connector.

SEE ALSO:

Google Analytics Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

Microsoft Dynamics CRM Connection

Create a remote connection using the Microsoft Dynamics CRM connector to sync data from your Microsoft Dynamics 365 account to Tableau CRM.

Create the Connection

1. In Tableau CRM, click the gear icon ( ) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Username	User ID registered with Microsoft Dynamics CRM.
Organization Name	Microsoft Dynamics CRM organization name. Organization names are case-sensitive.
Discovery Service URL	URL of the Microsoft Dynamics CRM service. Enter a URL in the format: `<http or https>://disco.crm<ID>.dynamics.com/XRMServices/2011/Discovery.svc` For example: `https://disco.crm1.dynamics.com/XRMServices/2011/Discovery.svc`
Domain	Domain to which the user belongs. Enter the complete domain name. For example, `msd.sampledomain.com`.
Authentication Type	Authentication type for the connection. Enter `Passport` Note: Tableau CRM supports only the Passport authentication type for Microsoft Dynamics CRM connections.
Password	Password to authenticate the user specified in Username.

Microsoft Dynamics CRM Connector Considerations

Keep these behaviors in mind when working with the Microsoft Dynamics CRM connector.

- This connector can sync up to 20 million rows or 10 GB per object, whichever limit it reaches first.

- The security settings of the user that you use to connect determine the data available for sync.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

Microsoft Dynamics CRM Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

NetSuite Connection

Create a remote connection using the NetSuite connector to sync data from NetSuite to Tableau CRM.

Connection Requirements

Before you connect to a NetSuite account, you must install the NetSuite V2 bundle in the account.

1. Log in to NetSuite using a Full Access account.

2. Go to **Customization** > **SuiteBundler** > **Search and Install Bundles**.

3. In the **KEYWORDS** field, enter `Netsuite_V2_Bundle` and click **Search**.

4. Click the **Netsuite_V2_Bundle** link.

5. Click **Install** and complete the installation.

NetSuite connections use token-based authentication to access NetSuite. To create access tokens, first create an integration in the NetSuite account and enable token-based authentication for it. You can then create access tokens for this integration.

Create the Connection

1. In Tableau CRM, click the gear icon ( ) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.

Setting	Description
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Consumer Secret	Client password associated with the NetSuite account. Generated when you create an integration and enable token-based authentication in NetSuite.
Consumer Key	Client key associated with the NetSuite account. Generated when you create an integration and enable token-based authentication in NetSuite.
Account ID	NetSuite account ID. Find this ID in Web Service Preferences in your NetSuite account.
Rest Domain	Rest domain name. Enter a URL in the format: `https://<Rest Domain Name>.netsuite.com` The rest domain name value depends on your environment or account. Valid options are: • `rest.na1.beta` for a beta environment • `rest.sandbox` for a sandbox account • `rest` or `rest.na1` for a production account For example: `https://rest.netsuite.com`
Token ID	Token ID for the access token generated in NetSuite.
Token Secret	Token secret for the access token generated in NetSuite.

NetSuite Connector Considerations

Keep these behaviors in mind when working with the NetSuite connector.

- This connector can sync up to 100,000 rows or 500 MB per object, whichever limit it reaches first.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

Netsuite Connector Known Issues and Knowledge Articles

Oracle Eloqua Connection

Create a remote connection using the Oracle Eloqua connector to sync data from Oracle Eloqua to Tableau CRM.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Base URL	URL to connect to the Eloqua application. Enter in the format `https://<Base Name>.eloqua.com` The data center that hosts your Eloqua account determines the base URL. For help with finding your account's base URL, see Determining base URLs in the Oracle Eloqua Developer Help Center. For example: ` https://secure.s04.eloqua.com `
Username	User name for the Eloqua account.
Time Zone Offset	Time zone in the Eloqua system settings relative to GMT. For example, if the Eloqua system time zone is GMT-06:00, enter `-06:00`
Domain Name	Company name for the Eloqua account, used at Eloqua sign-in.
Password	Password for the Eloqua account.

Oracle Eloqua Connector Considerations

Keep these behaviors in mind when working with the Oracle Eloqua connector.

- This connector can sync up to 10 million rows or 5 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- Object names must not contain special characters, such as spaces.
- You can select up to 250 fields for each connected object.
- You can export up to 5 million activity records from Oracle Eloqua. For more information, see the Oracle Eloqua export limits.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

Oracle Eloqua Connector Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

Database Connectors

Use these connectors to sync data from your external database services to Tableau CRM.

Amazon RDS Connection

Create a remote connection using one of the Amazon RDS connectors to bring data managed on the Amazon Relational Database Service into Tableau CRM.

Amazon Redshift Connection

Create a remote connection using the Amazon Redshift connector to sync data from Amazon Redshift to Tableau CRM.

Google BigQuery for Legacy SQL Connection

Create a remote connection using the Google BigQuery connector to sync data from Google BigQuery to Tableau CRM.

Google BigQuery Standard SQL Connection

Bring your large volumes of Google BigQuery data with standard SQL support into Tableau CRM using the Google BigQuery Standard SQL connector.

Google Cloud Spanner Connection

Create a remote connection using the Google Spanner connector to sync data from Google Cloud Spanner to Tableau CRM.

Heroku Postgres Connection

Create a remote connection using the Heroku Postgres connector to sync Heroku Postgres data to Tableau CRM.

Microsoft Azure SQL Data Warehouse Connection

Create a remote connection using the Microsoft Azure SQL Data Warehouse connector to sync data from Microsoft Azure SQL Data Warehouse to Tableau CRM.

Microsoft Azure SQL Database Connection

Create a remote connection using the Microsoft Azure SQL Database connector to sync data from Microsoft Azure SQL Database to Tableau CRM.

SAP HANA Connection

Create a remote connection using the SAP HANA connector to sync data from cloud-based SAP HANA to Tableau CRM.

Snowflake Connection

Create a remote connection using the Snowflake connector to sync data from Snowflake to Tableau CRM.

Amazon RDS Connection

Create a remote connection using one of the Amazon RDS connectors to bring data managed on the Amazon Relational Database Service into Tableau CRM.

Available Amazon RDS Connectors

- AWS RDS Aurora MySQL
- AWS RDS Aurora PostgresSQL
- AWS RDS MariaDB
- AWS RDS MySQL
- AWS RDS Oracle
- AWS RDS PostgreSQL
- AWS RDS SQL Server

📝 Note: Only SSL-enabled Amazon RDS instances are supported.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙️) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Database	Name of the database that you're connecting to.
JDBC Connection URL	URL schema for the database. Enter in the format: `<prefix>.<hostname>.<cluster>.rds.amazonaws.com:<port>` 📝 Note: The port is optional. If no port is specified, the default port is used.

Setting	Description
	For example:
	``` analyticsaws.csg5yuelzxbf.us-west-2.rds.amazonaws.com:5407 ```
Username	User name to connect to the database.
Schema 📝 Note: This setting is not available for MariaDB and MySQL connections.	Schema name for the database. Required if the JDBC connection URL doesn't provide enough context. Otherwise, this setting is optional. • Microsoft SQL Server—Use the schema name to specify the correct object. • PostgreSQL—Use the schema name to specify the correct object.
Password	Password to connect to the database.

Filter Data Synced to Tableau CRM

Exclude unnecessary or sensitive data from syncing to Tableau CRM with data sync filters. Filters run on the source object and speed up data sync by pulling only the data you need into Tableau CRM.

📝 Note: If you might use excluded data in the future, use a dataflow or recipe filter to limit the data shown in a dataset instead of a data sync filter.

When adding or modifying an object as described in Add and Remove Remote Objects and Fields from Data Sync, click ⌄▼. To filter an Amazon AWS RDS connector, enter a filter using the syntax described in its help.

- AWS RDS Aurora MySQL Help
- AWS RDS Aurora PostgresSQL Help
- AWS RDS MariaDB Help
- AWS RDS MySQL Help
- AWS RDS Oracle Help
- AWS RDS PostgreSQL Help
- AWS RDS SQL Server Help

Amazon RDS Connector Considerations

Keep these behaviors in mind when working with an Amazon RDS connector.

- This connector can sync up to 20 million rows or 10 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first. The AWS RDS SQL Server connector is not available to sync Salesforce Government Cloud org data.
- Unsigned INT and unsigned BIGINT fields aren't supported.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

Amazon RDS Connector Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

Amazon Redshift Connection

Create a remote connection using the Amazon Redshift connector to sync data from Amazon Redshift to Tableau CRM.

Connection Requirements

Before you create an Amazon Redshift connection, ensure that the Redshift cluster that you're connecting to is SSL enabled. For more information about using SSL, see Configure Security Options for Connections in Amazon Web Services documentation.

When you run a sync job on a Redshift object, the Redshift data is first staged in an Amazon S3 bucket and then pulled from the S3 bucket into Tableau CRM. To enable the connector to access the data, specify the name of the bucket, its secret access key, and its access key ID in the connection settings. You can only sync data included in the specified Amazon S3 bucket.

Create the Connection

1. In Tableau CRM, click the gear icon ( ) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Master Symmetric Key	Optional. Your Amazon S3 master symmetric key used to enable client-side data encryption. This must be a 256-bit AES encryption key in Base64 format.
AWS Secret Access Key	Your Amazon secret access key.
Username	User name for your Amazon Redshift account.

637

Setting	Description
Number Of Nodes in Cluster	The number of nodes in the Amazon Redshift cluster.
Schema	Your Amazon Redshift schema name. Only objects that exist within this schema are available for sync.
AWS Access Key ID	Your Amazon S3 bucket access key ID.
Cluster Node Type	Node type of the Amazon Redshift cluster.
S3 Bucket Name	The name of your Amazon S3 bucket.
JDBC URL	Your Amazon Redshift connection URL. Enter in the format: `jdbc:redshift://<cluster>.<hostname>.<region>.redshift.amazonaws.com:<port>/<database>` See Amazon Web Services' Obtain the JDBC URL for help with obtaining this URL. Use the exact URL specified in the cluster database properties. For example: `jdbc:redshift://examplecluster.de7bis04b.us-west-2.redshift.amazonaws.com:5631/dev`
Password	Password for your Amazon Redshift account.

Filter Data Synced to Tableau CRM

Exclude unnecessary or sensitive data from syncing to Tableau CRM with data sync filters. Filters run on the source object and speed up data sync by pulling only the data you need into Tableau CRM.

📝 **Note:** If you might use excluded data in the future, use a dataflow or recipe filter to limit the data shown in a dataset instead of a data sync filter.

When adding or modifying an object as described in Add and Remove Remote Objects and Fields from Data Sync, click 🔽 . For the Amazon Redshift connector, enter a filter in the syntax described in the Amazon Redshift help.

Amazon Redshift Connector Considerations

Keep these behaviors in mind when working with the Amazon Redshift connector.

- This connector can sync up to 100 million rows or 50 GB per object, whichever limit it reaches first.
- Ensure that the Redshift cluster that you're connecting to is SSL enabled. For more information about using SSL, see Configure Security Options for Connections in Amazon Web Services documentation.
- While connecting to a Redshift view, the field picker appears empty when a field's length exceeds 32,000 characters and the source data table and view are in different database schemas. Decrease the field length in Redshift, or move the table and view to the same schema.
- When syncing a Redshift object, the job quits if the query takes longer than the 60-minute threshold.
- Your Redshift node must be available to Salesforce data centers with these IP ranges. Nodes in a VPC or behind a firewall are not supported.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

Amazon Redshift Connector Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

Google BigQuery for Legacy SQL Connection

Create a remote connection using the Google BigQuery connector to sync data from Google BigQuery to Tableau CRM.

🛑 **Important:** Use the Google BigQuery Standard SQL Connection instead of this connector if using standard SQL.

Connection Requirements

The service account that you use to connect to Google Big Query must have the standard role "BigQuery Data Viewer" and a custom role with these permissions.

- bigquery.jobs.create
- bigquery.jobs.get
- bigquery.jobs.list
- bigquery.jobs.listAll
- bigquery.jobs.update
- bigquery.tables.create
- bigquery.tables.delete
- bigquery.tables.update
- bigquery.tables.updateData
- storage.buckets.list

Account creation generates a JSON file of account properties. Most of the necessary connection settings are in this file.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.

Setting	Description
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Schema	Name of the Google BigQuery Dataset ID you're connecting to.
Private Key	Enter the `private_key` value from the JSON file. 📝 Note: Don't include `"private_key":` `"-----BEGIN PRIVATE KEY-----\n` at the start of the key, and `\n-----END PRIVATE KEY-----\n"` at the end.
Client Email	Enter the `client_email` value from the JSON file.
Region Id	Optional setting. Enter the data location of the Google BigQuery dataset if the location isn't in North America. The dataset data location is found in the BigQuery console in the Dataset Info section. For example, `asia-northeast1`.
Storage Path	Required when using the Staging extract mode. Path in Google Cloud Storage where a local stage file is created to store the data temporarily. Enter the bucket name or the bucket name and folder name. For example, enter `<bucket_name>` or `<bucket_name>/<folder_name>`. 📝 Note: To write files to this staging area, the user specified in Client Email must have the list, read, and write permissions on the bucket.
Project ID	Enter the `project_id` value from the JSON file. If you have multiple projects with the same service account, enter the ID of the project containing the dataset that you want to connect to.
Extract Mode	Optional setting. Enter one of the following modes to extract data from Google BigQuery. Direct is the default if a mode isn't specified. **Direct** Use this mode to extract small data volumes. The connector extracts the data directly from Google BigQuery. **Staging** Use this mode to extract large data volumes, such as 10 million records. To increase performance when handling large data volumes, the connector stages the data in the specified Google Cloud storage path and then extracts the data from the storage path.

Setting	Description
	💡 **Tip:** If a direct extract fails, try using this extract mode.

Filter Data Synced to Tableau CRM

Exclude unnecessary or sensitive data from syncing to Tableau CRM with data sync filters. Filters run on the source object and speed up data sync by pulling only the data you need into Tableau CRM.

📝 **Note:** If you might use excluded data in the future, use a dataflow or recipe filter to limit the data shown in a dataset instead of a data sync filter.

When adding or modifying an object as described in Add and Remove Remote Objects and Fields from Data Sync, click [▼] . For the Google BigQuery connector, enter a filter in the syntax described in Google's BigQuery Help.

Google BigQuery Connector Considerations

Keep these behaviors in mind when working with the Google BigQuery connector.

- This connector can sync up to 100 million rows or 50 GB per object, whichever limit it reaches first.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

- BigQuery tables using data integrated from Google Drive aren't supported. The Google Drive data must be moved into BigQuery.

Nested and Repeated Fields

The connector flattens nested fields and syncs them as separate fields. In addition, the connector generates new rows for repeated fields. Let's look at an example. The contact data in this JSON file contains the current and previous address for each contact. The `addresses` field has nested `status`, `street`, `city`, and `state` fields. These nested fields appear twice for each contact: first for the current address, and then for the previous address.

```
{"id":"1","name":"James Park","addresses":[{"status":"current","street":"1 Harper
Alley","city":"Toledo","state":"OH"},{"status":"previous","street":"392 Parkside
Street","city":"Seattle","state":"WA"}]}
{"id":"2","name":"Lori Carr","addresses":[{"status":"current","street":"7501 Talisman
Court","city":"Fort Worth","state":"TX"},{"status":"previous","street":"3368 Anderson
Lane","city":"Moreno Valley","state":"CA"}]}
```

If you include the `addresses` field for sync, each nested field appears as a separate field in each row of the synced connected object. The repeated address fields result in two rows for each contact: one row for the current address, and one row for the previous address.

id	name	addresses.status	addresses.street	addresses.city	addresses.state
1	James Park	current	1 Harper Alley	Toledo	OH
1	James Park	previous	392 Parkside Street	Seattle	WA
2	Lori Carr	current	7501 Talisman Court	Fort Worth	TX

id	name	addresses.status	addresses.street	addresses.city	addresses.state
2	Lori Carr	previous	3368 Anderson Lane	Moreno Valley	CA

Temporary Destination Tables

The connector creates temporary destination tables in BigQuery to stage data before serving it to the client. The connector names these tables in the format `EA_TEMP_<5-character random string>_<objectName>`. Tables are usually deleted automatically. To avoid incurring extra data storage costs on Google BigQuery, check that all temporary tables have been deleted, and manually delete any tables that haven't been deleted.

Table Support

The Google BigQuery for Legacy SQL connector supports Standard SQL Tables, Legacy SQL Tables, and Legacy SQL Views, but doesn't support Standard SQL Views.

SEE ALSO:

Google BigQuery Connector Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

Google BigQuery Standard SQL Connection

Bring your large volumes of Google BigQuery data with standard SQL support into Tableau CRM using the Google BigQuery Standard SQL connector.

🛑 **Important:** Use the Google BigQuery for Legacy SQL Connection instead of this connector if using legacy SQL.

Connection Requirements

The service account that you use to connect to Google Big Query must have the standard role "BigQuery Data Viewer" and a custom role with these permissions.

- bigquery.jobs.create
- bigquery.jobs.get
- bigquery.jobs.list
- bigquery.jobs.listAll
- bigquery.jobs.update
- bigquery.tables.create
- bigquery.tables.delete
- bigquery.tables.update
- bigquery.tables.updateData
- storage.buckets.list

Account creation generates a JSON file of account properties. Most of the necessary connection settings are in this file.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.
2. In the data manager, click the **Connect** tab.
3. Click **Connect to Data**.
4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Schema	Name of the Google BigQuery Dataset ID you are connecting to.
Private Key	Enter the `private_key` value from the JSON file.
	📝 Note: Don't include `"private_key":` `"-----BEGIN PRIVATE KEY-----\n` at the start of the key, and `\n-----END PRIVATE KEY-----\n"` at the end.
Client Email	Enter the `client_email` value from the JSON file.
Region Id	Optional setting. Enter the data location of the Google BigQuery dataset if the location isn't in North America. The dataset data location is found in the BigQuery console in the Dataset Info section. For example, `asia-northeast1`.
Storage Path	Path in Google Cloud Storage where a local stage file is created to store the data temporarily. Enter the bucket name or the bucket name and folder name. For example, enter `<bucket_name>` or `<bucket_name>/<folder_name>`.
	📝 Note: To write files to this staging area, the user specified in Client Email must have the list, read, and write permissions on the bucket.
Project ID	Enter the `project_id` value from the JSON file.
	If you have multiple projects with the same service account, enter the ID of the project containing the dataset that you want to connect to.

Google BigQuery Standard SQL Connector Considerations

Keep these behaviors in mind when working with the Google BigQuery Standard SQL connector.

- This connector can sync up to 100 million rows or 50 GB per object, whichever limit it reaches first.

- Staging mode is supported. Direct mode is not supported.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

- BigQuery tables using data integrated from Google Drive are not supported. The Google Drive data must be moved into BigQuery.

- The connector does not flatten nested fields. When previewing nested data before load, only the top-level fields are shown.

Temporary Destination Tables

The connector creates temporary destination tables in BigQuery to stage data before serving it to the client. The connector names these tables in the format `EA_TEMP_<5-character random string>_<objectName>`. Tables are usually deleted automatically. To avoid incurring extra data storage costs on Google BigQuery, check that all temporary tables have been deleted, and manually delete any tables that haven't been deleted.

Table Support

The Google BigQuery Standard SQL connector supports Standard SQL Tables, Legacy SQL Tables, and Standard SQL Views, but does not support Legacy SQL Views.

Google Cloud Spanner Connection

Create a remote connection using the Google Spanner connector to sync data from Google Cloud Spanner to Tableau CRM.

Connection Requirements

A Google Cloud Spanner connection requires a private key and client email. You can generate a JSON file containing this information from the IAM & Admin page in the Google Cloud Platform Console.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.

Setting	Description
Description	Description of the connection for internal use.
Private Key	Enter the `private_key` value from the JSON file. 📝 **Note:** Don't include `"private_key":` `"-----BEGIN PRIVATE KEY-----\n` at the start of the key, and `\n-----END PRIVATE KEY-----\n"` at the end.
Instance Id	Enter the instance ID of the Google Cloud Spanner instance.
Database	Enter the database ID of the Google Cloud Spanner database.
Client Email	Enter the `client_email` value from the JSON file.
Project ID	Enter the `project_id` value from the JSON file. If you have multiple projects with the same service account, enter the ID of the project containing the dataset that you want to connect to.

Google Cloud Spanner Connector Considerations

Keep these behaviors in mind when working with the Google Cloud Spanner connector.

- This connector can sync up to 20 million rows or 10 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

Google Cloud Spanner Connector Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

Heroku Postgres Connection

Create a remote connection using the Heroku Postgres connector to sync Heroku Postgres data to Tableau CRM.

Connection Requirements

To access Heroku Postgres in Heroku Private Spaces, complete these steps before you create the connection.

1. Contact Heroku Support to activate Trusted IP ranges for data services in the space. Trusted IP ranges for data services is a beta feature.

2. Add the Salesforce IP ranges to the Trusted IP ranges in the space. See the Salesforce IP Addresses & Domains knowledge article for a list of Salesforce IP ranges.

Create the Connection

1. In Tableau CRM, click the gear icon (   ) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated. Go to the database in Heroku and click **View Credentials…** to access the values for the connection settings.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
JDBC Connection URL	URL schema for the database. Enter in the format: `jdbc:postgresql://<host>:<port>/<database>` Look for the host, port, and database details in the database credentials. For example: `jdbc:postgresql://ec1-23-45-67-89.compute-2.amazonaws.com:5112/a1b2c3`
Username	The database user, found in the database credentials.
Schema	The name of the schema you want to use for the database. If no schema names have been specified for the database, you can enter `public`.
Password	The database password, found in the database credentials.

Filter Data Synced to Tableau CRM

Exclude unnecessary or sensitive data from syncing to Tableau CRM with data sync filters. Filters run on the source object and speed up data sync by pulling only the data you need into Tableau CRM.

📝 Note: If you might use excluded data in the future, use a dataflow or recipe filter to limit the data shown in a dataset instead of a data sync filter.

When adding or modifying an object as described in Add and Remove Remote Objects and Fields from Data Sync, click [▼]. For the Heroku Postgres connector, enter a filter using the syntax described in the Heroku Postgres Help and your version of Postgres's Help.

Heroku Postgres Connector Considerations

Keep these behaviors in mind when working with the Heroku Postgres connector.

- This connector can sync up to 100 million rows or 50 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- The data available for sync is determined by the security settings of the user that you use to connect.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

Heroku Postgres Connector Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

Microsoft Azure SQL Data Warehouse Connection

Create a remote connection using the Microsoft Azure SQL Data Warehouse connector to sync data from Microsoft Azure SQL Data Warehouse to Tableau CRM.

Connection Requirements

The user that you use to connect to Microsoft Azure SQL Data Warehouse must have the db_owner privilege.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.
2. In the data manager, click the **Connect** tab.
3. Click **Connect to Data**.
4. Click **Add Connection**.
5. Click the connector's icon and enter its settings, as described in the Connection Settings section.
6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted

Setting	Description
	through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Azure DW JDBC URL	URL schema for the database. Enter in the format: `<host name>:<port>`
	To form the host name, append `.sql.azuresynapse.net` or `.database.windows.net` to the server name. The server name is found by following Microsoft's documentation. For example, if your server's name is *Blue_server*, the host name is *Blue_server.sql.azuresynapse.net*. The default port is 1433.
	Azure DW JDBC URL example:
	``` Blue_server.sql.azuresynapse.net:1433 ```
Azure DW JDBC Password	Password to connect to the Microsoft Azure SQL Data Warehouse account.
Azure Blob Account Name	Name of the Microsoft Azure Storage account to stage the files.
Database	Name of the Microsoft Azure database that you are connecting to.
Azure DW JDBC Username	User name to connect to the Microsoft Azure SQL Data Warehouse account.
Azure Blob Account Key	Microsoft Azure Storage access key to stage the files.
Azure DW Schema Name	Name of the schema in Microsoft Azure SQL Data Warehouse.

Filter Data Synced to Tableau CRM

Exclude unnecessary or sensitive data from syncing to Tableau CRM with data sync filters. Filters run on the source object and speed up data sync by pulling only the data you need into Tableau CRM.

> 📝 **Note:** If you might use excluded data in the future, use a dataflow or recipe filter to limit the data shown in a dataset instead of a data sync filter.

When adding or modifying an object as described in Add and Remove Remote Objects and Fields from Data Sync, click 🔻 . For the Microsoft Azure SQL Data Warehouse connector, enter a filter in the syntax described in the Azure SQL Database and Dynamic SQL documentation.

Microsoft Azure SQL Data Warehouse Connector Considerations

Keep these behaviors in mind when working with the Microsoft Azure SQL Data Warehouse connector.

- This connector can sync up to 20 million rows or 10 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

Microsoft Azure SQL Data Warehouse Connector Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

Microsoft Azure SQL Database Connection

Create a remote connection using the Microsoft Azure SQL Database connector to sync data from Microsoft Azure SQL Database to Tableau CRM.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Database	Microsoft Azure SQL database name.
JDBC Connection URL	URL schema for the database. Enter in the format: `<host name>:<port>` To form the host name, append `.database.windows.net` to the server name. The server name is found by following Microsoft's documentation. For example, if your server's name is *Blue_server*, the host name is *Blue_server.database.windows.net*. The default port is 1433.

Setting	Description
	URL example: `Blue_server.database.windows.net:1433`
Username	User name for the database login. Enter in the format: *<username>@<host name>* To form the host name, append *.database.windows.net* to the server name. The server name is found by following Microsoft's documentation. For example, if your server's name is *Blue_server*, the host name is *Blue_server.database.windows.net*.
Schema	Schema name for the database. You must enter a schema name if the JDBC connection URL doesn't provide enough context. The default value is *dbo*
Password	Password for the database login.

Filter Data Synced to Tableau CRM

Exclude unnecessary or sensitive data from syncing to Tableau CRM with data sync filters. Filters run on the source object and speed up data sync by pulling only the data you need into Tableau CRM.

📝 **Note:** If you might use excluded data in the future, use a dataflow or recipe filter to limit the data shown in a dataset instead of a data sync filter.

When adding or modifying an object as described in Add and Remove Remote Objects and Fields from Data Sync, click 🔽 . For the Microsoft Azure SQL Database connector,enter a filter in the syntax described in the Azure SQL Database and WHERE documentation.

Microsoft Azure SQL Database Connector Considerations

Keep these behaviors in mind when working with the Microsoft Azure SQL Database connector.

- This connector can sync up to 100 million rows or 50 GB per object, whichever limit it reaches first.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

Microsoft Azure SQL Database Connector Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

SAP HANA Connection

Create a remote connection using the SAP HANA connector to sync data from cloud-based SAP HANA to Tableau CRM.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
URL	SAP HANA database connection string. Must end in `.xsodata`. Enter in the format: `https://<Database Name>.ondemand.com/<Service Definition Path>` For example: `https://connectorshanatp.hanatrial.ondemand.com/Package/ProductDataService.xsodata`
Password	Password to connect to the SAP HANA database.
Username	User name to connect to the SAP HANA database.

SAP HANA Connector Considerations

Keep these behaviors in mind when working with the SAP HANA connector.

- This connector can sync up to 10 million rows or 5 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- This is an OData-based connector.
- Connection to on-premise SAP HANA is not supported.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

SAP HANA Connector Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

Snowflake Connection

Create a remote connection using the Snowflake connector to sync data from Snowflake to Tableau CRM.

Note: If you want to sync data from Tableau CRM to Snowflake, use the instead.

Create the Connection

1. In Tableau CRM, click the gear icon (   ) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Schema	Snowflake schema name.
Password	Password for your Snowflake account.
Database	Snowflake database name.
Role	Optional setting. Snowflake role assigned to the user that you are using to connect.
Additional JDBC URL Parameters	Optional setting. One or more JDBC connection parameters. Enter in the format *parameter1=value&parameter2=value&parameter3=value*

Setting	Description
	For example: `user=joates&warehouse=mywh&db=mydb&schema=public`
Warehouse	Snowflake warehouse name.
Username	User name for the Snowflake account.
Account	Name of your Snowflake account. 💡 **Tip:** The account name is the first segment in the domain in your Snowflake URL. For example, `123abc` is your account name in `https://123abc.snowflakecomputing.com`.

Filter Data Synced to Tableau CRM

Exclude unnecessary or sensitive data from syncing to Tableau CRM with data sync filters. Filters run on the source object and speed up data sync by pulling only the data you need into Tableau CRM.

📝 **Note:** If you might use excluded data in the future, use a dataflow or recipe filter to limit the data shown in a dataset instead of a data sync filter.

When adding or modifying an object as described in Add and Remove Remote Objects and Fields from Data Sync, click 🔽 . For the Snowflake connector, enter a SQL filter as described in the Snowflake WHERE documentation.

Snowflake Connector Considerations

Keep these behaviors in mind when working with the Snowflake connector.

- This connector can sync up to 100 million rows or 50 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- A Snowflake warehouse can be set to automatically resume or suspend. Review the Snowflake documentation for information about the optimal use of these features.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

Analytics Mulesoft Connectors

Pull your cloud-based data via CloudHub into Tableau CRM without code using Analytics Mulesoft connectors for Tableau CRM. After the connection to your data source via CloudHub is configured, choose the objects to sync to Tableau CRM. Once in Tableau CRM, your data can be analyzed independently and with other synced data from your Salesforce org or other remotely connected data sources.

How Analytics Mulesoft Connectors Work

When you create an Analytics Mulesoft connection, Salesforce generates a template in the associated CloudHub account. CloudHub is the integration platform as a service component of Mulesoft's Anypoint Platform, described further in Mulesoft's CloudHub documentation. The template selects the data source-specific driver and generates API calls that perform actions to securely connect with the source system. With the connection established, the metadata pull, sample data pull, and full data pull are available for Tableau CRM to configure the data sync.

Analytics Mulesoft Microsoft SQL Connector

Create a remote connection using the Analytics Mulesoft Microsoft SQL Server connector to sync data from Microsoft SQL Server on AWS RDS to Tableau CRM.

Analytics Mulesoft MySQL Connector

Create a remote connection using the Analytics Mulesoft MySQL connector to sync data from MySQL to Tableau CRM through Mulesoft.

Analytics Mulesoft Oracle Connector

Create a remote connection using the Analytics Mulesoft Oracle connector to sync data from Oracle to Tableau CRM through Mulesoft.

Analytics Mulesoft Microsoft SQL Connector

Create a remote connection using the Analytics Mulesoft Microsoft SQL Server connector to sync data from Microsoft SQL Server on AWS RDS to Tableau CRM.

Connection Requirements

A Mulesoft license, purchased or acquired separately, is required. The 30-Day Mulesoft Trial license is supported. Ask your Mulesoft admin for CloudHub credentials with access to deploy.

Create the Connection

1. In Tableau CRM, click the gear icon ( ) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Mulesoft Connections**.

5. Click **Add Connection**.

6. Click the connector's icon and enter its settings, as described in the Connection Settings section.

7. When you finish entering the settings, click **Save**.

📝 Note: Saving the connector prompts Anypoint to generate a Mule for the connection. Mule generation can take up to two minutes. If you attempt to open the new connection during Mule generation, an error message says that objects cannot be retrieved. You can check Mule status in Anypoint.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
	To simplify AnyPoint management, the application name of the Mule instance initiated by this connector is automatically added to the end of your Connection Name. For example, if you assigned the connection name `ABC_Systems` and the Mule's autogenerated name is `MyMulesoft-tzflx`, the connection `ABC_Systems (mule:MyMulesoft-tzflx)` is added to your list.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Database Port	Network port number used to connect to the database server.
CloudHub Env Name	CloudHub environment name where the connection is deployed. Typical values are Sandbox, Design, or Production.
Database Schema	Schema name for the database.
Database Username	User name to connect to the database.
CloudHub Username	User name to connect to the Mulesoft Anypoint account containing the CloudHub instance.
Database Name	Name of the Microsoft SQL Server database that you are connecting to.
Database Password	Password for the Microsoft SQL Server database.
CloudHub Password	Password to connect to the Mulesoft Anypoint account containing the CloudHub instance.
Database Host	URL schema for the database. Enter in the format: `<prefix>.<hostname>.<cluster>.rds.amazonaws.com` For example: `myrdssqlmule.ast7yapjzxxq.us-west-2.rds.amazonaws.com`

Analytics Mulesoft Microsoft SQL Server Connector Considerations

Keep these behaviors in mind when working with the Analytics Mulesoft Microsoft SQL Server connector.

- This connector can sync up to 1 million rows or 1 GB per object, whichever limit it reaches first.
- Your CloudHub administrator manages your CloudHub resources. If you are running multiple concurrent syncs or larger volumes of data, inform your CloudHub administrator so they can manage CloudHub allocations.
- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Analytics Mulesoft MySQL Connector

Create a remote connection using the Analytics Mulesoft MySQL connector to sync data from MySQL to Tableau CRM through Mulesoft.

Connection Requirements

A Mulesoft license, purchased or acquired separately, is required. The 30-Day Mulesoft Trial license is supported. Ask your Mulesoft admin for CloudHub credentials with access to deploy.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Mulesoft Connections**.

5. Click **Add Connection**.

6. Click the connector's icon and enter its settings, as described in the Connection Settings section.

7. When you finish entering the settings, click **Save**.

📝 Note: Saving the connector prompts Anypoint to generate a Mule for the connection. Mule generation can take up to two minutes. If you attempt to open the new connection during Mule generation, an error message says that objects cannot be retrieved. You can check Mule status in Anypoint.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections. To simplify AnyPoint management, the application name of the Mule instance initiated by this connector is automatically added to the end of your Connection Name. For example, if you assigned the connection name `ABC_Systems` and the Mule's autogenerated name is `MyMulesoft-tzflx`, the connection `ABC_Systems (mule:MyMulesoft-tzflx)` is added to your list.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Database Port	Network port number used to connect to the database server.

Setting	Description
Database Name	Name of the MySQL database that you are connecting to.
Database Password	Password for the MySQL database.
CloudHub Env Name	CloudHub environment name where the connection is deployed. Typical values are Sandbox, Design, or Production.
CloudHub Password	Password to connect to the Mulesoft Anypoint account containing the CloudHub instance.
Database Username	User name to connect to the database.
CloudHub Username	User name to connect to the Mulesoft Anypoint account containing the CloudHub instance.
Database Host	URL schema for the database. Enter in the format: `<prefix>.<hostname>.<cluster>.rds.amazonaws.com` For example: `mysqlmule.ast7yapjzxxq.us-west-2.rds.amazonaws.com`

Analytics Mulesoft MySQL Connector Considerations

Keep these behaviors in mind when working with the Analytics Mulesoft MySQL connector.

- This connector can sync up to 1 million rows or 1 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Analytics Mulesoft Oracle Connector

Create a remote connection using the Analytics Mulesoft Oracle connector to sync data from Oracle to Tableau CRM through Mulesoft.

Connection Requirements

A Mulesoft license, purchased or acquired separately, is required. The 30-Day Mulesoft Trial license is supported. Ask your Mulesoft admin for CloudHub credentials with access to deploy.

Create the Connection

1. In Tableau CRM, click the gear icon ( ) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Mulesoft Connections**.

5. Click **Add Connection**.

6. Click the connector's icon and enter its settings, as described in the Connection Settings section.

7. When you finish entering the settings, click **Save**.

📝 Note: Saving the connector prompts Anypoint to generate a Mule for the connection. Mule generation can take up to two minutes. If you attempt to open the new connection during Mule generation, an error message says that objects cannot be retrieved. You can check Mule status in Anypoint.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
	To simplify AnyPoint management, the application name of the Mule instance initiated by this connector is automatically added to the end of your Connection Name. For example, if you assigned the connection name `ABC_Systems` and the Mule's autogenerated name is `MyMulesoft-tzflx`, the connection `ABC_Systems (mule:MyMulesoft-tzflx)` is added to your list.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Database Port	Network port number used to connect to the database server.
CloudHub Env Name	CloudHub environment name where the connection is deployed. Typical values are Sandbox, Design, or Production.
Database Schema	Schema name for the database.
Database Username	User name to connect to the database.
CloudHub Username	User name to connect to the Mulesoft Anypoint account containing the CloudHub instance.
Database Name	Name of the Oracle database that you are connecting to.
Database Password	Password for the Oracle database.
CloudHub Password	Password to connect to the Mulesoft Anypoint account containing the CloudHub instance.
Database Host	URL schema for the database. Enter in the format: `<prefix>.<hostname>.<cluster>.rds.amazonaws.com` For example: `oracledbmule.ast7yapjzxxq.us-west-2.rds.amazonaws.com`

Analytics Mulesoft Oracle Connector Considerations

Keep these behaviors in mind when working with the Analytics Mulesoft Oracle connector.

- This connector can sync up to 1 million rows or 1 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Object Store and No SQL Connectors

Use these connectors to sync data from your object store and non-SQL services to Tableau CRM.

Amazon S3 Connection

Create a remote connection using the Amazon S3 connector to sync CSV data from an S3 bucket to Tableau CRM.

Amazon S3 Connection

Create a remote connection using the Amazon S3 connector to sync CSV data from an S3 bucket to Tableau CRM.

📝 Note: If you want to write from Tableau CRM to Amazon S3, use an Amazon S3 Output Connection (Beta) on page 737 instead.

Connection Details

Knowing your S3 bucket folder hierarchy is important to configuring this connector. Set the S3 bucket settings and parent's folder path when creating the connection. Then select one or more objects, which are the subfolders under the parent's folder path, that contain the CSV files you want to sync to Tableau CRM. When you sync the connected object, Tableau CRM looks for a CSV file called `schema_sample.csv` to detect the schema of the CSV data in the folder and to display a data preview. From this preview, you can view and change field attributes for all the files that you are loading from the folder.

Tableau CRM loads all CSV files in the subfolder that have the same fields as the `schema_sample.csv` and appends the rows in a single object.

The permissions for the Amazon S3 account used to create this connection must include ListBucket and GetObject. Permissions must include resource grants to the bucket and either "any object" or, if you chose to be specific, the appropriate ARNs that represent the path and objects being referenced. For example:

Create the Connection

1.
    In Tableau CRM, click the gear icon ( ) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

> **Note:** To avoid Amazon S3 connector setup timeout, choose a Folder Path with fewer than 5,000 entry options. An entry is the folder you intend to sync, the files within the folder, as well as all subfolders and subfiles. Recommended best practice is to create a bucket or folder within the bucket dedicated to Data Sync, populated with only the files to sync.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Master Symmetric Key	Optional setting. Your Amazon S3 master symmetric key used to enable client-side data encryption. This must be in a 256-bit AES encryption key in the Base64 format. Required when data is client-side encrypted with a master key stored within Salesforce.
Secret Key	Your Amazon secret access key.
Region Name	Optional setting. Region of your S3 service.
Folder Path	Path to the folder that you want to connect to. The path must start with the bucket name and can't include the name of the subfolder whose data you want to sync.
AWS Access Key ID	Your Amazon S3 bucket access key ID.

Amazon S3 Connector Considerations

Keep these behaviors in mind when working with the Amazon S3 connector.

- This connector can sync up to 100 million rows or 50 GB per object, whichever limit it reaches first.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

- This connector is not supported when your data is client-side encrypted with an AWS KMS managed key.
- The folder path that you specify in the connection settings must start with the bucket name and the parent folder. Don't include the folder that you want to connect to. Let's look at an example. You want to load files from the Call_Back_Logs_2018_Jan in S3. The folder path appears above the folder when you open it in S3.
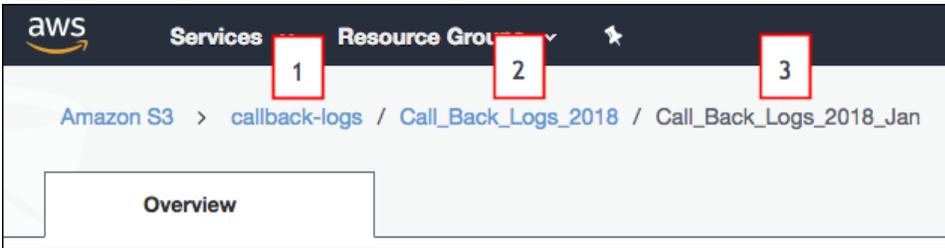
The folder path in S3 starts with the bucket name (1), followed by a folder name (2). In this example, subfolder (3) contains CSV data that you want to sync. When you specify this connection's folder path setting, you include only the bucket (1) and parent folder (2), not the subfolder (3) that you're syncing. After you create the connection, all subfolders directly under the parent folder (2), including Call_Back_Logs_2018_Jan, appear as possible objects that you can sync using this connection.
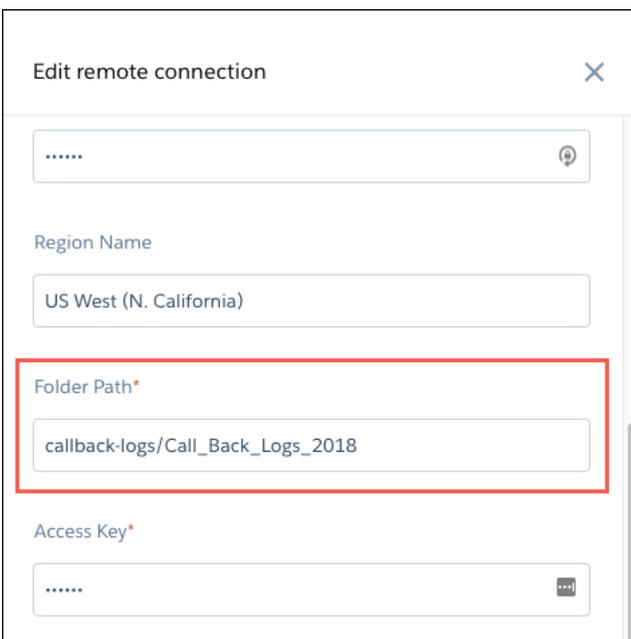
- Folder names in S3 must follow developer naming conventions, with no spaces or special characters.
- Rename one of the files in the subfolder to `schema_sample.csv`. Tableau CRM uses this file to detect the schema of the CSV data.

- Remove all empty rows from the `schema_sample.csv` file.

- Each file that you want to load from the S3 subfolder must have a file name ending in `.csv`. Each file must also have a header row and the same fields as the `schema_sample.csv` file. A file can have extra fields, but these fields aren't loaded. Tableau CRM ignores files that don't meet these requirements.

- Field names in the header row of each file must follow developer naming conventions, with no spaces or special characters. The header rows in your schema file and data files are case-sensitive, so use the same capitalization throughout.

- You can load only whole files, not parts of files.

Connection Example

👁 **Example:** To better understand how the S3 connector works under more complex scenarios, consider the following hierarchy in S3. The hierarchy has two buckets: Quarterly_Financial_Data and Call_Logs. The first bucket has one level of folders. The second bucket has two levels: parent folders and subfolders. Here's the S3 hierarchy.

- Quarterly_Financial_Data

  - 2018

    - 2018_quarter1_results.csv
    - 2018_quarter2_results.csv
    - 2018_quarter3_results.csv
    - 2018_quarter4_results.csv

  - 2019

    - 2019_quarter1_results.csv
    - 2019_quarter2_results.csv
    - 2019_quarter3_results.csv
    - 2019_quarter4_results.csv

- Call_Logs

  - 2018_Call_Logs

    - Q1_2018

      - call_logs_2018_01.csv
      - call_logs_2018_02.csv
      - call_logs_2018_03.csv

    - Q2_2018

      - call_logs_2018_04.csv
      - call_logs_2018_05.csv

- – call_logs_2018_06.csv

- Q3_2018

  - – call_logs_2018_07.csv

  - – call_logs_2018_08.csv

  - – call_logs_2018_09.csv

- Q4_2018

  - – call_logs_2018_10.csv

  - – call_logs_2018_11.csv

  - – call_logs_2018_12.csv

Let's look at how you can set up connections and connected objects to accomplish the following goals.

Goal	Action
Extract data from multiple buckets into Tableau CRM.	Because each connection is associated with a single bucket, create a separate connection for each bucket.
	For example, to extract data from the Quarterly_Financial_Data and Call_Logs buckets, create two connections, one for Quarterly_Financial_Data and another for Call_Logs. For each connection, set the appropriate bucket name in the Folder Path connection property.
Extract specific folders, but not all subfolders, from a parent folder.	Create a connection for the parent folder, and then use the connection to connect to specific subfolders.
	For example, to extract call logs from Q1_2018 and Q2_2018 folders, create one connection with folder path *Call_Logs/2018_Call_Logs*. Then create connected objects for Q1_2018 and Q2_2018.
Extract data from a subset of the CSV files under a folder.	Move the subset of files into a subfolder, create the connection on the parent folder, and then use the connection to connect to the subfolder.
	For example, to extract data from call_logs_2018_11.csv and call_logs_2018_12.csv, move these CSV files under a new subfolder under Q4_2018. Next, create a connection with folder path *Call_Logs/2018_Call_Logs/Q4_2018*. Finally, create a connected object based on the new subfolder.

SEE ALSO:

Amazon S3 Connector Known Issues and Knowledge Articles

Add and Remove Remote Objects and Fields from Data Sync

Pilot and Beta Connectors

You can use these connectors as a pilot or beta feature to sync external data with Tableau CRM. Before choosing a connector, review its pilot or beta instructions.

Google Cloud Storage Connection (Pilot)

Create a remote connection using the Google Cloud Storage connector to sync data from Google Cloud Storage to Tableau CRM.

Marketo Connection (Beta)

Create a remote connection using the Marketo connector to sync data from Marketo to Tableau CRM.

Analytics Mulesoft SAP SuccessFactors Connector (Pilot)

Create a remote connection using the Analytics Mulesoft SAP SuccessFactors connector to sync data from SAP SuccessFactors to Tableau CRM through Mulesoft.

Salesforce External Community Cloud Connection (Pilot)

Create a remote connection using the Salesforce External Community Cloud connector to sync Community Cloud data from another Salesforce org to Tableau CRM.

Salesforce External CPQ Connection (Pilot)

Create a remote connection using the Salesforce External CPQ connector to sync CPQ data from another Salesforce org to Tableau CRM.

Salesforce External Force.com Connection (Pilot)

Create a remote connection using the Salesforce External Force.com connector to sync Force.com data from another Salesforce org to Tableau CRM.

Salesforce External Sales Cloud Connection (Pilot)

Create a remote connection using the Salesforce External Sales Cloud connector to sync Sales Cloud data from another Salesforce org to Tableau CRM.

Salesforce External Service Cloud Connection (Pilot)

Create a remote connection using the Salesforce External Service Cloud connector to sync Service Cloud data from another Salesforce org to Tableau CRM.

SAP SuccessFactors Connection (Pilot)

Create a remote connection using the SAP SuccessFactors connector to sync data from SAP SuccessFactors to Tableau CRM.

SugarCRM Connection (Pilot)

Create a remote connection using the SugarCRM connector to sync data from SugarCRM to Tableau CRM.

Teradata Connection (Pilot)

Create a remote connection using the Teradata connector to sync data from Teradata to Tableau CRM.

Zendesk Connection (Pilot)

Create a remote connection using the Zendesk connector to sync data from Zendesk to Tableau CRM.

Zuora AQuA Connection (Pilot)

Create a remote connection using the Zuora AQuA connector to sync data from Zuora AQuA to Tableau CRM.

Google Cloud Storage Connection (Pilot)

Create a remote connection using the Google Cloud Storage connector to sync data from Google Cloud Storage to Tableau CRM.

Note: We provide the Google Cloud Storage Connector to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. The Google Cloud Storage Connector isn't generally available unless or until Salesforce

announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Google Cloud Storage Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Connection Details

Use a Google service account instead of a user account to grant access to your Google Cloud Storage data. For more information about Google service accounts, see Understanding Service Accounts.

Generate a JSON file containing the required client email and private key from the IAM & Admin page in the Google Cloud Platform Console.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Service Account Key	The `private_key` value from the JSON file. 📝 Note: Don't include `"private_key":` `"-----BEGIN PRIVATE KEY-----\n` at the start of the key and `\n-----END PRIVATE KEY-----\n"` at the end.
Service Account ID	The `client_email` value from the JSON file.
Project ID	The ID of the Google Cloud project you will sync data from the JSON file.

Google Cloud Storage Connector Considerations

When working with the Google Cloud Storage connector, keeps these behaviors in mind.

- This connector can sync up to 100,000 rows or 500 MB per object, whichever limit it reaches first.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Marketo Connection (Beta)

Create a remote connection using the Marketo connector to sync data from Marketo to Tableau CRM.

> **Note:** As a beta feature, Marketo Connector is a preview and isn't part of the "Services" under your master subscription agreement with Salesforce. Use this feature at your sole discretion, and make your purchase decisions only on the basis of generally available products and features. Salesforce doesn't guarantee general availability of this feature within any particular time frame or at all, and we can discontinue it at any time. This feature is for evaluation purposes only, not for production use. It's offered as is and isn't supported, and Salesforce has no liability for any harm or damage arising out of or in connection with it. All restrictions, Salesforce reservation of rights, obligations concerning the Services, and terms for related Non-Salesforce Applications and Content apply equally to your use of this feature. You can provide feedback and suggestions for the Marketo Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Connection Requirements

The Marketo connector accesses Marketo data using REST API. You need a Marketo custom service's Client ID, Client Secret, and your Marketo REST API endpoint for authentication. Ask your Marketo administrator or track down the credentials and endpoint in Marketo.

1. In Marketo, create a role with Access API permission and assign it to a user.

2. Create a custom service using the user's credentials.

3. Record the custom service's Client ID and Client Secret credentials. For instructions on generating the credentials, see Client ID and Client Secret credentials.

4. Record the REST API endpoint from the Web Services section. To locate your Marketo endpoint, see REST API endpoint.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Rest API URL	Marketo web services connection REST API endpoint string. Enter in the format: `https://<Database Name>.mktorest.com` Example: `https://123-ABC-456.mktorest.com`
Grant Type	Generates a token to access Marketo data. The value is prepopulated. The prepopulated `client_credentials` is the only valid option.
Client Secret	Generated by the Marketo custom service.
Client Id	Generated by the Marketo custom service.

Extraction and Filter Criteria

After creating the Marketo connection and choosing the object for extraction, set your filter criteria to limit the extraction and decrease import and processing time. Not all fields can be filtered.

Note: Filter properties for all objects and fields are shown regardless of the object selected for extraction. Filter properties have the name format `Field name (Object name)`, for example `Email Ids (Email)`.

Marketo Object	Filter Property Name	Required	Description
Campaign	Campaign Ids (Campaign)	No	Supports multiple values separated by commas.
	Campaign Names (Campaign)	No	Supports multiple values separated by commas.
Channel	Channel Names (Channel)	No	Supports multiple values separated by commas.
Custom, Opportunity, OpportunityRole, Company, and SalesPerson	Filter Type (Custom/Opportunity/ OpportunityRole/Company/SalesPerson)	No	Supports a single value from the object's searchableFields. Use Filter Type to specify which object field is the filter for the values entered in the Filter Values (Custom/Opportunity/OpportunityRole/Company/SalesPerson) property.
	Filter Values (Custom/Opportunity/ OpportunityRole/Company/SalesPerson)	No	Supports multiple values separated by commas. Use Filter Values alongside the Filter Type (Custom/Opportunity/ OpportunityRole/Company/SalesPerson) property to specify the values to filter.
Email	Email Ids (Email)	For Email object	Supports multiple values separated by commas.
Folder	Workspace (Folder)	No	A single value is supported, with a maximum folder depth of 2.
Lead	Filter Field (Lead)	For Lead object if not using Start Date and End Date properties	A single value is supported, like `email`. Use Filter Field to specify which lead field is the filter for the values entered in the Filter Values (Lead) property.
	Filter Values (Lead)	For Lead object if not using Start Date and End Date properties	Supports multiple values separated by commas. Filter Values must be used with the Filter Field (Lead) property to specify the values to filter. Example: Filter Field (Lead) is `email`, so Filter Values (Lead) is `abc@email.com, xyz@email.com`. Leads with the email address of `abc@email.com, xyz@email.com` are included in the extraction.
	Program Id (Lead)	No	A single value is supported.

Marketo Object	Filter Property Name	Required	Description
Lead and Lead Activity	Activity Type Ids (Lead/LeadActivity)	For Lead Activity object	Supports up to 10 values, separated by commas. Activity type IDs are unique to each Marketo instance. Example: ActivityType ID 12 represents new leads while ActivityType ID 13 is for a changed data value.
	End Date (Lead/LeadActivity)	For Lead object if not using Filter Field and Filter Values properties For Lead Activity object	Last date of the time period included in the extraction in YYYY-MM-DD format. Must be used with Start Date (Lead / LeadActivity).
	Start Date (Lead/LeadActivity)	For Lead object if not using Filter Field and Filter Values properties For Lead Activity object	Beginning of the time period included in the extraction in YYYY-MM-DD format. Use to capture details since a certain time, since the last extracted period, or for a specific time period when using the End Date (Lead / LeadActivity) property. To retrieve lead changes for a single day, specify the same date in Start Date and End Date.
Lead Activity	LeadActivity Ids (LeadActivity)	No	Supports multiple lead ID values separated by commas. Use LeadActivity Ids with Activity Type Ids (Lead / LeadActivity) property to retrieve only certain activities that belong to the designated leads.
List	List Ids (List)	No	Supports multiple values separated by commas.
	List Names (List)	No	Supports multiple values separated by commas.
	Program Names (List)	No	Multiple program name values, separated by commas, associated with lists.
Program	Program Ids (Program)	No	Supports multiple values separated by commas.
	Program Names (Program)	No	Supports multiple values separated by commas.
	Tag Types (Program)	No	Supports multiple values separated by commas. Use Tag Types to specify which tag types are the filters for the values entered in the Tag Values (Program) property.
	Tag Values (Program)	No	Supports multiple values separated by commas. Use Tag Values with the Tag Types (Program) property to specify the values to filter.
Tag	Tag Types (Tag)	No	Supports multiple values separated by commas.

Marketo Connector Considerations

When working with the Marketo connector, keeps these behaviors in mind.

- This connector can sync up to 100,000 rows or 500 MB per object, whichever limit it reaches first.

- You can extract up to 300 lead records at a time from Marketo when using the Filter Field and Filter Values properties.

- If your Marketo instance is synced with a CRM, Marketo disables connections to the Company, Opportunity, OpportunityRole, SalesPerson, and custom objects. If you try to extract from one of these objects when synced with a CRM, an error saying that the object API is disabled appears.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SEE ALSO:

    Marketo Connector Known Issues and Knowledge Articles

    Add and Remove Remote Objects and Fields from Data Sync

Analytics Mulesoft SAP SuccessFactors Connector (Pilot)

Create a remote connection using the Analytics Mulesoft SAP SuccessFactors connector to sync data from SAP SuccessFactors to Tableau CRM through Mulesoft.

📝 **Note:** We provide the Analytics Mulesoft SAP SuccessFactors Connector to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. The Analytics Mulesoft SAP SuccessFactors Connector isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Analytics Mulesoft SAP SuccessFactors Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Connection Requirements

A Mulesoft license, purchased or acquired separately, is required. The 30-Day Mulesoft Trial license is supported. Ask your Mulesoft admin for CloudHub credentials with access to deploy.

Create the Connection

1.
    In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Mulesoft Connections**.

5. Click **Add Connection**.

6. Click the connector's icon and enter its settings, as described in the Connection Settings section.

7. When you finish entering the settings, click **Save**.

📝 **Note:** Saving the connector prompts Anypoint to generate a Mule for the connection. Mule generation can take up to two minutes. If you attempt to open the new connection during Mule generation, an error message says that objects cannot be retrieved. You can check Mule status in Anypoint.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
	To simplify AnyPoint management, the application name of the Mule instance initiated by this connector is automatically added to the end of your Connection Name. For example, if you assigned the connection name *ABC_Systems* and the Mule's autogenerated name is *MyMulesoft-tzflx*, the connection *ABC_Systems (mule:MyMulesoft-tzflx)* is added to your list.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
CloudHub Env ID	CloudHub environment where the connection is deployed.
SuccessFactors Password	Password for the SAP SuccessFactors database.
SuccessFactors Company ID	ID value assigned to your SuccessFactors organization.
CloudHub Password	Password to connect to the Mulesoft Anypoint account containing the CloudHub instance.
CloudHub Username	User name to connect to the Mulesoft Anypoint account containing the CloudHub instance.
SuccessFactors Endpoint URL	API endpoint URL of your SuccessFactors instance. Enter in the format *https://<database name>.successfactors.com*
SuccessFactors Username	User name to connect to the database.

Analytics Mulesoft SAP SuccessFactors Connector Considerations

Keep these behaviors in mind when working with the Analytics Mulesoft SAP SuccessFactors connector.

- This connector can sync up to 1 million rows or 1 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Salesforce External Community Cloud Connection (Pilot)

Create a remote connection using the Salesforce External Community Cloud connector to sync Community Cloud data from another Salesforce org to Tableau CRM.

> 📝 Note: We provide the Salesforce External Community Cloud Connector to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. The Salesforce External Community Cloud Connector isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Salesforce External Community Cloud Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Create the Connection

1.
In Tableau CRM, click the gear icon (⚙️) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Username	User name used to connect to external Salesforce org for sync of objects and fields.
Service URL	Login URL for the external org. This is completed for you, but you can change it if the org is a sandbox org or uses a My Domain name or custom domain.
Password	Password for the user specified in Username. You may have to append the user's security token to the end of the password. For more information about obtaining a security token, see Reset Your Security Token.

Salesforce External Community Cloud Connector Considerations

Keep these behaviors in mind when working with the Salesforce External Community Cloud connector.

- This connector can sync up to 20 million rows or 10 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- The objects and fields available for sync in the external org are determined by the security settings of the user that you use to connect.
- Standard objects available in API versions 34.0 to 41.0 are eligible for data sync with this connector.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Salesforce External CPQ Connection (Pilot)

Create a remote connection using the Salesforce External CPQ connector to sync CPQ data from another Salesforce org to Tableau CRM.

📝 **Note:** We provide the Salesforce External CPQ Connector to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. The Salesforce External CPQ Connector isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Salesforce External CPQ Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.
2. In the data manager, click the **Connect** tab.
3. Click **Connect to Data**.
4. Click **Add Connection**.
5. Click the connector's icon and enter its settings, as described in the Connection Settings section.
6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.

Setting	Description
Username	User name used to connect to external Salesforce org for sync of objects and fields.
Service URL	Login URL for the external org. This is completed for you, but you can change it if the org is a sandbox org or uses a My Domain name or custom domain.
Password	Password for the user specified in Username. You may have to append the user's security token to the end of the password. For more information about obtaining a security token, see Reset Your Security Token.

Salesforce External CPQ Connector Considerations

Keep these behaviors in mind when working with the Salesforce External CPQ connector.

- This connector can sync up to 20 million rows or 10 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- The objects and fields available for sync in the external org are determined by the security settings of the user that you use to connect.
- Standard objects available in API versions 34.0 to 41.0 are eligible for data sync with this connector.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Salesforce External Force.com Connection (Pilot)

Create a remote connection using the Salesforce External Force.com connector to sync Force.com data from another Salesforce org to Tableau CRM.

🔖 Note: We provide the Salesforce External Force.com Connector to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. The Salesforce External Force.com Connector isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Salesforce External Force.com Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Create the Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Username	User name used to connect to external Salesforce org for sync of objects and fields.
Service URL	Login URL for the external org. This is completed for you, but you can change it if the org is a sandbox org or uses a My Domain name or custom domain.
Password	Password for the user specified in Username. You may have to append the user's security token to the end of the password. For more information about obtaining a security token, see Reset Your Security Token.

Salesforce External Force.com Connector Considerations

Keep these behaviors in mind when working with the Salesforce External Force.com connector.

- This connector can sync up to 20 million rows or 10 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- The objects and fields available for sync in the external org are determined by the security settings of the user that you use to connect.
- Standard objects available in API versions 34.0 to 41.0 are eligible for data sync with this connector.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Salesforce External Sales Cloud Connection (Pilot)

Create a remote connection using the Salesforce External Sales Cloud connector to sync Sales Cloud data from another Salesforce org to Tableau CRM.

> **Note:** We provide the Salesforce External Sales Cloud Connector to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs

are subject to change, and we can't guarantee acceptance. The Salesforce External Sales Cloud Connector isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Salesforce External Sales Cloud Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Create the Connection

1. In Tableau CRM, click the gear icon ( ) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Username	User name used to connect to external Salesforce org for sync of objects and fields.
Service URL	Login URL for the external org. This is completed for you, but you can change it if the org is a sandbox org or uses a My Domain name or custom domain.
Password	Password for the user specified in Username. You may have to append the user's security token to the end of the password. For more information about obtaining a security token, see Reset Your Security Token.

Salesforce External Sales Cloud Connector Considerations

Keep these behaviors in mind when working with the Salesforce External Sales Cloud connector.

- This connector can sync up to 20 million rows or 10 GB per object, whichever limit it reaches first.

- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.

- The objects and fields available for sync in the external org are determined by the security settings of the user that you use to connect.

- Standard objects available in API versions 34.0 to 41.0 are eligible for data sync with this connector.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Salesforce External Service Cloud Connection (Pilot)

Create a remote connection using the Salesforce External Service Cloud connector to sync Service Cloud data from another Salesforce org to Tableau CRM.

🗒 Note: We provide the Salesforce External Service Cloud Connector to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. The Salesforce External Service Cloud Connector isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Salesforce External Service Cloud Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Create the Connection

1.
In Tableau CRM, click the gear icon (⚙️) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Add Connection**.

5. Click the connector's icon and enter its settings, as described in the Connection Settings section.

6. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Username	User name used to connect to external Salesforce org for sync of objects and fields.

Setting	Description
Service URL	Login URL for the external org. This is completed for you, but you can change it if the org is a sandbox org or uses a My Domain name or custom domain.
Password	Password for the user specified in Username. You may have to append the user's security token to the end of the password. For more information about obtaining a security token, see Reset Your Security Token.

Salesforce External Service Cloud Connector Considerations

Keep these behaviors in mind when working with the Salesforce External Service Cloud connector.

- This connector can sync up to 20 million rows or 10 GB per object, whichever limit it reaches first.
- When using this connector, Salesforce Government Cloud org data is protected in transit with advanced encryption and can sync up to 10 million rows or 5 GB for each connected object, whichever limit is reached first.
- The objects and fields available for sync in the external org are determined by the security settings of the user that you use to connect.
- Standard objects available in API versions 34.0 to 41.0 are eligible for data sync with this connector.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SAP SuccessFactors Connection (Pilot)

Create a remote connection using the SAP SuccessFactors connector to sync data from SAP SuccessFactors to Tableau CRM.

📝 Note: We provide the SAP SuccessFactors Connector to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. The SAP SuccessFactors Connector isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the SAP SuccessFactors Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.

Setting	Description
Description	Description of the connection for internal use.
Password	Optional. Password for the SuccessFactors account used to sync.
Company ID	Optional. The value returned when you created the SuccessFactors account used to sync.
Service URL	SuccessFactors service root URL.
Username	Optional. Username for the SuccessFactors account used to sync.

SAP SuccessFactors Connector Considerations

When working with the SAP SuccessFactors connector, keeps these behaviors in mind.

- This connector can sync up to 100,000 rows or 500 MB per object, whichever limit it reaches first.
- Maximum number of characters in a field is 32,000.
- Maximum number of concurrent data sync runs is 3.
- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.
- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

SugarCRM Connection (Pilot)

Create a remote connection using the SugarCRM connector to sync data from SugarCRM to Tableau CRM.

> Note: We provide the SugarCRM Connector to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. The SugarCRM Connector isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the SugarCRM Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Username	User name to connect to the SugarCRM account.

Setting	Description
URL	Complete URL of the SugarCRM account. Enter in the format:*<http or https>://<instancename>.sugarondemand.com* For example: `https://mysugarinstance.sugarondemand.com`
Password	Password to connect to the SugarCRM account.

SugarCRM Connector Considerations

When working with the SugarCRM connector, keeps these behaviors in mind.

- This connector can sync up to 100,000 rows or 500 MB per object, whichever limit it reaches first.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Teradata Connection (Pilot)

Create a remote connection using the Teradata connector to sync data from Teradata to Tableau CRM.

> 📝 **Note:** We provide the Teradata Connector to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. The Teradata Connector isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Teradata Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.

Setting	Description
JDBC Connection URL	The URL used to connect to the database. Enter in the format: `https://<Database Name>.teradata.com` For example: `https://blueskyserver.teradata.com`
Username	The database user, found in the database credentials.
Password	The database password, found in the database credentials.

Teradata Connector Considerations

When working with the Teradata connector, keeps these behaviors in mind.

- This connector can sync up to 100,000 rows or 500 MB per object, whichever limit it reaches first.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Zendesk Connection (Pilot)

Create a remote connection using the Zendesk connector to sync data from Zendesk to Tableau CRM.

📝 Note: We provide the Zendesk Connector to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. The Zendesk Connector isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Zendesk Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Username	User name for the Zendesk account.

Setting	Description
URL	Complete URL of the Zendesk account. Enter in the format: `<http or https>://<Instance Name>.zendesk.com/<Instance Details>` For example: `https://blueskysystems.zendesk.com/api/v3`
Password	Password for the Zendesk account.

Zendesk Connector Considerations

When working with the Zendesk connector, keeps these behaviors in mind.

- This connector can sync up to 100,000 rows or 500 MB per object, whichever limit it reaches first.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Zuora AQuA Connection (Pilot)

Create a remote connection using the Zuora AQuA connector to sync data from Zuora AQuA to Tableau CRM.

> 📝 Note: We provide the Zuora AQuA Connector to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. The Zuora AQuA Connector isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Zuora AQuA Connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Username	User name for the Zuora AQuA account.

Setting	Description
Endpoint URL	The URL of the Zuora AQuA server. Enter in the format: `https://<instancename>.zuora.com` For example: `https://services123.zuora.com`
Password	Password for the Zuora AQuA account.

Zuora AQuA Connector Considerations

When working with the Zuora AQuA connector, keeps these behaviors in mind.

- This connector can sync up to 100,000 rows or 500 MB per object, whichever limit it reaches first.

- Maximum number of characters in a field is 32,000.

- Maximum number of concurrent data sync runs is 3.

- You can sync data from up to 100 objects in Tableau CRM. This total includes local and external objects.

- Connected object names must start with a letter and contain only letters, digits, or underscores. Object names cannot have consecutive underscores or end with an underscore.

Add and Remove Remote Objects and Fields from Data Sync

Update data sync for connected external data sources to keep up with your changing business by adding objects, removing objects, and managing which fields are synced. Synced data is stored in Tableau CRM as objects that can be used in dataflows and recipes.

> **Note:** For help with managing data sync for local Salesforce objects, see Add and Remove Local Salesforce Objects and Fields from Data Sync on page 616.

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. Click the **Connect** tab. All objects configured to sync are listed under their connection name.

3. To add an object and its fields:

   a. Select **Connect to Data**.

   b. Select the source for your object.

   c. Select the object, and click **Continue**. You can't select objects that are already connected.

   d. Select the fields that you want to sync, and click **Continue**.

   e. If you're using a connector that supports filtering, you can filter the data that is synced by clicking ▼ . Check the connector's help for required filter syntax.

   f. To view or edit a field's attributes, click the field's column header, and click ✎ . You can edit a field's label, type, and settings.

   g. Save your changes.

4. To add or remove a field from data sync for an object already configured for sync:

   a. Select the object's name.

    b. Check the box next to the fields to add, remove the box for fields to remove.

    c. Select **Continue**, then save your changes.

> 📝 Note:
> - Only field label changes appear in the preview data.
> - Field attributes for local Salesforce objects are read only.
> - The default value and precision must be consistent between combined data sources. Review other data sources' configuration when changing the value and precision.
> - For existing non-S3 connected objects, you can change measure and date field types to text. If you change a field to text and want to change it back, return to the field selection screen. Deselect the field, and select the field again. The field's original type is restored when you continue to the preview screen.
> - For new S3 connected objects, you can change any field type to any other field type. For existing connected S3 objects, you can change field types if you remove the object and fields then recreate them, or transform them after extraction.

> ⛔ Important: Connected objects are synced the next time sync runs for the connection. To make the data immediately available in a dataflow or recipe, sync manually.

SEE ALSO:

    Verify the Incremental Sync Settings for Salesforce Data

    Schedule, Run, and Monitor Data Sync

Stage Your Data for Recipes and Dataflows with Data Sync

Use data sync to decouple the extract of data from your recipes and dataflows, and sync this data to Tableau CRM on a separate schedule. By scheduling sync from Salesforce and remote systems ahead of time, your recipe and dataflow have less to do and run faster. To lighten the load even more, Tableau CRM can sync supported local Salesforce data incrementally by default, meaning that only data that's changed gets synced.

Use Data Sync with Recipes

Recipes only use data that synced to Tableau CRM. To expand the data available when you build and run a recipe, add objects and fields to data sync on page 685.

Use Data Sync with Dataflows

Data sync makes your dataflow runs faster. Without data sync, a dataflow performs a separate extract each time it needs data from a Salesforce object. Let's look at an example. Imagine your organization has three dataflows, extracting data from Salesforce objects as follows:

	Accounts	**Contacts**	**Opportunities**	**Campaigns**	**Leads**	**Cases**	**Users**
Default Dataflow	🏢	📇	👑			💼	👤
Sales Analytics App Dataflow	🏢	📇	👑	🎯	✴️	💼	👤

	Accounts	Contacts	Opportunities	Campaigns	Leads	Cases	Users
Service Analytics App Dataflow							

Every time these dataflows run, they must extract all this Salesforce data. And the more data there is, the longer the dataflow takes to run. In addition, the dataflows perform separate, duplicate, extracts from the same object. For example, all three dataflows extract Accounts data.

With data sync, all of these extracts are performed as a separate process, which you can schedule to take place before your dataflows run.

	Accounts	Contacts	Opportunities	Campaigns	Leads	Cases	Users
Data Sync							

This synced data is then available to all your dataflows, which run faster because they no longer have to extract any data—just load and transform.

Add, Remove, and Manage the Objects and Fields That Sync to Tableau CRM

As your analytics and business needs change, the data synced to Tableau CRM change, too. Add, remove, and change the settings for the objects and fields included in data sync.

Schedule, Run, and Monitor Data Sync

You can schedule sync to run automatically, manually run a sync, and monitor a sync's progress, all in the data manager.

Verify the Incremental Sync Settings for Salesforce Data

Before you run or schedule data sync, specify whether the sync extracts incremental changes or all records from each Salesforce object. By default, Tableau CRM performs an incremental sync. An incremental sync runs faster because it extracts only the latest changes to the Salesforce object.

Enable Data Sync and Connections

Data Sync is enabled by default if you turned on Tableau CRM after the Winter '20 release. If you turned on Tableau CRM before the Winter '20 release, manually enable Data Sync and Connections to optimize your dataflows and connect to external data.

Sync Out for Snowflake

Sync Out for Snowflake exports your raw local Salesforce data via Tableau CRM to Snowflake using the Tableau CRM output connector for Snowflake. With Sync Out for Snowflake, keep your Salesforce data in Snowflake up to date using scheduled Data Sync without the need for a third-party ETL tool. Fresh Salesforce data is vital if you maintain a central Snowflake data lake for processing, analysis, business automation, or storage. For example, give your shipping logistics team the freshest data by merging your account data from your system of record, Salesforce, with your ERP's shipping data in your data lake.

Data Sync Limits and Considerations

Here are some things to consider when you're working with data sync.

Add, Remove, and Manage the Objects and Fields That Sync to Tableau CRM

As your analytics and business needs change, the data synced to Tableau CRM change, too. Add, remove, and change the settings for the objects and fields included in data sync.

For help with managing data sync for local Salesforce objects, see Add and Remove Local Salesforce Objects and Fields from Data Sync on page 616.

For help with managing data sync for remote objects, see Add and Remove Remote Objects and Fields from Data Sync on page 683.

SEE ALSO:

> Configure the Dataflow Through the Definition File
>
> sfdcDigest Transformation

Schedule, Run, and Monitor Data Sync

You can schedule sync to run automatically, manually run a sync, and monitor a sync's progress, all in the data manager.

🛑 **Important:** Data Sync is enabled by default if you turned on Tableau CRM after the Winter '20 release. If you manually enable Data Sync, run sync for your Salesforce objects, and make sure that it completes before your dataflow's next run. Dataflows with sfdcDigest nodes fail until sync has run for the first time.

Run Data Sync Manually

Run data sync manually the first time to make the data available to build recipes and dataflows. Schedule subsequent syncs to regularly update the data used by recipes and dataflows.

Schedule Data Sync to Run Automatically

Schedule data syncs to run regularly. You schedule data sync for each connection, where all objects under the connection sync at the specified time, and not individual objects. To sync objects from the same data source on different intervals, create multiple connections to the data source, and set a unique schedule for each connection. To ensure that the latest source data is loaded into datasets, schedule data syncs to run before the corresponding recipes and dataflows.

Monitor a Data Sync Job

Monitor the progress of syncs in Data Manager.

Set Data Sync Notifications

Set Data Sync notifications to receive an email notification when a sync job has warnings, when sync fails, or every time the sync finishes.

Troubleshoot Canceled Data Sync, Recipe, and Dataflow Runs Due to Overlapping Schedules

Did you notice stale data in a dashboard due to a canceled recipe, data sync, or dataflow run? Or did you check the Monitor tab of the Data Manager and see occasional canceled scheduled runs? Canceled runs can result from too many of the same runs in the job queue due to tightly scheduled recipe, dataflow, or data syncs jobs. Usually you can avoid this situation if you extend the duration between a recipe's, data sync's, or dataflow's scheduled runs.

Run Data Sync Manually

Run data sync manually the first time to make the data available to build recipes and dataflows. Schedule subsequent syncs to regularly update the data used by recipes and dataflows.

> 📝 **Note:** Data sync jobs don't count towards your daily dataflow and recipe run limit.

1. In Tableau CRM, click **Data Manager** in the left pane to open Data Manager.

2. To display the list of objects enabled for sync, group by connection, click the **Connect** tab.

3. To show and hide the list of objects, click the arrow to the left of the connection name.

   If you can't see the Connect tab, you must enable data sync in your org. See Enable Data Sync and Connections.

4. To run sync for all objects in a connection, click ▾ to the right of the connection name, and select **Run Now**. The connection is queued to sync.

5. To run sync for a single remote object, click ▾ to the right of the object name, and select **Run Data Sync**.

6. To run sync for a single local Salesforce object, first select the connection mode you want to use. Click ▾ to the right of the local object name, and select **Edit Connection Mode**.

7. Choose from the following:

   a. **Incremental Sync** updates only rows that changed since the last sync. It's the fastest option.

   b. **Periodic Full Sync** updates rows incrementally and periodically overwrites all rows with records in the Salesforce object.

   c. **Full Sync** updates all rows with records in the Salesforce object.

8. Click **Save**.

9. Click ▾ again, and select **Run Data Sync**. The Monitor tab of Data Manager opens so you can see the status of your sync.

SEE ALSO:

Monitor a Data Sync Job

Tableau CRM Limits

Schedule Data Sync to Run Automatically

Schedule data syncs to run regularly. You schedule data sync for each connection, where all objects under the connection sync at the specified time, and not individual objects. To sync objects from the same data source on different intervals, create multiple connections to the data source, and set a unique schedule for each connection. To ensure that the latest source data is loaded into datasets, schedule data syncs to run before the corresponding recipes and dataflows.

> ✏️ **Note:** Data sync jobs don't count towards your daily dataflow and recipe run limit.

1. In Tableau CRM, click **Data Manager** in the left pane to open Data Manager.

2. Click the **Connect** tab. The Connect tab displays a list of objects enabled for sync, grouped by connection.

   If you can't see the Connect tab, you must enable data sync in your org. See Enable Data Sync and Connections.

3. Click ⏷ to the right of the connection that you want to schedule, and select **Schedule**. The scheduler appears.

4. Select the time to run the recipe. You can schedule it to run by minute, hour, week, or month. Tableau CRM runs the data sync according to the time zone of the user who set the schedule.

   > 💡 **Tip:** If you don't have a Tableau CRM Plus license but want to schedule a run every 15, 20, or 30 minutes, contact Salesforce Customer Support to request subhour scheduling. This feature isn't available in sandbox orgs.

5. If you schedule the recipe to run by minute or hour, select **Stop queuing at a specific time** to stop the recipe from running after a certain time. For example, to restrict runs to office hours, set a job to start at 8:00 am, run every hour, and stop at 6:00 pm.

6. Click **Save**. Schedule information for the connection displays in the connection header.

> 💡 **Tip:** To ensure that sync completes before the recipe or dataflow starts, use event-based scheduling for the recipe or dataflow.

> ✏️ **Note:** Sandbox and developer edition org schedules for data sync, dataflow, and recipes are removed 30 days after the last save. Users subscribed to its notifications receive an email notifying them when a schedule is removed. Set the schedule again anytime.

SEE ALSO:

Monitor a Data Sync Job

Tableau CRM Limits

Monitor a Data Sync Job

Monitor the progress of syncs in Data Manager.

To monitor syncs, perform the following steps.

1. In Tableau CRM, click **Data Manager** in the left pane to open Data Manager.
   Data Manager opens to the Monitor tab with the Jobs view selected.

2. In the data manager, click the **Monitor** tab.
   The Jobs subtab displays information about your sync jobs.

SEE ALSO:

Troubleshoot Canceled Data Sync, Recipe, and Dataflow Runs Due to Overlapping Schedules

Set Data Sync Notifications

Set Data Sync notifications to receive an email notification when a sync job has warnings, when sync fails, or every time the sync finishes.

1. In Tableau CRM, click the gear icon (⚙) and then click **Data Manager**.
   The data manager opens on the Monitor tab, with the Jobs tab open by default.

2. Click the **Connect** tab.

3. On the right of the connection, click ▼ and select **Notifications**.
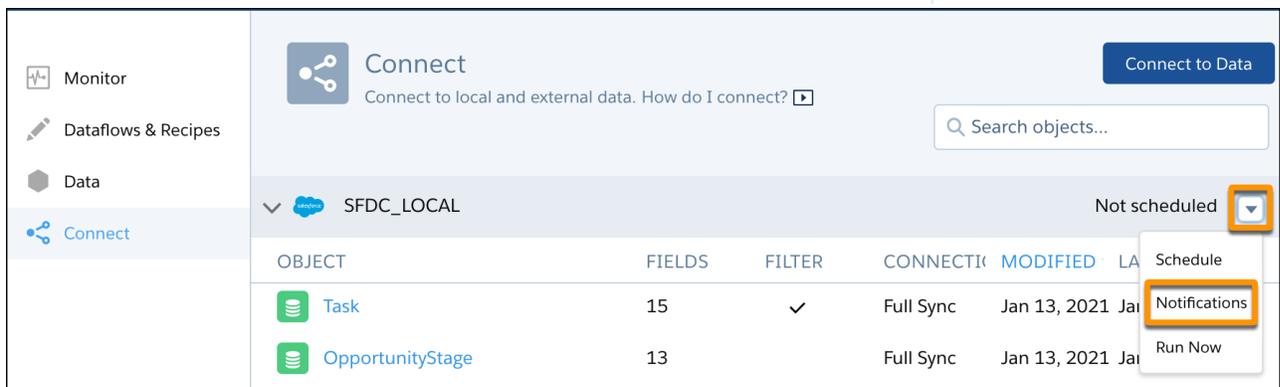
4. From the Notify Me About picklist, select what you want to be notified about.

> **Note:** By default, email notifications are sent only to the person setting the notifications. Other people with the Edit Analytics Dataflows permission can subscribe to a data sync by setting notifications.

Troubleshoot Canceled Data Sync, Recipe, and Dataflow Runs Due to Overlapping Schedules

Did you notice stale data in a dashboard due to a canceled recipe, data sync, or dataflow run? Or did you check the Monitor tab of the Data Manager and see occasional canceled scheduled runs? Canceled runs can result from too many of the same runs in the job queue due to tightly scheduled recipe, dataflow, or data syncs jobs. Usually you can avoid this situation if you extend the duration between a recipe's, data sync's, or dataflow's scheduled runs.

What You're Seeing

Canceled runs can result in datasets that do not update in the expected timeframe, which can result in stale data when viewed or explored. Delayed job runs push back all other queued jobs of the same type. This results in a cascade of delays that eventually leads to cancellation messages as the scheduling system cancels the queued jobs.

Best Practices to Avoid Canceled Jobs Due to Overlapping Schedules

- Schedule your recipe, dataflow, and data sync jobs with plenty of time between the runs to allow for potential delays.
- Periodically review your job runs to see how long an average job takes, and update the schedule to allow for potential delays.
- Split large data syncs into multiple smaller data syncs using additional remote or local connections to the same data source. You can set more frequent sync schedules for smaller groups of objects that require more frequent updates, and infrequent sync schedules for less-updated objects.
- Split large dataflows into multiple smaller dataflows.
- Enable priority scheduling to automatically queue shorter/smaller runs before longer/larger runs.

Why Your Overlapping Jobs Are Canceled

Tableau CRM data sync, recipes, and dataflows use asynchronous processing, also known as Bulk API. Asynchronous processing queues up requests and runs them in first-in, first-out order as server resources become available (unless you enable priority scheduling). Synchronous requests have priority over asynchronous processes, so these jobs sometimes pause as server resources reprioritize them. This means that recipe, dataflow, and data sync jobs have no expectations for queue or processing time and no SLA for completion.

The Analytics scheduler allows any job to be simultaneously running and queued once. Another attempt to run that job while an earlier request is still in queue results in the scheduling system canceling the new job request.

Based on the licenses provisioned, recipes and dataflows have a concurrency limit as described in Tableau CRM Limits under ELT Job Limits. If two recipes or dataflows perform a write (sfdcRegister) operation on the same dataset, they do not run concurrently—one runs

and the other gets queued. Sandbox environments are limited to a dataflow concurrency of 1. No more than 3 data sync jobs can run at the same time.

SEE ALSO:

Set Dataflow Notifications

Troubleshoot Canceled Data Sync, Recipe, and Dataflow Runs Due to Overlapping Schedules

Verify the Incremental Sync Settings for Salesforce Data

Before you run or schedule data sync, specify whether the sync extracts incremental changes or all records from each Salesforce object. By default, Tableau CRM performs an incremental sync. An incremental sync runs faster because it extracts only the latest changes to the Salesforce object.

📝 **Note:** During the first sync of an object, Tableau CRM always performs a full sync. Switching the site or migrating the org also triggers the object to undergo a full sync.

1. In Analytics Cloud, click the gear icon (⚙), and then click **Data Manager** to open the data manager.

2. In the data manager, click the **Connect** tab.

   The Connect tab displays a list of objects enabled for sync, grouped by connection.

3. Under a local connection like SFDC_LOCAL, click [▼] to the right of the Salesforce object that you want to sync, and then select **Edit Connection Mode**.
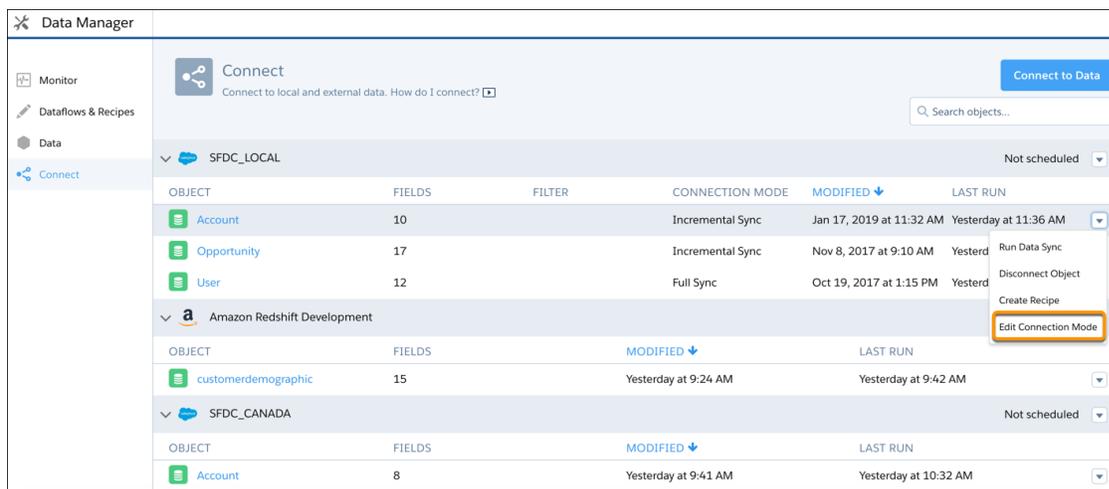
📝 **Note:** Connection mode doesn't apply to objects with other connection types.

4. Select one of the following modes.

Connection Mode	Description
Incremental Sync	Pulls only new, updated, and deleted records to match the changes in the Salesforce object since the previous sync. Use this method to run faster syncs.

Connection Mode	Description
Periodic Full Sync	Runs an incremental sync on each scheduled sync. Also runs a full sync on the first sync that happens after Friday 11 PM in your org's time zone.
Full Sync	Pulls all records from the Salesforce object, and overwrites the records from the previous sync.

5. Click **Save**.

Considerations When Syncing Data Incrementally
Consider the following limitations when using incremental data syncs.

- Incremental sync isn't supported for Salesforce big objects or the following objects.
  - CallCenter
  - CaseTeamMember
  - CategoryNode
  - CollaborationGroup
  - CollaborationGroupMember
  - CollaborationGroupMemberRequest
  - Division
  - Domain
  - DomainSite
  - Group
  - GroupMember
  - ModelFactor
  - Profile
  - Site
  - Territory
  - Topic
  - User
  - UserRole
  - UserTerritory

- Even if incremental sync is enabled, Tableau CRM performs a full sync at runtime for the following objects.
  - The object performs hard deletes or cascade hard deletes on records.
  - The object is known to fail when queried without a specific filter or where condition.
  - The object doesn't have a datetime system field.
  - The object has fields or field attributes (such as scale and precision) specified for the sync that don't match the previously synced dataset. If you make field changes in an sfdcDigest dataflow node, such as adding or removing fields, or changing attributes, Tableau CRM triggers a full sync the next time sync runs.
  - The object is configured for a periodic full sync.

- Verify that your org doesn't have conflicting sync settings for the same object. For example, if incremental sync is explicitly turned off in an sfdcDigest node, and not explicitly turned on in later nodes for the same object in the same dataflow, a full sync is performed. To enable incremental sync, set the `incremental` parameter to `true` in the dataflow JSON.

- If multiple dataflows have conflicting sync settings for the same object, Tableau CRM uses the object's sync settings from the last saved dataflow.

- Run full sync for objects containing formula fields. With incremental sync, formula fields can become out of sync with your synced object.

- Run a full sync manually if incremental sync fails for objects with many updates. If incremental syncs continue to fail, set the object's connection mode to **Full Sync**.

- Consider periodically running a full sync to ensure that all changes are picked up. To configure a periodic full sync on an object, set the connection mode to **Periodic Full Sync**.

Enable Data Sync and Connections

Data Sync is enabled by default if you turned on Tableau CRM after the Winter '20 release. If you turned on Tableau CRM before the Winter '20 release, manually enable Data Sync and Connections to optimize your dataflows and connect to external data.

> ⛔ **Important:** Before you enable Data Sync, we recommend that you read Understand What Happens When You Enable Data Sync and Connections.

1. From Setup, enter *Analytics* in the Quick Find box, then select **Settings**.

2. Select **Enable Data Sync and Connections**.

3. Save your changes.

> ⛔ **Important:** After you enable Data Sync and Connections, make sure that you run data sync before your dataflows next run. Dataflows with sfdcDigest nodes fail until data sync has run and completed for the first time. See Schedule, Run, and Monitor Data Sync.

Understand What Happens When You Enable Data Sync and Connections

Data Sync is enabled by default if you turned on Tableau CRM after the Winter '20 release. If you turned on Tableau CRM before the Winter '20 release, you can manually enable Data Sync. When you enable the Data Sync and Tableau CRM Connections setting, Tableau CRM looks at the sfdcDigest nodes in your existing scheduled dataflows to see which Salesforce objects and fields you're currently extracting. Using this information, Tableau CRM enables sync for each object to extract the data separately. Understanding exactly what happens when you enable Data Sync helps you identify other actions to consider to ensure that your sync runs successfully.

Understand What Happens When You Enable Data Sync and Connections

Data Sync is enabled by default if you turned on Tableau CRM after the Winter '20 release. If you turned on Tableau CRM before the Winter '20 release, you can manually enable Data Sync. When you enable the Data Sync and Tableau CRM Connections setting, Tableau CRM looks at the sfdcDigest nodes in your existing scheduled dataflows to see which Salesforce objects and fields you're currently extracting. Using this information, Tableau CRM enables sync for each object to extract the data separately. Understanding exactly what happens when you enable Data Sync helps you identify other actions to consider to ensure that your sync runs successfully.

Here's what happens, and the action we recommend that you take.

Your Existing Dataflows Are Validated

Because data sync is based on your existing dataflows, Tableau CRM validates your scheduled dataflows to ensure that they are error free. Although dataflows with errors are not considered for data sync, Tableau CRM alerts you to any errors in sfdcDigest nodes when you enable data sync.

Recommended Action

Before you enable Data Sync, review your scheduled dataflows for errors. In particular, look for incorrect object or field names in sfdcDigest nodes. If you see errors displayed when you enable Data Sync, fix them in your dataflows before you continue. When you fix errors in a dataflow, Tableau CRM considers it for a sync when you save the dataflow or upload the definition file.

⚠ **Important:** Tableau CRM doesn't consider unscheduled dataflows when you enable Data Sync. Make sure that any dataflow that you want to be considered is scheduled. See Schedule a Dataflow to Run Automatically.

All Fields in Multiple sfdcDigest Nodes for an Object Are Included in Data Sync

If your dataflows have multiple sfdcDigest nodes for the same object, each with a different set of fields, a superset of all these fields is included in data sync. In addition, if the same field has different attributes, such as `type` or `isMultivalue`, the attributes from the most recently created dataflow are used for data sync. Consider the following example dataflows, each containing an sfdcDigest node for the Opportunity object:

Dataflow A (Created 08/01/2016)	Dataflow B (Created 08/15/2016)

<pre>{
    "Extract_OpportunitiesA": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Opportunity",
            "fields": [
                { "name": "Id"},
                { "name": "Name" },
                { "name": "StageName" },
                { "name": "CloseDate",
"fiscalMonthOffset": 9 },
                { "name": "AccountId" },
                { "name": "OwnerId" }
            ]
        }
    }</pre>
	<pre>{
	    "Extract_OpportunitiesA": {
	        "action": "sfdcDigest",
	        "parameters": {
	            "object": "Opportunity",
	            "fields": [
	                { "name": "Id"},
	                { "name": "Name" },
	                { "name": "StageName" },
	                { "name": "CloseDate",
	"fiscalMonthOffset": 3 }
	            ]
	        }
	    }</pre>

In the resulting data sync settings for this object, the Id, Name, StageName, CloseDate, AccountId, and OwnerId are included. The `fiscalMonthOffset` attribute for CloseDate is set to `3` from the most recently created dataflow.

Recommended Action

Review the fields in the sfdcDigest nodes of your dataflows. If the resulting data sync settings include fields that you don't require, you can disable them in the settings for that object. If you have field attribute conflicts, ensure that the attributes you want to use for data sync are in the most recently created dataflow.

Advanced Filters on sfdcDigest Nodes Are Included in Data Sync

If an sfdcDigest node includes a `complexFilterConditions` parameter to apply an advanced filter, Tableau CRM adds this filter to the data sync settings for that object. When there are multiple sfdcDigest nodes with complex filters for the same object, only the filter from the most recently created or updated dataflow is added to the data sync settings. Tableau CRM ignores structured filters applied through `filterConditions` parameters.

Recommended Action

After you enable Data Sync, review each object's data sync settings to verify that filters have been applied correctly. To use multiple filters for an object, we recommend that you remove any filter from data sync settings, and add the filters to the corresponding dataflow using the filter transformation (see filter Transformation).

SEE ALSO:

Configure the Dataflow Through the Definition File

sfdcDigest Transformation

Filtering Records Extracted from a Salesforce Object

Sync Out for Snowflake

Sync Out for Snowflake exports your raw local Salesforce data via Tableau CRM to Snowflake using the Tableau CRM output connector for Snowflake. With Sync Out for Snowflake, keep your Salesforce data in Snowflake up to date using scheduled Data Sync without the need for a third-party ETL tool. Fresh Salesforce data is vital if you maintain a central Snowflake data lake for processing, analysis, business automation, or storage. For example, give your shipping logistics team the freshest data by merging your account data from your system of record, Salesforce, with your ERP's shipping data in your data lake.

> 📝 **Note:** Use the Snowflake Output Connection on page 743 to push transformed data from Tableau CRM to Snowflake using a Data Prep output node. If you want to sync data from Snowflake to Tableau CRM, use the Snowflake Input Connection on page 652 instead.

> 🔆 **Warning:** The full Salesforce object data synced to Tableau CRM is pushed to Snowflake with Sync Out. Update your Snowflake access controls to manage data access.

Enable Sync Out

1. From Setup, enter *Analytics* in the Quick Find box.

2. Select **Settings** under **Tableau CRM**.

3. Select **Enable Snowflake output Connection** and **Save**.

4. Turn on and configure the Snowflake output connector as described in Snowflake Output Connection on page 743. The output connection is used to link Tableau CRM to Snowflake. You don't add or update a Data Prep recipe output node to use Sync Out, as the push happens during Data Sync.

    > 📝 **Note:** The Snowflake account used for the output connection needs privileges to create and own tables; insert, update, truncate, and merge data; and create temporary stages.

Configure Sync Out for a Salesforce Local Connection

1. On the Connect tab of the Data Manager, select the down arrow next to the Salesforce Local Connection you want to use with Sync Out.

    > 💡 **Tip:** Do you want to sync out only some of this connection's objects? Add another local connection using these instructions on page 617, then reassign the objects for sync out objects to the local connection.

2. Click **Sync Out**. This option is only available after you contact Salesforce to enable Sync Out.

3. Click the slider to enable Sync Out.

4. Choose the target connection.

5. Select **Save**.

With Sync Out enabled, data writes to Snowflake with each manual or scheduled run of Data Sync. You don't set up or modify a Data Prep recipe.

Turn Off Sync Out for a Salesforce Local Connection

1. On the Connect tab of the Data Manager, select the down arrow next to the Salesforce Local Connection you want to turn off Sync Out.

2. Select to disable Sync Out.

3. Select **Save**.

Sync Out for Snowflake Considerations

Keep these behaviors in mind when working with Sync Out for Snowflake.

* Sync Out is only available for your local Salesforce objects. To sync data from other objects, use the Snowflake Output Connection on page 743 and a Data Prep recipe.

* Up to 10 million rows of data are written per run of Data Sync with a maximum of 50 million rows written per 24-hour period.

* Sync Out for Snowflake extends Data Sync runs. Evaluate your Data Sync run times, then update scheduled recipes and dataflows to begin after Data Sync completes.

* When the prior run's data is deleted in preparation for the current run, the earlier version of the data becomes inaccessible. Set up a process to copy or use the output after each run.

* Don't modify or delete the Snowflake schema. Sync Out creates and manages the tables in Snowflake. Tables are recreated with each full sync.

* Use periodic full sync to avoid data drift.

* Object data pushed in Sync Out uses Salesforce's field attributes, like precision and scale, instead of those defined in Data Sync.

* To investigate Sync Out run status, expand the Data Sync nodes on the Monitor page.

* To receive updates about Sync Out, change your Data Sync Notifications to All instead of Warning.

* A Sync Out failure doesn't always cause the Data Sync run to fail. View the Sync Out status and message on the Monitor page by expanding the Data Sync job.

* To remove a field from Sync Out, remove it from Data Sync.

Data Sync Limits and Considerations

Here are some things to consider when you're working with data sync.

* You can include up to 100 objects for sync. If you are currently extracting more than 100 objects in your dataflows, contact Salesforce Customer Support before you enable data sync.

* Data Sync only performs one sync per object. Keep this limit in mind if you must perform multiple syncs from the same object. For example, when you sync filtered and unfiltered data from an object.

Considerations When Creating or Updating Dataflows with Data Sync

Keep these behaviors in mind when creating or updating a dataflow with Data Sync enabled. Data Sync is enabled by default if you turned on Tableau CRM after the Winter '20 release.

Considerations When Using Tableau CRM Templated Apps with Data Sync

If you use template-based Tableau CRM apps, such as Sales Analytics or Service Analytics, here are some things to consider before and after you enable Data Sync. Data Sync is enabled by default if you turned on Tableau CRM after the Winter '20 release.

Considerations When Creating or Updating Dataflows with Data Sync

Keep these behaviors in mind when creating or updating a dataflow with Data Sync enabled. Data Sync is enabled by default if you turned on Tableau CRM after the Winter '20 release.

- When you upload a dataflow definition file or update a dataflow in the dataflow editor, Tableau CRM validates the definition file or dataflow. If you see errors displayed, correct them, and upload the file or update the dataflow again.

- Dataflow definition file uploads can take longer because Tableau CRM is validating the file and using the sfdcDigest nodes to define sync settings.

- When you remove an object's sfdcDigest node from the definition file, sync is not disabled for that object. If necessary, disconnect the object from sync on the Connect tab of the data manager.

- When you remove a field from an sfdcDigest node, the field is still included for sync. If necessary, exclude the field in the objects sync settings.

- When you add a field in an sfdcDigest node, the field is included for data sync and a sync is triggered for the object when the dataflow next runs.

- When you unschedule the dataflow, Tableau CRM does not disconnect the Salesforce objects and fields from sync.

Considerations When Using Tableau CRM Templated Apps with Data Sync

If you use template-based Tableau CRM apps, such as Sales Analytics or Service Analytics, here are some things to consider before and after you enable Data Sync. Data Sync is enabled by default if you turned on Tableau CRM after the Winter '20 release.

Enabling Data Sync When You're Already Using Tableau CRM Templated Apps

If you already have Tableau CRM apps when you enable Data Sync, it's important that the sync includes all the Salesforce objects and fields used in these apps. If it doesn't, the affected Tableau CRM app dataflows fail. Before enabling Data Sync, ensure that all the required objects and fields are included.

1. Before you enable Data Sync, set a recurring schedule on each Tableau CRM app dataflow. When you first enable Data Sync, Tableau CRM considers objects and fields only in scheduled dataflows.

2. After you enable Data Sync, check sync settings to verify that all the required objects and fields are included. See Add and Remove Local Salesforce Objects and Fields from Data Sync.

3. Run sync to ensure that all objects and fields in the Tableau CRM app dataflows have been synced.

4. Run your Tableau CRM app dataflows, and verify that they run to completion.

5. Schedule sync to run before all your dataflows to keep your Tableau CRM app data up to date.

Creating a Tableau CRM Templated App After Enabling Data Sync

When you create a Tableau CRM templated app, objects and fields from the new dataflow are included for sync, if they're not already included. Complete these steps after you create a Tableau CRM app.

1. Check sync settings, and verify that all the required objects and fields are included. See Add and Remove Local Salesforce Objects and Fields from Data Sync.

2. Run sync, and ensure that all objects in the new dataflow synced.

3. Run the new Tableau CRM app dataflow.

4. Schedule the new Tableau CRM app dataflow to run after sync.

Design Datasets from Synced Data and Other Datasets with Recipes and Dataflows

You can use a recipe or dataflow to design a dataset based on input data. In each recipe and dataflow, select the input data (connected objects or existing datasets), add data preparation logic to transform that data, and specify the dataset to load the results into. For example, you can use a recipe to combine data from different sources, clean the data to make it consistent, and then load the results into a new dataset.

The following recipe combines data from a data source and an existing dataset, and loads the results into another dataset.
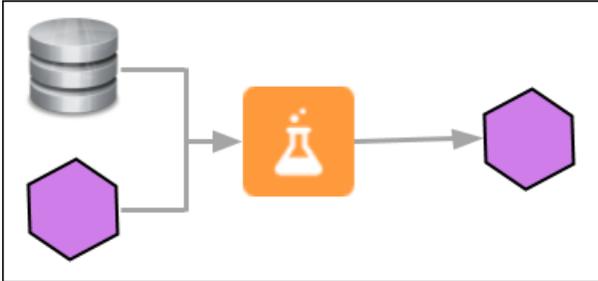
Should I Use a Recipe or Dataflow?

Although recipes and dataflows both prepare data, each approach offers a unique set of transformations that manipulate data. To determine which approach to use, figure out which transformations your data prep project requires. For example, recipes have more join types and smart transformations, such as Predict Missing Values and Detect Sentiment, that aren't available in dataflows. Recipes can also aggregate data to a higher level.

Design Datasets with Data Prep Recipes

A recipe is a sequence of transformations that Tableau CRM performs on source data before loading it into a dataset. For example, you can add transformations that combine data from multiple datasets, categorize data into buckets, create new columns based on calculations using other columns' data (formula columns), and clean the data to make column values consistent. You can then output the resulting data to a new target dataset.

Design Datasets with Dataflows and the Dataset Builder

Use a dataflow to create one or more datasets based on source data from existing datasets or synced data. A dataflow is a set of instructions that specifies what input data to include, how to transform that data, and which datasets to load the transformed data into.

SEE ALSO:

Prepare and Load Data into Datasets with Recipes and Dataflows

Should I Use a Recipe or Dataflow?

Although recipes and dataflows both prepare data, each approach offers a unique set of transformations that manipulate data. To determine which approach to use, figure out which transformations your data prep project requires. For example, recipes have more join types and smart transformations, such as Predict Missing Values and Detect Sentiment, that aren't available in dataflows. Recipes can also aggregate data to a higher level.

> **Note:** Dataflows and recipes aren't mutually exclusive. You can use both to meet more complex data preparation requirements. For example, you can use a dataflow to generate an intermediate dataset, and then use that dataset as the source for a recipe to perform additional transformations.

Recipe

Recipes are great for Salesforce admins with little to no data integration experience who want suggestions on how to clean up data and to preview transformation results. Recipes are also great for experienced data wranglers who just want to get stuff done without learning about the dataflow.

With a recipe you can:

- Preview your data and how it changes as you apply each transformation.
- Quickly remove columns or change column labels.
- Analyze the quality of your data with column profiles.
- Get smart suggestions about how to improve and transform your data.
- Bucket values without having to write complex SAQL expressions.
- Create calculated columns with a visual formula builder.
- Use a point-and-click interface to easily transform values to ensure data consistency. For example, you can bucket, trim, split, and replace values without a formula.
- See the history of all your changes, and back up or move forward to replay it.

Dataflow

Dataflows are great for advanced Tableau CRM developers who prefer to work in a visual editor and, if necessary, JSON.

With a dataflow you can:

- Design complex data preparation flows with the visual Dataflow Editor.
- Edit the underlying JSON for finer control.
- Perform calculations across rows to derive new data for trending analysis.
- Apply complex filters.

Transformation Comparison at a Glance

Still unsure which tool to choose? Use this table to compare which transformations are available in recipes and dataflows.

> **Note:** Tableau CRM currently supports two versions of Data Prep. Data Prep is the newest version that replaces the previous version, Data Prep Classic.

Feature	Recipe	Dataflow
Aggregate	✅	
Append	✅	✅
Bucket	✅	✅ (computeExpression transformation)
Calculate Expressions (across rows)	✅ (Formula transformation)	✅ (computeRelative transformation)
Calculate Expressions (same row)	✅ (Formula transformation)	✅ (computeExpression transformation)

Feature	Recipe	Dataflow
Convert Column Types	✅	✅
Delta		✅
Detect Sentiment	✅	
Drop Columns	✅	✅ (sliceDataset transformation)
Extract Dataset Data	✅ (Input node)	✅ (edgemart transformation)
Extract Salesforce Data	✅ (Input node)	✅ (sfdcDigest transformation)
Extract Synced Data	✅ (Input node)	✅ (digest transformation)
Filter	✅	✅
Flatten Hierarchies	✅	✅
Format Dates	✅	
Join	✅	
Lookup	✅	✅ (augment transformation)
Predict Missing Values	✅	
Predict Values	✅	✅ (prediction transformation)
Profile Column	✅	
Update Values		✅ (update transformation)

SEE ALSO:

Design Datasets with Data Prep Recipes

Design Datasets with Dataflows and the Dataset Builder

Design Datasets with Data Prep Recipes

A recipe is a sequence of transformations that Tableau CRM performs on source data before loading it into a dataset. For example, you can add transformations that combine data from multiple datasets, categorize data into buckets, create new columns based on calculations using other columns' data (formula columns), and clean the data to make column values consistent. You can then output the resulting data to a new target dataset.

[Should I Use Data Prep Classic or Data Prep?](#)

Tableau CRM provides two versions of Data Prep: Data Prep Classic and Data Prep. Data Prep is the latest version that replaces the previous version, Data Prep Classic. Recipes built in Data Prep can't be opened in Data Prep Classic. When you open a Data Prep Classic recipe, it is upgraded to and opens in Data Prep. As of the Winter '21 release, you can no longer create recipes in Data Prep Classic.

[Clean, Transform, and Load Data with Data Prep](#)

Data Prep provides an intuitive, visual interface that allows you to easily point-and-click your way to build recipes that prepare data and load it into a target. Use the graph of a recipe to see at a glance where data comes from and how it flows through the recipe to the target. To validate the recipe as you build, preview how raw data is transformed at every step of the way.

[Clean, Transform, and Load Data with Data Prep Classic](#)

Use a recipe to clean and combine data from multiple datasets or connected objects. Add bucket and formula fields, filter rows, transform field values, convert field types, and standardize date formats. You can then output the results to a new target dataset.

[Manage Recipes](#)

After you create a recipe, use the Dataflows and Recipes tab to edit and delete them.

SEE ALSO:

[Should I Use a Recipe or Dataflow?](#)

Should I Use Data Prep Classic or Data Prep?

Tableau CRM provides two versions of Data Prep: Data Prep Classic and Data Prep. Data Prep is the latest version that replaces the previous version, Data Prep Classic. Recipes built in Data Prep can't be opened in Data Prep Classic. When you open a Data Prep Classic recipe, it is upgraded to and opens in Data Prep. As of the Winter '21 release, you can no longer create recipes in Data Prep Classic.

Clean, Transform, and Load Data with Data Prep

Data Prep provides an intuitive, visual interface that allows you to easily point-and-click your way to build recipes that prepare data and load it into a target. Use the graph of a recipe to see at a glance where data comes from and how it flows through the recipe to the target. To validate the recipe as you build, preview how raw data is transformed at every step of the way.

Watch a Demo: ▶ Introducing Data Prep (English Only)

In Data Prep, a recipe consists of nodes. You can add different types of nodes to a recipe to bring in, manipulate, and write data to a target. Data Prep displays a graph of the recipe and its nodes in the Graph area.

Unlike a dataflow graph, which can be a bit cluttered, a Data Prep graph doesn't show every transformation. It shows only inputs (source data), appends, aggregates, filters, joins, and the output (where the data is written). It also shows transforms, which are groups of transformations that change the raw data, such as concatenation and column type conversions. By hiding lower-level data changes, the graph provides a higher-level, easier-to-read view of the flow of data. We call each object shown in the graph a *node*. A recipe can have multiple Input, Append, Aggregate, Filter, Join, Transform, and Output nodes.

Although the graph doesn't show individual data transformations, you can select a Transform node (1) in the Graph area to see its transformations in the Details area (2).

You can select a transformation step in the left panel of the Details area to preview the results of that transformation in the Preview tab. Similarly, you can select any node in the graph to preview the results of that node.

When you run a recipe, Tableau CRM applies the function of each node on the input data, and then outputs the results to the specified target. If the target is a dataset, you can then use that dataset for exploration or in a dashboard, or as input to other recipes or dataflows.

Data Prep Access Based on Your User Type

Not all Data Prep users need access to all data prep features. For example, a business analyst doesn't need to change dataset security to grant themselves and others access to more data. To support advanced and limited Data Prep user types, Tableau CRM comes with two user permissions: Edit Analytics Dataflows and Edit Dataset Recipes.

Create a Recipe with Data Prep

Use a Data Prep recipe to clean, transform, and enrich data before loading data into one or more targets. To ensure you're building the right logic, preview the results as you build it. When you're done, run the saved recipe to write the results to the targets. You can't open recipes created with Data Prep in Data Prep Classic.

Preview Results in a Data Prep Recipe

As a best practice, catch mistakes early by reviewing the results of each node and transformation that you add to a Data Prep recipe. Each user can initiate up to 4,000 previews per hour.

Profile Columns to Understand Data in a Data Prep Recipe

Run column profiling on sample data to estimate key stats about columns, such as the frequency of values and percent of columns with missing values. Column profiling is especially useful when you are combining data from different sources, where inconsistencies are often introduced.

Limitations When Using Data Prep

Consider the following Data Prep limitations before building recipes.

Data Prep Access Based on Your User Type

Not all Data Prep users need access to all data prep features. For example, a business analyst doesn't need to change dataset security to grant themselves and others access to more data. To support advanced and limited Data Prep user types, Tableau CRM comes with two user permissions: Edit Analytics Dataflows and Edit Dataset Recipes.

To have full access to data prep features, you must be assigned the Edit Analytics Dataflows user permission. Among other things, it enables you to change dataset security on datasets, allowing you to grant yourself and other users access to more Salesforce data in datasets.

If you're assigned the Edit Dataset Recipes user permission, you have limited access, including:

- When you create a recipe, you can't use datasets with security predicates as a source for the recipe.
- To prevent you from granting yourself access to rows that you currently don't have permission to see, you can't add or change security predicates on target datasets.
- You can't create, edit, run, or schedule recipes that have target datasets defined with security predicates.
- To prevent you from seeing data that you don't have access to view, you can't view connected objects on the Data tab or preview them as sources in a recipe.

To understand all differences between the user permissions, check out this table.

Action	Edit Dataset Recipes	Edit Analytics Dataflows
View, create, edit, and delete local connections. Creating a local connection includes adding connected objects. Editing a local connection includes changing the connected objects, moving objects between local connections, and filtering rows from connected objects.		X
View, create, edit, and delete output connections. To create an output connection, the org admin must enable the output connection in setup, and you need the Edit Analytics Dataflows and Add Remote Connections user permissions.		X
Run, schedule, unschedule, and set notifications for data syncs. You can also change the schedule and disconnect an object from a data sync.		
Add a local or remote connected object as a source to a recipe.		X
Write recipe results to a CSV.	X	X
Write recipe results to multiple output nodes		X
Write recipe results to an output connection. You must create the output connection before you can select it in an output node of a recipe.		X
Sync out—exports local Salesforce data from a connected object to a remote connection. To sync out data, the org admin must enable the output connection in		X

Action	Edit Dataset Recipes	Edit Analytics Dataflows
setup, and you need the Edit Analytics Dataflows and Add Remote Connections user permissions.		
Preview recipe data from a remote input or output connection in Data Prep.		X
Add, change, and remove security predicates for datasets that contain Salesforce data.		X
View, create, edit, delete, run, schedule, unschedule, and set notifications for a dataflow.		X
View, create, and delete a recipe.	X	X
Edit, run, schedule, unschedule, and set notifications for a recipe that writes to a dataset without a security predicate.	X	X
Edit, run, schedule, unschedule, and set notifications for a recipe that writes to a dataset with a security predicate.		X
Monitor a dataflow or recipe job. Download the dataflow error log for the job. If you don't own the dataflow, you also need the View All Data profile permission to download the error log.		X

SEE ALSO:

Salesforce Data Access in Tableau CRM

Set Up Dataset Security to Control Access to Rows

Create a Recipe with Data Prep

Use a Data Prep recipe to clean, transform, and enrich data before loading data into one or more targets. To ensure you're building the right logic, preview the results as you build it. When you're done, run the saved recipe to write the results to the targets. You can't open recipes created with Data Prep in Data Prep Classic.

Watch a Demo: ▶ Create a Data Prep Recipe (English Only)

> 📝 Note: Government Cloud users must use Data Prep Classic to create and edit recipes. To create a Data Prep Classic recipe, see Create a Recipe with Data Prep Classic.

1. In Tableau CRM, click **Data Manager** in the left pane.
   Data Manager opens in a new browser tab.

2. In Data Manager, click **Dataflows & Recipes**.

3. In the Dataflows & Recipes tab, click **Recipes**.
   The Recipes subtab displays a list of recipes created with Data Prep Classic and Data Prep.

4. To create a recipe, click **Create Recipe**.

5. Review the Data Prep welcome mat, and then click **X** to close it.
   Optionally, click **Learn More** to get more information about Data Prep.

6. Click **Select Data** to choose the initial input data for the recipe.
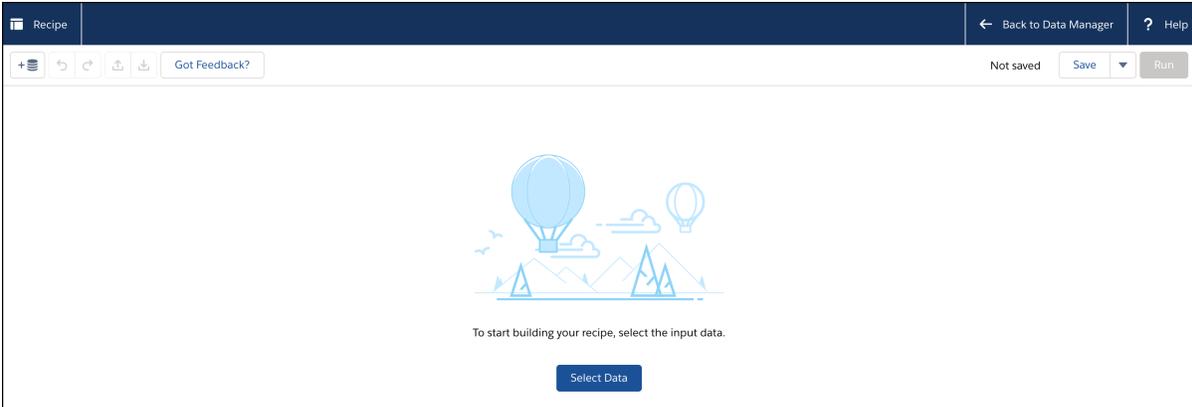
7. Select the check box next to each dataset or connected object that you want to add as input for the recipe.

    You can't use a trended dataset as input data for a recipe. (To prepare data from trended datasets, use a dataflow instead.) If needed, you can add more data to the recipe later.

8. In the Selected Columns area on the right, choose which columns to include from the selected input data.
    By default, all columns are included. If you added multiple data source objects, select each row, one at a time, to choose its columns. The Selected Columns area shows columns for the highlighted row only.

9. Click **Next**.

Data Prep adds a separate Input node to the graph for each input data selection.

10. To add a node to the recipe, click the Add Node button (⊕) between two nodes or at the end of the recipe, and then select the node type. To show the Add Node button between two nodes, hover the cursor on the connecting line.

Add nodes to transform, filter, aggregate, join, and append data.

11. To ensure you're adding the right data prep logic, preview the results of each node and transformation that you add to the recipe. To preview the results, select the node or transformation in the graph.

For example, selecting an Aggregate node shows the following preview results.

12. To edit or delete a node, select the node, click ⋯, and then select the appropriate action.

13. To edit or delete a transformation in a Transform node, select the Transform node, select the transformation step, and then click ✏️ or 🗑️.

14. To split the recipe into two branches, hover the cursor over the node where the branch begins, click ⌁, and then select the node type to start the new branch.

You can branch a recipe to write results to multiple output nodes or to transform some of the rows before appending them together again later.

15. To connect nodes in the graph, drag the Add Node button (**+**) on top of the target node in the recipe graph, and then on top of the **Connect Nodes** option.
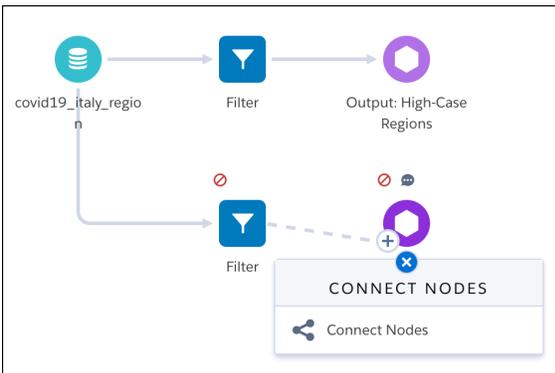
16. End the recipe with one or more Output nodes to tell the recipe where to write the results.

17. To save the recipe, click **Save**.

18. Enter a recipe name and description, and then click **Save** again.

19. To change the recipe name, click the Edit Recipe Name button (pencil icon) and enter a new label.
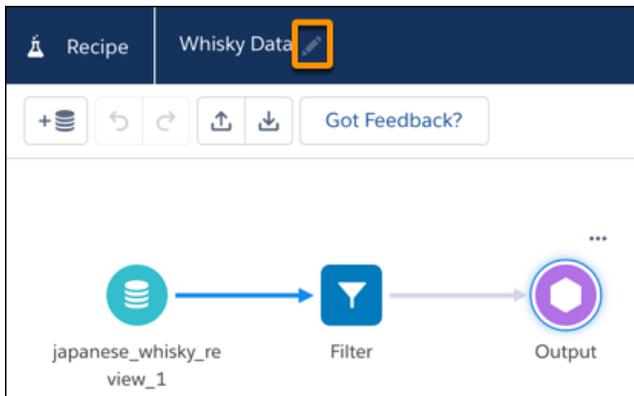
As you build recipes, feel free to share your feedback with us. Click **Got Feedback** in Data Prep to tell us what's on your mind.

Try Out Pilot Features in a Data Prep Recipe

Join the Smart Transform Pilot Program to try out all pilot features in Data Prep recipes. Pilot features can change with each release as existing features become generally available or are retired and new pilot features are added to the program.

Nodes for Data Prep Recipes

Data Prep provides nodes that allow you to bring in data, transform it, and write the results to a target. For example, you can use a Join node to add columns from a related object to a recipe. You add nodes in the Graph area of Data Prep. You can string together multiple nodes to sequentially change the flow of the data.

Transformations for Data Prep Recipes

Data Prep provides transformations that allow you to prepare, clean, and transform your data. For example, you can use a transformation to create a calculated column based on a formula. You add transformations inside a Transform node. You can string together multiple transformations to manipulate data sequentially.

SEE ALSO:

[Run a Recipe](#)

[Open and Edit a Recipe](#)

[Delete a Recipe](#)

[Preview Results in a Data Prep Recipe](#)

Try Out Pilot Features in a Data Prep Recipe
Join the Smart Transform Pilot Program to try out all pilot features in Data Prep recipes. Pilot features can change with each release as existing features become generally available or are retired and new pilot features are added to the program.

We provide the Smart Transform Pilot Program to selected customers who agree to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. Each Data Prep pilot feature isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Smart Transform Pilot Program in the [Trailblazer Community](#).

To use the Smart Transform Pilot Program features, you need to be assigned the Enable Data Prep Pilot Features user permission in addition to your org being a part of the Smart Transform Pilot Program.

The program includes these pilot features.

- [Time Series Forecasting transformation](#)

- Cluster transformation
- Pivot functionality in the Aggregate node

Nodes for Data Prep Recipes

Data Prep provides nodes that allow you to bring in data, transform it, and write the results to a target. For example, you can use a Join node to add columns from a related object to a recipe. You add nodes in the Graph area of Data Prep. You can string together multiple nodes to sequentially change the flow of the data.

A recipe can have multiple Input, Append, Aggregate, Filter, Join, Transform, and Output nodes.
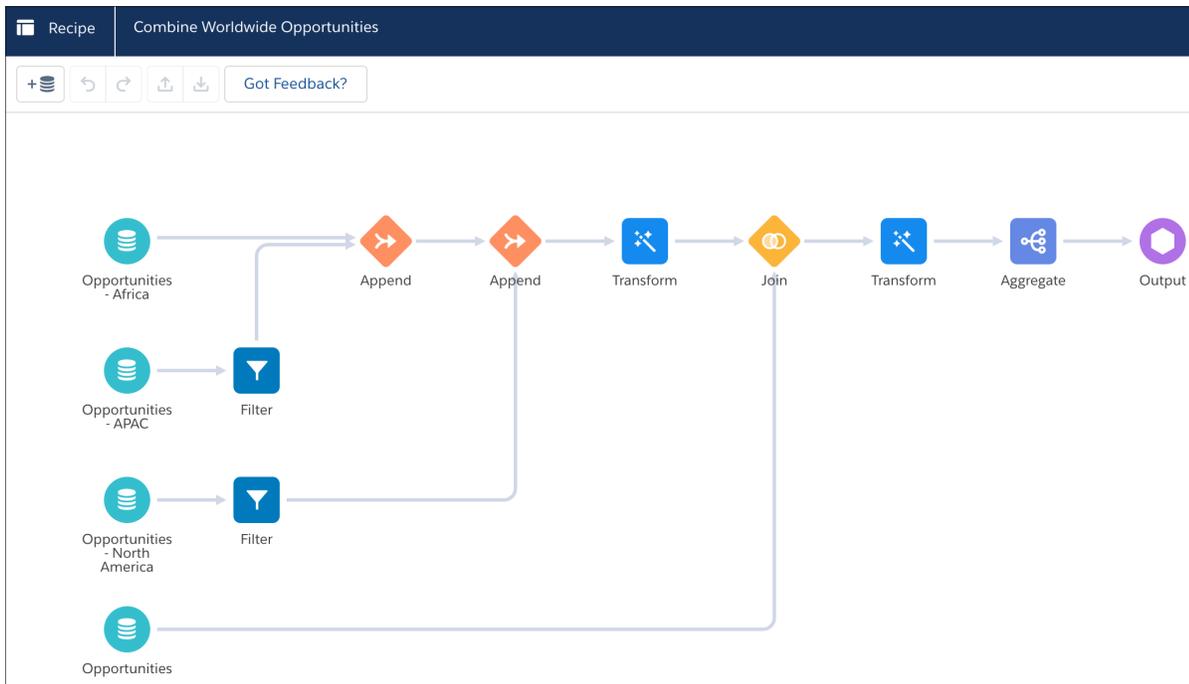
Input Node: Add Data to the Recipe

Although you choose input data when you create a recipe, if needed, you can add more data to an existing recipe. Data Prep shows a separate Input node for each set of data that you add to the recipe.

Aggregate Node: Roll Up Data to a Higher Level

Use an Aggregate node to roll up data to a higher granularity. You can aggregate using: unique, sum, average, count, maximum, and minimum. You can group by any dimension or date column. You can also pivot dimensions as columns. For example, you can group opportunities by account to get account-level details, like total opportunity amount, average number of cases, and the average days to close an opportunity for each account. You can then pivot on account type to analyze each combination of account and account type.

Append Node: Stack Rows from Different Sets of Data

Use an Append node in a Data Prep recipe to stack rows from multiple sets of input data into one dataset. For example, use this node to combine sales records from two different Salesforce orgs, each containing sales transactions from a specific region.

Filter Node: Filter Rows

Use a Filter node to remove rows that you don't need in your target. For example, you can filter case records to focus on escalated cases only. Filter values are case sensitive.

Join Node: Add Related Columns of Data to the Recipe

Use a Join node to add columns of data from related objects to existing data in a recipe. Depending on how you want to combine the new data, use one of the following methods: lookup, left join, right join, inner join, or full outer join.

Transform Node: Transform the Data Before Loading It Into a Target

Use a Transform node to manipulate data based on the transformations added to the node. You can add one or more transformations to each Transform node. For example, you can add transformations to concatenate two columns, standardize the formats in a date column, and detect the sentiments of comments in a Feedback column. Each transformation modifies the data in a unique way. To manipulate data at multiple stages of a recipe, add a separate Transform node at each stage.

Output Node: Write Recipe Results to a Dataset or External System

Use Output nodes with Data Prep recipes to write recipe results to a target, such as a dataset or an Amazon S3 connection. A recipe can have only one Output node. If the recipe writes to a dataset, you can change the API name and label of the dataset in the Output node. If an existing dataset uses the specified API name, the recipe overwrites the existing dataset when it runs.

Input Node: Add Data to the Recipe

Although you choose input data when you create a recipe, if needed, you can add more data to an existing recipe. Data Prep shows a separate Input node for each set of data that you add to the recipe.

1. In the recipe, select the Add Data button ( ).

2. Select the input data (1) and then choose which columns to include (2). You can select a dataset or connected object as the input data.

   📝 **Note:** You can't use a trended dataset as input data for a recipe. To prepare data from a trended dataset, use a dataflow.

   By default, all columns are included. If you select multiple sets of input data, select each row, one at a time, to choose its columns. The Selected Columns area shows columns for the selected row only.

3. Click **Next**.

   Data Prep adds a separate Input node to the graph for each input data selection.

4. Save the recipe.

When you add an Input node to a Data Prep recipe, all columns after the first 100 are hidden from preview, by default. If needed, you can unhide them to preview them.

Aggregate Node: Roll Up Data to a Higher Level

Use an Aggregate node to roll up data to a higher granularity. You can aggregate using: unique, sum, average, count, maximum, and minimum. You can group by any dimension or date column. You can also pivot dimensions as columns. For example, you can group opportunities by account to get account-level details, like total opportunity amount, average number of cases, and the average days to close an opportunity for each account. You can then pivot on account type to analyze each combination of account and account type.

> 📝 **Note:** Join the Smart Transform Pilot Program to try out all pilot features in Data Prep recipes. Pilot features can change with each release as existing features become generally available or are retired and new pilot features are added to the program. We provide the Smart Transform Pilot Program to selected customers who agree to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. Each Data Prep pilot feature isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Smart Transform Pilot Program in the Trailblazer Community.

1. In the recipe that already has at least one Input node, select the Add Node button (➕) between two nodes or at the end of the recipe. To show the Add Node button between two nodes, hover the cursor on the connecting line.

2. In the Add Node dialog box, select **Aggregate**.
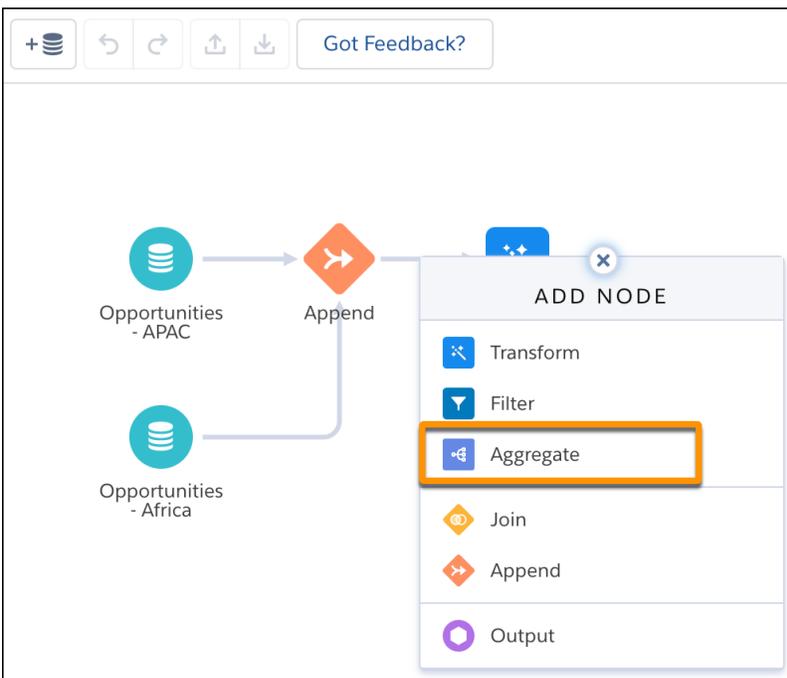
3. Add at least one aggregate.

Before you can add a grouping, add an aggregate. You can select one of the following aggregate functions.

Aggregate Function	Description
Unique	Counts only unique values in the selected column.
Sum	Calculates the total by adding all values in the selected column.
Average	Calculates the average value of the selected column.
Count	Counts all values in the selected column.
Maximum	Calculates the maximum value of the selected column.
Minimum	Calculates the minimum value of the selected column.
Stddevp	Calculates the population standard deviation. Population standard deviation measures the spread of data distribution—the typical distance between each data point and the average.
Stddev	Calculates the sample standard deviation. Sample standard deviation measures the spread of data distribution for a given sample. Use sample standard deviation to estimate the population standard deviation based on a sample.
Varp	Calculates the population variance. Population variance indicates how spread out data points are in a specific population.
Var	Calculates the sample variance. Sample variance indicates how spread out data points are for a given sample. Use sample variance to estimate the population variance based on a sample.
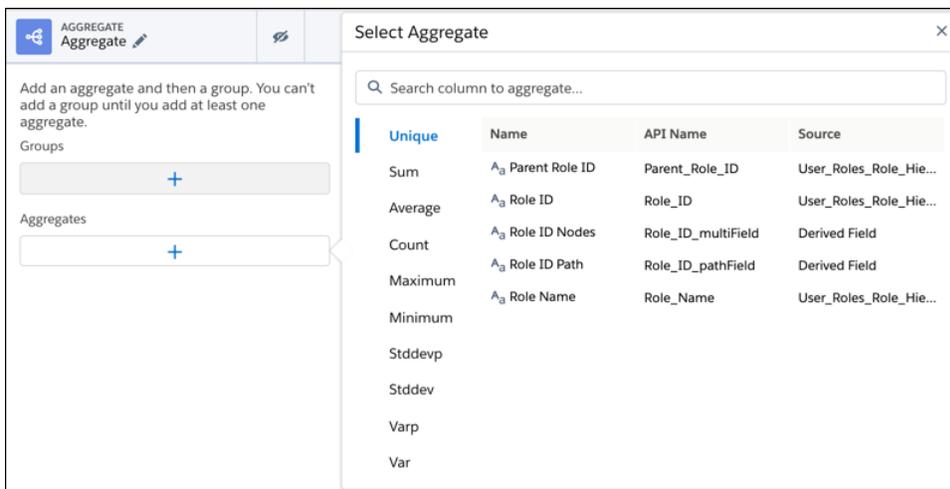
4. To group rows, add one or more dimension or date columns in the Group Rows field.
For example, you can group by account name.
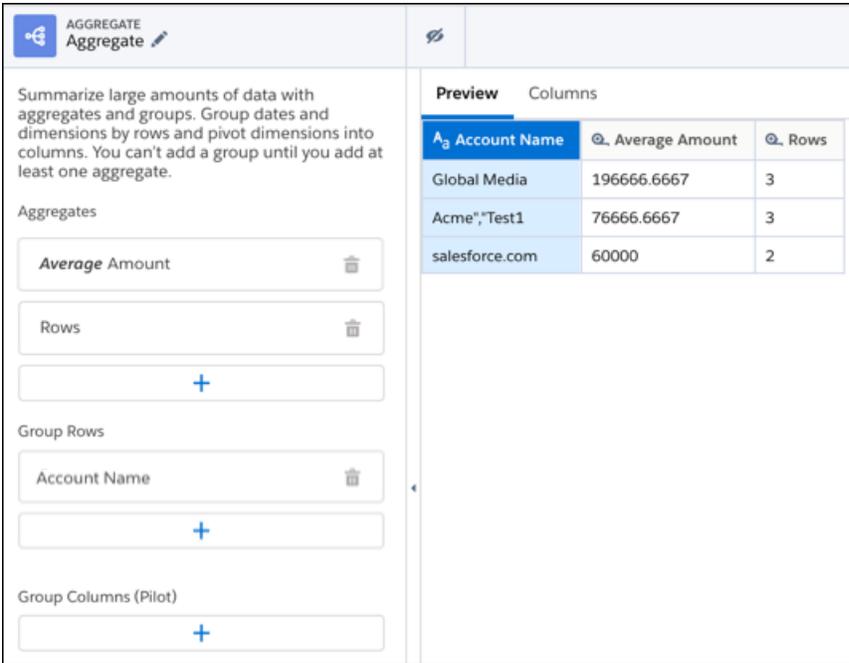
5. To pivot dimensions as columns, add up to two dimension columns in the Group Columns (Pilot) field, select the values for each dimension column, and click **Apply**.

   For example, you can group columns by opportunity type. Each combination of opportunity type and aggregate appears as a separate column.
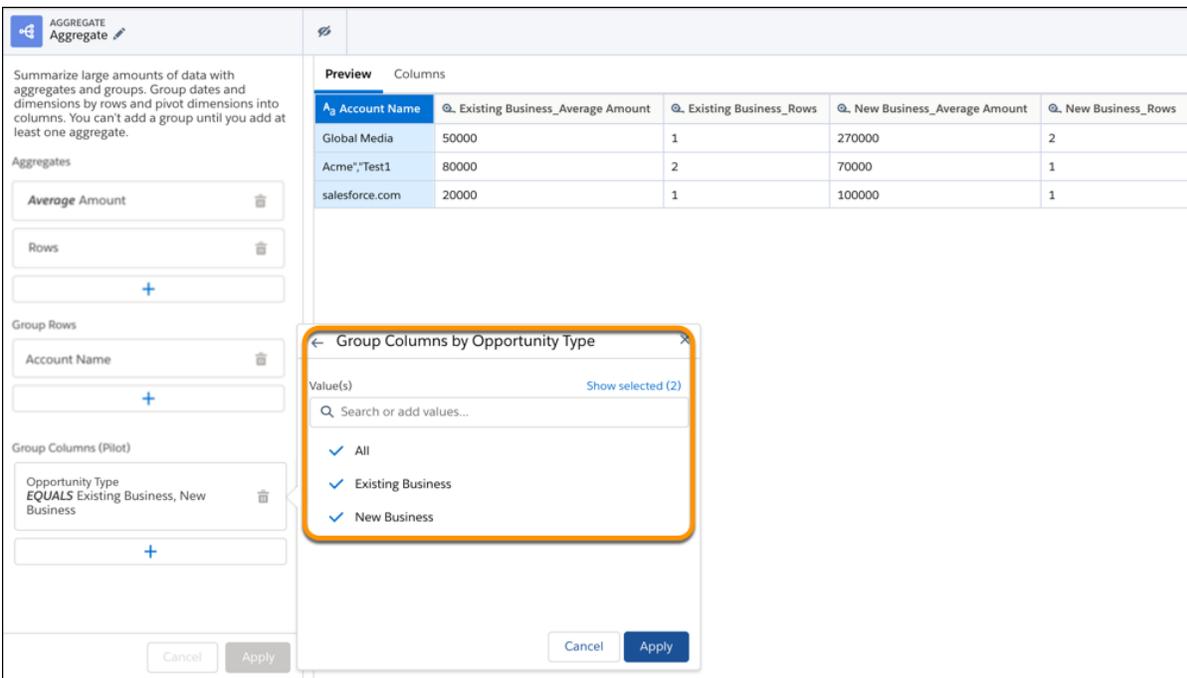
> 📝 Note: The Group Columns (Pilot) field is visible only if your org participates in the Smart Transform Pilot Program and you're assigned the Enable Data Prep Pilot Features user permission.

6. To add the Aggregate node to the recipe, click **Apply**.

7. Save the recipe.

📝 Note: Because pivoting increases the number of columns, keep in mind that the Aggregate node can create up to 5,000 columns. If the node exceeds the maximum, you can reduce the number of columns by changing the aggregates or row and column groupings.

Append Node: Stack Rows from Different Sets of Data

Use an Append node in a Data Prep recipe to stack rows from multiple sets of input data into one dataset. For example, use this node to combine sales records from two different Salesforce orgs, each containing sales transactions from a specific region.

Let's look at an example of how you can use append in a recipe. Imagine you're a company that uses separate Salesforce orgs for your APAC and Africa sales, but you want to report on worldwide sales. You sync the APAC opportunities to Tableau CRM, add the data to a recipe, then use an Append node to combine it with the Africa sales data. After you match the columns from both sets of input data, both sets of sales records are combined into the same worldwide opportunity dataset.
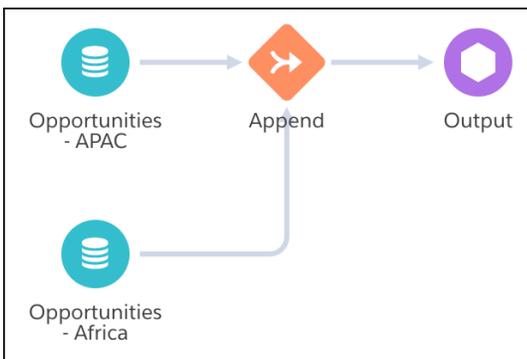
💡 Tip: If a recipe column contains unique IDs that identify each row, you must ensure that the appended data doesn't duplicate the IDs. The Append node doesn't remove duplicate records or validate the uniqueness of columns.

1. In a Data Prep recipe that already has at least one Input node, select the Add Node button (➕) between two nodes or at the end of the recipe. To show the Add Node button between two nodes, hover the cursor on the connecting line.

2. In the Add Node box, select **Append**.
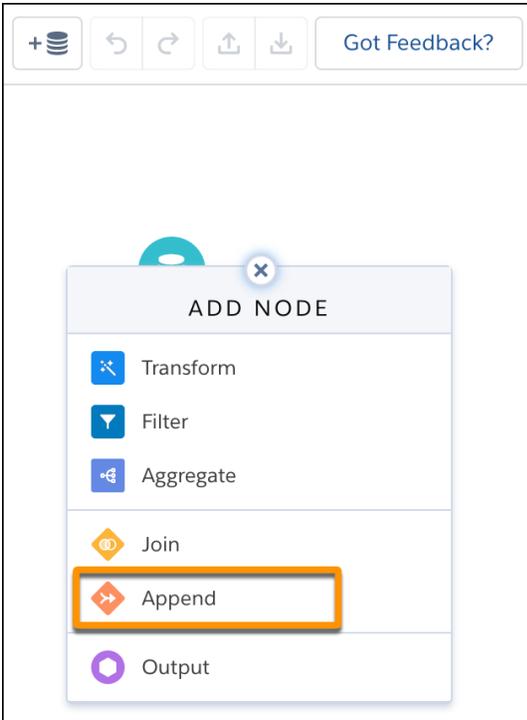
3. Select the data to append (1), and then choose which columns to include (2).
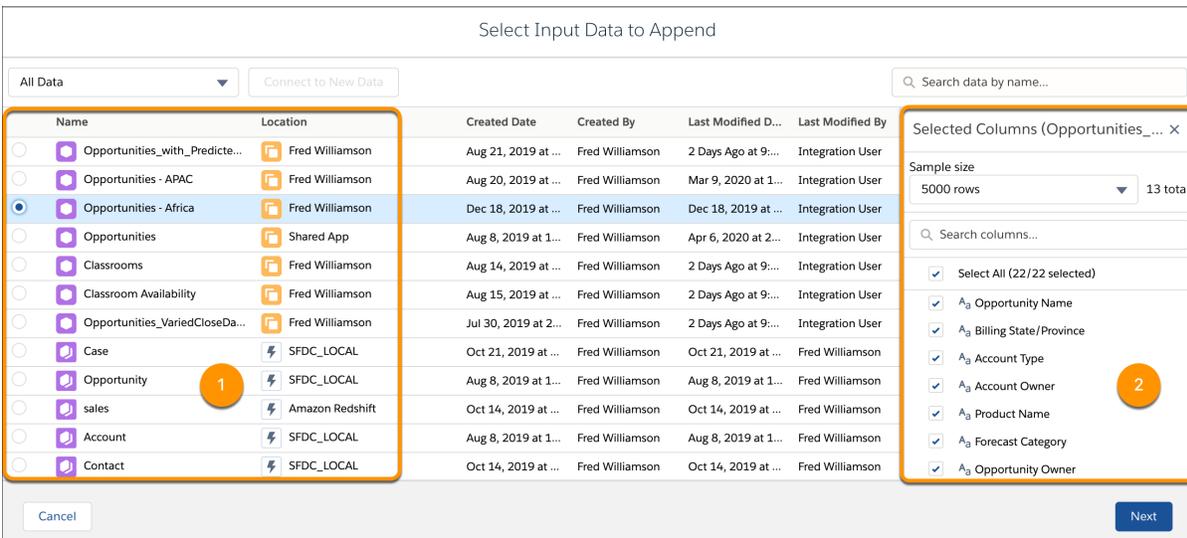
4. Click **Next**.

Data Prep automatically maps columns from both sets of input data based on matching column names. You can change the mappings, manually map unmapped columns, or leave columns unmapped.

If a column is mapped for one source and not the other, the Append node inserts null values for all source rows to which the column doesn't apply. For example, the Product ID and Region columns don't apply to Legacy Sales Data source. As a result, the preview shows nulls for these columns for all rows coming from Legacy Sales Data (1).

> **Note:** If a measure column isn't mapped to a source and null measure handling isn't enabled, the Append node inserts zeroes instead of nulls for those source rows.

5. To map two columns, enter the column name from the appended rows next to its corresponding recipe column.

6. To add a column from the appended rows that doesn't exist in the recipe, click **+** (below the mapped columns) and select the column. Leave the recipe column blank.

7. Click **Apply** to add the node to the recipe.

8. Save the recipe.

When you run the recipe, the Append node combines rows from both sets of input data into the same dataset.

Filter Node: Filter Rows

Use a Filter node to remove rows that you don't need in your target. For example, you can filter case records to focus on escalated cases only. Filter values are case sensitive.

1.

   In the recipe that already has at least one Input node, select the Add Node button (+) between two nodes or at the end of the recipe. To show the Add Node button between two nodes, hover the cursor on the connecting line.

2. In the Add Node dialog box, select **Filter**.
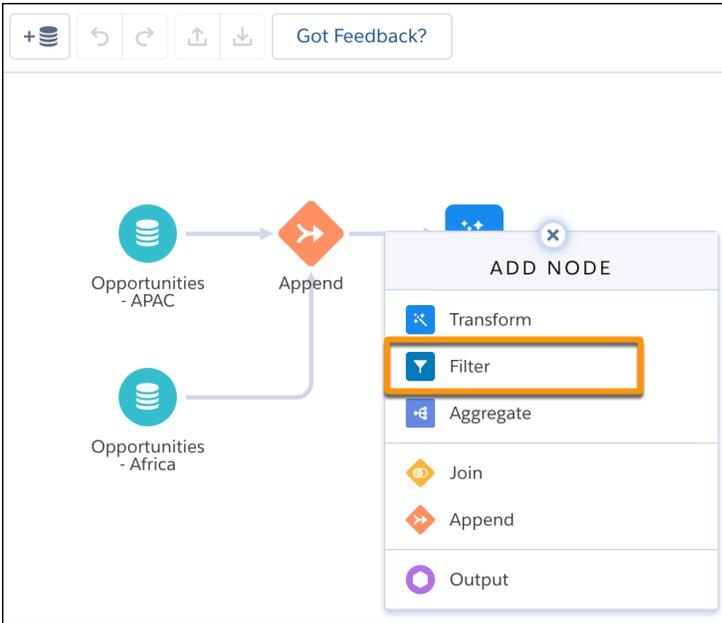
3. Enter the filter conditions.

   If you enter a filter value in a condition, know that filter values are case sensitive.

   If you define a filter condition on a dimension column, you can select a filter value from the list. The list of available values comes from the sampled rows, not all rows. If a value doesn't appear in the list, change your sample size or manually enter the value in the filter condition. Then, click **Add new value**.
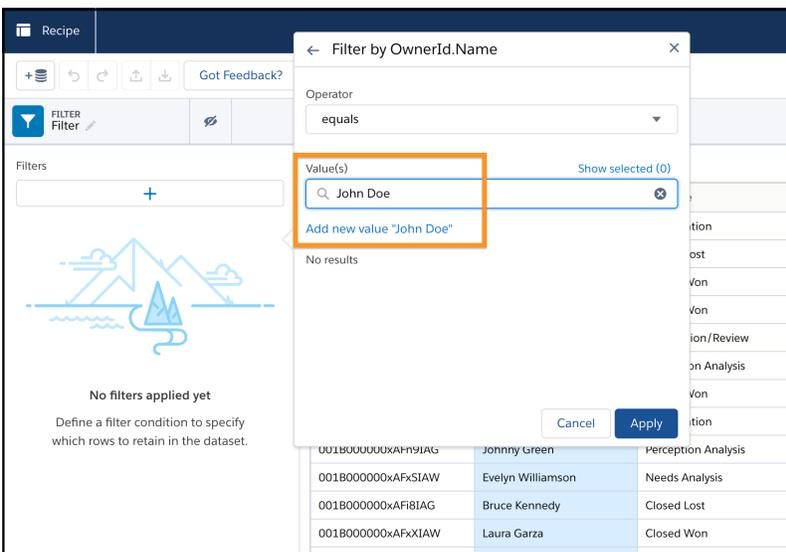
4. Click **Apply**.

5. Save the recipe.

Join Node: Add Related Columns of Data to the Recipe

Use a Join node to add columns of data from related objects to existing data in a recipe. Depending on how you want to combine the new data, use one of the following methods: lookup, left join, right join, inner join, or full outer join.

1. In a Data Prep recipe that already has at least one Input node, select the Add Node button (+) between two nodes or at the end of the recipe. To show the Add Node button between two nodes, hover the cursor on the connecting line.

2. In the Add Node box, select **Join**.
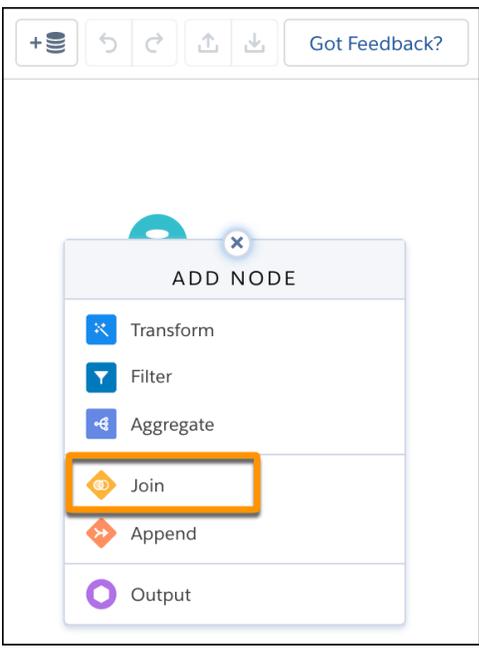
3. Select the data to join (1), and then choose which columns to include (2).
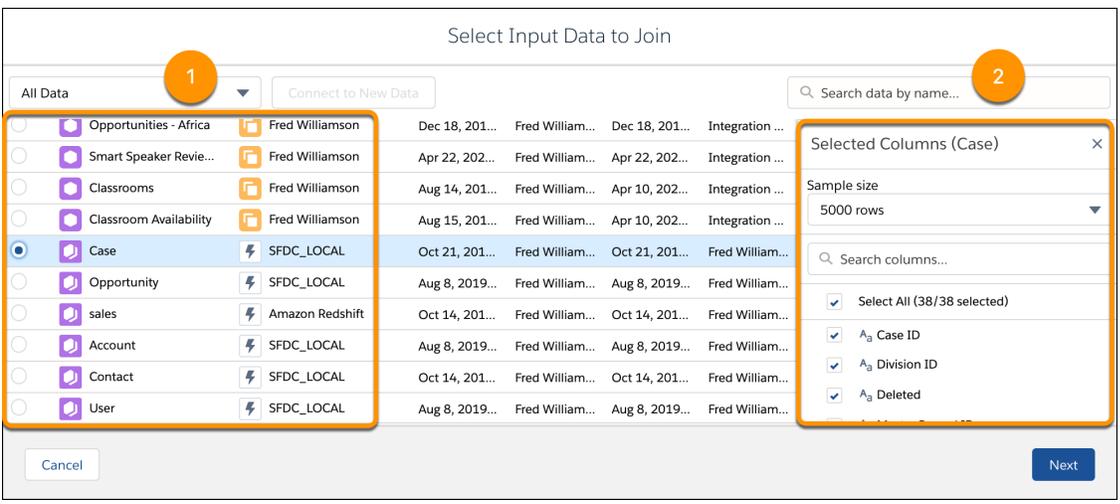
4. Click **Next**.

5. Choose the join type.

By default, Lookup is selected.
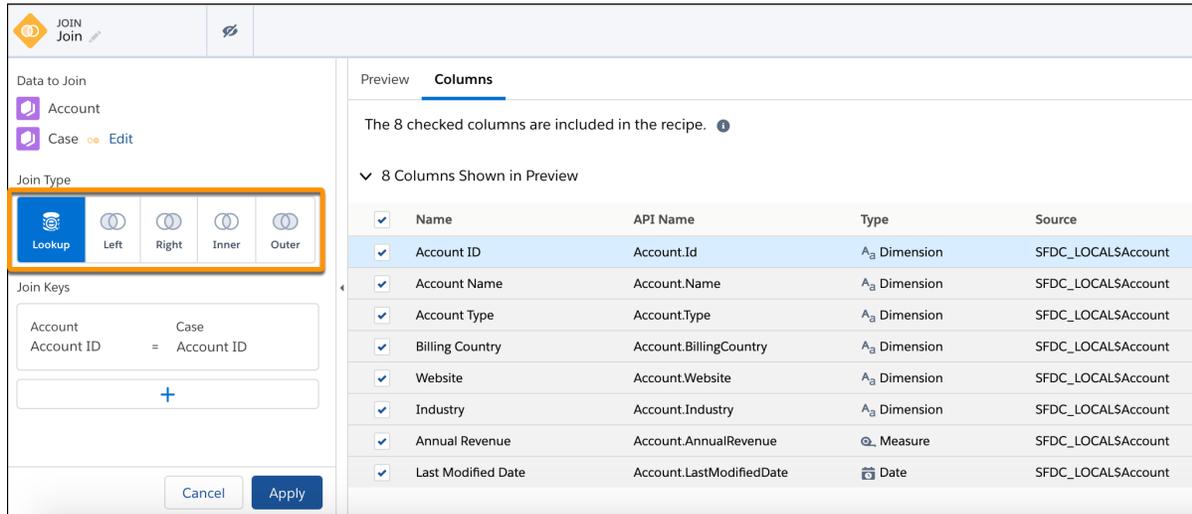
6. If needed, change the join keys.

Data Prep adds suggested join keys from both sets of input data. If needed, use multiple keys to match records. For example, imagine you're adding contact information to lead data. Using names alone to match can result in duplicate matches when different leads have the same name. Use two keys to match on name and company to ensure you're matching the right data. You can use up to 5 keys.

> 🛑 **Important:** Tableau CRM datasets contain date and time component fields (such as Year, Quarter, and Month) that are derived automatically from existing date and time fields. You can't use these component fields as join keys.

7. To prevent API name conflicts between columns in the left and right data sources, enter a prefix for columns coming from the right data source.

The Join node prepends the prefix to the API names of all columns from the right. For example, if you specify Case as the prefix, all right column API names begin with "Case."
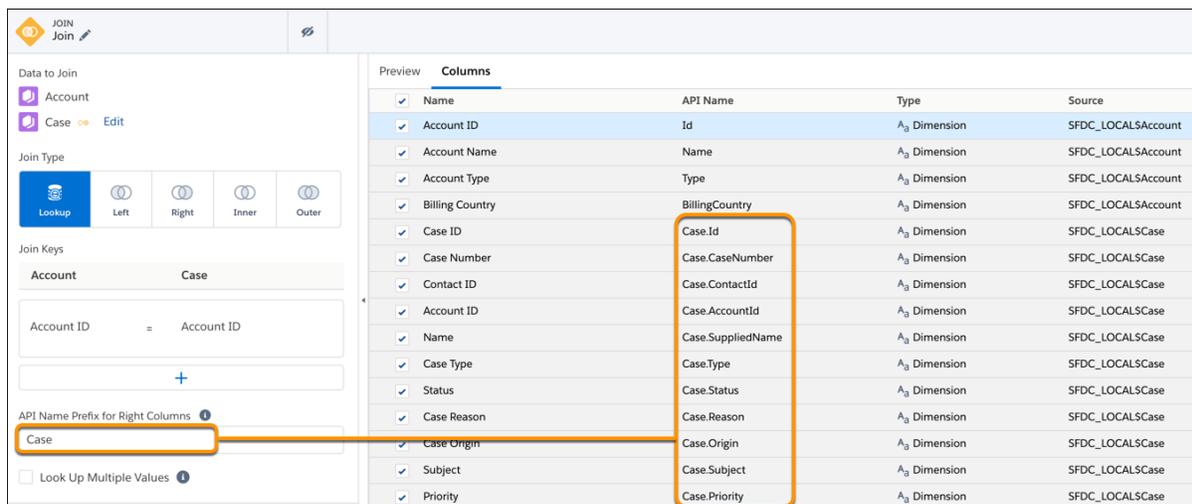
8. If you chose the Lookup join type, select **Lookup Multiple Values** for the lookup to return all matching rows from the right data stream. If you don't select this option, the lookup returns only one of the matching rows.

For more information about single-match and multiple-match lookups, see Lookup on page 724.

9. In the Columns tab in the right panel, select the columns to include in the recipe. The Source column in the table shows the object where each column comes from.

By default, the recipe only adds the join key columns. If a join key column isn't included in the recipe, you can't apply the transformation to the recipe.

10. To preview all included columns and their data, click **Preview**.
The joined columns show in Preview.

11. To hide columns from the preview, click the Hide button (⊘).

The preview can display up to 100 columns. Hiding columns from the preview doesn't affect whether the columns are included in the recipe.

12. Click **Apply** to add the node to the recipe.

13. Save the recipe.

Lookup

A lookup returns all rows from the left data stream (recipe data) and only matching rows from the right data stream (lookup source). If multiple matches are found in the right data stream, you can set the lookup to return either a single row or all matching rows. To ensure that the grain of the results doesn't change, the lookup outputs one row for each row in the left data stream.

Left Join

Like a lookup, a left join includes all rows from the left (recipe data) and only matching rows from the right. Unlike a lookup, a join includes all matched rows in the target when multiple rows match.

Right Join

A right join includes all rows from the right and only matching rows from the left (recipe data). The join includes all matched rows in the target when multiple rows match.

Inner Join

An inner join includes only matching rows from the left (recipe data) and right. The join includes all matched rows in the target when multiple rows match.

Full Outer Join

A full outer join includes all rows from the left (recipe data) and right, regardless of whether they have matches. The join includes all matched rows in the target when multiple rows match.

Considerations When Using Joins

Unlike a lookup, a join creates a separate record for each match in the target dataset when multiple rows match. Before using a join, ensure that you understand the implications of duplicate rows.

Lookup

A lookup returns all rows from the left data stream (recipe data) and only matching rows from the right data stream (lookup source). If multiple matches are found in the right data stream, you can set the lookup to return either a single row or all matching rows. To ensure that the grain of the results doesn't change, the lookup outputs one row for each row in the left data stream.

📝 **Note:** Data Prep Classic supports single-match lookups only.

👁 **Example:** Our company's marketing team captures demographic data in Redshift for opportunities stored in Salesforce. To help create more targeted campaigns, the team is building a dashboard that segments our current opportunities by the customers demographics. Let's use a lookup to add the demographic data to each opportunity record.

Consider the following two data streams. The left data stream contains opportunity records. The right contains demographic data. To illustrate how the lookup handles special cases, both data streams contain unmatched rows and duplicate keys.

If the lookup is configured to return a single match when multiple matches are found, the lookup produces the following results based on the matching keys.

Aa Cust_ID (Left Key)	Aa Customer Name	Q Amount	Aa Cust_id (Right Key)	Aa Education Level	Q Average Income
1	Alice	5000	1	High School	50000
2	Amy	4500	2	Bachelors	76000
2	Bob	6000	2	Bachelors	76000
3	Claire	10000	3	Masters	100000
4	David	15000			0

All rows from the left, including rows without a match, are included in the results. Although Cust_ID 1 has two matches on the right, the lookup returns only the first matching record. Also, because Cust_ID 4 doesn't have a match, the Cust_id and Education Level columns are null and Average Income is 0 for that record. Unmatched dimensions are set to null. Unmatched measures are set to 0.

If the lookup is configured to return all matching values when multiple matches are found, the lookup produces the following results based on the matching keys.

Cust_ID (Left Key)	Customer Name	Amount	Cust_id (Right Key)	Education Level	Avg Income
1	Alice	5000	1	High School, Secondary School	98000
2	Amy	4500	2	Bachelors	76000
2	Bob	6000	2	Bachelors	76000
3	Claire	10000	3	Masters	100000
4	David	15000	null	null	0

A multiple-match lookup returns the same results as the single-match lookup, except for left rows that have multiple matches. Unlike a single-match lookup, a multiple-match lookup returns all matches and combines them. For dimension columns, it generates a multivalue column with all dimension values—notice how High School and Secondary School are both in the Education Level column for Cust_ID 1. For measure columns, it adds the measure values from all matched records. For Cust_ID 1, the Avg Income column is 98,000 (50,000 + 48,000).

> **Note:** The Preview tab doesn't show all values in a multivalue column. Instead, it shows the first value only. Keep this behavior in mind when you preview the results of a multiple-match lookup. In the example, notice that the Preview tab shows only High School in the Education Level column even though the column contains High School and Secondary School.

Aₐ Cust_ID (Left Key)	Aₐ Customer Name	Q Amount	Aₐ Cust_id (Right Key)	Aₐ Education Level	Q Average Income
1	Alice	5000	1	High School	98000
2	Amy	4500	2	Bachelors	76000
2	Bob	6000	2	Bachelors	76000
3	Claire	10000	3	Masters	100000
4	David	15000			0

Left Join

Like a lookup, a left join includes all rows from the left (recipe data) and only matching rows from the right. Unlike a lookup, a join includes all matched rows in the target when multiple rows match.

> **Example:** Our company's marketing team captures demographic data in Redshift for opportunities stored in Salesforce. To help create more targeted campaigns, the team is building a dashboard that segments our current opportunities by the Education Level demographic. Let's use a left join to add demographic data to the opportunity data.
>
> Consider the following two data streams that feed the recipe's target dataset. To illustrate how this recipe function handles unmatched rows and duplicate keys, we included them in both data streams.

Left

Opp_ID (Left Key)	Customer Name
1	Alice
2	Amy
2	Bob
3	Claire
4	David

Right

Opp_id (Right Key)	Education Level
1	High School
1	Secondary School
2	Bachelors
3	Masters
5	PhD

After performing the left join based on the matching keys, the recipe produces the following target dataset.

Opp_ID (Left Key)	Customer Name	Opp_id (Right Key)	Education Level
1	Alice	1	High School
1	Alice	1	Secondary School
2	Amy	2	Bachelors
2	Bob	2	Bachelors
3	Claire	3	Masters
4	David	*null*	*null*

All rows from the left, including those without a match and those with multiple matches, are included in the target. Because Opp_ID 1 has two matches on the right (High School and Secondary School), the target contains a separate record for each education level. Also, because Opp_ID 4 doesn't have a match, the Opp_id and Education Level columns are null for that record.

Right Join

A right join includes all rows from the right and only matching rows from the left (recipe data). The join includes all matched rows in the target when multiple rows match.

Example: Our company's marketing team captures demographic data in Redshift for opportunities stored in Salesforce. To help create more targeted campaigns, the team is building a dashboard that segments our current opportunities by the Education Level demographic. Let's use a right join to add the opportunity data to the demographic data.

Consider the following two data streams that feed the recipe's target dataset. To illustrate how this recipe function handles unmatched rows and duplicate keys, we included them in both data streams.

Left

Opp_ID (Left Key)	Customer Name
1	Alice
2	Amy
2	Bob
3	Claire
4	David

Right

Opp_id (Right Key)	Education Level
1	High School
1	Secondary School
2	Bachelors
3	Masters
5	PhD

After performing the right join based on the matching keys, the recipe produces the following target dataset.

Opp_ID (Left Key)	Customer Name	Opp_id (Right Key)	Education Level
1	Alice	1	High School
1	Alice	1	Secondary School
2	Amy	2	Bachelors
2	Bob	2	Bachelors
3	Claire	3	Masters
null	*null*	5	PhD

All rows from the right, including those without a match and those with multiple matches, are included in the target. Because Opp_id 2 has two matches on the left (Amy and Bob), the target contains a separate record for each of them. Also, because Opp_id 5 doesn't have a match, the Opp_ID and Customer Name columns are null for that record.

⚠ Warning: If duplicate records contain measures (as shown for Opp_id 1 in the following screenshot), don't double count the measures when aggregating the records. To prevent duplicate records, use a lookup instead of a join.

Opp_ID (Left Key)	Customer Name	Opp_id (Right Key)	Education Level	Annual Income
1	Alice	1	High School	90000
1	Alice	1	Secondary School	90000
2	Amy	2	Bachelors	70000
2	Bob	2	Bachelors	80000
3	Claire	3	Masters	60000
4	David	*null*	*null*	75000

Inner Join

An inner join includes only matching rows from the left (recipe data) and right. The join includes all matched rows in the target when multiple rows match.

👁 **Example:** Our company's marketing team captures demographic data in Redshift for opportunities stored in Salesforce. To compare opportunities by education level of the customer, let's first combine both sets of data into a dataset using an inner join.

Consider the following two data streams that feed the recipe's target dataset. To illustrate how this recipe function handles unmatched rows and duplicate keys, we included them in both data streams.

Left

Opp_ID (Left Key)	Customer Name
1	Alice
2	Amy
2	Bob
3	Claire
4	David

Right

Opp_id (Right Key)	Education Level
1	High School
1	Secondary School
2	Bachelors
3	Masters
5	PhD

After performing the inner join based on the matching keys, the recipe produces the following target dataset.

Opp_ID (Left Key)	Customer Name	Opp_id (Right Key)	Education Level
1	Alice	1	High School
1	Alice	1	Secondary School
2	Amy	2	Bachelors
2	Bob	2	Bachelors
3	Claire	3	Masters

All rows that have a match are included in the target. Because of multiple matches, you see two records with Opp_ID 2 (Amy and Bob) and two records with Opp_id 1 (High School and Secondary School). Notice also that Opp_ID 4 and Opp_id 5 are excluded from the target because neither have a match.

Full Outer Join

A full outer join includes all rows from the left (recipe data) and right, regardless of whether they have matches. The join includes all matched rows in the target when multiple rows match.

👁 **Example:** Our company's marketing team captures demographic data in Redshift for opportunities stored in Salesforce. To help create more targeted campaigns, the team is building a dashboard that segments our current opportunities by the Education Level demographic. Let's use a full outer join to combine the opportunity and demographic data.

Consider the following two data streams that feed the recipe's target dataset. To illustrate how this recipe function handles unmatched rows and duplicate keys, we included them in both data streams.

Left		Right	
Opp_ID (Left Key)	**Customer Name**	**Opp_id (Right Key)**	**Education Level**
1	Alice	1	High School
2	Amy	1	Secondary School
2	Bob	2	Bachelors
3	Claire	3	Masters
4	David	5	PhD

After performing the full outer join based on the matching the keys, the recipe produces the following target dataset.

Opp_ID (Left Key)	Name	Opp_id (Right Key)	City
1	Alice	1	High School
1	Alice	1	Secondary School
2	Amy	2	Bachelors
2	Bob	2	Bachelors
3	Claire	3	Masters
4	David	*null*	*null*
null	*null*	5	PhD

All rows—matched or not—are included in the target. Because of multiple matches, you see two records with Opp_ID 2 (Amy and Bob) and two more records with Opp_id 1 (High School and Secondary School). Opp_ID 4 and Opp_id 5 are also included despite not having a match.

By including unmatched left records, we now know that we have to find out David's education level to complete this dataset. By including unmatched right records, we see that no customers have a PhD.

Considerations When Using Joins

Unlike a lookup, a join creates a separate record for each match in the target dataset when multiple rows match. Before using a join, ensure that you understand the implications of duplicate rows.

Review the following tips.

Don't double count measures when aggregating records from a join.

Consider the following two data streams that feed the recipe's target dataset. Both input data streams have duplicate key values.

Left

Opp_ID (Left Key)	Customer Name	Opp Amount
1	Alice	1000
2	Amy	20000
2	Bob	9000
3	Claire	6000

Right

Opp_id (Right Key)	Education Level
1	High School
1	Secondary School
2	Bachelors
3	Masters

A left join duplicates the Opp_ID 1 record in the left data stream because it has multiple matches in the right data stream.

Opp_ID (Left Key)	Customer Name	Opp Amount	Opp_id (Right Key)	Education Level
1	Alice	1000	1	High School
1	Alice	1000	1	Secondary School
2	Amy	20000	2	Bachelors
2	Bob	9000	2	Bachelors
3	Claire	6000	3	Masters

Notice that the duplicated records repeat the opportunity amount for Alice. If you added all opportunity amounts to get the total, you'd double count the amount for Alice. To prevent duplicate records, use a lookup instead of a join.

Refrain from using joins when the join keys have a many-to-many relationship.

When the join keys have a many-to-many relationship, the target dataset can become significantly larger than the input data streams. For instance, if four records on the left and five records on the right have the same key value, the join adds 20 (4*5) records to the target dataset. In a more extreme case, if 10,000 records on the left and 5,000 on the right share the same key value, the join creates 50 million records in the target dataset.

To illustrate why this occurs, consider the following two data streams that feed the recipe's target dataset. Both input data streams have duplicate key values.

Left			Right	
Opp_ID (Left Key)	**Customer Name**		**Opp_id (Right Key)**	**Education Level**
1	Alice		1	High School
1	Amy		1	Secondary School
1	Bob		1	Bachelors
1	Claire		1	Masters
			1	PhD

A left join duplicates each record in the left data stream five times because it has five matches in the right data stream.

Opp_ID (Left Key)	Customer Name	Opp_id (Right Key)	Education Level
1	Alice	1	High School
1	Alice	1	Secondary School
1	Alice	1	Bachelors
1	Alice	1	Masters
1	Alice	1	PhD
1	Amy	1	High School
1	Amy	1	Secondary School
1	Amy	1	Bachelors
1	Amy	1	Masters
1	Amy	1	PhD
1	Bob	1	High School
1	Bob	1	Secondary School
1	Bob	1	Bachelors
1	Bob	1	Masters
1	Bob	1	PhD
1	Claire	1	High School
1	Claire	1	Secondary School
1	Claire	1	Bachelors
1	Claire	1	Masters
1	Claire	1	PhD

To prevent duplicate records, use a lookup instead of a join. If you must use a join, try adding more key fields to make the keys have more unique values.

Transform Node: Transform the Data Before Loading It Into a Target

Use a Transform node to manipulate data based on the transformations added to the node. You can add one or more transformations to each Transform node. For example, you can add transformations to concatenate two columns, standardize the formats in a date column, and detect the sentiments of comments in a Feedback column. Each transformation modifies the data in a unique way. To manipulate data at multiple stages of a recipe, add a separate Transform node at each stage.

Watch a Demo: ⊙ Prep Your Data with the Transform Node (English Only)

1.
   In the recipe, select the Add Node button (➕) between two nodes or at the end of the recipe. To show the Add Node button between two nodes, hover the cursor on the connecting line.

2. In the Add Node dialog box, select **Transform**.
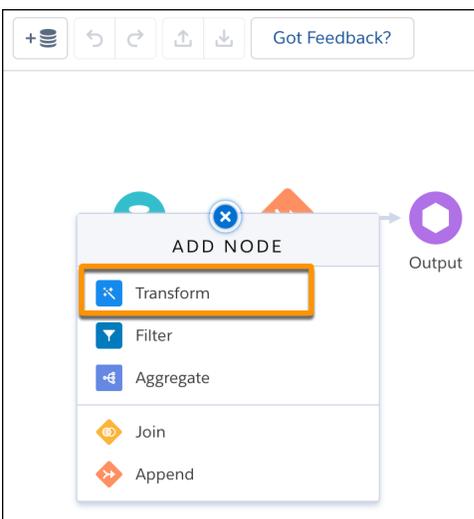
3. In the Preview tab of the details section, select one or more columns for which you want to transform their data.

4. Use the Transform toolbar (1) to add a transformation based on the selected column(s).
   Each time you add a transformation to the Transform node, it appears in the left panel (2). We refer to each card in the left panel as a *step*.

5. To preview the results of a transformation in the Preview tab, select the step in the left panel.

6. To view the Graph area, click the Collapse button ( ).

7. If needed, select a node in the graph to preview the results of the node.

It's always a great idea to preview results to ensure that each node and transformation returns expected results.

8. If needed, add more transformations to the Transform node.

9. Save the recipe.

SEE ALSO:

[Transformations for Data Prep Recipes](#)

Output Node: Write Recipe Results to a Dataset or External System

Use Output nodes with Data Prep recipes to write recipe results to a target, such as a dataset or an Amazon S3 connection. A recipe can have only one Output node. If the recipe writes to a dataset, you can change the API name and label of the dataset in the Output node. If an existing dataset uses the specified API name, the recipe overwrites the existing dataset when it runs.

1.
In the recipe, select the Add Node button ( ) at the end of the recipe.

2. In the Add Node dialog box, select **Output**.
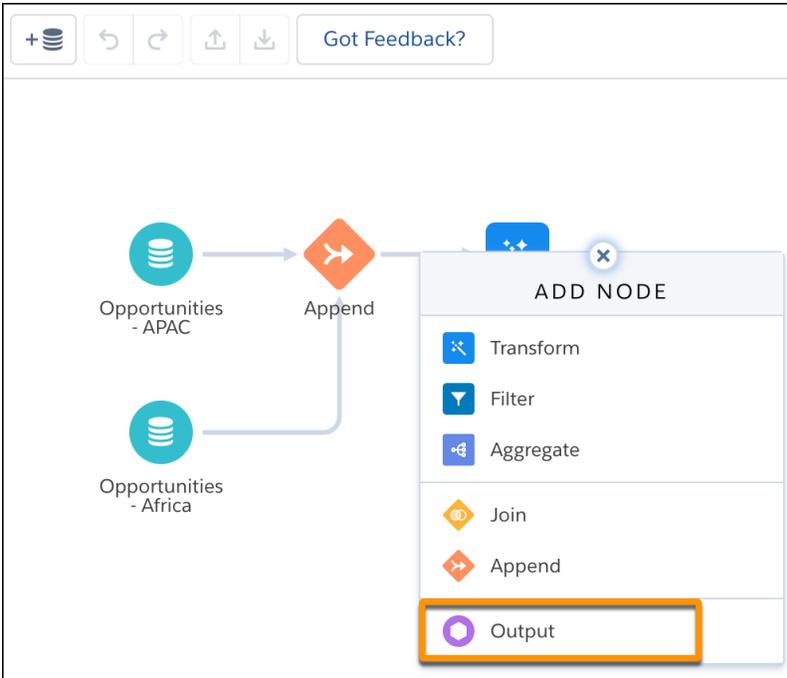
3. Fill in the target details.

For example, you can fill in the label, API name, and app of the dataset.

> **Note:** The Dataset API Name begins with a letter, contains alphanumeric characters and underscores, is fewer than 80 characters, is case-sensitive , and doesn't end with an underscore or contain 2 consecutive underscores.

4. Click **Apply**.

5. Save the recipe.

Run the recipe to create the dataset. As you create more datasets, keep in mind that your org has a row limit for all datasets in your org.

Recipe Output to .csv

Use the Data Prep output node to save your prepared data from Tableau CRM as a data and schema file in a Salesforce BPO (storage database). Then, you can download the data locally using the public API as a .csv file. This data push lets you transfer prepared, cleaned, and improved data from Tableau CRM into your systems for analysis, storage, or to inform business processes.

Amazon S3 Output Connection (Beta)

Create a remote connection using the Amazon S3 output connector to write CSV data from Tableau CRM to an Amazon S3 bucket when using Data Prep.

Salesforce Output Connection (Beta)

Push your prepared dataset data into any Salesforce org from Tableau CRM with the Tableau CRM Salesforce Output connector and Data Prep. With your prepared data back in Salesforce, you can integrate external data, apply the suite of Salesforce automation tools to act on the data, and allow non-Tableau CRM users access to the data through reports and dashboards.

Snowflake Output Connection

Create a remote connection using the Snowflake output connector to write data from Tableau CRM to a Snowflake table. Then, use Sync Out to push raw data from Tableau CRM to Snowflake, or a Data Prep recipe output node to push transformed data.

Tableau Online Output Connection (Beta)

Push your prepared data from Tableau CRM into Tableau Online with the Tableau Online Output connector and Data Prep recipes. Your transformed, merged, and cleaned data is pushed to Tableau Online as a .hyper file for further analysis. For example, prepare historical Opportunity data for struggling products in Tableau CRM, and then use Tableau Online's analytics tools to identify helpful trends to optimize deals.

SEE ALSO:

Run a Recipe

Tableau CRM Limits

Set Up Dataset Security to Control Access to Rows

Recipe Output to .csv

Use the Data Prep output node to save your prepared data from Tableau CRM as a data and schema file in a Salesforce BPO (storage database). Then, you can download the data locally using the public API as a .csv file. This data push lets you transfer prepared, cleaned, and improved data from Tableau CRM into your systems for analysis, storage, or to inform business processes.

Push Data to the Salesforce Database

1. Build your recipe with Data Prep and add an output node.

2. Select to write to **CSV**.

3. Enter the user name of the user who downloads the file.

4. Optionally, select to **Specify CSV partition settings** to set the maximum file size and maximum number of rows per partitioned section of the CSV file.

5. Select **Apply** and save the recipe.

Pull Data from the Salesforce Database

When the recipe runs, Tableau CRM writes the output to the Salesforce database. To retrieve the data from Salesforce, use a tool such as Workbench to find the dataset ID and the Salesforce CLI to download the file. Talk to your Salesforce developer about using these tools and consult this Salesforce blog for a similar process.

.csv Output Considerations

Keep these behaviors in mind when working with the .csv output from the Data Prep recipe output node.

- Output connections are only available for recipes built with Data Prep.

- Data stored is automatically removed after 48 hours.

- If the data is larger than the maximum file size or number of rows specified in the output node's partition settings, there will be more than one .csv file. All partitioned .csv files can function independently, but you can download and merge the files to recreate the dataset.

Amazon S3 Output Connection (Beta)

Create a remote connection using the Amazon S3 output connector to write CSV data from Tableau CRM to an Amazon S3 bucket when using Data Prep.

> **Note:** As a beta feature, the Amazon S3 output connector is a preview and isn't part of the "Services" under your master subscription agreement with Salesforce. Use this feature at your sole discretion, and make your purchase decisions only on the basis of generally available products and features. Salesforce doesn't guarantee general availability of this feature within any particular time frame or at all, and we can discontinue it at any time. This feature is for evaluation purposes only, not for production use. It's offered as is and isn't supported, and Salesforce has no liability for any harm or damage arising out of or in connection with it. All restrictions, Salesforce reservation of rights, obligations concerning the Services, and terms for related Non-Salesforce Applications and Content apply equally to your use of this feature. You can provide feedback and suggestions for the Amazon S3 output connector in the Einstein Analytics Pilot and Beta Connectors group in the Trailblazer Community.

> **Note:** If you want to sync data from an Amazon S3 bucket to Tableau CRM, use an Amazon S3 Connection on page 659 instead.

How Tableau CRM Output Connectors Work

Tableau CRM connectors allow you to write the outcome of a recipe to an external system for further analysis, business automation, and storage. After you configure an output connector to the data source and folder path using the steps in this guide, create a recipe using Data Prep. Add an Output node to the recipe and select the Output connection. When the recipe runs, Tableau CRM creates a folder based on the name of the specified folder path within the directory. For example, if you chose the folder path `...support/servicedata`, a new folder *servicedata* is created at the path `...support/servicedata/servicedata`. The new folder contains these files:

- The output dataset CSV file, with the naming standard *folderpath_0.csv*. In our example, this is *servicedata_0.csv*. For larger datasets, Tableau CRM automatically partitions the data into CSVs with increasing numbers appended to the name. In our example, there could be a *servicedata_1.csv*, a *servicedata_2.csv*, and more until all data is written. This file doesn't have a header row.

- The metadata file *schema.json*, with information on each column including the name, data type, and precision.

- The _SUCCESS file, generated when the CSV is completely written.

When you run the recipe again, the files previously generated are deleted and the new version of the files are written.

Enable the Amazon S3 Output Connector

1. From Setup, enter `Analytics` in the Quick Find box.

2. Select **Settings** under **Analytics**.

3. Select **Enable Amazon S3 output connection** and **Save**.

Create the Output Connection

1. In Tableau CRM, click the gear icon ( ) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Output Connections**.

5. Click **Add Connection**.

6. Click the connector's icon and enter its settings, as described in the Connection Settings section.

7. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. The developer name can't include spaces, and you can't change it after you create the connection.
Description	Description of the connection for internal use.
Secret Key	Your Amazon secret access key.
Region Name	Optional setting. Region of your S3 service. The default region is US East(N. Virginia).
Folder Path	Path to the folder that you want to connect to. The path must start with the bucket name and can't include the name of the subfolder whose data you want to sync.
Access Key	Your Amazon S3 bucket access key ID.

Amazon S3 Output Connector Considerations

Keep these behaviors in mind when working with the Amazon S3 output connector.

- Each output connection can be used once per recipe.
- Output connections are only available for recipes built with Data Prep.
- Up to 10 GB of data are written externally per run of a recipe per output connector. Up to 50 GB of data can be written per 24-hour period.
- Each output connection only supports one folder path. To output to another folder path, create another output connection.
- When the prior run's files are deleted in preparation for the current run, the earlier version of an output dataset is inaccessible. Set up a process to copy or use the output after each run, using the _SUCCESS file as an indication that the write is complete.

SEE ALSO:

Output Node: Write Recipe Results to a Dataset or External System

Salesforce Output Connection (Beta)

Push your prepared dataset data into any Salesforce org from Tableau CRM with the Tableau CRM Salesforce Output connector and Data Prep. With your prepared data back in Salesforce, you can integrate external data, apply the suite of Salesforce automation tools to act on the data, and allow non-Tableau CRM users access to the data through reports and dashboards.

> 📝 **Note:** As a beta feature, the Salesforce Output Connector is a preview and isn't part of the "Services" under your master subscription agreement with Salesforce. Use this feature at your sole discretion, and make your purchase decisions only on the basis of generally available products and features. Salesforce doesn't guarantee general availability of this feature within any particular time frame or at all, and we can discontinue it at any time. This feature is for evaluation purposes only, not for production use. It's offered as is and isn't supported, and Salesforce has no liability for any harm or damage arising out of or in connection with it. All restrictions, Salesforce reservation of rights, obligations concerning the Services, and terms for related Non-Salesforce Applications and Content apply equally to your use of this feature. You can provide feedback and suggestions for the Salesforce Output Connector in the Tableau CRM Pilot and Beta Connectors group in the Trailblazer Community.

> 📝 **Note:** Use the Salesforce External Connection on page 621 to sync data from another Salesforce org to Tableau CRM.

How the Tableau CRM Salesforce Output Connector with Output Node Works

Tableau CRM Salesforce Output connectors allow you to write the outcome of a recipe to any Salesforce org for further analysis, business automation, and expanded access. After you configure an output connector to the Salesforce org, create a recipe using Data Prep. Add an Output node to the recipe, select the Salesforce Output connection, and configure the node with the object, mapping, and other settings. When the recipe runs, Tableau CRM can INSERT, UPDATE, or UPSERT the output dataset to the currently supported objects: Account, Opportunity, and custom objects.

Enable the Salesforce Output Connector

1. From Setup, enter *Analytics* in the Quick Find box.

2. Select **Settings** under **Analytics**.

3. Select **Enable Salesforce output connection** and **Save**.

Create the Output Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Output Connections**.

5. Click **Add Connection**.

6. Click the connector's icon and enter its settings, as described in the Connection Settings section.

7. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted

Setting	Description
	through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Username	User name used to connect to external Salesforce org for sync of objects and fields.
Service URL	Login URL for the external org. This is completed for you, but you can change it if the org is a sandbox or it uses a My Domain name or custom domain. Enter in the format: `<http or https>://<MyDomainName.my or CustomDomain>.salesforce.com/services/Soap/<Identifier>/<Version Number>.0` For example: `https://login.salesforce.com/services/Soap/u/34.0`
Password	Password for the user specified in *Username*. Depending on your security settings, append the user's security token to the end of the password. For more information about obtaining a security token, see Reset Your Security Token.

Push Data to Salesforce

With the Salesforce Output Connector configured, build a Data Prep recipe that merges and transforms the data to push to Salesforce. Add an output node and configure it to use the Salesforce Output connector.

1. Select to write to an **Output Connection** (1).

2. Select the connection name of the Salesforce Output connection you created (2).

3. Select the destination Salesforce object (3). Account, Opportunity, and custom objects are supported. Objects available to the User ID used to configure the connector are listed.

4. Choose whether to INSERT (add new records), UPDATE (change data of existing records), or UPSERT (add new records and update existing records). (4)

5. Map recipe columns to their equivalent Salesforce object columns. All columns must be mapped to apply the output node. Id is only mapped for UPDATE and UPSERT.

6. Select **Apply**.

7. Save the recipe.

Salesforce Output Connector Considerations

Keep these behaviors in mind when working with the Salesforce output connector and using a Data Prep recipe output node.

- To troubleshoot unsuccessful or partial failed runs, see Investigate Unsuccessful Salesforce Output Connector Runs on page 742.

- Each output connection can be used one time per recipe.

- Output connections are only available for recipes built with Data Prep.

- Up to 1 GB, or 1 million rows, of data is written externally per run of a recipe per output connector. Up to 1 GB, or 1 million rows, of data can be written per 24-hour period.

- This connector supports INSERT, UPDATE, and UPSERT to the Account, Opportunity, and custom Salesforce objects.

- This connector doesn't support objects that are the parent or child in a parent-child relationship. If there are two custom objects with a dependency, manually insert the header record, get the header ID, then insert the detail record with those details.

Investigate Unsuccessful Salesforce Output Connector Runs

The Tableau CRM Salesforce Output Connector pushes data from Tableau CRM into a Salesforce org as a Bulk API job. If a portion of the data push fails, use Salesforce Setup and the Workbench to troubleshoot.

Investigate Unsuccessful Salesforce Output Connector Runs
The Tableau CRM Salesforce Output Connector pushes data from Tableau CRM into a Salesforce org as a Bulk API job. If a portion of the data push fails, use Salesforce Setup and the Workbench to troubleshoot.

1. In the org that initiated the job, click **Setup.**

2. Enter `Bulk Data` in the `Quick Find` box, then select **Bulk Data Load Jobs**.

3. Select the job corresponding to your data push.
   The information describing the outcome of your run including processing time, object written to, and more, are displayed.

4. Record the Job ID and API Version.

5. Review the Records Failed column of the Auto-Chunked Batches section.

Auto-Chunked Batches

View Request	View Result	Batch ID	Start Time	End Time	Total Processing Time (ms)	API Active Processing Time (ms)	Apex Processing Time (ms)	Records Processed	Records Failed	Retry Count	State Message	Status
		7511F000005BKVV	3/8/2021 6:08 PM	3/8/2021 6:08 PM	0	0	0	0	0	0		Completed
		7511F000005BKVa	3/8/2021 6:08 PM	3/8/2021 6:08 PM	99	43	0	9	1	0		Completed

If any records failed, investigate further in Workbench. Rows fail because the row's data wasn't able to create or update a record in Salesforce. This can happen for a number of reasons, including incorrect date formatting or a mismatch between the source row's and record field's datatype.

6. Sign into Workbench using the credentials for your Salesforce org.

7. Click **utilities**.

8. Click **REST Explorer**.

9. To find failed rows, create the URI based on the format
   `/services/data/vXX.X/jobs/ingest/jobID/failedResults/` described in Get Job Failed Record Results. Fill in your API version into "vXX.X", and your Job ID into "jobID".

For example, `/services/data/v51.0/jobs/ingest/751B000000JLPLt/failedResults/`

10. Click **Execute**.

Within the raw response there is a representation of your failed rows at the bottom starting with `sf__Id`. The first text after the row names is your error reason. For example, here's a raw response.

```
"sf__Id","sf__Error",Id,Account__c,Checkbox__c,Date__c,Text__c,Currency__c
"","ENTITY_IS_DELETED:entity is
deleted:--","a1h5G0000000OhEXTJ","003M000000YAEHJLA5","false","","","41040.0"
```

The error component is "ENTITY_IS_DELETED:entity is deleted".

11. To find successful rows, create the URI based on the instructions in Get Job Successful Record Results.

For more information on the Salesforce Bulk API and Workbench check out the Use Bulk API Trailhead module.

Snowflake Output Connection

Create a remote connection using the Snowflake output connector to write data from Tableau CRM to a Snowflake table. Then, use Sync Out to push raw data from Tableau CRM to Snowflake, or a Data Prep recipe output node to push transformed data.

> 📝 **Note:** Use Sync Out for Snowflake on page 695 to push raw data from Tableau CRM to Snowflake after setting up the connection. If you want to sync data from Snowflake to Tableau CRM, use the Snowflake Connection on page 652 instead.

How Tableau CRM Output Connectors with Output Nodes Work

Tableau CRM connectors allow you to write the outcome of a recipe to an external system for further analysis, business automation, and storage. After you configure an output connector to the data source, create a recipe using Data Prep. Add an Output node to the recipe, select the Output connection, and choose the Snowflake table name from the list of objects. When the recipe runs, Tableau CRM writes the output dataset to the selected table. When you run the recipe again, the data previously written is deleted and the new data is written.

Enable the Snowflake Output Connector

1. From Setup, enter *Analytics* in the Quick Find box.

2. Select **Settings** under **Tableau CRM**.

3. Select **Enable Snowflake output connection** and **Save**.

Create the Output Connection

1.
In Tableau CRM, click the gear icon (⚙️) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Output Connections**.

5. Click **Add Connection**.

6. Click the connector's icon and enter its settings, as described in the Connection Settings section.

7. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Schema	Snowflake schema name.
Password	Password for your Snowflake account.
Database	Snowflake database name.
Role	Optional setting. Snowflake role assigned to the user that you're using to connect.
Warehouse	Snowflake warehouse name.
Username	User name for the Snowflake account.
Account	Name of your Snowflake account.
	💡 Tip: The account name is the first segment in the domain in your Snowflake URL. For example, `123abc` is your account name in `https://123abc.snowflakecomputing.com`.

Push Data to Snowflake

With the Snowflake Output Connector configured, you have two options to push data to Snowflake from Tableau CRM.

- To push raw data, without augmentation or transformation, use Sync Out for Snowflake on page 695. Data is pushed with each Data Sync run. You don't use a Data Prep recipe.
- To push augmented and transformed data, build a Data Prep recipe and configure the output node on page 734 to use the Snowflake output connector.

Snowflake Output Connector Considerations

Keep these behaviors in mind when working with the Snowflake output connector and using a Data Prep recipe output node.

- Each output connection can be used one time per recipe. To push again from the same recipe, add another connection with the same credentials
- Output connections are only available for recipes built with Data Prep.
- Up to 10 GB, or 10 million rows, of data is written externally per run of a recipe per output connector. Up to 50 GB, or 50 million rows, of data can be written per 24-hour period.

- When the prior run's data is deleted in preparation for the current run, the earlier version of an output dataset is inaccessible. Set up a process to copy or use the output after each run.

SEE ALSO:

[Output Node: Write Recipe Results to a Dataset or External System](#)

Tableau Online Output Connection (Beta)

Push your prepared data from Tableau CRM into Tableau Online with the Tableau Online Output connector and Data Prep recipes. Your transformed, merged, and cleaned data is pushed to Tableau Online as a .hyper file for further analysis. For example, prepare historical Opportunity data for struggling products in Tableau CRM, and then use Tableau Online's analytics tools to identify helpful trends to optimize deals.

> **Note:** As a beta feature, the Tableau Online Output Connector is a preview and isn't part of the "Services" under your master subscription agreement with Salesforce. Use this feature at your sole discretion, and make your purchase decisions only on the basis of generally available products and features. Salesforce doesn't guarantee general availability of this feature within any particular time frame or at all, and we can discontinue it at any time. This feature is for evaluation purposes only, not for production use. It's offered as is and isn't supported, and Salesforce has no liability for any harm or damage arising out of or in connection with it. All restrictions, Salesforce reservation of rights, obligations concerning the Services, and terms for related Non-Salesforce Applications and Content apply equally to your use of this feature. You can provide feedback and suggestions for the Tableau Online Output Connector in the [Tableau CRM Pilot and Beta Connectors group](#) in the Trailblazer Community.

How the Tableau Online Output Connector Works

The Tableau Online Output connector, from Tableau CRM, allows you to write the outcome of a recipe to a Tableau Online project as a .hyper file. After you configure an output connector to Tableau Online, create a recipe using Data Prep. Add an Output node to the recipe, select the Tableau Online Output connection, and configure the node with the object (Tableau Online project). When the recipe runs, Tableau CRM writes the output dataset file with the name Extract.hyper to the selected project. When the recipe runs again we overwrite the Extract.hyper file, but you can view the historical versions of files in Tableau Online.

Tableau Online Credential Permission Requirements

The credentials used to create the Tableau Online output connection must use the Creator license. The Creator role provides needed access.

Enable the Tableau Online Output Connector

1. From Setup, enter `Analytics` in the Quick Find box.

2. Select **Settings** under **Analytics**.

3. Select **Enable Tableau Online output connection** and **Save**.

Create the Output Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Output Connections**.

5. Click **Add Connection**.

6. Click the connector's icon and enter its settings, as described in the Connection Settings section.

7. When you finish entering the settings, either click **Save** or **Save & Test**. Save & Test validates your settings by attempting to connect to the source. If the connection fails, Tableau CRM shows possible reasons.

Connection Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name can't include spaces. The API name is used in your dataflows to reference data extracted through this connection. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Server URL	The base URL for your Tableau Online server, including the pod name. To isolate the Server URL, sign into Tableau Online and pull the bolded portion of your URL: **https://us-west-5a.online.tableau.com**/#/site/exampleurl5a/home
Username	User name for the Tableau Online account.
Content URL	The subpath portion of your Tableau Online URL. To isolate the Content URL, sign into Tableau Online and pull the bolded portion of your URL: https://us-west-5a.online.tableau.com/#/site/**exampleurl5a**/home
Password	Password for your Tableau Online account.

Push Data to Tableau Online

With the Tableau Online Output Connector configured, build a Data Prep recipe that merges and transforms the data to push to Tableau Online. Add an output node and configure it to use the Tableau Online Output connector.
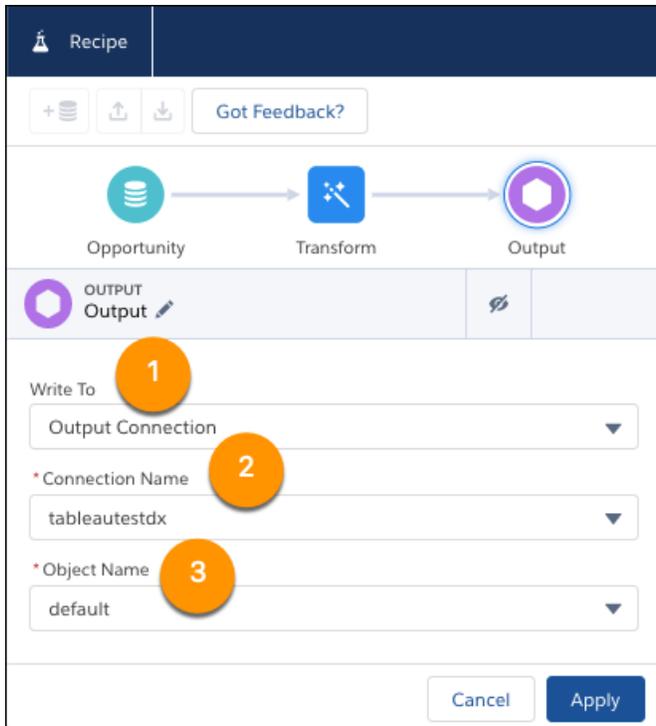
1. Select to write to an **Output Connection** (1).

2. Select the connection name of the Tableau Online Output connection you created (2).

3. Select the destination Tableau Online project as the Object Name (3). Projects available to the username used to create the output connector are listed.

4. Select **Apply**.

5. Save the recipe.

Tableau Online Output Connector Considerations

Keep these behaviors in mind when working with the Tableau Online output connector and using a Data Prep recipe output node.

- Each output connection can be used one time per recipe. To push again from the same recipe, add another connection with the same credentials

- Output connections are only available for recipes built with Data Prep.

- Up to 1 GB, or 1 million rows, of data is written externally per run of a recipe per output connector. Up to 1 GB, or 1 million rows, of data can be written per 24-hour period.

- The Free Trial version of Tableau Online isn't supported by this connector.

- This connector requires the presence of your Tableau Online default project.

- If a job run takes longer than 12 hours, the Tableau Online security token may reset and cancel the run. Before the 12th hour, to prevent reset of the security token, sign into Tableau Online with the credentials used to configure the connector.

Transformations for Data Prep Recipes
Data Prep provides transformations that allow you to prepare, clean, and transform your data. For example, you can use a transformation to create a calculated column based on a formula. You add transformations inside a Transform node. You can string together multiple transformations to manipulate data sequentially.

Bucket Transformations: Categorize Column Values

With Data Prep, you can bucket your measure, dimension, and date fields to group and organize data.

Cluster Transformation: Segment Your Data (Pilot)

Use the Cluster transformation in a Data Prep recipe to segment rows of data into distinct clusters based on common characteristics. For example, you can cluster accounts based on account industry, number of employees, rating, and annual revenue. Using the clusters, you can identify products and services to upsell to each account based on other accounts in the same cluster.

Data Type Conversion Transformations: Convert Column Types

The data type of a dataset column determines how you can query that column's data. For example, you can filter and group by a dimension or date column, or perform math calculations on a measure column. When you load data into a dataset, Tableau CRM sometimes tags a dataset field with the wrong type. If needed, use the column-type conversion transformations in a Data Prep recipe to convert columns to the correct types.

Date and Time Transformations: Calculate on Dates Fields

Your date fields are key to understanding trends over time or keeping teams aware of upcoming milestones and deadlines. Use the Date and Time transformations to calculate based on dates or make them more helpful to your analysis. With the `Now` option, you can insert a column with the current date and time in a specified format. Use `Date Difference` to calculate the duration between two selected date columns as days, months, or years. And use the `Add or Subtract Days or Months` function to add or subtract days or months from a date column.

Detect Sentiment Transformation: Determine the Sentiment of Text

You can find valuable information in text fields, such as product reviews and social media posts. Use the Detect Sentiment transformation in a Data Prep recipe to quickly categorize that information into sentiments: positive, negative, and neutral. For example, detect the sentiment of survey responses to evaluate how customers feel about your product support. If more than a certain percentage—say 30%—of the comments are negative, escalate the feedback to support management.

Drop Columns Transformation: Drop Columns from the Recipe

Drop unwanted columns from a Data Prep recipe. For example, you can add a Drop Columns transformation after a Formula transformation to drop input columns used for a calculated column.

Discovery Predict Transformation: Get Einstein Discovery Predictions

Use the Discovery Predict transformation to populate your datasets with predictive and prescriptive intelligence. When you run a Data Prep recipe with a Discovery Predict node, Einstein calculates and saves predicted outcomes on a row-by-row basis. You can optionally store descriptions of top predictors and improvements. With the Discovery Predict node, you can quickly evaluate predictions across a large set of data, assess multiple models before deploying them into production, and aggregate this information in a dashboard.

Extract Transformation: Get a Date Component

Use the Extract transformation in a Data Prep recipe to pull a selected component from a date field into a new field. For example, extract the hour component from the Case Created Date column to analyze case creation by hour of the day.

Edit Attributes Transformation: Change the Column Names and Value Formats

To make column names more descriptive and apply consistent formats to column values, use the Edit Attributes transformation in a Data Prep recipe. You can set the labels and API names of all columns. You can also set the precision and scale for measure columns, character length for dimension columns, and date formats for date columns.

Flatten Transformation: Flatten Hierarchies

The Flatten transformation flattens hierarchical data. For example, you can flatten the Salesforce role hierarchy to implement row-level security on a dataset based on the role hierarchy.

Format Dates Transformation: Standardize the Date Format in a Column

If a dimension column contains dates in different formats, use the Format Dates recipe transformation in a Data Prep recipe to standardize the format for all dates in the column. A consistent format enables you to correctly filter and group records by date, including filtering by date component, such as month. It also ensures that you can successfully convert the column type from dimension to date. Optionally, you can convert the column type from Dimension to Date.

Formula Transformation: Create a Calculated Column Based on an Expression

Create a column in a Data Prep recipe that displays values based on a formula calculation. The calculation can include input from other fields in the same row or across rows. For example, you can create a Profit column based on input from Revenue and Cost columns. Enter formulas in EA-SQL format. EA-SQL is a collection of standard and custom functions for numeric, string, and date data.

Predict Missing Values Transformation: Fill In Missing Values

Use the Predict Missing Values transformation in a Data Prep recipe to complete your data by filling in missing values in a dimension column. Tableau CRM intelligently predicts values based on values in other strongly correlated columns in your data.

Split Transformation: Break Up Column Values

You can split the strings in a dimension column into two values by specifying a delimiter. To split column values into more than two parts, add multiple instances of the Split transformation. For instance, you can use 3 Split transformations in a Data Prep recipe to split the full address into the following components: street address, city, state, and zip code.

Time Series Forecasting Transformation: Forecast Measures (Pilot)

Make decisions today based on forecasts about tomorrow with time series forecasting. A time series forecast takes an ordered series of points and then intelligently forecast what the next values will be. For example, estimate units sold for the next four quarters based on the last 5 years of sales. Use the Time Series Forecasting transformation in a Data Prep recipe to run forecasts based on historical data.

Bucket Transformations: Categorize Column Values
With Data Prep, you can bucket your measure, dimension, and date fields to group and organize data.

Categorize Date Column Values into Buckets

Use a bucket column in a Data Prep recipe to categorize and organize date values. For example, you can create buckets Current Period and Past Periods to group dates.

Categorize Dimension Column Values into Buckets

Use a bucket column in a Data Prep recipe to categorize and organize text values. For example, you can create buckets West, Central, and East to group regions.

Categorize Measure Column Values into Buckets

Use a bucket column in a Data Prep recipe to categorize and organize numeric values. For example, you can create buckets High, Middle, Low to group numbers of cases per account.

Categorize Date Column Values into Buckets
Use a bucket column in a Data Prep recipe to categorize and organize date values. For example, you can create buckets Current Period and Past Periods to group dates.

1. In the Transform node of a Data Prep recipe, select the date column in the Preview tab.

2. In the Transform toolbar, click the Bucket button ( ) to define the buckets.

3. Select whether to use absolute or relative dates.
   Absolute dates are defined dates on a calendar, like July 25, 2023. Relative dates are a time period in relation to the day the data is viewed, like last quarter, or next month.

USER PERMISSIONS
To manage and create a recipe:
• Edit Analytics Dataflows OR Edit Dataset Recipes

4. Select ➕.

5. Enter a label for the first bucket.
   The first bucket is for the earliest date range. Each subsequent bucket must have a later date range than the previous bucket.

6. Select the date range for this bucket.

7. Click **Add**.

8. To create another bucket with a later date range, click ➕ and specify the details.

9. To place null values into their own bucket, select **Bucket null values**.

10. Under Show Results In, select whether the new bucket values should appear in a new column and what should happen to the original column.

11. If you elect to create a column, set its label under Column Label.

12. Click **Apply** to add the transformation to the Transform node.
The Preview tab shows the new bucket column.

13. To view the Graph area, click the Collapse button ( ).

14. Save the recipe.

After you run the recipe, you can view the bucket assigned to each row in the bucket column.

Categorize Dimension Column Values into Buckets
Use a bucket column in a Data Prep recipe to categorize and organize text values. For example, you can create buckets West, Central, and East to group regions.

1. In the Transform node of a Data Prep recipe, select the dimension column in the Preview tab.

2. In the Transform toolbar, click the Bucket button ( ) to define the buckets.

3. Select .

4. Enter a label for the bucket.

Define Range 1 ✕

Bucket Label

Enter label...

Values Included in Bucket

Search or manually enter values... 🔍

Cancel　　Add

5. In the Values Included in Bucket box, search for and select the values to be assigned to this bucket. For values not found in search, manually enter the value and press **Enter**.

6. Click **Add**.

7. To create more buckets, click ➕ and select a label and the values to assign to the bucket.

8. To place null values into their own bucket, select **Bucket null values**.

9. Under Show Results In, select whether the buckets should appear in a new column and what should happen to the original column.

10. If you elect to create a column, set the label under Column Label.

11. Click **Apply** to add the transformation to the Transform node.
The Preview tab shows the new bucket column.

12. To view the Graph area, click the Collapse button (⬙).

13. Save the recipe.

After you run the recipe, you can view the bucket assigned to each row in the bucket column.

Categorize Measure Column Values into Buckets
Use a bucket column in a Data Prep recipe to categorize and organize numeric values. For example, you can create buckets High, Middle, Low to group numbers of cases per account.

1. In the Transform node of a Data Prep recipe, select the measure column in the Preview tab.

2. In the Transform toolbar, click the Bucket button (🗲) to define the buckets.

3. For Range 1, enter a value and label for the first bucket. Rows with values below the Range 1 value are assigned to this bucket.

4. If your bucket column has more than two buckets, click **Add Range>>** to add a middle range.

5. For each additional middle range that you add, enter the range of values and the bucket label.

Rows with values greater than or equal to the first value and less than the second value are assigned this bucket.

6. For the final range, enter a value and bucket label.
Rows with values greater than or equal to this value are assigned to this bucket.

Bucket (Probability (%))

Organizes values into buckets based on
ranges. For example, organize values
below 100 in a Low Yield bucket and
values above 100 in a High Yield
bucket.

Range 1 <

| Low
| 30   ✕ |

Add Range >>

Range 2 -

| Middle
| 30 - 70   ✕ |

Add Range >>

Range 3 >=

| High
| 70   ✕ |

☐ Bucket null values

Cancel   **Apply**

7. To place null values into their own bucket, select **Bucket null values**.

8. Under Show Results In, select whether the buckets should appear in a new column and what should happen to the original column.

9. If you elect to create a column, set the label under Column Label.

10. Click **Apply** to add the transformation to the Transform node.
The Preview tab shows the new bucket column.

11. To view the Graph area, click the Collapse button (⬕).

12. Save the recipe.

After you run the recipe, you can view the bucket assigned to each row in the bucket column.

Cluster Transformation: Segment Your Data (Pilot)
Use the Cluster transformation in a Data Prep recipe to segment rows of data into distinct clusters
based on common characteristics. For example, you can cluster accounts based on account industry,
number of employees, rating, and annual revenue. Using the clusters, you can identify products
and services to upsell to each account based on other accounts in the same cluster.

> 📝 Note: Join the Smart Transform Pilot Program to try out all pilot features in Data Prep recipes.
> Pilot features can change with each release as existing features become generally available
> or are retired and new pilot features are added to the program. We provide the Smart
> Transform Pilot Program to selected customers who agree to specific terms and conditions.
> To be nominated to participate in the program, contact Salesforce. Pilot programs are subject
> to change, and we can't guarantee acceptance. Each Data Prep pilot feature isn't generally
> available unless or until Salesforce announces its general availability in documentation or in
> press releases or public statements. We can't guarantee general availability within any particular
> time frame or at all. Make your purchase decisions only on the basis of generally available

USER PERMISSIONS

To create a recipe:
- Edit Analytics Dataflows
  OR Edit Dataset Recipes

To use pilot features:
- Enable Data Prep Pilot
  Features

-

products and features. You can provide feedback and suggestions for the Smart Transform Pilot Program in the Trailblazer Community.

1. In a Transform node of a Data Prep recipe, select any column in the Preview tab.

2. To add a Cluster transformation to the Transform node, click the Cluster button (🛰️).
   The Cluster panel opens.

3. In Number of Clusters, enter the number of clusters to create.

   The number of clusters must be between 2 and 20, inclusive.

4. In Columns to Cluster, select the measure and dimension columns to determine the clusters.

   The clustering algorithm determines clusters based on the collective set of values from the selected columns. For example, accounts with the same industry and rating can be placed in different clusters if the number of employees and annual revenues don't fall into the same ranges determined by the algorithm.

5. If needed, change the label of the cluster column, which will store the cluster for each row.

   The cluster column in preview shows "Cluster TBD."

6. To add the transformation to the Transform node, click **Apply**.

7. To view the Graph area, click the Collapse button (🔷).

8. Save the recipe.

9. Run the recipe to assign and view a cluster for each row.

   To process billions of rows, the Cluster transformation uses the K-Means clustering algorithm. The clusters are detected dynamically during the recipe job's runtime, but aren't persisted. The clusters can change between recipe runs even if the data and the clustering configuration doesn't change.

Example: For months, you've noticed that some accounts have lots of deals with lower average opportunity amounts and other accounts that have larger deals, but fewer of them. To find ways to drive larger deal sizes and more deals, you decide to cluster your accounts.

You use the Cluster transformation to create three clusters based on the account industry, number of employees, rating, and annual revenue.

Account Cluster	Account Name	Number of Employees	Annual Revenue	Industry ↓	Rating
Cluster 1	US Wirefree	112,327	1,350,000,000	Utilities	Hot
Cluster 1	Tru Vue	71,721	1,110,000,000	Utilities	Cool
Cluster 1	Softura	100,997	1,500,000,000	Utilities	Hot
Cluster 1	Bbk Holding Company	103,478	1,440,000,000	Utilities	Warm
Cluster 2	Internap	38	5,150,000,000	Utilities	Warm
Cluster 2	Vertical Venture DOT Co	153	5,030,000,000	Utilities	Hot
Cluster 2	Digicon Imaging	61	6,530,000,000	Utilities	Warm
Cluster 2	Springer-Miller Systems	8,986	8,630,000,000	Utilities	Cool

While analyzing the clusters in a dashboard, you notice that Cluster 3 has the highest average annual revenue (about $40 billion), but only three accounts are in that cluster.
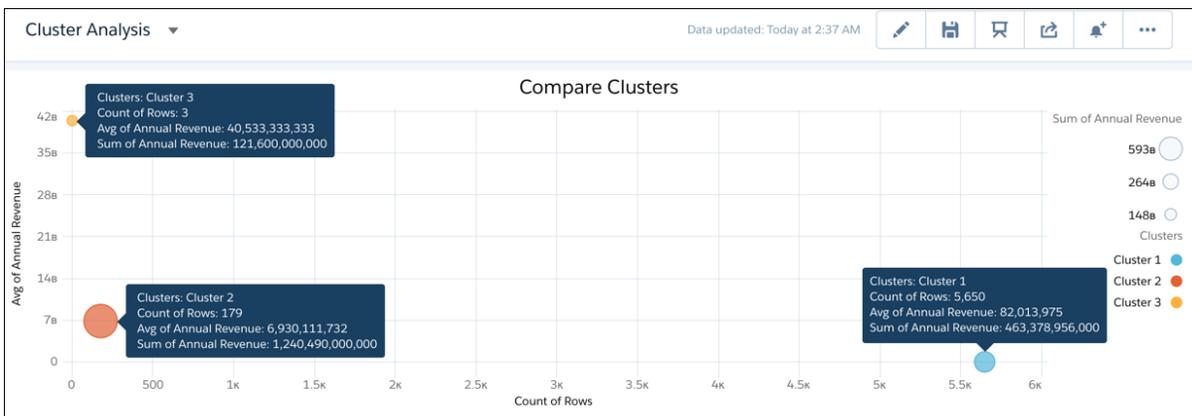
Why doesn't Cluster 3 have more deals? Is it due to repeat business? Let's find out.

Use the Aggregate node to group total spend by account and pivot the results by opportunity type: Add-On Business, New Business, Renewal, and Services.

Looking at spend by opportunity type for Cluster 2, we see lots of repeat business. Great!
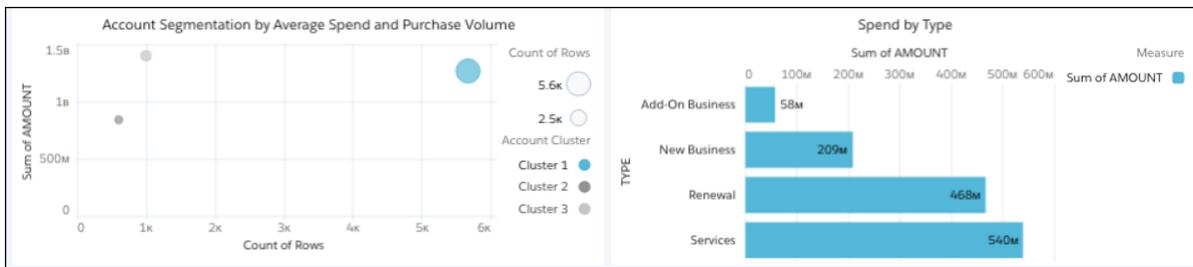
But Cluster 3 has new business only. These accounts buy once and don't return for additional business—hence the lower number of deals.
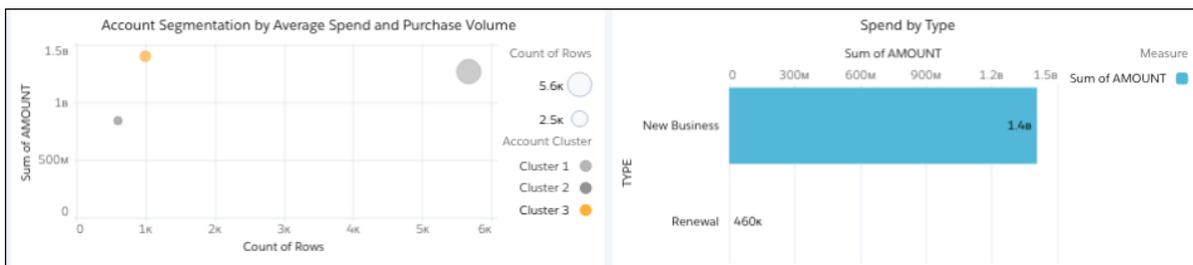
We found gold! To increase the number of deals from Cluster 3 accounts, we can push renewals, services, and add-on business.

Data Type Conversion Transformations: Convert Column Types

The data type of a dataset column determines how you can query that column's data. For example, you can filter and group by a dimension or date column, or perform math calculations on a measure column. When you load data into a dataset, Tableau CRM sometimes tags a dataset field with the wrong type. If needed, use the column-type conversion transformations in a Data Prep recipe to convert columns to the correct types.

Dimension to Date Transformation: Convert the Column Type

When you load data into a dataset, Tableau CRM tags a date column as a dimension if it contains unexpected string values or dates in different formats. To use date functionality, such as grouping by month, use the Dimension To Date transformation to change the Dimension column type to Date.

Dimension to Measure Transformation: Convert the Column Type

When you load data into a dataset, Tableau CRM tags a measure column as a dimension if it contains unexpected strings or characters. To perform a mathematical calculation on the column, such as calculating the average, use the Dimension To Measure transformation to change the Dimension column type to Measure. You can perform mathematical calculations on measure columns only.

Measure to Dimension Transformation: Convert the Column Type

When you load data into a dataset, Tableau CRM tags a dimension column as a measure if it contains only numbers. For example, Tableau CRM tags a numerical Room Number column as a measure, even though it's actually a dimension that identifies rooms. To group and filter by a measure column, use the Measure To Dimension transformation to change a Measure column type to Dimension. You can group and filter by dimension columns only.

Dimension to Date Transformation: Convert the Column Type

When you load data into a dataset, Tableau CRM tags a date column as a dimension if it contains unexpected string values or dates in different formats. To use date functionality, such as grouping by month, use the Dimension To Date transformation to change the Dimension column type to Date.

1. In the Transform node of a Data Prep recipe, select the dimension column that you want to convert to the Date column type.

For example, you standardized the dates in a dimension column and now want to convert the column type to Date.

2. In the Transform toolbar, select **Dimension to** > **Date**.

3. In the Date Format field, select the date format to display the dates.

4. Under Show Results In, select whether the new values appear in a new column and whether to keep the original column.

5. If you elect to create a column, set the label under Column Label.

6. Click **Apply** to add the transformation to the Transform node.
   The Preview tab shows the new date component column.

7. To view the Graph area, click the Collapse button ( ).

8. Save the recipe.

Run the recipe to convert the column type to Date and the date values to the specified format. If the transformation can't parse a date value, it replaces the value with null.

Dimension to Measure Transformation: Convert the Column Type
When you load data into a dataset, Tableau CRM tags a measure column as a dimension if it contains unexpected strings or characters. To perform a mathematical calculation on the column, such as calculating the average, use the Dimension To Measure transformation to change the Dimension column type to Measure. You can perform mathematical calculations on measure columns only.

During the column type conversion, the Dimension to Measure transformation rounds decimals to the nearest whole number. For example, 300.2939 becomes 300.

1. In the Transform node of a Data Prep recipe, select the dimension column that you want to convert to the Measure column type.

2. In the Transform toolbar, select **Dimension to** > **Measure**.

USER PERMISSIONS

To manage and create a recipe:
- Edit Analytics Dataflows OR Edit Dataset Recipes

3. Under Show Results In, select whether the new values appear in a new column and whether to keep the original column.

4. If you elect to create a column, set the label under Column Label.

5. To change more column attributes, click the Column Attributes box, change them, and click **Apply**.

Based on the output column's datatype, you can set these attributes.

Attribute	Description
API Name	The API name of the column. The column API name can each be up to 255 characters. The API name must also be unique within the recipe.

Attribute	Description
Precision	The maximum number of digits in a numeric column. Includes all numbers to the left and to the right of the decimal point (but excludes the decimal point character). Enter an integer between 1 and 18, inclusive. 📝 Note: If a numeric value exceeds the specified precision, the value is set to 0 or null, depending on whether null measure handling is enabled. For example, if precision is set to 5 and null measure handling is enabled, the numeric value 123456.78 is replaced with null.
Scale	The number of digits to the right of the decimal point in a numeric column. Enter an integer between 1 and 17, inclulsive. The scale must be less than the precision. For example, if the numeric value is 123456.789 and you set the scale to 2, the number appears as 123456.78.
Length	The maximum number of characters in a text column. Enter an integer between 1 and 32000, inclusive. The default length is 255 characters.
Date Format	The date format of a date column.

For more guidelines, see the "Field Name Restrictions" section in this help topic.

6. Click **Apply** to add the transformation to the Transform node.
The Preview tab shows the measure column.

7. To view the Graph area, click the Collapse button (￭).

8. Save the recipe.

Run the recipe to convert the column type to measure.

Measure to Dimension Transformation: Convert the Column Type
When you load data into a dataset, Tableau CRM tags a dimension column as a measure if it contains only numbers. For example, Tableau CRM tags a numerical Room Number column as a measure, even though it's actually a dimension that identifies rooms. To group and filter by a measure column, use the Measure To Dimension transformation to change a Measure column type to Dimension. You can group and filter by dimension columns only.

1. In the Transform node of a Data Prep recipe, select the measure column that you want to convert to the Dimension column type.

2. In the Transform toolbar, select **Measure to** > **Dimension**.

USER PERMISSIONS

To manage and create a recipe:
• Edit Analytics Dataflows OR Edit Dataset Recipes

3. Under Show Results In, select whether the new values appear in a new column and whether to keep the original column.

4. If you elect to create a column, set the label under Column Label.

5. To change more column attributes, click the Column Attributes box, change them, and click **Apply**.

Based on the output column's datatype, you can set these attributes.

Attribute	Description
API Name	The API name of the column. The column API name can each be up to 255 characters. The API name must also be unique within the recipe.

Attribute	Description
Precision	The maximum number of digits in a numeric column. Includes all numbers to the left and to the right of the decimal point (but excludes the decimal point character). Enter an integer between 1 and 18, inclusive.
	📝 **Note:** If a numeric value exceeds the specified precision, the value is set to 0 or null, depending on whether null measure handling is enabled. For example, if precision is set to 5 and null measure handling is enabled, the numeric value 123456.78 is replaced with null.
Scale	The number of digits to the right of the decimal point in a numeric column. Enter an integer between 1 and 17, inclusive. The scale must be less than the precision. For example, if the numeric value is 123456.789 and you set the scale to 2, the number appears as 123456.78.
Length	The maximum number of characters in a text column. Enter an integer between 1 and 32000, inclusive. The default length is 255 characters.
Date Format	The date format of a date column.

For more guidelines, see the "Field Name Restrictions" section in this help topic.

6. To add the transformation to the Transform node, click **Apply**.
   The Preview tab shows the dimension column.

7. To view the Graph area, click the Collapse button (◪).

8. Save the recipe.

Run the recipe to convert the column type to Dimension.

Date and Time Transformations: Calculate on Dates Fields
Your date fields are key to understanding trends over time or keeping teams aware of upcoming milestones and deadlines. Use the Date and Time transformations to calculate based on dates or make them more helpful to your analysis. With the `Now` option, you can insert a column with the current date and time in a specified format. Use `Date Difference` to calculate the duration between two selected date columns as days, months, or years. And use the `Add or Subtract Days or Months` function to add or subtract days or months from a date column.

Add or Subtract Days or Months
Returns the column date with the specified days or months added or removed.

Date Difference
Calculate the duration between two selected date columns as days, months, or years.

Now
Create a column in a Data Prep recipe that displays the current date and time in a specified format.

Add or Subtract Days or Months
Returns the column date with the specified days or months added or removed.

1. In a Transform node of a Data Prep recipe, select one date column in the Preview tab.

2. Select the Formula button (*fx*) in the Transform toolbar.

3. Select **Add or Subtract Days or Months**.

USER PERMISSIONS

To create a recipe:
- Edit Analytics Dataflows OR Edit Dataset Recipes

4.

Configure the transformation.

a. Start Date Column is automatically filled with the date column you previously chose. Cancel and restart to choose a different Start Date.

b. Choose **Days** or **Months** to add or subtract.

c. Select the number to add or subtract with the plus and minus buttons.

d. Choose the date format of the output column.

5. Optionally, change the label of the column that stores the result data in the Column Label field.

6. Click **Apply** to add the transformation to the Transform node.
The Preview tab shows the new calculated date column.

7. To view the Graph area, click the Collapse button ().

8. Save the recipe.

Date Difference

Calculate the duration between two selected date columns as days, months, or years.

1. In a Transform node of a Data Prep recipe, select one date column in the Preview tab.

2. Select the Formula button (fx) in the Transform toolbar.

3. Select **Date Difference**.

4.

Configure the transformation.

a. Start Date is automatically filled with the date column you previously chose. Cancel and restart to choose a different Start Date.

b. End Date is the value used to calculate the duration from the start date. Choose a date column from your data or the **Now** option to use the current date and time.

c. Select whether to return duration as days, months, or years. Days are rounded up to the nearest value, while months and years include the selected number of decimal places.

5. Optionally, change the label of the column that stores the result in the Column Label field.

6. Click **Apply** to add the transformation to the Transform node.
   The Preview tab shows the new Now column.

7. To view the Graph area, click the Collapse button ( ).

8. Save the recipe.

After you run the recipe, you can view the calculated duration. The Now option uses the date and time when the recipe last ran, based on UTC timezone.

Now

Create a column in a Data Prep recipe that displays the current date and time in a specified format.

This transformation works without interacting with any other columns.

1. In a Transform node of a Data Prep recipe, select the Formula button (fx) in the Transform toolbar.

2. Select **Now**.

3. Select the date format.

Formula Now

Returns the current date and time in the specified date format. Learn More

Date Format

Search Formats...

Show Results In

New Column (and Keep Original) ▼

Column Label

Now
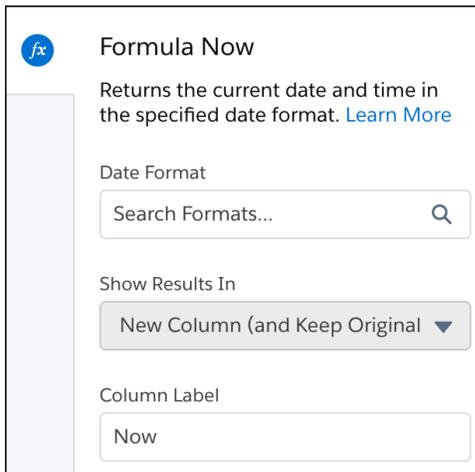
4. Optionally, change the label of the column that stores the Now data in the Column Label field.

5. Click **Apply** to add the transformation to the Transform node.
The Preview tab shows the new Now column.

gender	emp_no	birth_date	last_name	hire_date	Now
M	10001	1953-09-02T00:00:00.000Z	Facello	1986-06-26T00:00:00.000Z	20-08-14
F	10002	1964-06-02T00:00:00.000Z	Simmel	1985-11-21T00:00:00.000Z	20-08-14
M	10003	1959-12-03T00:00:00.000Z	Bamford	1986-08-28T00:00:00.000Z	20-08-14
M	10004	1954-05-01T00:00:00.000Z	Koblick	1986-12-01T00:00:00.000Z	20-08-14
M	10005	1955-01-21T00:00:00.000Z	Maliniak	1989-09-12T00:00:00.000Z	20-08-14

6. To view the Graph area, click the Collapse button ( ).

7. Save the recipe.

Note: The Now option uses the date and time when the recipe last ran, based on UTC timezone.

Detect Sentiment Transformation: Determine the Sentiment of Text

You can find valuable information in text fields, such as product reviews and social media posts. Use the Detect Sentiment transformation in a Data Prep recipe to quickly categorize that information into sentiments: positive, negative, and neutral. For example, detect the sentiment of survey responses to evaluate how customers feel about your product support. If more than a certain percentage—say 30%—of the comments are negative, escalate the feedback to support management.

This transformation works on a single dimension column and supports English text only. It processes non-English text as English and ignores images, including emojis. The transformation returns null when the input value is null and returns neutral when the input value is an empty string (''). Typically, the transformation generates the correct sentiments, but the results can vary based on the data. For example, text without sentiment (such as IDs, nouns, addresses, and alphanumeric values) can return unexpected results.

1. In a Transform node of a Data Prep recipe, select the dimension column in the Preview tab.

2. In the Transform toolbar, click the Detect Sentiment button.
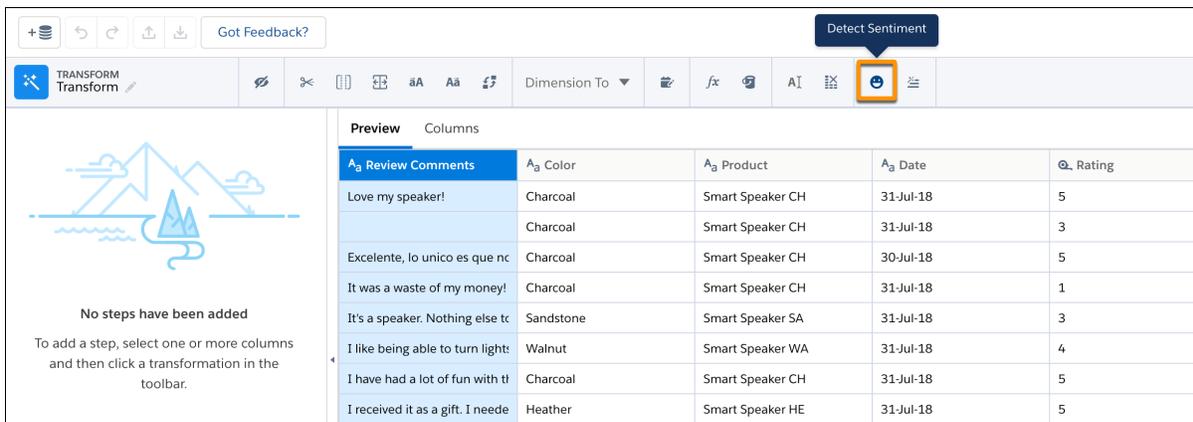
3. Optionally, change the label of the column that will store the generated sentiments in the Column Label field.
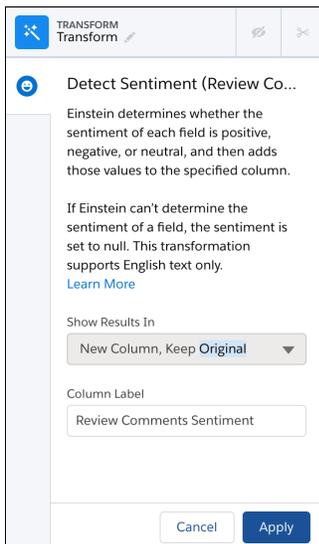
4. Click **Apply** to add the transformation to the Transform node.
   The Preview tab shows the new sentiment column. Currently, the column shows the temporary value Sentiment TBD.
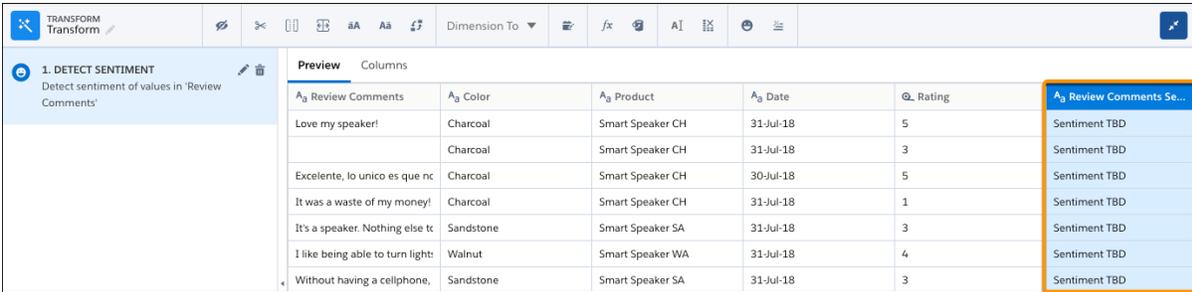
5. To view the Graph area, click the Collapse button ( ).

6. Save the recipe.

After you run the recipe, you can view the generated sentiments in the sentiment column.

Detect Sentiment Model Card
The Detect Sentiment transformation uses the model to determine whether the sentiment of text is positive, negative, or neutral. Use this model card to better understand the model, how it's trained, its capabilities, its intended use, and its limitations.

For questions about the model, contact smart-transforms-analytics@salesforce.com.

Model Details
Get basic information about the Detect Sentiment model.

The following table provides details about the current model.

Detail	Description
Model Developer	Tableau CRM
Model Date	August 12, 2020
Model Version	4
Model Type	Sentiment Analysis
Model Input	Text column
Model Output	For each field in a text column of a Data Prep recipe, the model returns the sentiment: positive, negative, or neutral. The model returns null if the input is null and returns neutral if the input is an empty string ('').
License	Detect Sentiment is available to customers with any Tableau CRM user license.

Minor changes to the model can occur throughout the release. Major changes can occur and are communicated via Release Notes.

Model Training
We trained the Detect Sentiment model based on the GloVe dictionary and training data. To learn sentiments of text, we used the GloVe dictionary to convert text from the training data to word vectors, and then identified the sentiment based on the rating associated with each comment in the training data.

We used the following process to train the model.
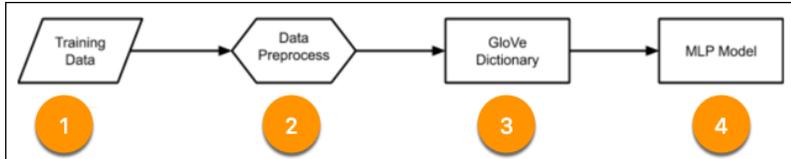
1. Gather the training data. The training data consists of product review comments and their ratings, like 4 stars out of 5.

2. Use a preprocess to massage the training data into the format required by GloVe.

3. Run the training data through GloVe to convert it to vectors. We used the Wikipedia 2014 + English Gigaword Fifth Edition GloVe dictionary to convert each string to a vector.

4. Pass the vectors and their associated ratings through a Multilayer Perceptron (MLP) neural network model to train the model to predict the sentiment of text.

For more information about the GloVe dictionary, see this website.

The GloVe dictionary is made available under the Public Domain Dedication and License v1.0 whose full text can be found at: http://opendatacommons.org/licenses/pddl/1.0/.

Intended Use
Understand the primary intended and out-of-scope use cases associated with the Detect Sentiment model.

Primary intended use

Detect the sentiment of unstructured text passed to the Detect Sentiment transformation in a Data Prep recipe. Unstructured text can include customer surveys, product and service feedback, and unstructured communication, such as chat messages, text messages, email correspondence, and social media posts.

Out-of-scope use cases

- The Detect Sentiment transformation can't be invoked outside of Data Prep recipes.

- The input text can't bypass Salesforce data centers. The data must be available in Salesforce to be evaluated by the Detect Sentiment transformation.

- Detect Sentiment doesn't support non-English input text. It processes non-English text as English and ignores images, including emojis.

Factors
Understand relevant factors associated with the Detect Sentiment model.

After considering the deep-learning and lexicon-based approaches for sentiment analysis, we decided to use a hybrid of both approaches to mitigate bias in the sentiment model. (See Ethical Considerations on page 770 for more details).

We chose to train the model using publicly available, labeled product review datasets over other datasets because of its relevance to business use cases. Other publicly available, labeled datasets, such as movie reviews or social media posts, are generally not as relevant and can hamper accuracy. Because the sentiment model is trained on specific sets of data, the accuracy of the sentiment can vary based on how similar your data is to the training data. (See Metrics on page 769 for more details.) Also, because the training data currently is in English, the Detect Sentiment model supports only English text at this time.

Metrics
To learn how well the model works with different domains of data, check out the accuracies and F1 scores.

When we evaluated the model based on product reviews and customer service survey responses, the model achieved sentiment accuracy between 69.09%-83.12% and an F1 score between 0.34 and 0.9. When we performed biased testing, the model achieved 92.31% accuracy and an F1 score of 0.9.

If you're curious why the results vary, it's because each data domain can have unique terms that weren't used to train the Detect Sentiment model. As a result, the model's accuracy can vary based on the domain of the data.

Training and Evaluation Data
Get the basics about the data used to train and evaluate the Detect Sentiment model.

The model is trained using 80% of openly available product review data, which consists of about 100,000 ratings and reviews. The sentiments of the training data are determined based on the ratings. For example, highly rated comments are considered positive.

To evaluate the model, we compared the generated sentiments of the remaining 20% of the product reviews against their actual ratings. We also evaluated the model using proprietary survey responses.

Ethical Considerations
We have taken additional measures and precautions to correct bias against terms representing gender identity, sexual orientation, religious affiliation, and race. Eleven identity terms were included in this mitigation (aetheist, queer, gay, lesbian, homosexual, feminist, black, white, heterosexual, islam, muslim, and bisexual). Using synthetic data, we trained the model to always predict neutral sentiments for these terms. Although text surrounding these identity terms can change the sentiment, these terms alone have a neutral sentiment. With the introduction of mostly handcrafted, high quality data, we were able to reduce bias for the specified terms to a great extent without harming model performance. We recognize that the list of identity terms included in this mitigation is not exhaustive, and that future work should be undertaken to expand the list to be more inclusive of a broader range of stakeholders.

Caveats and Recommendations
The model typically generates the correct sentiments, but the results can vary based on the data. For example, the model might not accurately process new language, slang, or sarcasm. It also returns unexpected results for text without sentiment, such as IDs, nouns, addresses, and alphanumeric values. Because the accuracy can vary, consider making decisions based on trends in sentiment scores over time rather than on scores at a particular moment in time.

Related Resources
For more details about the Detect Sentiment model card, check out the following resources.

- GloVe: Global Vectors for Word Representation

- Detect Sentiment Transformation: Determine the Sentiment of Text (online help)

Drop Columns Transformation: Drop Columns from the Recipe
Drop unwanted columns from a Data Prep recipe. For example, you can add a Drop Columns transformation after a Formula transformation to drop input columns used for a calculated column.

USER PERMISSIONS

To create a recipe:
- Edit Analytics Dataflows OR Edit Dataset Recipes

1. In a Transform node of a Data Prep recipe, select the column in the Preview tab.

2. In the Transform toolbar, click the Drop Columns button ( ).
   The dropped column appears in the Drop Columns step.
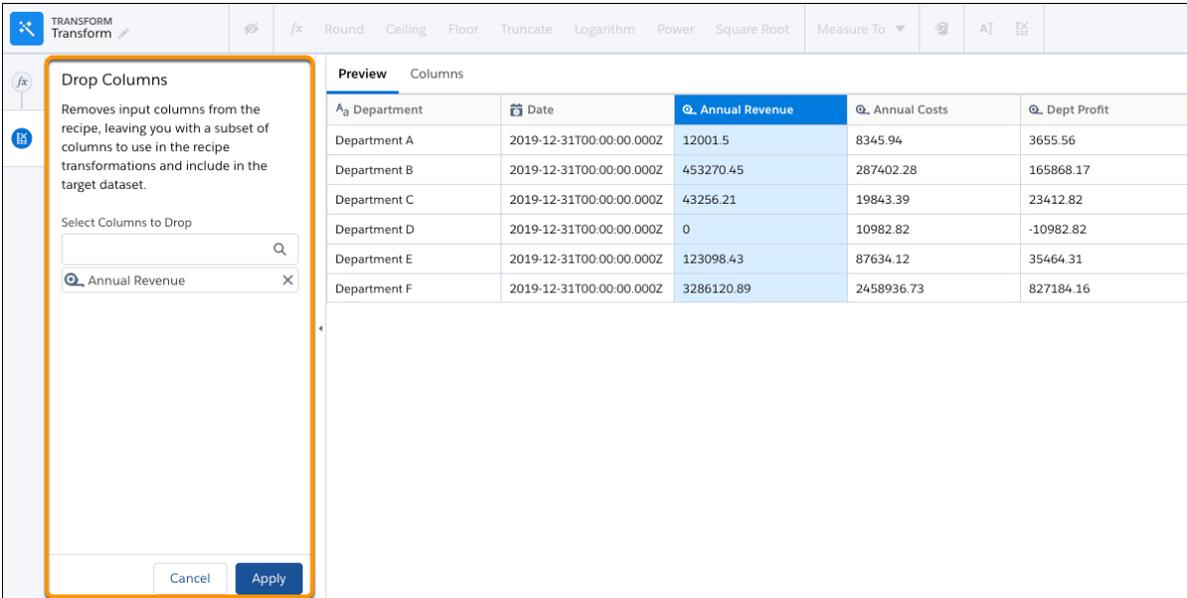
3. To drop additional columns, enter their label in the Select Columns to Drop field.

4. Click **Apply** to add the transformation to the Transform node.
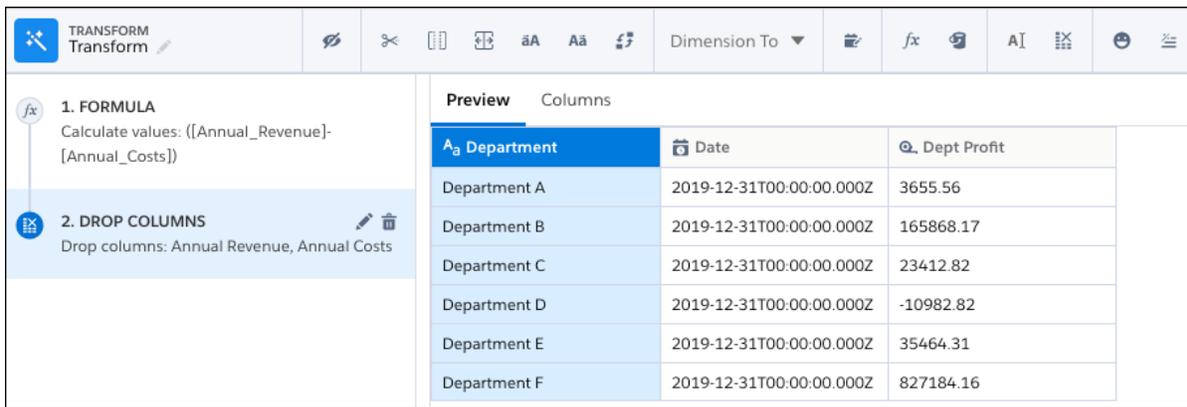
5. To view the Graph area, click the Collapse button ( ).

6. Save the recipe.

From this point forward, dropped columns can't be used in recipe transformations and nodes. They also no longer appear in Preview.

Discovery Predict Transformation: Get Einstein Discovery Predictions
Use the Discovery Predict transformation to populate your datasets with predictive and prescriptive intelligence. When you run a Data Prep recipe with a Discovery Predict node, Einstein calculates and saves predicted outcomes on a row-by-row basis. You can optionally store descriptions of top predictors and improvements. With the Discovery Predict node, you can quickly evaluate predictions across a large set of data, assess multiple models before deploying them into production, and aggregate this information in a dashboard.

1. Edit a Data Prep recipe.

2. Where you want to insert the node, click Add Node (+) and choose **Discovery Predict**.
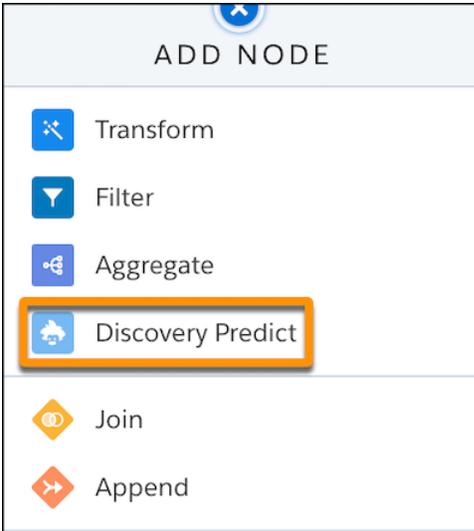
3. Select a deployed prediction, then click **Next**.

4. Map all fields from the prediction to the recipe.

5. If you want to include prediction and improvement text in the dataset, click the **Options** subtab.

Issue	Description
Top predictors	Enable **Top predictors** and specify the number of entries you want to save to the dataset.
Top improvements	Enable **Top improvements** and specify the number of entries you want to save to the dataset.
	✎ Note: This option is available for predictions that use only the Generalized Linear Model (GLM) algorithm. If the selected prediction has any models that use a different algorithm, then getting top improvements is not an available option. To learn more about algorithms in Einstein Discovery story settings, see Edit Story Settings on page 1639.

6. Click **Apply**.
   Einstein adds the Discovery Predict node to your recipe.

7. Save the recipe.

Run the recipe to get predictions.

Extract Transformation: Get a Date Component

Use the Extract transformation in a Data Prep recipe to pull a selected component from a date field into a new field. For example, extract the hour component from the Case Created Date column to analyze case creation by hour of the day.

1. In the Transform node of a Data Prep recipe, select the date column that you want to extract a component from.

2. In the Transform toolbar, click the Extract button ( ) to set the extract settings.

3. Select which date component to extract.

4. Under Show Results In, select whether the new values appear in a new column and whether to keep the original column.

5. If you elect to create a column, set the label under Column Label.

6. Click **Apply** to add the transformation to the Transform node.
The Preview tab shows the new date component column.

7. To view the Graph area, click the Collapse button ( ).

8. Save the recipe.

After you run the recipe, the selected date component is extracted in the specified dimension column.

Edit Attributes Transformation: Change the Column Names and Value Formats

To make column names more descriptive and apply consistent formats to column values, use the Edit Attributes transformation in a Data Prep recipe. You can set the labels and API names of all columns. You can also set the precision and scale for measure columns, character length for dimension columns, and date formats for date columns.

1. In a Transform node of a Data Prep recipe, select the column in the Preview tab.

2. In the Transform toolbar, click the Edit Attributes button ( ).
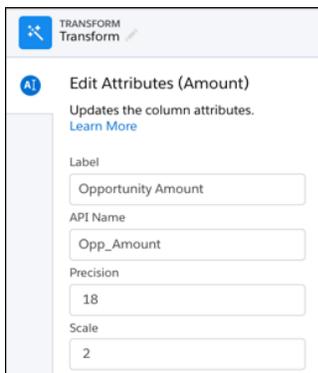
3. Change the column attributes.

Based on the column datatype, you can set these attributes.

Attribute	Description
Label	The display label for the column, the one that appears in charts and dashboards. The column label can each be up to 255 characters.
API Name	The API name of the column. The column API name can each be up to 255 characters. The API name must also be unique within the recipe.
Precision	The maximum number of digits in a numeric column. Includes all numbers to the left and to the right of the decimal point (but excludes the decimal point character). Enter an integer between 1 and 18, inclusive. 📝 Note: If a numeric value exceeds the specified precision, the value is set to 0 or null, depending on whether null measure handling is enabled. For example, if precision is set to 5 and null measure handling is enabled, the numeric value 123456.78 is replaced with null.
Scale	The number of digits to the right of the decimal point in a numeric column. Enter an integer between 1 and 17, inclusive. The scale must be less than the precision. For example, if the numeric value is 123456.789 and you set the scale to 2, the number appears as 123456.78.
Length	The maximum number of characters in a text column. Enter an integer between 1 and 32000, inclusive. The default length is 255 characters.
Date Format	The date format of a date column.

For more guidelines, see the "Field Name Restrictions" section in this help topic.

4. Click **Apply** to add the transformation to the Transform node.
   If applicable, Preview shows the new column label.

5. To view the Graph area, click the Collapse button (⬕).

6. Save the recipe.

Flatten Transformation: Flatten Hierarchies

The Flatten transformation flattens hierarchical data. For example, you can flatten the Salesforce role hierarchy to implement row-level security on a dataset based on the role hierarchy.

When you configure the Flatten transformation, you specify the field that contains every node in the hierarchy and the field that contains their corresponding parent based on the hierarchy. The Flatten transformation generates one record for each hierarchy node, which we refer to as the "child node." Each record contains two generated columns that store the hierarchy for each child node. One column contains a list of all ancestors for each node in the hierarchy. The other column contains the hierarchy path.

See the Roles and Role Path columns in the following flattened dataset to see how ancestors are stored.

Role ID (child node)	Role Name	Parent Role ID	Roles	Role Path
1	Salesperson 1	10	10, 20, 30	/10/20/30

Role ID (child node)	Role Name	Parent Role ID	Roles	Role Path
2	Salesperson 2	10	10, 20, 30	/10/20/30
3	Salesperson 3	11	11, 20, 30	/11/20/30
10	Regional Manager 1	20	20, 30	/20/30
11	Regional Manager 2	20	20, 30	/20/30
20	Vice President 1	30	30	/30
21	Vice President 2	30	30	/30
30	CEO	Not applicable	Not applicable	Not applicable

You can also configure the Flatten transformation to include the child node in the generated hierarchy columns. The following dataset shows the child node in bold.

Role ID (child node)	Role Name	Parent Role ID	Roles	Role Path
1	Salesperson 1	10	**1**, 10, 20, 30	/**1**/10/20/30
2	Salesperson 2	10	**2**,10, 20, 30	/**2**/10/20/30
3	Salesperson 3	11	**3**,11, 20, 30	/**3**/11/20/30
10	Regional Manager 1	20	**10**,20, 30	/**10**/20/30
11	Regional Manager 2	20	**11**,20, 30	/**11**/20/30
20	Vice President 1	30	**20**,30	/**20**/30
21	Vice President 2	30	**21**,30	/**21**/30
30	CEO	Not applicable	**30**	**30**

1. In a Transform node of a Data Prep recipe, select the dimension column that contains all child nodes in the hierarchy.
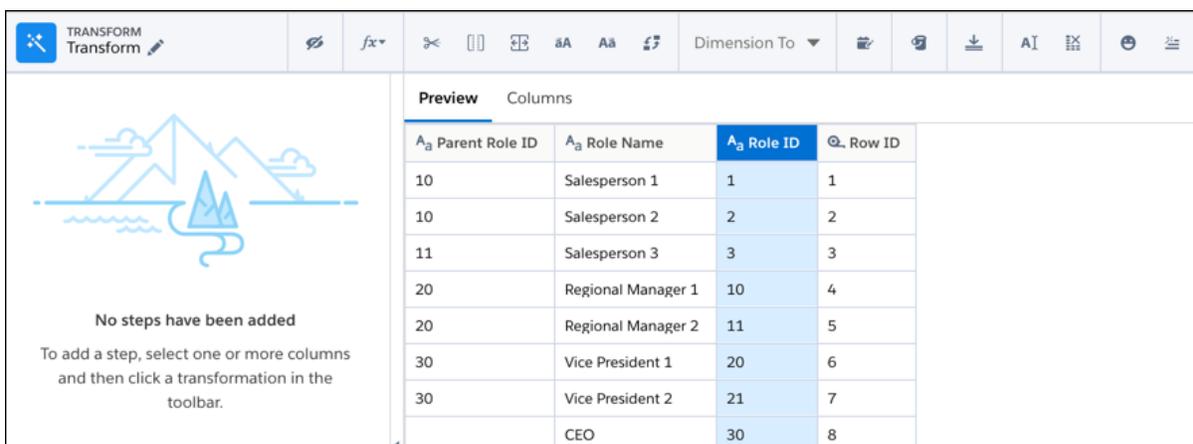
2. In the Transform toolbar, click the Flatten button (⬇).

3. Enter the following transformation details.

Property	Description
Parent Column	Column that contains the parent of each node in the hierarchy. For example, the Regional Manager 1 role is the parent of the Salesperson 1 role in a role hierarchy.
Include child node	Indicates whether to include the child node in the generated hierarchy nodes and path columns.
Hierarchy Nodes Column Label	Name of the multivalue output column that contains a list of all ancestors in the hierarchy, in order from the lowest to the highest level. The Flatten transformation creates this column and generates the list of ancestors for each node in the hierarchy. For example, for Salesperson 1 role, the hierarchy of ancestors is: `Sales Manager 1, Regional Manager 1,Vice President 1, CEO` . To hide this column from queries and dataset exploration, select **Is a system column**. By default, this column isn't created as a system column.
Hierarchy Path Column Label	String that contains the hierarchical path of all ancestors in the hierarchy, in order from the lowest to the highest level. Nodes are separated by forward slashes. The Flatten transformation creates this output column and generates the ancestry path for each node in the hierarchy. For example, for a salesperson role in a role hierarchy, the value is: `Sales Manager 1/Regional Manager 1/Vice President 1/CEO` . To hide this column from queries and dataset exploration, select **Is a system column**. By default, this column isn't created as a system column.

4. Preview the transformation results to check that the flattened hierarchy is right.

Preview shows only the child node and its parent in the hierarchy nodes column and only the child node in the hierarchy path column. You have to run the recipe to see all ancestors in both columns.

Aa Parent Role ID	Aa Role Name	Aa Role ID	Q Row ID	Aa Role ID Path	Aa Role ID Nodes
10	Salesperson 1	1	1	1/10	1
10	Salesperson 2	2	2	2/10	2
11	Salesperson 3	3	3	3/11	3
20	Regional Manager 1	10	4	10/20	10
20	Regional Manager 2	11	5	11/20	11
30	Vice President 1	20	6	20/30	20
30	Vice President 2	21	7	21/30	21
	CEO	30	8	30	30

5. Click **Apply** to add the transformation to the Transform node.

6. To view the Graph area, click the Collapse button (⬚).

Run the recipe to generate the hierarchy nodes and hierarchy path columns in the dataset.

📝 Note: Only the first value in the hierarchy nodes column appears when exploring the dataset in a lens. In general, Explorer shows only the first value of a multivalue column.

Format Dates Transformation: Standardize the Date Format in a Column

If a dimension column contains dates in different formats, use the Format Dates recipe transformation in a Data Prep recipe to standardize the format for all dates in the column. A consistent format enables you to correctly filter and group records by date, including filtering by date component, such as month. It also ensures that you can successfully convert the column type from dimension to date. Optionally, you can convert the column type from Dimension to Date.

USER PERMISSIONS

To create a recipe:
- Edit Analytics Dataflows OR Edit Dataset Recipes

1. In a Transform node of a Data Prep recipe, select the dimension column that contains dates in the Preview tab.

2. In the Transform toolbar, click the Format Dates button (![icon]).

3. Enter the original date formats in the selected column.

> **Note:** The order of the date formats in the Original Formats box matters. For example, if a date is 01/06/2011, the date format could be dd/MM/yyyy or MM/dd/yyyy. The Date Format transformation assumes the first matching format shown the Original Date Format box is the right one.

4. Select the desired date format in the Change To field.

5. To convert a dimension column with dates to a Date column type, select **Convert column type to Date**.

6. Under Show Results In, select whether the new values appear in a new column and whether to keep the original column.

7. If you elect to create a column, set the label under Column Label.

8. Click **Apply** to add the transformation to the Transform node.

9. To view the Graph area, click the Collapse button (![icon]).

When you run the recipe, the transformation replaces the dates in the existing column with dates in the specified format. If Tableau CRM can't determine the original format or the date doesn't have a value for a date component, the transformation replaces the date with null. For example, if you standardize on the `MM/dd/yyyy:hh:mm:sssz` format and a date value doesn't have the seconds date component (`sssz`), Tableau CRM replaces the date with null.

Formula Transformation: Create a Calculated Column Based on an Expression

Create a column in a Data Prep recipe that displays values based on a formula calculation. The calculation can include input from other fields in the same row or across rows. For example, you can create a Profit column based on input from Revenue and Cost columns. Enter formulas in EA-SQL format. EA-SQL is a collection of standard and custom functions for numeric, string, and date data.

USER PERMISSIONS

To create a recipe:
- Edit Analytics Dataflows OR Edit Dataset Recipes

1. In a Transform node of a Data Prep recipe, select a column in the Preview tab.

2. To create the formula for the calculated column, select the Formula button (fx) in the Transform toolbar and select **Custom Formula**. The formula editor appears.
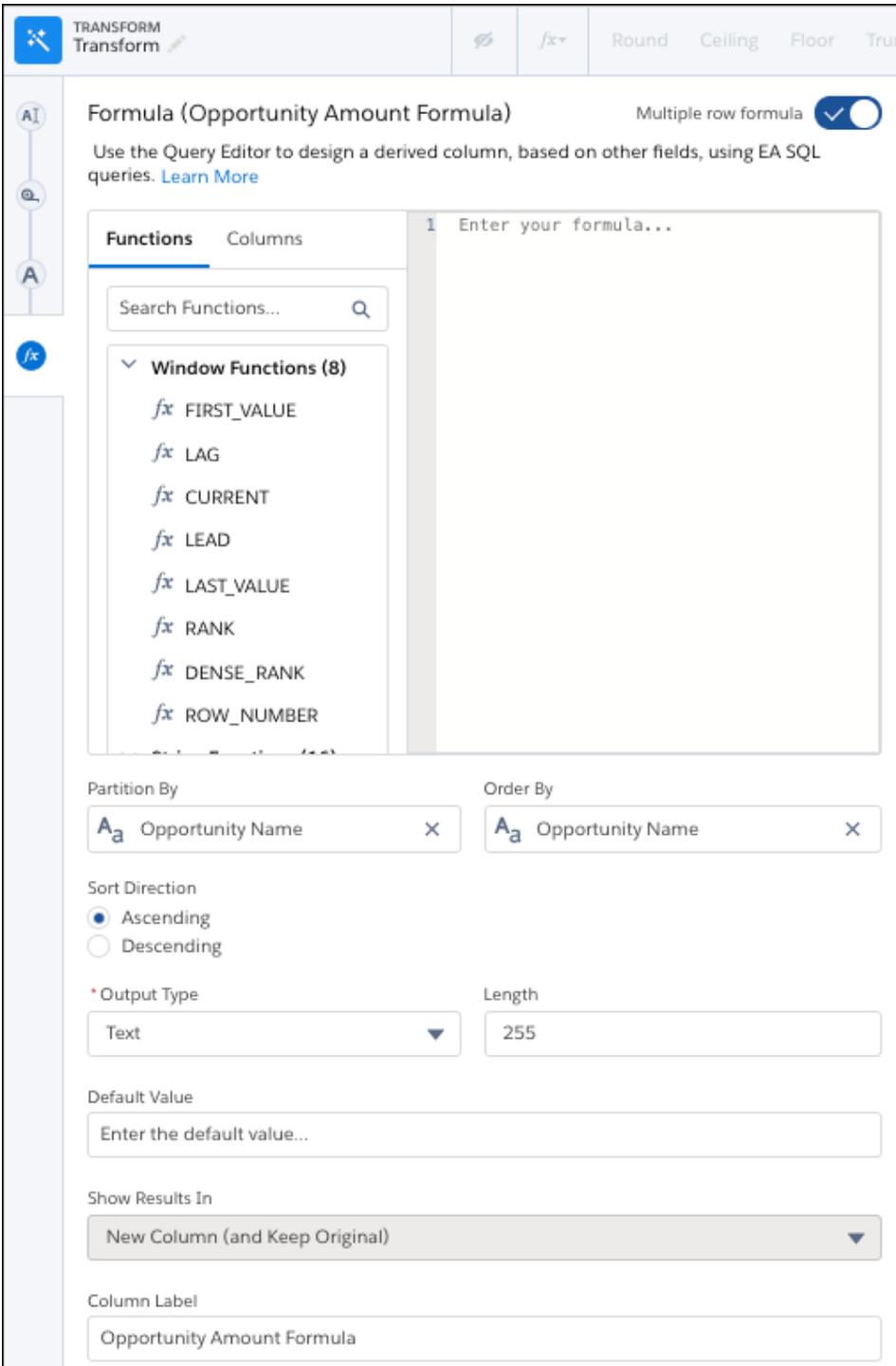
3. In the formula editor, enter the formula and add parameters using the correct syntax.

As you type in the formula editor, functions and columns matching the name appear for you to select to add to the formula.
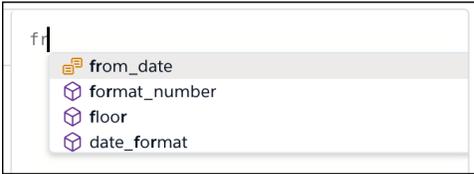
To get help, check out all functions and columns in the Functions and Columns tabs. To add one to the formula, select it and click **+**.

Functions are listed in the Functions tab by category: Window Functions, String Functions, Numeric Functions, Date/Time Functions, and Additional Functions. To view the window functions, enable **Multiple row formula**.

4. For formulas with window functions, specify the column used to partition the rows in **Partition By**, column used to sort the data in **Order By**, and sort direction in **Sort Direction**.

For example, set Partition By to Opportunity Name, Order By to Last Modified Date, and Sort Direction to Ascending. Based on these settings, the transformation combines all records into partitions (think buckets) based on the same opportunity name. It then sorts the records in each partition in ascending order based on the last modified date.

5. Identify each column in the formula by its API name. Type in the API name or select the column from the **Columns** tab.
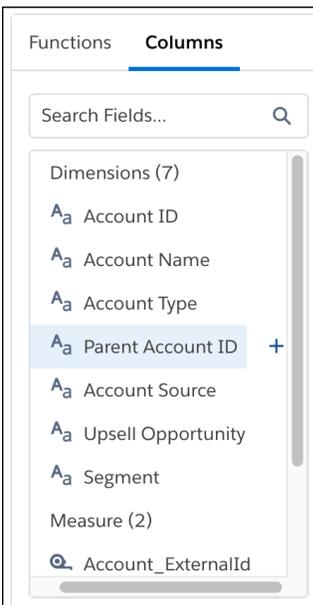
6. In **Output Type**, select whether the calculated column's data is text, numeric, date, or datetime.

7. Based on the output type, you can set these attributes.

Attribute	Description
Precision	The maximum number of digits in a numeric column. Includes all numbers to the left and to the right of the decimal point (but excludes the decimal point character). Enter an integer between 1 and 18, inclusive. 🖉 Note: If a numeric value exceeds the specified precision, the value is set to 0 or null, depending on whether null measure handling is enabled. For example, if precision is set to 5 and null measure handling is enabled, the numeric value 123456.78 is replaced with null.
Scale	The number of digits to the right of the decimal point in a numeric column. Enter an integer between 1 and 17, inclulsive. The scale must be less than the precision. For example, if the numeric value is 123456.789 and you set the scale to 2, the number appears as 123456.78.
Length	The maximum number of characters in a text column. Enter an integer between 1 and 32000, inclusive. The default length is 255 characters.
Date Format	The date format of a date column.

8. In **Default Value**, choose the value to show if your expression returns no results.

9. If needed, you can change the label of the calculated column.

10. To add the formula as a step in the Transform node, click **Apply**.
    The calculated column appears in the recipe preview.

11. To view the Graph area, click the Collapse button ( ).

12. Save the recipe.

You can bucket a formula column or reference it in later formulas defined in the recipe.

💡 Tip: Common functions have their own buttons in the Transform toolbar that automatically fill in the formula with the column and predefined syntax. The toolbar shows appropriate functions based on the type and number of selected columns. To use one, click the function button in the Transform toolbar (1). The formula is started for you (2), using the column and function that you selected.

Date Functions for Formulas

Use date functions to adjust or calculate values from dates in your recipe. For example, you can add time, find time until, and extract a day of the week from date fields. The arguments for date functions are date columns.

Numeric Operators and Functions for Formulas

Use numeric functions to calculate values from measures in your recipe. For example, you can round, truncate, and determine the absolute value of a number. The arguments for numeric functions can be numbers or measure columns.

String Functions for Formulas

Use string functions to create values based on other strings. For example, you can change the casing of text strings, concatenate values from multiple columns, and replace values. The arguments for string functions must be text strings or dimension fields.

Window Functions for Formulas

Use window functions in a Data Prep recipe to perform calculations across rows. For each row of input data, perform a calculation on a selection—or window—of related rows. Unlike aggregation that groups rows, window functions return a result for each row. For example, you can calculate the changes to the opportunity amount throughout the stages of each opportunity to see if you can hit your sales target.

Additional Functions for Formulas

These versatile functions can be used with the data types text, number, date_only, and datetime to build logic into your formulas.

Upgrade or Edit Formulas for Converted Recipes

You see a different formula editor if your recipe was converted from Data Prep Classic to Data Prep and your formulas couldn't automatically upgrade. You're still able to edit and use these formulas, but to get the most formula options through EA-SQL and best experience, recreate the formula in the Data Prep recipe editor.

Date Functions for Formulas

Use date functions to adjust or calculate values from dates in your recipe. For example, you can add time, find time until, and extract a day of the week from date fields. The arguments for date functions are date columns.

When entering a formula use the API name of the column in the expression, not the label. The parameters section is surrounded by parentheses. For example, you can enter the following expression.

```
datediff(ClosedDate, CreatedDate)
```

Date Function	Description	Syntax
add_months	Returns the date with the specified number of months after the start date.	add_months(startDate,num_months) startDate is the date field calculated on. num_months is the number of months added to the startDate.
current_date	Returns the current date.	current_date()
current_timestamp	Returns the current date and time.	current_timestamp()
datediff	Returns the number of days between start date and end date.	datediff(endDate,startDate) endDate is the later of the two dates you are finding the difference between.

Date Function	Description	Syntax
		`startDate` is the earlier of the two dates you are finding the difference between.
`date_add`	Returns the date with the specified number of days after the start date.	`date_add(startDate,num_days)` `startDate` is the date field calculated on. `num_days` is the number of days added to the startDate.

`date_format`	Converts the timestamp to a specified date format.	`date_format(field,'format')`
		
		 `field` is a date field/timestamp or string to be converted to the given format.
		
		 `'format'` is the date and time format pattern to follow.
		
		 Example:
		
		 ```date_format('2016-04-08', 'y')```
		
		 returns 2016

`date_sub`	Removes the specified number of days from the start date.	`date_sub(startDate,num_days)` `startDate` is the date field calculated on. `num_days` is the number of days subtracted from the startDate.
`day`	Returns the day of the month component of the date/timestamp.	`day(date)` `date` is the date field to extract the day of the month from.
`dayofmonth`	Returns the day of the month component of the date/timestamp.	`dayofmonth(date)` `date` is the date field to extract the day of the month from.
`dayofweek`	Returns the day of the week component of the date/timestamp. 1 = Sunday, 2 = Monday, ..., 7 = Saturday.	`dayofweek(date)` `date` is the date field to extract the day of the week from.
`dayofyear`	Returns the day of year component of the date/timestamp.	`dayofyear(date)` `date` is the date field to extract the day of the year from.
`months_between`	Returns the number of months between two timestamps. Calculates the difference based on 31 days per month rounded to 8 digits unless roundOff=false. If timestamp1 is later than timestamp2, the result is positive. If timestamp1 and timestamp2 are on the same day of the month, or both are the last day of the month, the result is an integer and the time of day is ignored.	`months_between(timestamp2,timestamp1,roundOff(optional))` `timestamp2` is one of the timestamps to compare. `timestamp1` is the other timestamp to compare. `roundOff(optional)` is a boolean that determines whether to round the result. This parameter is optional.

Date Function	Description	Syntax
`now`	Returns the current date and time in the specified format.	`now()`
`to_date`	Converts the date string to the specified date format. Returns null with invalid input.	`to_date(field,'format(optional)')` `field` is a date string field that you want to convert. `'format(optional)'` is the date and time format pattern to follow. This parameter is optional. The function follows casting rules to a date if format isn't included.
`to_timestamp`	Converts the timestamp string to the specified timestamp format.	`to_timestamp(field,'format(optional)')` `field` is a timestamp string field that you want to convert. `'format(optional)'` is the timestamp format pattern to follow. This parameter is optional. The function follows casting rules to a timestamp if format isn't included.

`to_unix_timestamp`	Returns the UNIX timestamp of the specified time.	`to_unix_timestamp(timestamp,format(optional))`
		
		`timestamp` is a date/timestamp or string which is returned as a UNIX timestamp.
		
		`format(optional)` is the date/time format pattern to follow. `format(optional)` is ignored if `timestamp` is not a string. Default value is "yyyy-MM-dd HH:mm:ss". This parameter is optional.
		
		Example:
		
		`to_unix_timestamp('2016-04-08', 'yyyy-MM-dd')`
		
		returns
		
		`1460098800`

`trunc`	Replaces the specified portion of the timestamp, and all portions after, with zeroes.	`trunc(field,'format(optional)')`
		
		`field` is a timestamp field.
		
		`'format(optional)'` specifies the part of the timestamp to truncate. Accepted values are "YEAR", "YYYY", "YY", "MON", "MONTH", "MM", "DAY", "DD", "HOUR", "MINUTE", "SECOND", "WEEK", "QUARTER"]
		
		Example:
		
		`trunc('2015-03-05T09:32:05.359', 'MM')`
		
		returns
		
		`2015-03-01T00:00:00`

Date and Time Function Considerations

Keep these behaviors in mind when working with date and time functions.

- These functions are relative to when the job runs.

  - `now`
  - `current_date`
  - `current_timestamp`
  - `to_unix_timestamp`

- All date and datetime functions are based on the UTC timezone.

Numeric Operators and Functions for Formulas

Use numeric functions to calculate values from measures in your recipe. For example, you can round, truncate, and determine the absolute value of a number. The arguments for numeric functions can be numbers or measure columns.

When entering a formula, use the API name of the column in the expression, not the label. The parameters section is surrounded by parentheses. For example, you can enter the following expression.

```
round(Total_Amount/Unit_Price, 2)
```

You can use the following math operators.

Operator	Description
+	Calculates the sum of two values.
-	Calculates the difference between two values.
*	Multiplies the two values.
/	Divides the first value by the second.
()	Specifies that the expressions within the parentheses are evaluated first. All other expressions are evaluated using standard operator precedence.

You can also use the following numeric functions in an expression.

Numeric Function	Description	Syntax
abs	Calculates the absolute value of a number. The absolute value of a number is the number without its positive or negative sign.	`abs(field)` `field` is a number or measure field that you want to determine the absolute value of. Example: `abs(Amount)`
ceil	Rounds a number up to the nearest integer.	`ceil(field)` `field` is a number or measure field that you want to round up to the nearest integer.
exp	Returns a value for e raised to the power of a number you specify.	`exp(field)` `field` is a number or measure field to which you want to raise e.

Numeric Function	Description	Syntax
`floor`	Returns a number rounded down to the nearest integer.	`floor(field)` `field` is a number or measure field that you want to round down to the nearest integer.
`format_number`	Formats a number by rounding to the specified number of decimal places or by applying a certain format, for example #,###,###.##.	`format_number(expr,format)` `expr` is a number or measure field that you want to change the format of. `format` is the new format to be applied to the numbers in the column. The parameter can be in the format composed as '#,###,###.##' or actual numerals like '12,332.1235'. This parameter is optional. If you don't specify a format, the field is truncated to an integer. Example: `format_number(12332.123456, '###,###,###,###,###,###.###')` returns `12,332.123`
`log`	Returns the logarithm of the number in the base that you specify.	`log(value,field)` `value` is a number or measure field used as the base of the logarithm. `field` is a number or measure field that you want to take the logarithm of.
`power`	Raises a number to the power of another number.	`power(field,value)` `field` is a number or measure field that you want to raise to the specified power. `value` is a number or measure field that is the power that you want to raise the specified number to.
`round`	Returns the nearest number to a number you specify, constraining the new number by a specified number of digits.	`round(field,numberOfDigits)` `field` is a number or measure field that you want to round. `numberOfDigits` is a number or measure field that specifies the number of digits to round the specified number to.
`sqrt`	Returns the positive square root of a given number.	`sqrt(field)` `field` is a number or measure field that you want to find the square root of.
`truncNumber`	Truncates the specified number of decimal values from the number, or converts to an integer.	`truncNumber(field,decimalValueToRemove)` `field` is a number or measure field that you want to truncate.

Numeric Function	Description	Syntax
		`decimalValueToRemove` is the number of decimal values you want to remove from the number. This parameter is optional. If you don't specify a decimalValueToRemove, the field is truncated to an integer. Example: `truncNumber(13.111, 1)` returns `13.1`

String Functions for Formulas

Use string functions to create values based on other strings. For example, you can change the casing of text strings, concatenate values from multiple columns, and replace values. The arguments for string functions must be text strings or dimension fields.

When entering a formula, use the API name of the column in the expression, not the label. The parameters section is surrounded by parentheses. Also, text strings must be enclosed in double straight quotes (`"This is a string."`).

For example, you can nest string functions in an expression to combine the sales territory and country into a single column. Then you can apply title casing to the results. `title(concat(Sales_Territory, " ",Country))`

You can use the following string functions in a formula expression.

String Function	Description	Syntax

`concat`	Returns a string by merging the values of the specified columns and input strings.	`concat(field1,field2,...fieldN)`
		
		`field1` is the first dimension field or text string to include in the concatenated value.
		
		`field2` is the second dimension field or text string to include.
		
		`fieldN` is any number of extra dimension fields or text strings to include.
		
		Example: To create the full name, concatenate the first and last names with a space in between them.
		
		`concat(OwnerId.FirstName," ",OwnerId.LastName)`

`endsWith`	Returns true if the specified string is found at the end of the column value. This function is only supported in `case` functions because it returns as a boolean.	`ends_with(fieldName,literal)` `fieldName` is the dimension field to be searched. `literal` is the value to search for at the end of the field.

`lower`	Returns the string with all characters in lowercase. If the input string is null, then the result is null.	`lower(field)`
		
		`field` is the dimension field or text string to convert to lowercase.
		
		Example: The Account.Industry column uses different casing for industries, like "Media" and "media." To apply consistent casing to the columns values, convert them to lowercase.
		
		`lower(Account.Industry)`

String Function	Description	Syntax

ltrim	Removes the specified substring from the beginning of a string. If no substring is provided, remove the leading space characters from a string.	`ltrim(field,valueToBeRemoved)`
		
		`field` is the dimension field or text string to remove the leading spaces or specified substring from.
		
		`valueToBeRemoved` is the value to remove from the string. This parameter is optional. If you don't specify a valueToBeRemoved, the transformation removes the leading spaces.
		
		Example: Trim the leading space characters from a string.
		
		`ltrim("     5 Spaces Before")`

replace	Replaces a substring with another string. If any of the arguments are null, then the function returns null. This function is case-sensitive.	`replace(field,searchString,replacementString)`
		
		`string` is the dimension field or text string that contains the substring to be replaced.
		
		`searchString` is the substring to replace. If searchString is an empty string, the function returns null.
		
		`replacementString` is the value that replaces the substring.
		
		Example: Change the account name from "salesforce.com" to "Salesforce" in the Account_Name column.
		
		`replace(Account_Name,"salesforce.com","Salesforce")`

rtrim	Removes the specified substring from the end of a string. If no substring is provided, remove the trailing space characters from the end of a string.	`rtrim(field,valueToBeRemoved)`
		
		`field` is the dimension field or text string to remove the trailing spaces or specified substring from.
		
		`valueToBeRemoved` is the value to remove from the string. This parameter is optional. If you don't specify a valueToBeRemoved, the transformation removes the trailing spaces.
		
		Example: Trim the 2 trailing space characters from a string.
		
		`rtrim("2 Spaces After  ")`

startsWith	Returns true if the specified string is found at the beginning of the column value. This function is only supported in `case` functions because it returns as a boolean.	`startsWith(fieldName,literal)` `fieldName` is the dimension field to be searched. `literal` is the value to search for at the beginning of the field.
string	Converts the date or number value to a string data type.	`string(expression)` `expression` is the date or number field to convert.

String Function	Description	Syntax
substr	Returns characters from the string, starting at the specified position and of the specified length. Also called `substring`.	`substr(field,position,len)` `field` is the dimension field or text string to extract the substring from. `position` is the starting character position of the substring. The first character in a string is at position 1. If position is negative, then the position is relative to the end of the string. A position of -1 denotes the last character. `len` is the length, or number of characters, to return. If len is 0, the output is an empty string. If len is negative, then the function returns null. This parameter is optional. Example: Return the 13th, 14th, and 15th character from each account ID. <pre>substr(Account_Id,13,3)</pre>
title	Returns the string with the initial character of every word in uppercase and the remaining characters in lowercase. For example, "united states" becomes "United States."	`title(field)` `field` is the dimension field or text string on which to apply title casing. Example: Apply title casing on the opportunity names to ensure consistent casing on these values. <pre>title(Opp_Name)</pre>
trim	Removes the specified substring from the beginning and end of a string. If no string is specified, remove the space characters from the beginning and end of a string.	`trim(field,valueToBeRemoved)` `field` is the dimension field or text string to remove the specified substring from. `valueToBeRemoved` is the value removed from the string. This parameter is optional. If you don't specify a valueToBeRemoved, the transformation removes the leading and trailing spaces. Example: Remove the two leading spaces before and after a string. <pre>trim(" Spaces Before and After ")</pre>
upper	Returns the string with all characters in uppercase. If string is null, then the result is null.	`upper(field)` `field` is the dimension field or text string to convert to uppercase. Example: You have a column First_Name that accepts freeform entry for users to enter their first name. As a result, some names are in uppercase, some in lowercase, and others in mixed case. Convert them to all to uppercase characters to make the casing consistent. <pre>upper(First_Name)</pre>

📝 **Note:** Functions that don't handle null values explicitly won't convert NULL values to be empty (for strings) or 0 (for numeric). For example, a `concat` output returns NULL if any of the concatenated fields have a null value on that row. So if the formula looks like `concat(first_name, last_name)`, the returned data with a NULL value for last_name is:

```
First Name | Last Name | Output
Alice      | Smith     | Alice Smith
Bob        | NULL      | NULL
```

To handle null values, you can use the `concat` function with another function. For example: `concat(first_name, coalesce(last_name, ''))`

Window Functions for Formulas

Use window functions in a Data Prep recipe to perform calculations across rows. For each row of input data, perform a calculation on a selection—or window—of related rows. Unlike aggregation that groups rows, window functions return a result for each row. For example, you can calculate the changes to the opportunity amount throughout the stages of each opportunity to see if you can hit your sales target.

When entering a formula, use the API name of the column in the expression, not the label. Surround the parameters with parentheses.

You can use the following window functions in a formula expression.

String Function	Description	Syntax

current	Returns the value from the current record in the partition.	`current(field)`
		
		 `field` is the field that you want to get the current value from.
		
		 Example: Get the value of the Amount field for the current row.
		
		 `current(Amount)`

dense_rank	Returns the rank of each record in the partition based on order. Ranks are consecutive—they don't repeat when the values match. For example, if the first three rows have the same value, then the first four ranks are 1, 1, 1, and 2.	`dense_rank()` Example: Calculate the dense rank for rows within each partition. `dense_rank()`

first_value	Returns the value from the first record in the partition.	`first_value(field)`
		
		 `field` is the field that you want to get the first value from.
		
		 Example: Get the original quote value for the Quote column.
		
		 `first_value(Quote)`

String Function	Description	Syntax

`lag`	Returns the value from the previous record in the partition.	`lag(field)`
		
		`field` is the field for which you want to obtain the previous value.
		
		Example: Calculate the percentage change in profit between the current and previous records.
		
		`(((current(Profit)-lag(Profit)))/current(Profit))*100`

`last_value`	Returns the value from the last record in the partition.	`last_value(field)`
		
		`field` is the field that you want to get the last value from.
		
		Example: Calculate the difference between the original quote and final quote.
		
		`first_value(Quote)-last_value(Quote)`

`lead`	Returns the value from the next record in the partition.	`lead(field)`
		
		`field` is the field for which you want to obtain the next value.
		
		Example: Get the next value for the Stage column.
		
		`lead(Stage)`

`rank`	Returns the rank of each record in the partition based on order. Repeats the rank when the values are the same, and skips as many on the next non-match. For example, if the first three rows have the same value, then the first four ranks are 1, 1, 1, and 4.	`rank()` Example: Rank the records in each partition. `rank()`
`row_number`	Returns the row number in the partition. Increments by 1 for every row in the partition. For example, if the first three rows have the same value, then the first four	`row_number()` Example: Assign a row number to each record in each partition. `row_number()`

String Function	Description	Syntax
	ranks are 1, 2, 3, and 4.	

Additional Functions for Formulas

These versatile functions can be used with the data types text, number, date_only, and datetime to build logic into your formulas.

When entering a formula use the API name of the column in the expression, not the label. The parameters section is surrounded by parentheses.

case Function

The `case` function handles if/then logic for formulas. This function can be used with TEXT, NUMBER, DATE_ONLY, and DATETIME data. The syntax is:

```
case primary_expr(optional)
 when condition
 then result_expr
 ...
 else
 default_expr
end
```

The syntax includes these keywords and parameters:

- **case** and **end** keywords begin and close the expression

- **when** and **then** keywords define a conditional statement. You can include one or more conditional statements.

- `condition` is a logical expression that can be evaluated to true or false. Construct this expression with any values, identifiers, logical operator, comparison operator, or scalar functions (including date and math functions) supported by EA SQL. Example:

  ```
  eg xInt < 5 or  price > 1000 and price <= 2000
  ```

- `result_expr` is an expression that can be evaluated by the EA SQL engine. You can include values, identifiers, and scalar functions, including date and math functions. The expression may evaluate to any data type, but the data type must be consistent among all conditional expressions. So if result_expr is of NUMERIC type, then result_expr2 … result_exprN must also be of NUMERIC type. Example:

  ```
  abs(price)
  ```

- `else` allows a default expression to be specified. The else statement must follow the conditional when/then statement. There can be only one else statement. This parameter is optional.

- `default_expr` is an expression that can be evaluated by the EA SQL engine. This is the outcome if the condition expression evaluates as false.

Here's a `case` function example. Type is a field in the dataset. "New Business" and "Existing Business" are values of the Type field.

```
case
    when Type = "New Business"
    then "NB"
    when Type = "Existing Business"
    then "EB"
    else
    Type
end
```

coalesce Function

The `coalesce` function returns the first non-null value in an expression. Otherwise, returns null. This function can be used with TEXT, NUMBER, DATE_ONLY, and DATETIME data. The syntax is: `coalesce(expr1,expr2)`

The syntax includes these parameters:

- `expr1` is the first field or text string to search for a non-null value
- `expr2`is the next field or text string to search for a non-null value

Upgrade or Edit Formulas for Converted Recipes
You see a different formula editor if your recipe was converted from Data Prep Classic to Data Prep and your formulas couldn't automatically upgrade. You're still able to edit and use these formulas, but to get the most formula options through EA-SQL and best experience, recreate the formula in the Data Prep recipe editor.

1. In a Data Prep recipe that was converted from Data Prep Classic to Data Prep, select a transformation with a formula.
   A formula that was not converted automatically looks like this. Continue to use these instructions to upgrade or edit your formula.

If the editor instead looks like this one, your formula was converted successfully and you use the Data Prep recipe formula editor instructions instead.
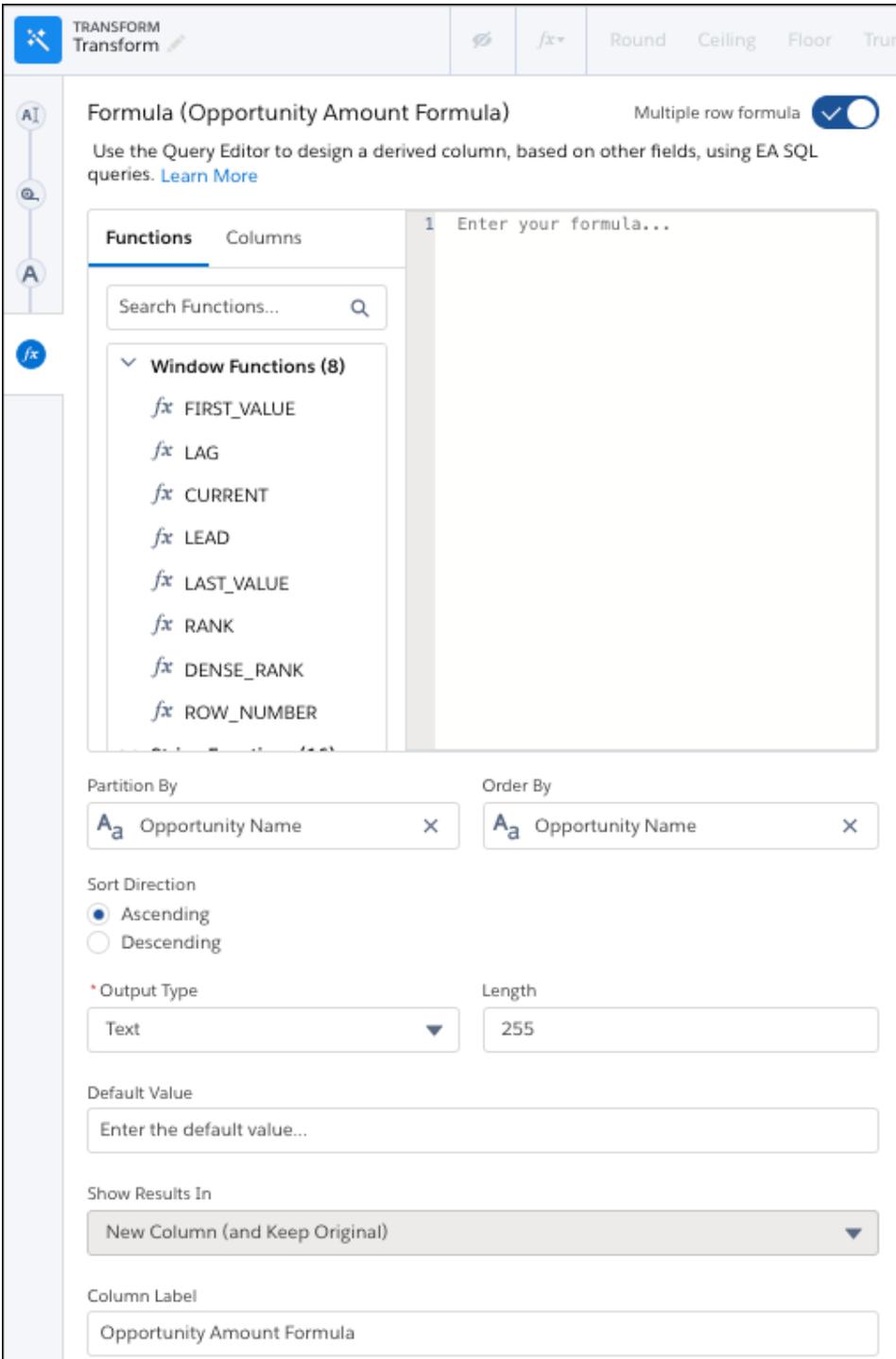
2. To upgrade your formula:

   a. Copy and save your original formula and select **Switch To SQL Editor**. The Data Prep recipe formula editor opens.

   b. Recreate your original formula using the correct syntax described in the Data Prep recipe formula editor instructions.

3. To edit your formula without upgrading, make your edits following the same syntax used when the formula was originally written. Do not select to switch to the SQL editor.

4. If needed, change the label of the calculated column.

5. When you have completed the formula, click **Apply** to add the transformation to the Transform node.
The new formula column appears in the recipe preview.

6. To view the Graph area, click the Collapse button ( ).

7. Save the recipe.

Numeric Operators and Functions for Unconverted Formulas

Use numeric functions to calculate values from measures in your recipe. For example, you can round, truncate, and determine the absolute value of a number. The arguments for numeric functions can be numbers or measure columns. If your recipe was upgraded to Data Prep, but your recipe could not convert, this help is for your unconverted recipe.

String Functions for Unconverted Formulas

Use string functions to create values based on other strings. For example, you can change the casing of text strings, concatenate values from multiple columns, and replace values. The arguments for string functions must be text strings or dimension columns. If your recipe was upgraded to Data Prep, but your recipe could not convert, this help is for your unconverted recipe.

Numeric Operators and Functions for Unconverted Formulas
Use numeric functions to calculate values from measures in your recipe. For example, you can round, truncate, and determine the absolute value of a number. The arguments for numeric functions can be numbers or measure columns. If your recipe was upgraded to Data Prep, but your recipe could not convert, this help is for your unconverted recipe.

📝 Note: To get a list of numeric functions for a Data Prep Classic recipe, see Numeric Functions for Formula Fields.

When entering a formula, column names must be enclosed in square brackets. Also, use the API name of the column in the expression, not the label. For example, you can enter the following expression.

```
round([Total_Amount]/[Unit_Price], 2)
```

You can use the following math operators.

Operator	Description
+	Calculates the sum of two values.
-	Calculates the difference between two values.
*	Multiplies the two values.
/	Divides the first value by the second.
()	Specifies that the expressions within the parentheses are evaluated first. All other expressions are evaluated using standard operator precedence.

You can also use the following numeric functions in an expression.

Numeric Function	Description	Syntax
abs	Calculates the absolute value of a number. The absolute value of a number is the number without its positive or negative sign.	`abs(number)` `number` is a number or measure column that you want to determine the absolute value of. Example: `abs([Amount])`
ceil	Rounds a number up to the nearest integer.	`ceil(number)` `number` is a number or measure column that you want to round up to the nearest integer.
exp	Returns a value of e raised to the exponent that you specify.	`exp(number)` `number` is a number or measure column to which you want to raise e.
floor	Returns a number rounded down to the nearest integer.	`floor(number)` `number` is a number or measure column that you want to round down to the nearest integer.
log	Returns the logarithm of the number in the base that you specify.	`log(base,number)` `base` is a number or measure column used as the base of the logarithm. `number` is a number or measure column that you want to take the logarithm of.
power	Raises a number to the power of another number.	`power(number,power)` `number` is a number or measure column that you want to raise to the specified power. `power` is a number or measure column that is the power that you want to raise the specified number to.
round	Returns the nearest number to a number you specify, constraining the new number by a specified number of digits.	`round(number,numberOfDigits)` `number` is a number or measure column that you want to round. `numberOfDigits` is a number or measure column that specifies the number of digits to round the specified number to.
sqrt	Returns the positive square root of a given number.	`sqrt(number)` `number` is a number or measure column that you want to find the square root of.
trunc	Truncates a number to an integer.	`trunc(number)` `number` is a number or measure column that you want to truncate to an integer.

String Functions for Unconverted Formulas

Use string functions to create values based on other strings. For example, you can change the casing of text strings, concatenate values from multiple columns, and replace values. The arguments for string functions must be text strings or dimension columns. If your recipe was upgraded to Data Prep, but your recipe could not convert, this help is for your unconverted recipe.

📝 **Note:** To get a list of string functions for a Data Prep recipe, see String Functions for Formula Fields.

For example, you can nest string functions in an expression to combine the sales territory and country into a single column. Then you can apply title casing to the results.

```
title(concat([Sales_Territory], " ",[Country]))
```

When you enter a formula, text strings must be enclosed in double straight quotes (`"This is a string."`). Column names must be enclosed in square brackets (`[Opportunity_Name]`).

You can use the following string functions in a formula expression.

String Function	Description	Syntax

concat	Returns a string by concatenating the values of the specified columns and input strings. For example, to display the close date as MM-DD-YYYY, concatenate the Close_Date_Month column, Close_Date_Day column, and Close_Date_Year column, and add a dash between each of them.	`concat(string1,string2,...)`
		
		`string1` is the first dimension column or text string to include in the concatenated value.
		
		`string2` is the second dimension column or text string to include.
		
		Example: To create the full name, concatenate the first and last names with a space in between them.
		
		`concat([OwnerId.FirstName]," ",[OwnerId.LastName])`

lower	Returns a string with all characters from the input string in lowercase. If the input string is null, then the result is null.	`lower(string)`
		
		`string` is the dimension column or text string to convert to lowercase.
		
		Example: The Account.Industry column uses different casing for industries, like "Media" and "media." To apply consistent casing to the columns values, convert them to lowercase.
		
		`lower([Account.Industry])`

ltrim	Removes the specified substring or leading spaces from the beginning of a string.	`ltrim(string,[substring]` `string` is the dimension column or text string to remove the leading spaces or specified substring from. `substring` is the value to remove from the string. This parameter is optional. If you don't specify a substring, the transformation removes the leading spaces.

String Function	Description	Syntax
		Example: Trim the leading space characters from a string. ```ltrim(" 5 Spaces Before")```

replace	Replaces a substring with the specified characters. If any of the arguments are null, then the function returns null. This function is case-sensitive.	`replace(string,searchString,replacementString)`
		
		`string` is the dimension column or text string that contains the substring to be replaced.
		
		`searchString` is the substring to replace. If searchString is an empty string, the function returns null.
		
		`replacementString` is the value that replaces the substring.
		
		Example: Change the account name from "salesforce.com" to "Salesforce" in the Account_Name column.
		
		```replace([Account_Name],"salesforce.com","Salesforce")```

rtrim	Removes the specified substring or trailing spaces from the end of a string.	`rtrim(string,[substring]`
		
		`string` is the dimension column or text string to remove the trailing spaces or specified substring from.
		
		`substring` is the value to remove from the string. This parameter is optional. If you don't specify a substring, the transformation removes the trailing spaces.
		
		Example: Trim the 2 trailing space characters from a string.
		
		```rtrim("2 Spaces After ")```

substr	Returns characters from the string, starting at the specified position and of the specified length.	`substr(string,position,[length])`
		
		`string` is the dimension column or text string to extract the substring from.
		
		`position` is the starting character position of the substring. The first character in a string is at position 1. If position is negative, then the position is relative to the end of the string. A position of -1 denotes the last character.
		
		`length` is the number of characters to return. If length is 0, the output is an empty string. If length is negative, then the function returns null. This parameter is optional.
		
		Example: Return the 13th, 14th, and 15th character from each account ID.
		
		```substr([Account_Id],13,3)```

title	Returns the string with the initial character of every word in uppercase and the remaining characters in lowercase. For example, "united states" becomes "United States."	`title(string)`
		
		`string` is the dimension column or text string on which to apply title casing.
		
		Example: Apply title casing on the opportunity names to ensure consistent casing on these values.
		
		```title([Opp_Name])```

String Function	Description	Syntax

trim	Removes the specified substring (or leading and trailing spaces) from the string.
	
	💡 **Tip:** Use the trim function in a SAQL query to trim other types of characters.
		`trim(string,[substring])`
		
		`string` is the dimension column or text string to remove the specified substring from.
		
		`substring` is the value removed from the string. This parameter is optional. If you don't specify a substring, the transformation removes the leading and trailing spaces.
		
		Example: Remove the two leading spaces before and after a string.
		
		`trim("  Spaces Before and After  ")`

upper	Returns the string with all characters in uppercase. If string is null, then the result is null.	`upper(string)`
		
		`string` is the dimension column or text string to convert to uppercase.
		
		Example: You have a column First_Name that accepts freeform entry for users to enter their first name. As a result, some names are in uppercase, some in lowercase, and others in mixed case. Convert them to all to uppercase characters to make the casing consistent.
		
		`upper([First_Name"])`

Predict Missing Values Transformation: Fill In Missing Values

Use the Predict Missing Values transformation in a Data Prep recipe to complete your data by filling in missing values in a dimension column. Tableau CRM intelligently predicts values based on values in other strongly correlated columns in your data.

Consider these limitations before using this feature.

- If there aren't enough rows to make accurate predictions, Tableau CRM doesn't insert predicted values.
- You can't apply transformations on predicted columns.
- Recipes that predict values can take longer to run.

1. In a Transform node of a Data Prep recipe, select the dimension column with missing values in the Preview tab.

2. In the Transform toolbar, click the Predict Missing Values button (⬚).

USER PERMISSIONS
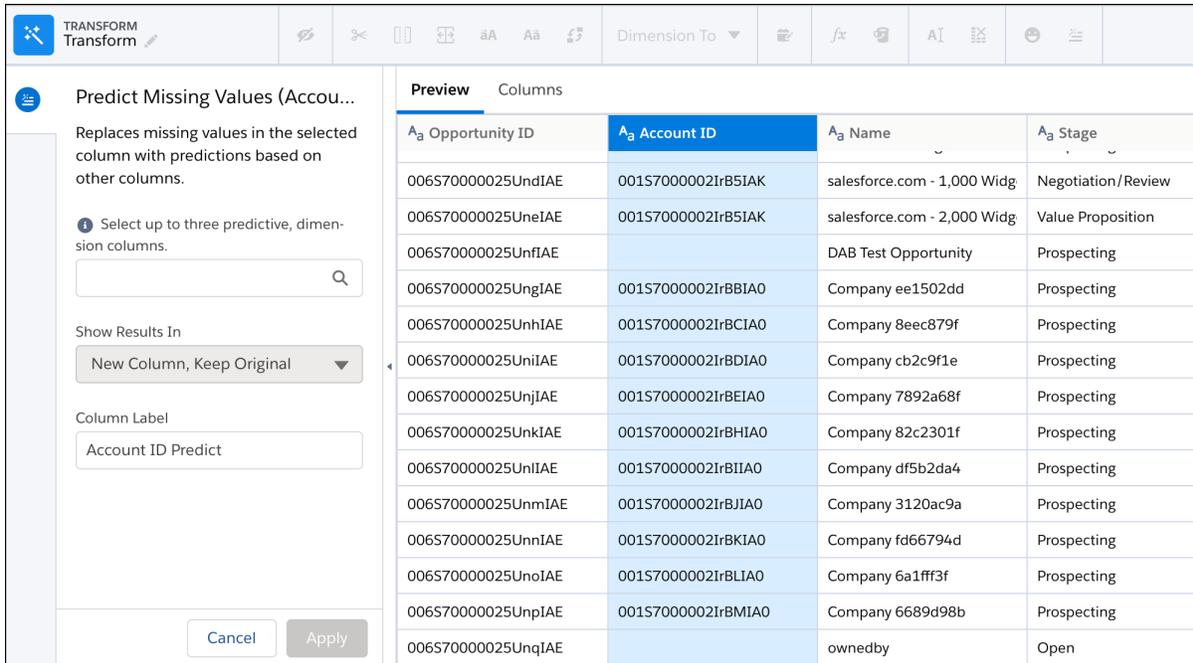
To create a recipe:
- Edit Analytics Dataflows OR Edit Dataset Recipes

3. Select up to three dimension columns to use to predict the missing values for the selected column.

> 💡 **Tip:** To make an accurate prediction, each column must have less than 200 unique values. Also, verify that these predictive columns contain clean, quality data. For example, you have an Education predictive column that contains values such as "Bachelors Degree" and "Bachelors." Use the bucket transformation to bucket field values with the same meaning. Then use the column with the clean data as a predictive column. For more information about bucketing, see Categorize Dimension Column Values into Buckets.

4. If needed, change the column label. This label appears as the column header in the dataset.

5. Click **Apply** to add the transformation to the Transform node.

   The preview shows the original column with the missing values and the new column with "Predict" at the end of the header. The preview shows "Prediction TBD" for predicted values in the new column. The predicted values don't appear until after you run the recipe.
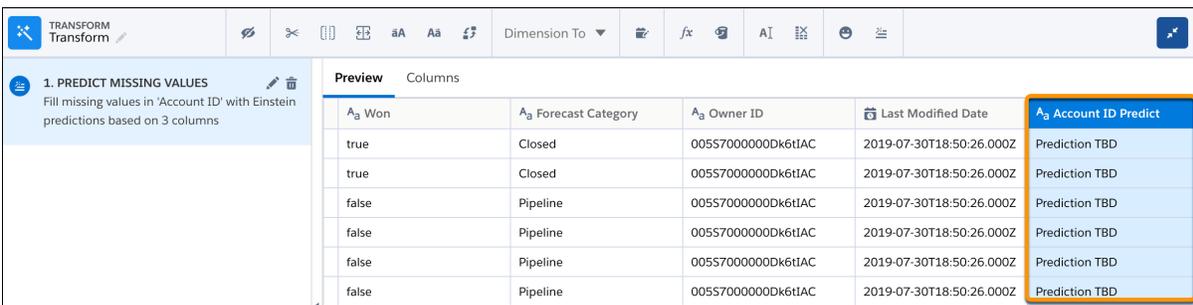
6. To view the Graph area, click the Collapse button ( ).

Run the recipe to generate the predictions. You can view the dataset as a values table to see the predictions. If needed, add the Drop Columns transformation after the Predict Missing Values transformation to drop the original column from the dataset.

Split Transformation: Break Up Column Values

You can split the strings in a dimension column into two values by specifying a delimiter. To split column values into more than two parts, add multiple instances of the Split transformation. For instance, you can use 3 Split transformations in a Data Prep recipe to split the full address into the following components: street address, city, state, and zip code.

1. In a Transform node of a Data Prep recipe, select the dimension column in the Preview tab.

2. In the Transform toolbar, click the Split button (⬅⬆).
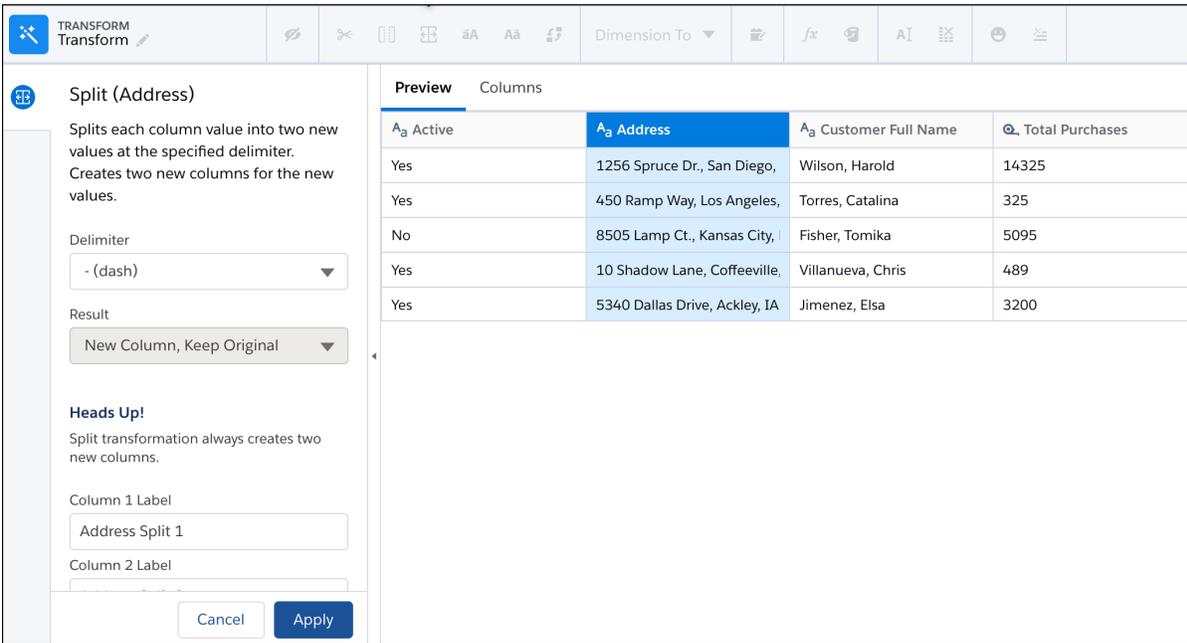
3. Select the delimiter in the Delimiter field.

   For example, if address components are separated by a comma in the Address field, specify comma as the delimiter.

4. Specify the column labels for both new columns.

5. Click **Apply** to add the transformation to the Transform node.
   Going back to the address example, the Address field is split into two. The Split transformation splits the value at the first occurrence of the delimiter. In this case, the first comma appears after the street address. So the full address is broken into the street address and the rest (city, state, and ZIP code).
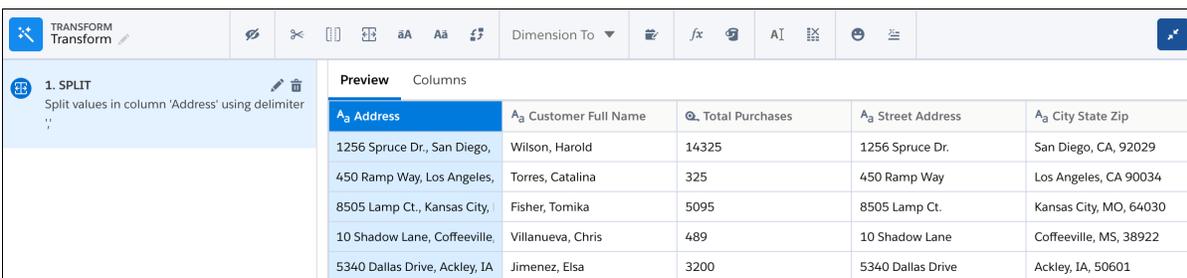
6. If needed, add the Split transformation again to further split the column values.

   For example, to finish splitting the city, state, and ZIP code, we can add two more Split transformations.

7. To view the Graph area, click the Collapse button ( ).

8. Save the recipe.

If you don't need the original column that you split, add the Drop Columns transformation after the Split transformation to drop it from the recipe. From that point, the original column doesn't appear in Preview anymore.

Time Series Forecasting Transformation: Forecast Measures (Pilot)

Make decisions today based on forecasts about tomorrow with time series forecasting. A time series forecast takes an ordered series of points and then intelligently forecast what the next values will be. For example, estimate units sold for the next four quarters based on the last 5 years of sales. Use the Time Series Forecasting transformation in a Data Prep recipe to run forecasts based on historical data.

> **Note:** Join the Smart Transform Pilot Program to try out all pilot features in Data Prep recipes. Pilot features can change with each release as existing features become generally available or are retired and new pilot features are added to the program. We provide the Smart Transform Pilot Program to selected customers who agree to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. Each Data Prep pilot feature isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features. You can provide feedback and suggestions for the Smart Transform Pilot Program in the Trailblazer Community.

Before you create a Time Series Forecast transformation, aggregate the measure that you want to forecast and group it by the forecast period. For example, you want to forecast the total amount and number of opportunities for the next six months. In an Aggregate node, add both measures as aggregates (1) and group rows by the Close Date column's year and month (2).

When the recipe runs, the Aggregate node creates a row for each time period of data. If the aggregated data is missing data for a particular time period, the Time Series Forecasting transformation fills in the missing row behind the scenes, entering 0 for forecasted columns and null for other columns.

1. In a Transform node of a Data Prep recipe, select any column in the Preview tab.

2. To add a Time Series Forecasting transformation to the Transform node, click the Time Series Forecast button (⬚).
   The Time Series Forecasting panel opens.

3. In the Group Dates By field, choose the time period by which you want to analyze the timeseries data, such as week or month.

4. Then select the dimension columns that identify the time period.

    For example, if you group dates by year-month, select the year and month columns. You created these date part columns in the Aggregate node in the prerequisite step.

5. Ignore the Seasonality field. This feature isn't available yet.

6. In Forecast Length, specify the number of periods to forecast.

    For instance, if you group dates by year-month and set the forecast length to 12, you get forecasts for the next 12 months.

7. In Columns to Forecast, choose which measure columns to forecast—you can forecast up to five columns.

8. In Model, select the forecast model to use.

    You can use one of these models.

- Additive uses Holt-Winters Additive method. Use this method when the seasonal variations are roughly constant throughout the series.
- Multiplicative uses Holt-Winters Multiplicative method. Use this method when the seasonal variations change proportionally to the average (level of the series).

9. If the data in the last period is incomplete, select **Ignore Last Time Period**.

For example, don't use the last period if you're currently in the middle of that period. Because the data for that period is incomplete, the forecast can be misleading.

10. To add the transformation to the Transform node, click **Apply**.

11. To view the Graph area, click the Collapse button ( ).

12. Save the recipe.

Run the recipe to see the forecasts. Each forecast column shows the results for every time period, including the historical ones. For example, the Rows Forecast and the Sum of Amount Forecast columns show the forecasts. To combine the forecasts for future periods with the historical values in a single column, use a Coalesce formula for each measure—similar to this one.

```
coalesce(COUNT_Rows, COUNT_Rows_forecast)
```

The historical values and forecasts appear in a single column, as shown here.

CLOSEDATE Month ↑	CLOSEDATE Year ↑	Rows	Rows Forecast	Actual and Forecasted Rows
02	2021	48	210	48
03	2021	25	113	25
04	2021	10	62	10
05	2021	10	33	10
06	2021	10	22	10
07	2021	12	18	12
08	2021	6	17	6
09	2021	15	13	15
10	2021	6	17	6
11	2021	7	13	7
12	2021	0	12	12
01	2022	0	15	15
02	2022	0	18	18
03	2022	0	21	21
04	2022	0	24	24
04	2022	0	27	27

> 📝 **Note:** The Time Series Forecasting transformation and timeseries SAQL function can produce different results because the underlying algorithm optimizations are different. To see which time series algorithm best fits your data, graph the forecasted values against the historical values for both methods.

Preview Results in a Data Prep Recipe

As a best practice, catch mistakes early by reviewing the results of each node and transformation that you add to a Data Prep recipe. Each user can initiate up to 4,000 previews per hour.

Watch a Demo: ▶ Preview Recipe Changes to Data (English Only)

The preview shows a sample of the data—up to 5,000 rows. It shows the latest available data, except when you preview a connected object. When previewing a connected object, the preview might show outdated sample data. The sample data for a connected object updates when you run data sync and the sample data is more than 24 hours old. For example, if the data sync just ran, but your sample data isn't more than 24 hours old, you will not see the updated sample data right away.

> **Note:** The column profile is also based on the sample data. It also doesn't include changes to the connected object data within the last 24 hours.

1. In the Data Prep recipe, select a node in the graph to view its results.
   For example, selecting an Aggregate node shows the following preview results.

> **Note:** If you add a Filter node, the preview filters the currently sampled rows. For example, the recipe processes 20,000 rows. The preview shows 5,000 sample rows for the Aggregate node. If you add a Filter node after the Aggregate node, you might see only 4,000 of the same sampled rows. The preview doesn't retrieve an updated sample of 5,000 rows for the Filter node.

2. If needed, edit the node to ensure it produces the correct results.

3. In a Transform node, select a transformation step.
   For example, selecting a Formula transformation that concatenates the billing state and country shows the following results.

4. If needed, edit the transformation to ensure it produces the correct results.

5. To hide a column from the preview, select a column in the Preview tab and click the Hide button ( ).

You can hide columns to focus on columns that are impacted by a node or transformation. Hiding a column from preview doesn't remove it from the recipe.

6. To view the columns only and hide the data, click the Columns tab.

7. To save the recipe, click **Save**.

Profile Columns to Understand Data in a Data Prep Recipe

Run column profiling on sample data to estimate key stats about columns, such as the frequency of values and percent of columns with missing values. Column profiling is especially useful when you are combining data from different sources, where inconsistencies are often introduced.

Column profiling works on the latest available data, except when you profile columns from a connected object. If you profile a column from a connected object, the profile might be based on outdated sample data. The sample doesn't include changes to the connected object data within the last 24 hours. The sample data for a connected object updates when you run data sync and the sample data is more than 24 hours old. For example, if the data sync just ran, but your sample data isn't more than 24 hours old, the sample data isn't updated right away.

To run a column profile, select the column in the Preview or Columns tab, and then click ⓠ. The Profile tab shows different results based on the data type of the column. Use the results to understand the shape of the data and identify potential data quality issues. The statistics provided for column profiles are based on sample rows, not all rows in the recipe.
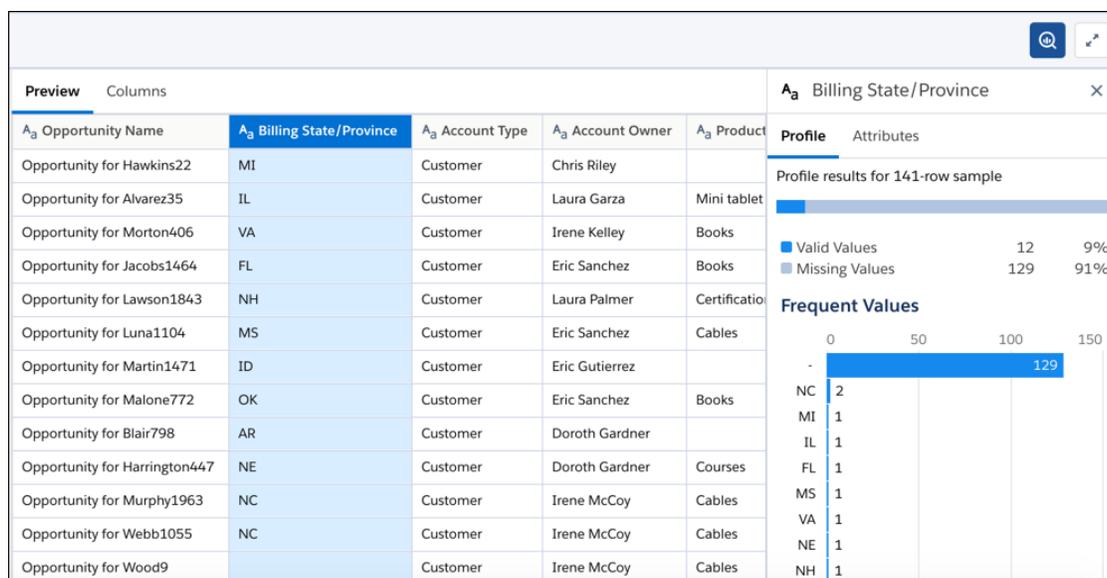
The Profile tab shows the following information.

- Valid values—Valid values are values that aren't missing.

- Missing values—Missing values, which include nulls and empty strings (""), can lead to skewed calculations. For more information, see Handle Missing Values.

- Zero values—For measures, the column profile shows the count of zeroes.

- Frequent values—The Frequent Values chart shows the top dimension values by count (based on the sample), sorted in descending order. To see the percentage of the sample total, hover over a bar. In the following example, Florida (FL) represents 8.7% of all sample values in the Billing State/Province column.
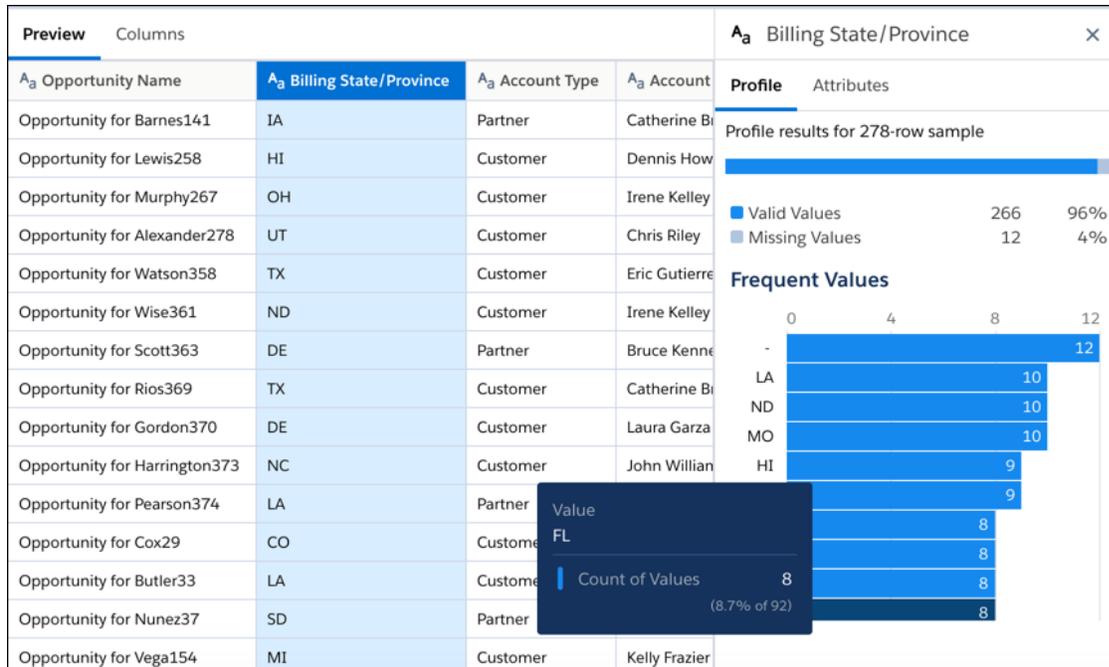
- Histogram—The histogram shows the shape of the sample data for measure and date columns. *Shape* refers to the distribution of the data over equal-length ranges. Use the histogram to understand the shape of the data and identify outliers. Outliers are important because they can distort calculation results. For a measure column profile, the histogram shows the count of rows, maximum value, minimum value, median value, and average value for each range. To see the percentage of the sample total, hover over a bar. For a date column profile, the histogram shows the count of rows for the selected date grouping. You can group by different time frames, including Year-Month, Year-Quarter, Year, Quarter, and Month.

The Attributes tab shows the label, API name, and data type of the selected column. For measure columns, it also shows the scale. For dimension columns, it also indicates whether the column contains multiple values.

Limitations When Using Data Prep

Consider the following Data Prep limitations before building recipes.

Some functionality isn't available, including the following features.

* You can't use Data Prep Classic, the old version, to open recipes created with Data Prep.

Clean, Transform, and Load Data with Data Prep Classic

Use a recipe to clean and combine data from multiple datasets or connected objects. Add bucket and formula fields, filter rows, transform field values, convert field types, and standardize date formats. You can then output the results to a new target dataset.

Watch a Demo: ▶ Get Started with the Recipe Editor (English Only)

When you're ready, follow these steps to create a recipe.

Create a Recipe with Data Prep Classic
Use a Data Prep Classic recipe to transform data before loading it into a dataset. Only Government Cloud users can create recipes with Data Prep Classic. All other users must use Data Prep to create recipes.

Set Up the Preview in a Recipe
A recipe displays a preview of your data on the Preview tab as you work. You can set the columns and number of rows that appear in this preview. If you are working with a large set of data, reducing the size of the preview speeds up the data refresh as you add steps to your recipe.

Add More Data in a Recipe
You can add columns of data from related objects to existing data in a recipe. Depending on how you want to combine the new data, use one of the following methods: lookup, left join, right join, inner join, or full outer join.

Add Rows in a Recipe with Append
Use append to add rows to the data in your recipe from another dataset or connected object. Fields are mapped automatically, or you can map them manually.

Clean and Prepare Data Intelligently with Column Profiles and Smart Suggestions
The column profile gives you key insights into the quality of your data and suggests additional transformations to help you clean and prepare it. This profile is especially useful when you are combining data from different sources, where inconsistencies are often introduced.

Add a Filter in a Recipe

Filter data in a recipe to remove rows that you don't need in your target dataset.

Bucket a Measure Field in a Recipe

Add a field to a recipe to bucket values in a specified measure field.

Bucket a Dimension Field in a Recipe

Add a field to a recipe to bucket values in a specified dimension field.

Bucket a Date Field in a Recipe

Add a field to a recipe to bucket values in a specified date field.

Add a Formula Field in a Recipe

Add a formula field to calculate new values from measures and dimensions in your recipe.

Aggregate and Group Data to a Different Grain

Large datasets can be hard to digest due to the amount of information and low-grain details. Aggregation allows these datasets to be rolled up to a higher granularity, thus allowing users to create recurring summary statistics and join datasets with different granularities. You can also aggregate data to perform calculations on grouped records without aggregating the measures. For instance, group by website session IDs and then calculate the average time on each page and total number of clicks.

Transform Fields in a Recipe

Data is not always consistent, especially when you combine data from different sources. In a recipe, you can transform fields to ensure that values are consistent in your target dataset. Change case, split values to get just the parts you need, and replace incorrect values.

Standardize Date Formats

If a dimension field contains dates in different formats, use the Format Dates recipe transformation to standardize the format for all values in the field. A consistent format enables you to correctly filter and group records by date, including filtering by date component, such as month. It also ensures that you can successfully convert the field type from dimension to date.

Convert Field Types in a Recipe

The type assigned to a dataset field determines how you can query that field's data. For example, you can filter and group by a dimension or date field, or perform math calculations on a measure field. When you load data into a dataset, Tableau CRM sometimes tags a dataset field with the wrong type. If needed, convert fields to the correct types.

Predict Missing Values in Dimension Columns

When a dataset or connected object has missing values in a dimension column, Tableau CRM can fill in missing values to complete your data. Tableau CRM intelligently predicts values based on values in other strongly correlated columns in your data.

Navigate Columns in a Recipe

As you add data and transform fields, the number of columns in your recipe preview increases. This makes it harder to find the columns that you want to work with. Use the column view to quickly find the columns that you need, and hide the ones you don't.

Navigate and Edit Recipe Steps

As you prepare your data, each change you make appears as a recipe step on the left. Think of these steps as your recipe history. You can move back and forward through this history to see how the data looks at different stages of the recipe. If you don't like what you see, you can edit or remove any step.

Save a Recipe

If you're not ready to create the target dataset, save the recipe and come back to it later. That saves your steps without you having to create the dataset.

Create a Recipe with Data Prep Classic

Use a Data Prep Classic recipe to transform data before loading it into a dataset. Only Government Cloud users can create recipes with Data Prep Classic. All other users must use Data Prep to create recipes.

> **Note:** To start creating a Data Prep recipe, see Create a Recipe with Data Prep.

1. In Analytics Cloud, click the gear icon (⚙) and then click **Data Manager** to open the data manager.
The data manager opens in a new browser tab.

2. In the data manager, click the Dataflows & Recipes tab.

3. In the Dataflows & Recipes tab, click the Recipes subtab.
The Recipes subtab displays a list of your existing recipes.

4. Click **Create Recipe**.

5. From the list of datasets, select the one that contains the data you want to prepare. Alternatively, you can click the **Connected Data** tab to select a connected object if your org has Data Sync enabled. Data Sync is enabled by default if you turned on Tableau CRM after the Winter '20 release.

> **Note:** Make sure the connected object that you select has synced and completed at least once.

You are prompted to enter a name for the recipe.

6. Enter a recipe name and then click **Next**.

> **Note:** The recipe name you enter here is also used for the target dataset name.

The recipe page opens, displaying a preview of the data from your selected dataset or connected object.

Set Up the Preview in a Recipe

A recipe displays a preview of your data on the Preview tab as you work. You can set the columns and number of rows that appear in this preview. If you are working with a large set of data, reducing the size of the preview speeds up the data refresh as you add steps to your recipe.

> **Note:** To speed up the preview when you create a recipe, multi-value fields aren't included, by default. The recipe editor hides these fields from the preview because they sometimes cause recipes to yield unexpected results. For example, a recipe with multi-value fields may result in query timeouts or fail to load or run. If needed, you can manually add these fields to the recipe.

If you're working with a connected object of more than 10,000 rows, Tableau CRM loads a sample of 10,000 rows into the recipe. As you work in a recipe, the sample size used in the preview is always visible at the top of the page. Use the selector to choose to see the first 1000, 5000, or 10000 rows.

> **Note:** Filters that you add in a recipe only apply to the sample data. They don't result in the recipe editor retrieving an updated sample.

To help you distinguish between nulls and empty strings, the preview shows null values as "null" in gray italics and empty strings as blank values.

If you don't need a column in your preview, select **Hide Column** from the column menu.
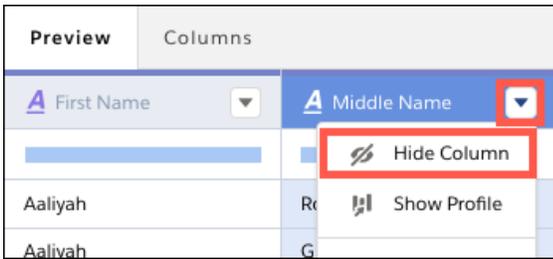
If you're working with numerous columns, click the columns tab to work from a list view of columns without the data.
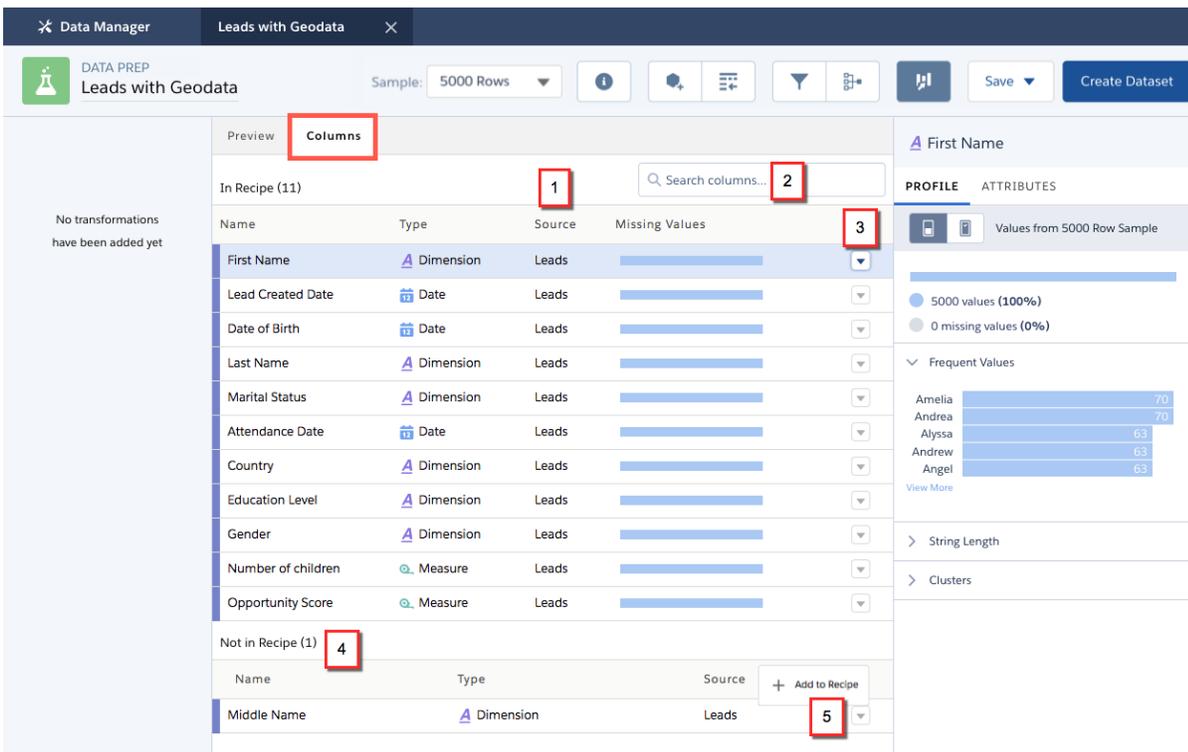
All the columns included in your recipe appear first (1), color-coded by source, with their type, the dataset they come from, and a missing values indicator. To find a column, start typing the name in the search box (2) to narrow down the list. To hide a column, click its menu button and select **Hide Column**. Columns that you hide in the recipe appear in a separate list (4). To add one of these columns back, select **Add to Recipe** from its menu (5).

To work on a column in either the Preview or Columns tab, click it. The column menu options and smart suggestions available are the same on both tabs.

> **Note:** The recipe preview can display up to 100 fields. If you remove a field from the preview, by default it's not included in the target dataset. However, you can reselect any field when you create the dataset.

Add More Data in a Recipe

You can add columns of data from related objects to existing data in a recipe. Depending on how you want to combine the new data, use one of the following methods: lookup, left join, right join, inner join, or full outer join.

> 📝 **Note:** To add columns from a related object in a Data Prep recipe, see Join Node: Add Related Columns of Data to the Recipe. To do it in a dataflow, see augment Transformation.

Watch a Demo: ▶ Add Data in a Recipe (English Only)

When you add data to a recipe, you select the dataset or connected object that contains the fields that you want to add. You also select the key fields to match the records between the existing and new data streams. For example, to add User fields to Opportunity records, use the User Id field from the User object and Opportunity Owner Id field from the Opportunity object as the keys. When defining the key, you can use a single field or, if necessary, multiple fields. Note that multi-value fields can't be used as keys.

Let's look at an example to illustrate how you add data. Imagine you started a recipe from an opportunities dataset, and you want to add more opportunity details from another dataset. You can use the opportunity ID field from each dataset as the keys.

1. On the recipe page, click the Add Data button (🔾₊).

2. Select either the Datasets or Connected Data tab, and click the dataset or object that you want to add.

3. Choose the join type. By default, Lookup is selected.

4. Select the keys from the left and right data streams. The left data stream is the existing data. The right is the new data that you want to add. Tableau CRM suggests keys if it finds related fields.

5. To select additional keys, click **+Add Key Pair**.

If needed, use multiple keys to match records. For example, imagine you're adding contact information to lead data. Using names alone to match can result in duplicate matches when different leads have the same name. Use two keys to match on name and company to ensure you're matching the right data. You can use up to 5 keys.

> 🛑 **Important:** Tableau CRM datasets contain date and time component fields, such as Year, Quarter, and Month, that are derived automatically from existing date and time fields. Use of these component fields as keys isn't supported.

6. In the **Columns to Include** section, select the columns to include in the recipe.

Only the key columns are selected by default. So make sure that you select all the columns that you need in the recipe. The columns are color coded to indicate their source.

📝 **Note:** The results preview can display up to 100 columns. Modify the preview or deselect columns to ensure that you can see just the columns that you need.

7. Click **Done**.

Each time you add data, a new step is created in your recipe, and the added fields appear in your recipe preview. Hover over a step and

click ⌄ to access the Edit and Remove options.

SEE ALSO:

> Lookup
>
> Left Join
>
> Right Join
>
> Inner Join
>
> Full Outer Join
>
> Considerations When Using Joins

Add Rows in a Recipe with Append

Use append to add rows to the data in your recipe from another dataset or connected object. Fields are mapped automatically, or you can map them manually.

📝 **Note:** To append rows in a Data Prep recipe, see Append Node: Stack Rows from Different Sets of Data. To do it in a dataflow, see append Transformation.

Let's look at an example of how you can use append in a recipe. Imagine you're a US company that uses separate Salesforce orgs for its US and Canada operations. You sync the US opportunities in your local org, and use a remote Salesforce connection to sync your Canada opportunities. You want to combine these US and Canada opportunities to create a single North America opportunities dataset.

Opportunity Name	Billing Country	Currency	Stage	Amount	Close Date	Lead Source
Opportunity for Vasquez1005	USA	USD	Closed Won	269,400	8/24/17	Trade Show
Opportunity for Collier1376	USA	USD	Prospecting	89,230	2/22/18	Trade Show
Opportunity for Moran1395	USA	USD	Closed Won	234,500	11/5/16	Word of mouth
Opportunity for Keller1431	USA	USD	Needs Analysis	2,894,390	2/10/18	Trade Show

Opportunity Name	Billing Country	Currency	Stage	Value	Close Date
Opportunity for Frazier644	Canada	CAD	Qualification	1,528,750	2/28/18
Opportunity for Weber1078	Canada	CAD	Value Proposition	36,960	3/17/18
Opportunity for Todd162	Canada	CAD	Prospecting	465,500	3/28/18
Opportunity for Barnett1692	Canada	CAD	Needs Analysis	4,708,010	1/21/18

You've already started the recipe from your synced US opportunities (1), to which you now want to append data from your synced Canada opportunities dataset (2). Notice that the fields are not exactly the same. The US data has an Amount field and the Canada data has a Value field. The US data has a Lead Source field, which the Canada data doesn't have. These kinds of differences affect how appended fields can be mapped to fields in the recipe.

- Appended fields are mapped automatically to recipe fields with the same name and type. You can change these mappings. In this example, the Currency field will be mapped automatically.

- Appended fields with different names are not mapped automatically, but you can map them to any fields of the same type. In this example, you have to manually map the Value field to the Amount field.

- Fields with different types can't be mapped.

- Recipe fields that are not in the appended data are added to the appended rows and populated with nulls. In this example, the Lead Source field will be added to the appended rows.

  📝 Note: If null measure handling in datasets isn't enabled for your org, measure fields are populated with zeros when they're added to appended rows.

Follow these steps to append data in a recipe.

1.
   On the recipe page, click the Append Data button (⮐).

2. Select either the DATASETS or CONNECTED DATA tab, and click the dataset or object that you want to add. In our example, you would select the Opportunity dataset from the CanadaOrg connection.

3. On the map fields dialog, review how Tableau CRM has mapped the recipe fields in the first column to fields from the data to append in the second column. Remember, fields with the same name and type are mapped automatically.

> 📝 **Note:** By default, hidden fields in your recipe aren't mapped. To map hidden fields, click the **Show hidden fields** link and accept or change the mappings.

4. To change a field mapping, click the field in the FIELD TO APPEND column, and select a field. You can only select from fields of the same type as the recipe field that you're mapping to.

In our example, you don't have a field to map to the Lead Source field, so you leave that as it is. The Amount field has not been mapped because the corresponding field in the Canada data was named Value. So you map the Value field to the Amount field.
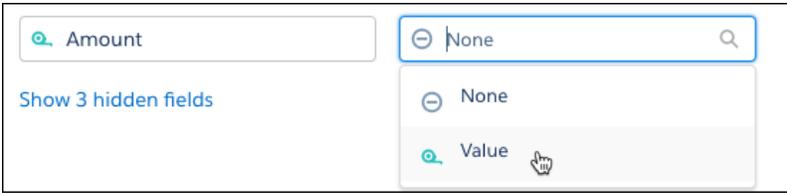
5. When you finish mapping fields, click **Continue**.

6. On the confirmation message, click **Append**.

> **Note:** Rows that you append in a recipe do not appear in the recipe preview, but are added to the target dataset when you run the recipe.

When you append data, a new step is created in your recipe.
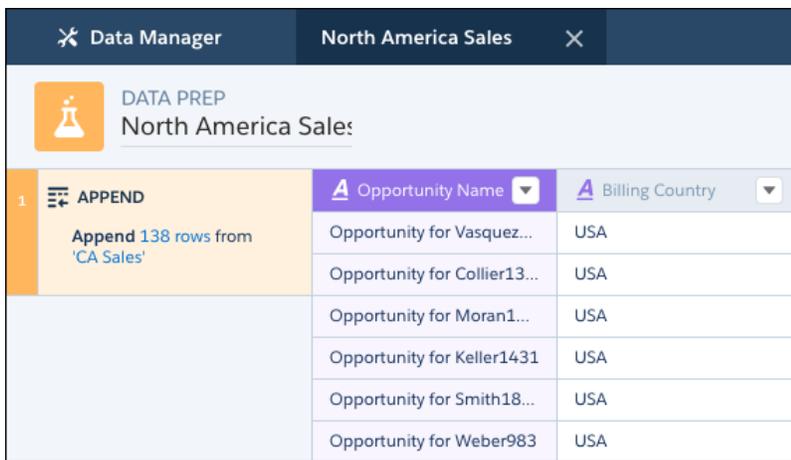
When you run the recipe, the rows are appended in the target dataset.

Opportunity Name	Billing Country	Currency	Lead Source	Amount	Stage
Opportunity for Morton591	USA	USD	Word of mouth	1,741,100	Proposal/Price Quote
Opportunity for Butler582	USA	USD	Partner	224,800	Prospecting
Opportunity for Frazier644	Canada	CAD	-	1,528,750	Qualification
Opportunity for Weber1078	Canada	CAD	-	36,960	Value Proposition

Here, the Lead Source field has been added to the Canada rows and populated with null values. The Value values from the Canada rows now appear in the Amount column.

Clean and Prepare Data Intelligently with Column Profiles and Smart Suggestions

The column profile gives you key insights into the quality of your data and suggests additional transformations to help you clean and prepare it. This profile is especially useful when you are combining data from different sources, where inconsistencies are often introduced.

Note: To profile columns in a Data Prep recipe, see Profile Columns to Understand Data in a Data Prep Recipe.

Let's look at an example of how you can use the profile to review the quality of data in a set of leads. Select the column you want to analyze to see its profile on the right of the recipe editor. To open the column profile if it isn't visible, click ⏹ or select Show Profile from the column's menu.
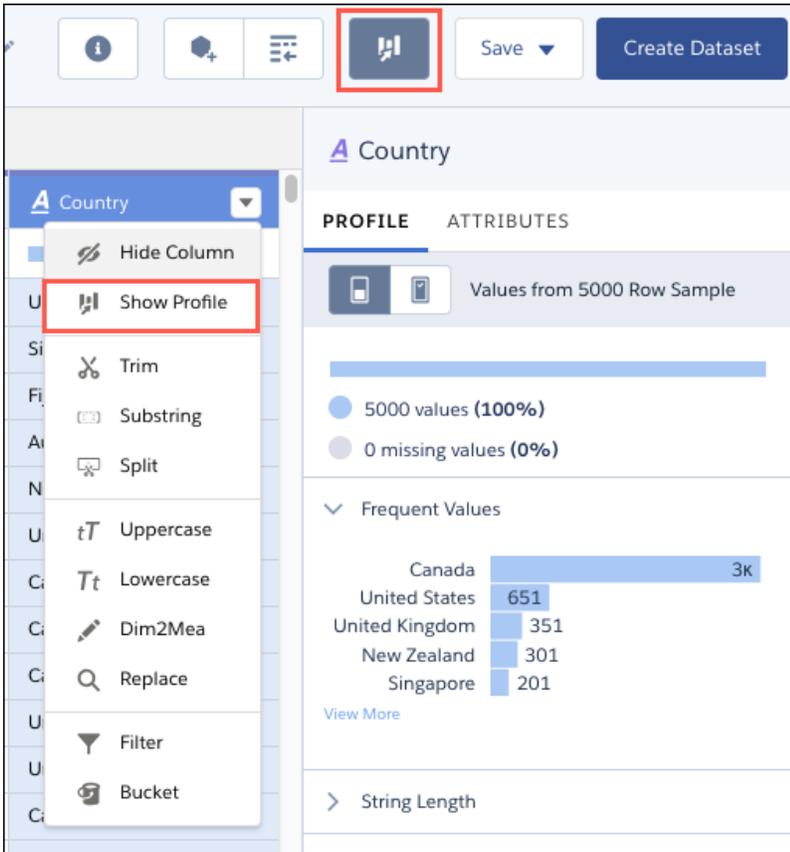
Note: The sections that you see in the profile depend on the type of column.

Here are the sections that you see in the column profile.

Sample Size

This section shows the number of rows used to generate the column profile data. To use a larger sample, click

⏹ .

Note: The option to use a larger sample of rows is available only for non-calculated dimension fields. The option doesn't appear if the sample size is greater than the number of rows in the recipe.

Valid Values
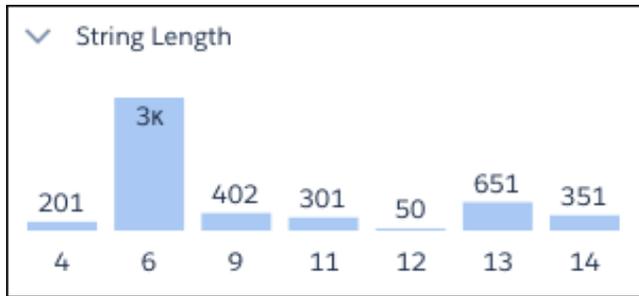
This section gives you an idea of how the column is populated. A high proportion of null values can indicate lower quality data of limited usefulness in lenses and dashboards.

Frequent Values
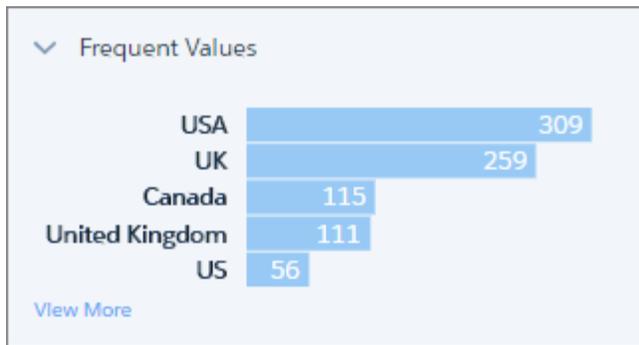
This section shows the count for each of the top five values in the column. To see the count for all values in the sample, click **View More**. This information gives you a quick insight into the variation in your data.

This example shows different values for the same country, which can impact grouping and filtering in a lens or dashboard.

For dimension and date columns, click one or more bars to see suggested transformations for those values in the Suggestions bar. In this example, if you select the US value, Tableau CRM suggests replacing it or filtering it out.

String Length (Dimension Columns Only)
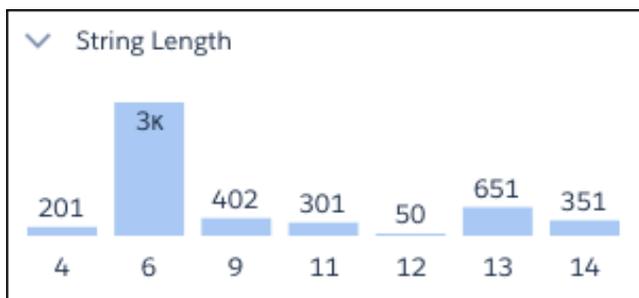
This section gives you a breakdown of the number of characters in dimension values. A wider variation than expected can indicate inconsistency issues with data from mixed sources.

Clusters (Dimension Columns Only)
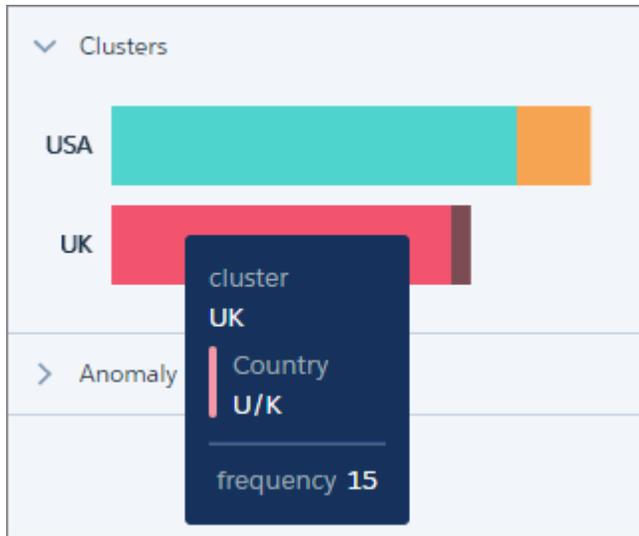
Clusters are groupings of similar values presented as a stacked bar. To see which value a segment represents, hover over it.

In this example, Tableau CRM has found instances of "USA" and "US", and "UK" and "U/K" in the Country column. These values might be genuinely different, but they could also indicate inconsistencies in data from different sources, or simply typos.

📝 **Note:** Clustering works best with English language values. Values in other languages can produce unexpected results.

To help you with your data cleansing, Tableau CRM suggests the Bucket By Cluster transformation whenever it identifies clusters. This transformation works just like regular bucketing, but gets you started by suggesting the buckets and values.
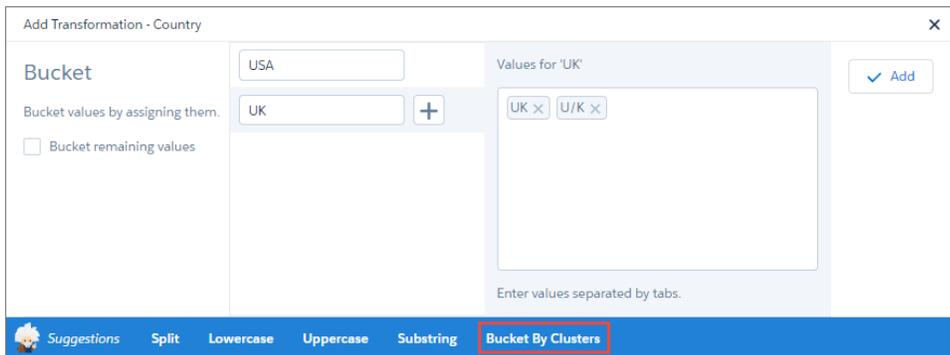
You can add the transformation as is, or add more buckets and values.

Anomalies (Dimension Columns Only)
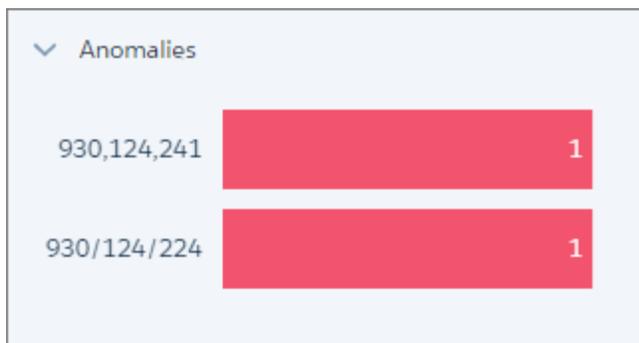
This section displays values that are not consistent with other values in the column. In this example, Tableau CRM has identified two values in a reference column that in which most values are in the format 123-456-789.

Again, to help you clean your data, Tableau CRM suggests transformations when you click one or more bars. For example, you can replace the values or filter them out.

Histogram (Measure and Date Columns Only)
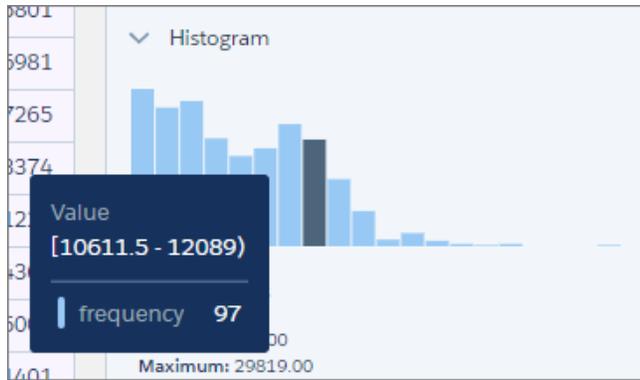
The histogram shows the distribution of values in a measure or date column. Each bar represents a predefined range, and the bar height shows the frequency of values falling within that range. Hover over a bar to see the range and frequency of the values it represents.

Use the histogram to see if the data falls within expected ranges. Low-frequency outlying values could indicate data inconsistencies or data that could impact aggregate calculations.

Measure columns also display statistical information that you can use in your data preparation.
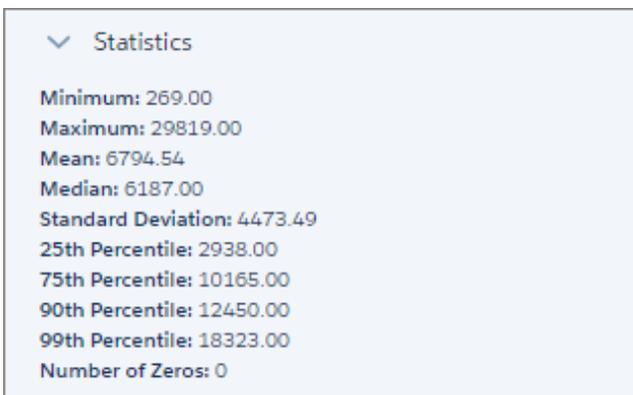
With date columns, you can select the time frame for grouping values by year or by quarter.
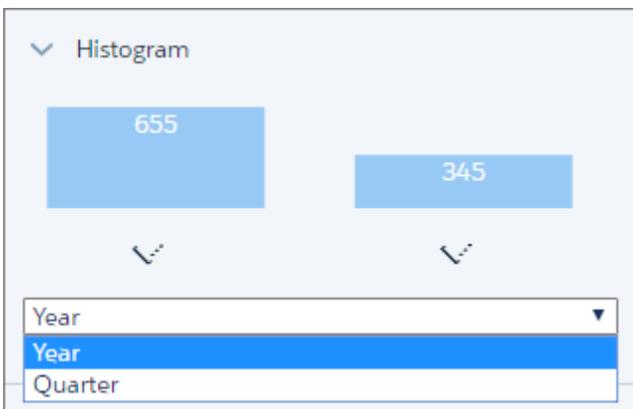
Add a Filter in a Recipe

Filter data in a recipe to remove rows that you don't need in your target dataset.

> **Note:** To filter rows in a Data Prep recipe, see Filter Node: Filter Rows. To do it in a dataflow, see filter Transformation.

1. On the recipe page, click ▼.
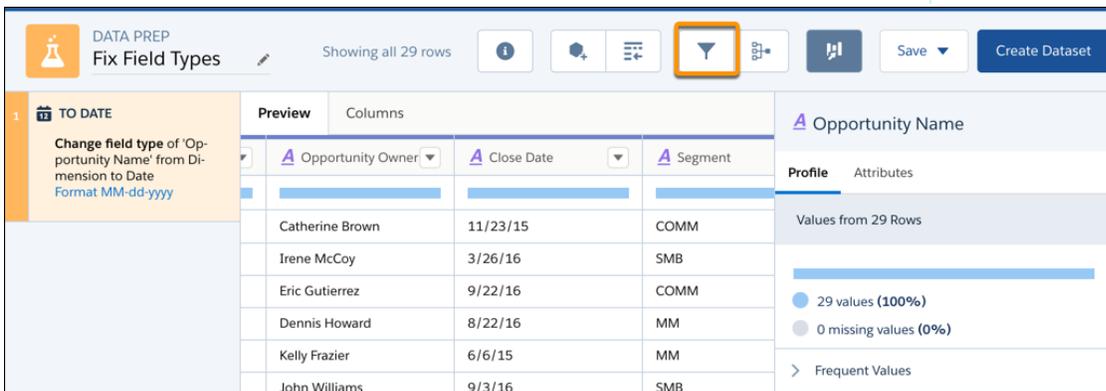
2. Click **+** and select the column to filter by.
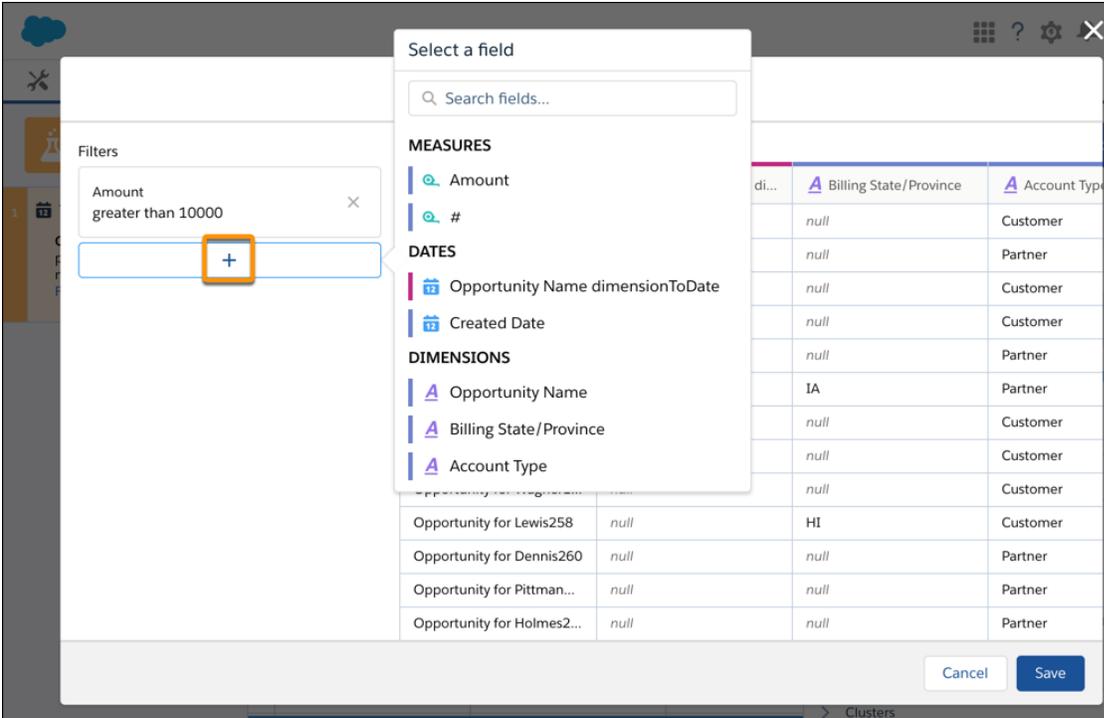
3. Enter the filter condition.

For a dimension column, you can enter your own value instead of selecting an existing column value.
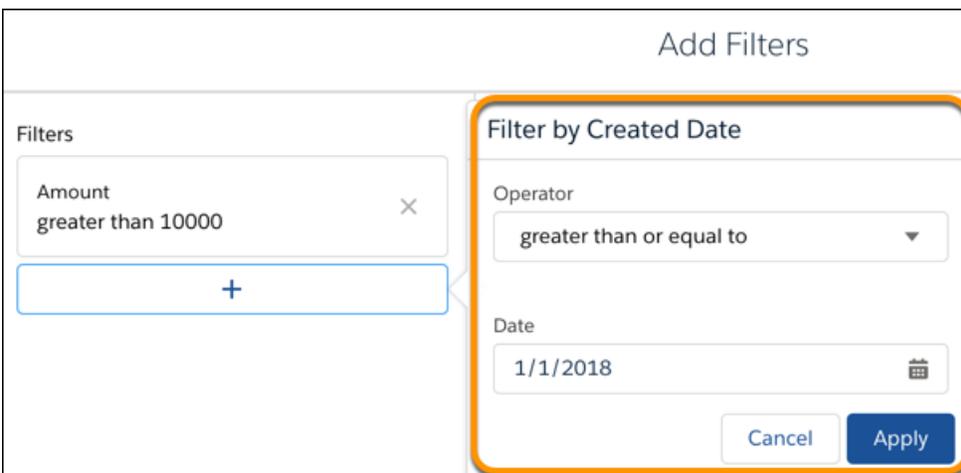
4. lick **Apply**.

5. To enter another condition, click .

6. When you have added all your conditions, click **Save**.
   The filter appears as a step in the left pane of the recipe.
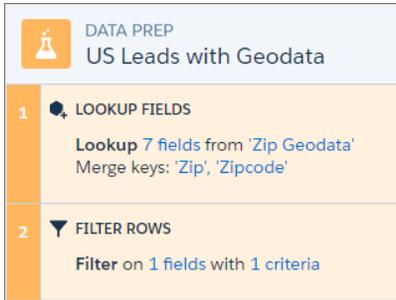
7. Save the recipe.

📝 **Note:** Filters in recipes filter rows that meet all specified conditions.

Bucket a Measure Field in a Recipe

Add a field to a recipe to bucket values in a specified measure field.

📝 **Note:** To bucket numeric values in a Data Prep recipe, see Categorize Measure Column Values into Buckets.

A bucket field lets you create new values based on existing values in another field. For example, use a bucket field to categorize households as small, medium, or large, based on the number of people in the household.

No. of People in Household	Size Category
1–2	Small
3–5	Medium
6 or more	Large

1. On the column header of the measure field you want to bucket, click ⯆ , and then select **Bucket**.
Bucket settings appear in the Add Transformation panel below the preview data.
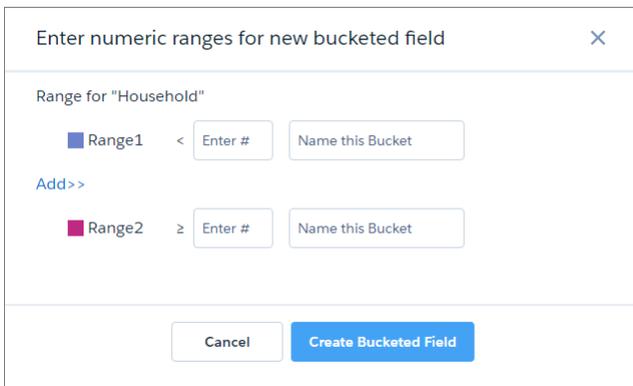
2. For Range1, enter a value and name for the first bucket. Rows with values below the Range1 value are assigned this bucket name. In this example, rows with values less than *3* are assigned a value of *Small* in the new bucket field.

The Range2 value populates with the value you enter for Range1.

3. If your bucket field has more than two buckets, click **Add>>** to add another range.

4. For each additional range you add, enter the range of values and the bucket name. Rows with values greater than or equal to the first value, and less than the second value are assigned the bucket name. In this example, rows with values *3*, *4*, or *5* are assigned a value of *Medium* in the new bucket field.

5. Repeat steps 4–5 to add any additional buckets that you need.

6. For the final range, enter a value and bucket name. Rows with values greater than or equal to this value are assigned this bucket name. In this example, rows with values greater than or equal to *6* are assigned a value of *Large* in the new bucket field.

7. When you have completed all the ranges, click **Add**.
The new bucket field appears next to the source field in the recipe preview.

💡 **Tip:** You can use a bucket field to filter rows in a recipe.

Bucket a Dimension Field in a Recipe

Add a field to a recipe to bucket values in a specified dimension field.

> 📝 **Note:** To bucket dimension values in a Data Prep recipe, see Categorize Dimension Column Values into Buckets.

A bucket field lets you create new values based on existing values in another field. For example, use a bucket field to categorize products in support cases according to their type.

Product	Product Category
HVAC Ctrl, HeatWave, Home A/C	HVAC
Humidity Ctrl, Humidity Unit, Home Humidity	Humidity
Compressor, Condenser, Thermostat	Parts

1. On the column header of the dimension field you want to bucket, click [▼], and then select **Bucket**.

Bucket settings appear in the Add Transformation panel below the preview data.

2. In the Bucket1 field, enter a name for the first bucket.

3. In the **Values for...**box, enter the values to be assigned to this bucket. Press Tab after each value to enter the next.

In this example, rows containing *HVAC Ctrl*, *HeatWave,* or *Home A/C* product values are assigned a value of *HVAC* in the new bucket field.

4. To add more buckets, click [+] and repeat steps 2–3.

5. To bucket remaining values that you have not bucketed, select **Bucket remaining values** and enter a bucket for these values. If you don't select this option, rows with unbucketed values keep the same value in the new bucket field.

6. When you have completed all the buckets, click **Add**.
The new bucket field appears next to the source field in the recipe preview.

Tip: You can use a bucket field to filter rows in a recipe.

Bucket a Date Field in a Recipe

Add a field to a recipe to bucket values in a specified date field.

Note: To bucket date values in a Data Prep recipe, see Categorize Date Column Values into Buckets.

A bucket field lets you create new values based on existing values in another field. For example, use a bucket field to categorize support cases according to the season they were opened.

1. On the column header of the date field you want to bucket, click ⬇, and then select **Bucket**. Bucket settings appear in the Add Transformation panel below the preview data. The default mode is Absolute, which lets you specify a date range and name for each bucket.

2. Enter a start date and end date for Bucket1, and enter a name for the bucket.

In this example, you want each bucket to be a season, so you enter the date range for winter and enter the bucket name.

| 12/1/0016 | 🗓 | - | 2/28/2017 | 🗓 | Winter | ➕ |

3. To add more buckets, click ➕ and repeat step 2.

4. To bucket remaining values that you have not bucketed, select **Bucket remaining values** and enter a bucket for these values. If you don't select this option, rows with unbucketed values keep the same value in the new bucket field.

5. When you have completed all the buckets, click **Add**.

The new bucket field appears next to the source field in the recipe preview.

🗓 CREATED DATE	A CREATED DATE DATEBUCKET
2017-02-22T23:51:17.000Z	Winter
2017-02-22T23:51:17.000Z	Winter
2017-02-22T23:51:17.000Z	Winter

In Relative mode, each bucket is a period of time relative to the date that the recipe runs. You can create bucket periods based on days, weeks, months, quarters, or even years. For example, create a bucket field from the case created date to categorize cases by when they were opened.

When you create a relative date bucket field, manually enter the start and end of each period, or use the sliders.

> 💡 **Tip:** You can use a bucket field to filter rows in a recipe.

Add a Formula Field in a Recipe

Add a formula field to calculate new values from measures and dimensions in your recipe.

> 📝 **Note:** To create a calculated column based on a formula in a Data Prep recipe, see Formula Transformation: Create a Calculated Column Based on an Expression. To do it in a dataflow, see computeExpression Transformation.

A formula field lets you calculate new values based on existing values in fields. To help you, a formula builder suggests fields and functions as you type.

1. To start creating a formula field, choose one of these methods.

   - To create the formula from scratch, click ⏷ on any field column header, and then select **Formula**.

     The formula builder appears in the Add Transformation panel below the recipe preview.
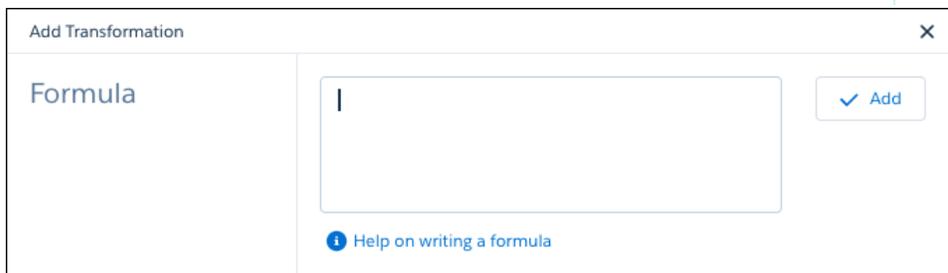
   - To use a suggested function, click the field that you want to use in the recipe preview, and then click a function in the suggestions bar below the preview.

     The formula is started for you, using the field and function that you selected.

- To select multiple fields for use in your formula, while pressing Shift or Ctrl (Cmd on Mac), click the fields that you want to use in the recipe preview. Then click an function in the suggestions bar below the preview.

The formula is started for you, using the fields and operator that you selected.

2. In the formula builder, enter the formula. If the formula has been started for you, you can edit it here.

> Note: Here are some tips for entering or editing your formula.
>
> - Just start typing to see matching functions and fields.
> - To select a match, either click it, or use your cursor up and down keys to select it and then press **Enter** or **Tab**.
> - Use your keyboard to enter parentheses, arithmetic operators, and numbers as needed.
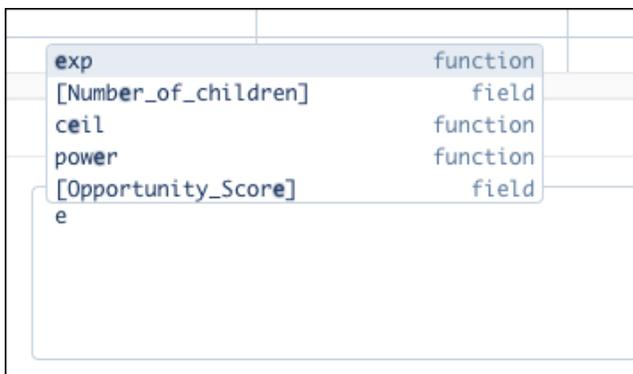
3. When you have completed the formula, click **Add**.
   The new formula field appears in the recipe preview.

💡 **Tip:** You can bucket a formula field or reference it in other formulas in the recipe.

Numeric Functions for Formula Fields

Use numeric functions to calculate new values from measures in your recipe. For example, you can round, truncate, and determine the absolute value of a number. The arguments in each function can be numbers or measure columns.

String Functions for Formula Fields

Use string functions to create values in a formula field based on one or more dimension columns in your recipe or strings. For example, you can change the casing of text strings, concatenate values from multiple columns, and replace values.

Numeric Functions for Formula Fields

Use numeric functions to calculate new values from measures in your recipe. For example, you can round, truncate, and determine the absolute value of a number. The arguments in each function can be numbers or measure columns.

📝 **Note:** To get a list of numeric functions for a Data Prep recipe, see Numeric Operators and Functions for Formulas.

abs Function

Calculates the absolute value of a number. The absolute value of a number is the number without its positive or negative sign.

ceil Function

Rounds a number up to the nearest integer.

exp Function

Returns a value for e raised to the power of a number you specify.

floor Function

Returns a number rounded down to the nearest integer.

log Function

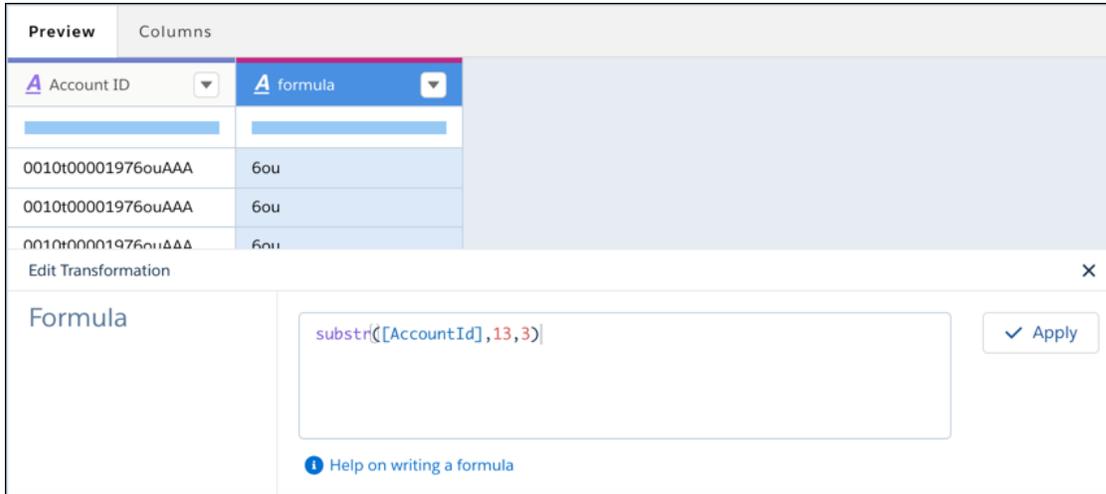Returns the logarithm of a number in the base you specify.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

abs Function

Calculates the absolute value of a number. The absolute value of a number is the number without its positive or negative sign.

Syntax

```
abs(number)
```

Arguments

Argument	Description
number	Literal value or measure field of which you want to determine the absolute value.

ceil Function

Rounds a number up to the nearest integer.

Syntax

```
ceil(number)
```

Arguments

Argument	Description
number	Literal value or measure field that you want to round up to the nearest integer.

exp Function

Returns a value for e raised to the power of a number you specify.

Syntax

`exp(`*`number`*`)`

Arguments

Argument	Description
number	Literal value or measure field to which you want to raise e.

floor Function

Returns a number rounded down to the nearest integer.

Syntax

`floor(`*`number`*`)`

Arguments

Argument	Description
number	Literal value or measure field that you want to round down to the nearest integer.

log Function

Returns the logarithm of a number in the base you specify.

Syntax

`log(`*`base`*`, `*`number`*`)`

Arguments

Argument	Description
base	Literal value or measure field used as the base of the logarithm.
number	Literal value or measure field that you want to take the logarithm of.

power Function

Raises a number to the power of another number.

Syntax

```
power(number, power)
```

Arguments

Argument	Description
number	Literal value or measure field that you want to raise to the specified power.
power	Literal value or measure field that is the power that you want to raise the specified number to.

round Function

Returns the nearest number to a number you specify, constraining the new number by a specified number of digits.

Syntax

```
round(number, numberOfDigits)
```

Arguments

Argument	Description
number	Literal value or measure field that you want to round.
numberOfDigits	Literal value or measure field that specifies the number of digits to which the specified number will be rounded.

sqrt Function

Returns the positive square root of a given number.

Syntax

```
sqrt(number)
```

Arguments

Argument	Description
number	Literal value or measure field that you want to find the square root of.

trunc Function

Truncates a number to an integer.

Syntax

`trunc(number)`

Arguments

Argument	Description
number	Literal value or measure field that you want to truncate to an integer.

String Functions for Formula Fields

Use string functions to create values in a formula field based on one or more dimension columns in your recipe or strings. For example, you can change the casing of text strings, concatenate values from multiple columns, and replace values.

> **Note:** To get a list of string functions for a Data Prep recipe, see String Functions for Formulas.

You can use a formula to manipulate strings, instead of dataflow transformations, to perform multiple string manipulations at the same time. For instance, you can nest string functions in the formula editor to combine the sales territory and country into a single field and apply title casing to the values.

`title(concat([Sales_Territory], " ",[Country]))`

When entering a formula, text strings must be enclosed in double straight quotes (`"This is a string."`). Column names must be enclosed in square brackets (`[Opportunity_Name]`).

concat Function

Returns a string by concatenating the values of the specified columns and input strings. For example, to display the close date as MM-DD-YYYY, concatenate the Close_Date_Month column, Close_Date_Day column, and Close_Date_Year column, and add a dash between each of them.

lower Function

Returns a string with all characters from the input string in lowercase. If the input string is null, then the result is null.

ltrim Function

Removes the specified substring from the beginning of a string. To remove leading spaces, do not specify a substring.

replace Function

Replaces a substring with the specified characters. If any of the arguments are null, then the function returns null. This function is case-sensitive.

rtrim Function

Removes the specified substring from the end of a string. To remove trailing spaces, do not specify a substring.

substr Function

Returns characters from the string, starting at the specified position and of the specified length.

title Function

Returns the string with the initial character of every word in uppercase and the remaining characters in lowercase. For example, "united states" becomes "United States."

trim Function

Removes the specified substring from the beginning and end of a string. To remove leading and trailing spaces, do not specify a substring.

upper Function

Returns the string with all characters in uppercase. If string is null, then the result is null.

concat Function

Returns a string by concatenating the values of the specified columns and input strings. For example, to display the close date as MM-DD-YYYY, concatenate the Close_Date_Month column, Close_Date_Day column, and Close_Date_Year column, and add a dash between each of them.

Syntax

```
concat(string1,string2,...)
```

Arguments

Argument	Description
string1	First dimension field or text string to include in the concatenated value.
string2	Second dimension field or text string to include.

📝 **Note:** This function also appears in the Suggestion bar when you select multiple dimension columns in the recipe editor.
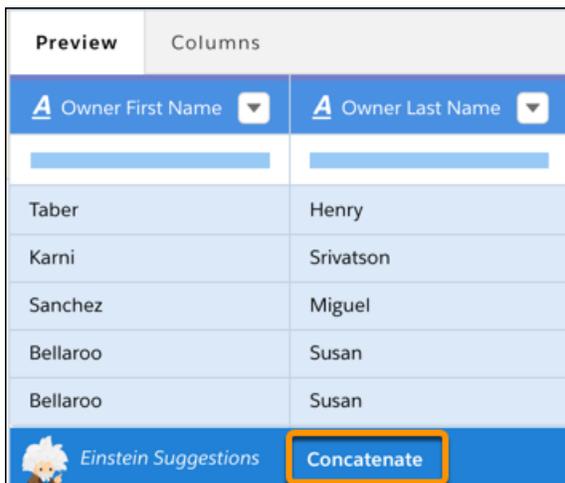
👁 **Example:** To create the full name, concatenate the first and last name with a space in between them as shown in the following formula.

```
concat([OwnerId.FirstName]," ",[OwnerId.LastName])
```
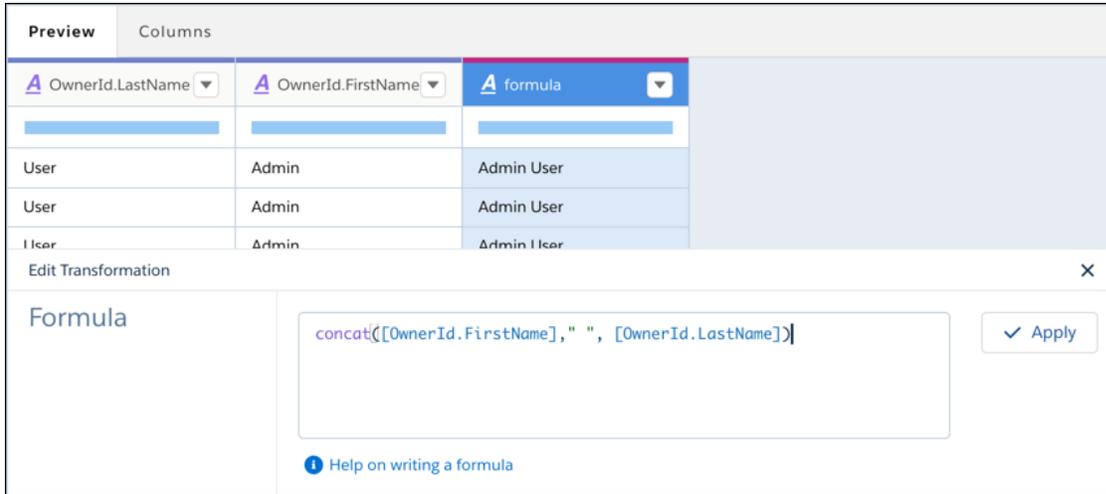
lower Function

Returns a string with all characters from the input string in lowercase. If the input string is null, then the result is null.

Syntax

```
lower(string)
```

Arguments

Argument	Description
string	Dimension field or text string to convert to lowercase.

Example: The AccountId.Industry column uses different casing for industries, like "Media" and "media." To apply consistent casing to the columns values, convert them to lowercase using the following formula.

```
lower([AccountId.Industry])
```
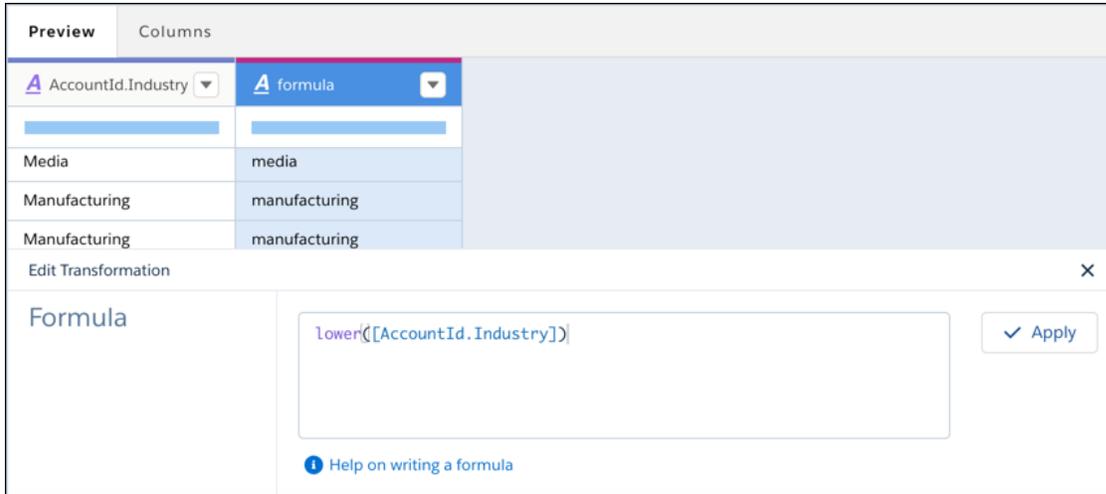
ltrim Function

Removes the specified substring from the beginning of a string. To remove leading spaces, do not specify a substring.

Syntax

```
ltrim(string,[substring])
```

Arguments

Argument	Description
string	Dimension field or text string to remove the leading spaces or specified substring from.
substring	Optional. The value removed from the string.

Example: You want to trim a string with 5 leading space characters.

```
ltrim("    5 Spaces Before")
```
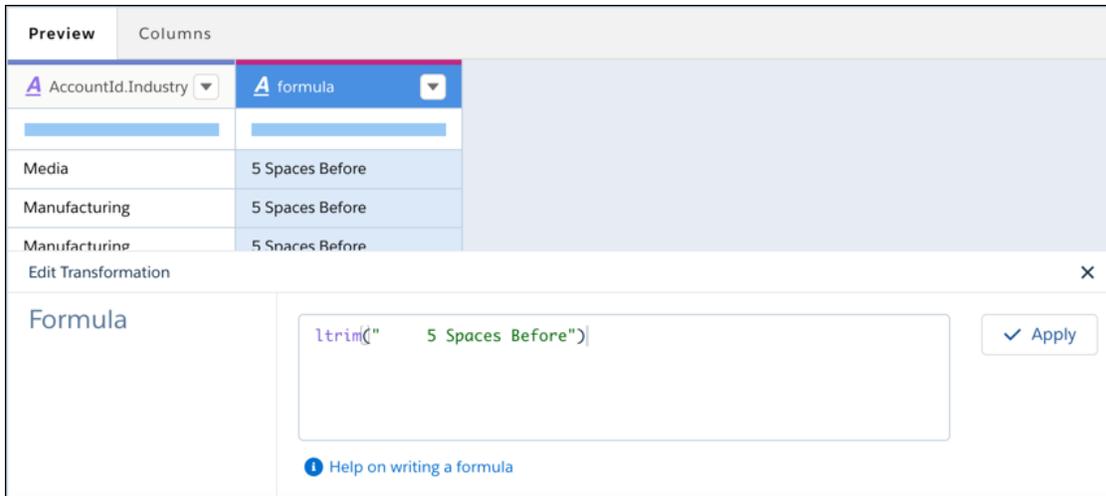
replace Function

Replaces a substring with the specified characters. If any of the arguments are null, then the function returns null. This function is case-sensitive.

Syntax

```
replace(string,searchString,replacementString)
```

Arguments

Argument	Description
string	Dimension field or text string that contains the substring to be replaced.
searchString	Substring to replace. If searchString is an empty string, the function returns null.
replacementString	The value that replaces the substring.

💡 Tip: To replace null values in a dimension or date column, use the Replace transformation and select the **Search for null** option.

👁 Example: To change the account name from "salesforce.com" to "Salesforce" in the Account_Name column, use the following formula.

```
replace([Account_Name],"salesforce.com","Salesforce")
```
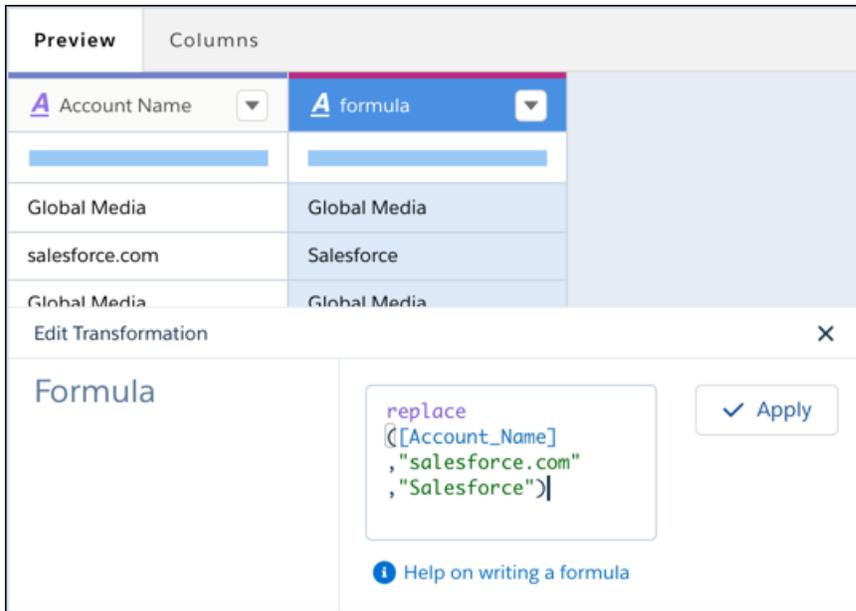
rtrim Function

Removes the specified substring from the end of a string. To remove trailing spaces, do not specify a substring.

Syntax

```
rtrim(string,[substring])
```

Arguments

Argument	Description
string	Dimension field or text string to remove the trailing spaces or specified substring from.
substring	Optional. The value to remove from the string.

👁 **Example:** You want to trim the 2 trailing space characters from a string.

```
rtrim("2 Spaces After  ")
```
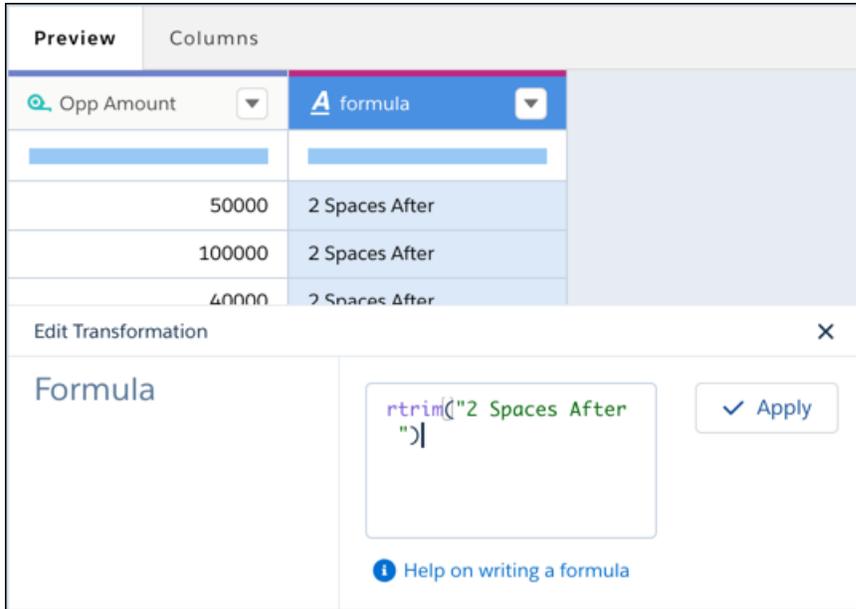
substr Function

Returns characters from the string, starting at the specified position and of the specified length.

Syntax

```
substr(string,position,[length])
```

Arguments

Argument	Description
string	Dimension field or text string to extract the substring from.
position	The starting character position of the substring. The first character in a string is at position 1. If position is negative, then the position is relative to the end of the string. So a position of -1 denotes the last character.
length	Optional. The number of characters to return. If length is 0, the output is an empty string. If length is negative, then the function returns null.

👁 **Example:** You want to return the 13th, 14th, and 15th character from each account ID.

```
substr([Account_Id],13,3)
```

title Function

Returns the string with the initial character of every word in uppercase and the remaining characters in lowercase. For example, "united states" becomes "United States."

Syntax

```
title(string)
```

Arguments

Argument	Description
string	Dimension field or text string on which to apply title casing.

Example: You want to apply title casing on the opportunity names to ensure consistent casing on these values.

```
title([Opp_Name])
```
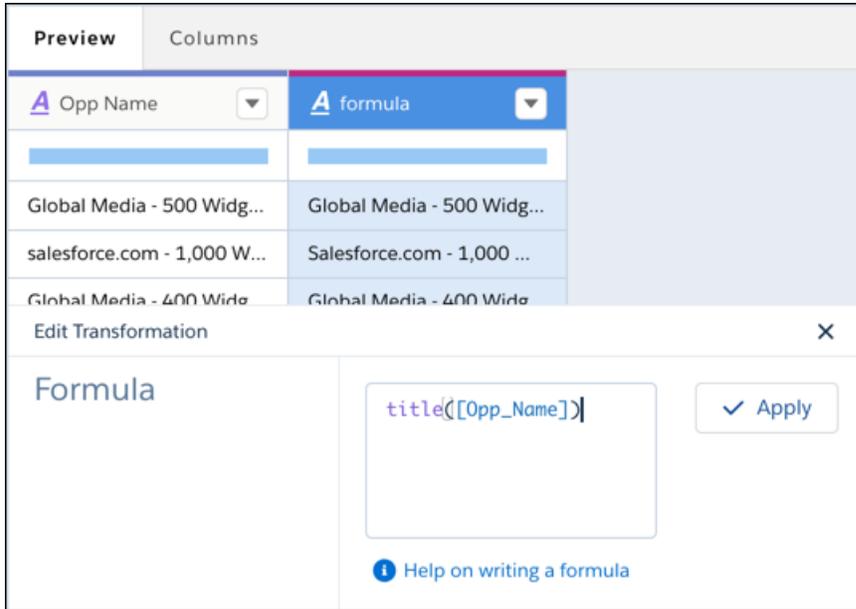
trim Function

Removes the specified substring from the beginning and end of a string. To remove leading and trailing spaces, do not specify a substring.

💡 **Tip:** Use the trim function in a SAQL query to trim other types of characters.

Syntax

```
trim(string,[substring])
```

Arguments

Argument	Description
string	Dimension field or text string to remove the specified substring from.
substring	Optional. The value removed from the string.

👁 **Example:** You want to remove the two leading spaces before and after a string.

```
trim("  Spaces Before and After  ")
```
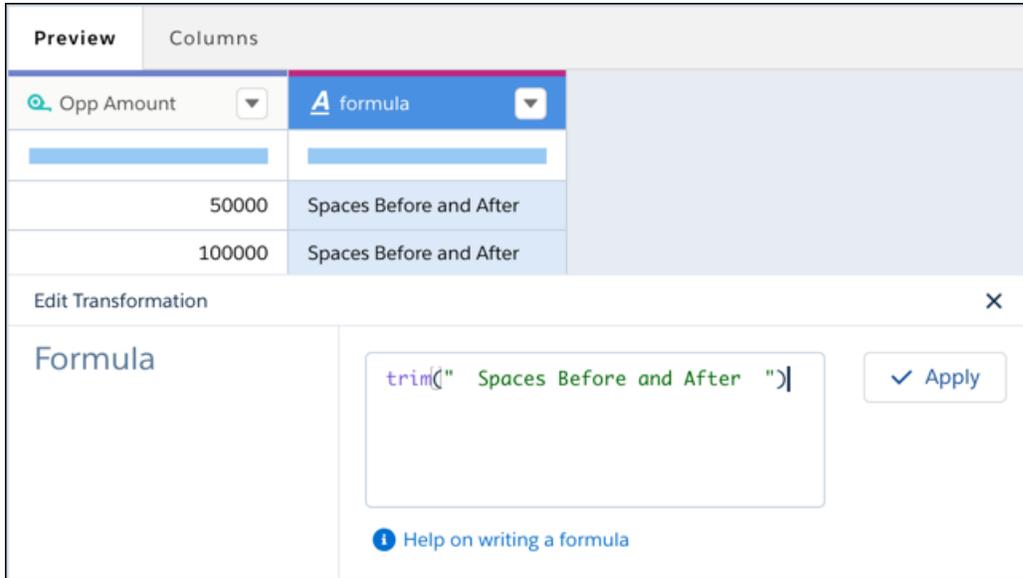
upper Function

Returns the string with all characters in uppercase. If string is null, then the result is null.

Syntax

```
upper(string)
```

Arguments

Argument	Description
string	Dimension field or text string to convert to uppercase.

👁 **Example:** You have a column Owner_First_Name that accepts freeform entry for users to enter their first name. As a result, some names are in uppercase, some in lowercase, and others in mixed case. To convert them to all uppercase characters to make the casing consistent, use the following formula.

```
upper([Owner_First_Name"])
```

Aggregate and Group Data to a Different Grain

Large datasets can be hard to digest due to the amount of information and low-grain details. Aggregation allows these datasets to be rolled up to a higher granularity, thus allowing users to create recurring summary statistics and join datasets with different granularities. You can also aggregate data to perform calculations on grouped records without aggregating the measures. For instance, group by website session IDs and then calculate the average time on each page and total number of clicks.

> 📝 **Note:** To aggregate rows in a Data Prep recipe, see Aggregate Node: Roll Up Data to a Higher Level.

You can add groups, aggregates, or both. Group by dimensions or date windows to roll up the records to a higher grain. For example, group daily logs by month so you can join that data with your monthly datasets. Aggregate measures to the new grouping granularity. You can use the following aggregate functions on measure columns: sum, unique, avg, count, max, and min.

> 📝 **Note:** To prevent double counting, exclude aggregated values that don't match the grain of the dataset. For example, the following dataset's grain is opportunity—each record represents an opportunity. However, the grain of the Account Annual Revenue aggregated value is account. If a user doesn't know the dataset grain and adds all Account Annual Revenue values, he would double count annual revenue for accounts with multiple records. To prevent double counting, move Account Annual Revenue to a different dataset where the grain is account.

1. On the recipe page, click the Aggregation button (⊞).

2. To change the granularity of the data, add groups. For example, group by account name to aggregate metrics by this dimension.

3. To view aggregated metrics for each grouping, add aggregates. For example, calculate the average age for opportunities to close, total amounts, average amounts, and total number of deals. Because these opportunities are grouped by account name, these aggregates provide key metrics about each account.

4. Click **Done**.
   Notice that the aggregates and group-by fields are the only fields included in the output. All other fields from the source dataset are excluded—you can't perform transforms on them in this recipe anymore.

5. Click **Save** > **Save Recipe**.

When you run the recipe, Tableau CRM generates a new dataset that contains the aggregates specified in the recipe and a grain determined by the groups. In the example, we see the aggregates for each account.

Transform Fields in a Recipe

Data is not always consistent, especially when you combine data from different sources. In a recipe, you can transform fields to ensure that values are consistent in your target dataset. Change case, split values to get just the parts you need, and replace incorrect values.

> 📝 **Note:** To transform column values in a Data Prep recipe, see Formula Transformation: Create a Calculated Column Based on an Expression. To accomplish these tasks in a dataflow, see computeExpression Transformation.

When you transform a field, a new field is added with the new values to the right of the original field. You can choose to keep or remove the original field.

To transform a field, click ▼ in the column header (1), and then click the transformation that you want to use. Alternatively, select a transformation from the Suggestions bar at the bottom of the recipe.

Transformation settings appear in the Add Transformation panel below the preview data.

You can apply these field transforms.

Transform	Description	Example
Extract	Extract the selected component from a date field into a new field.	Extract the hour component from the case created date to analyze case creation by hour of the day.
Lowercase	Convert values in selected field to lowercase.	Correct lead email addresses captured at a trade show. *JSMITH@FORRESTER.COM* becomes *jsmith@forrester.com*.
Replace	Find a specific value in a field and replace it with a new value. A new field is created containing both replaced and unchanged values. You can also replace nulls with a value. For example, to allow users to group transactions that don't have a product, replace all nulls in the Product field with "N/A".	Make country values consistent. **Find** all instances of *US* and **Replace** them with *USA*.
Split	Divide values into multiple parts at the delimiter and create a field for each part.	Divide phone numbers into area code and number. Select the – delimiter to split *925–900 2123* into two new values *925* and *900 2123*.
Substring	Starting at the character **Position** in a value, extract the number of characters in **Length** into a new field.	Extract the month from date values. Enter a **Position** of *4* and a **Length** of *2* to extract the month value *10* from the month value *2016–10–23*.
Trim	Remove leading and trailing whitespaces.	Remove whitespaces to ensure that values from one source are consistent with values from another source. " *CA* " becomes "*CA*".
Uppercase	Convert values in selected field to uppercase.	Correct US state values. *ca* becomes *CA*.

⊘ **Important:** Tableau CRM datasets contain date and time component fields, such as Year, Quarter, and Month, that are derived automatically from existing date and time fields. Transformations on these component fields are not supported.

SEE ALSO:

Standardize Date Formats

Convert Field Types in a Recipe

Standardize Date Formats

If a dimension field contains dates in different formats, use the Format Dates recipe transformation to standardize the format for all values in the field. A consistent format enables you to correctly filter and group records by date, including filtering by date component, such as month. It also ensures that you can successfully convert the field type from dimension to date.

Note: To format dates in a Data Prep recipe, see Format Dates Transformation: Standardize the Date Format in a Column.

1. On the column header of the dimension field, click [▼].
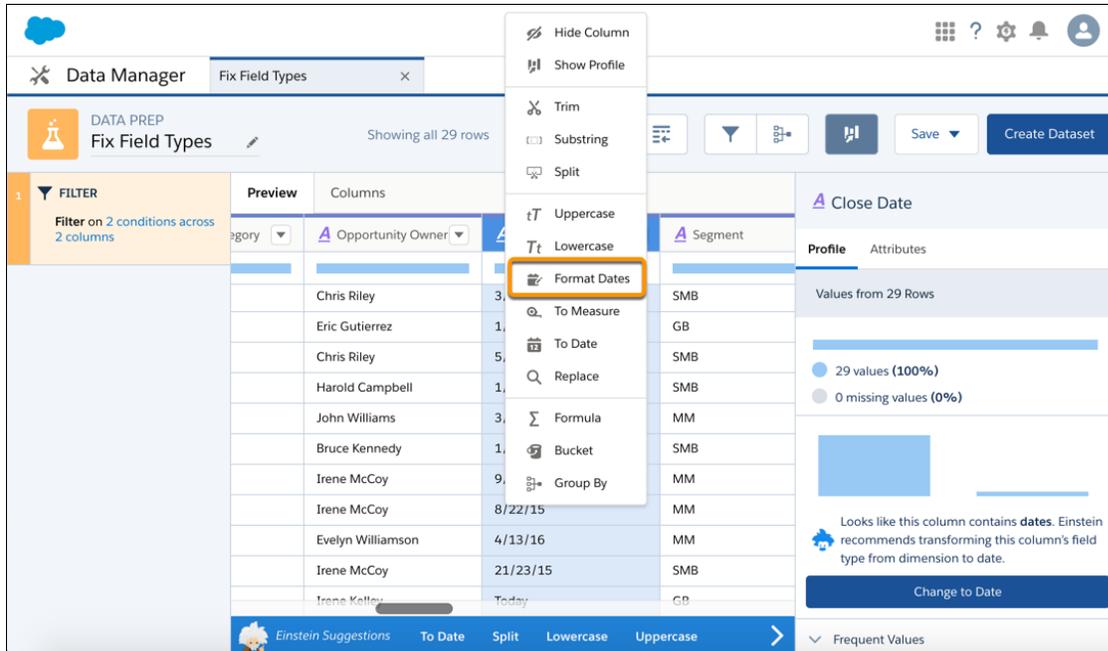
2. Select **Format Dates**.
   Analytics detects the formats of the dates in the field.
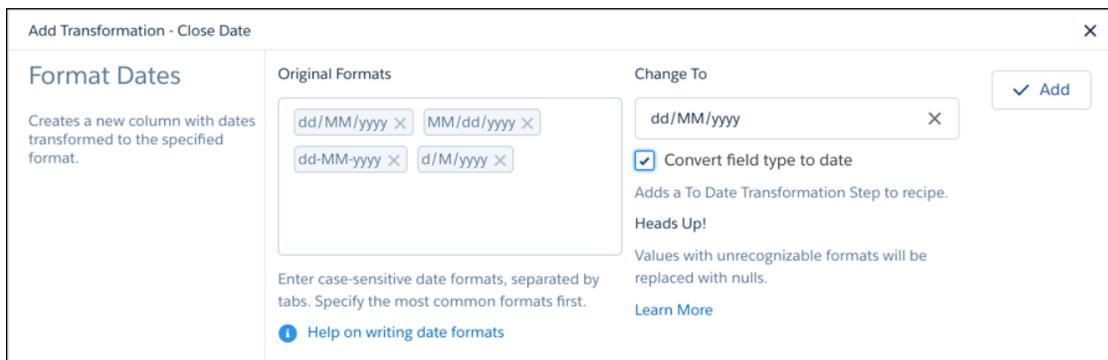
Note: The order of the date formats in the Original Formats box matters. For example, if a date is 01/06/2011, the date format could be dd/MM/yyyy or MM/dd/yyyy. The Date Format transformation assumes the first matching format shown the Original Date Format box is the right one.

3. If a date format isn't detected, enter it in the Original Formats box.

4. To convert a dimension field with dates to a date field type, select **Convert field type to date**.

5. Select the desired date format in the Change To field.

6. Click **Add** and save the recipe.

When you run the recipe, the transformation creates a field with the dates standardized in the selected format. If Tableau CRM can't determine the original format or the date doesn't have a value for a date component, the transformation replaces the date with null. For example, if you standardize on the `MM/dd/yyyy: hh:mm:sssz` format and a date value doesn't have the seconds date component (`sssz`), Tableau CRM replaces the date with null.

Guidelines for Formatting Dates
Review the following guidelines when working with dates.

Guidelines for Formatting Dates
Review the following guidelines when working with dates.

- Tableau CRM supports the following date formats. Formats are case sensitive.

Format	Sample Value
yyyy-MM-dd'T'HH:mm:ss.SSS'Z'	2014-04-29T16:53:34.000Z
yy-MM-dd'T'HH:mm:ss.SSS'Z'	14-04-29T16:53:34.000Z
yyyy-MM-dd'T'HH:mm:ss'Z'	2014-04-29T16:53:34Z
yy-MM-dd'T'HH:mm:ss'Z'	14-04-29T16:53:34Z
yyyy-MM-dd HH:mm:ss	2014-06-03 11:31:45
yy-MM-dd HH:mm:ss	14-06-03 11:31:45
dd.MM.yyyy HH:mm:ss	03.06.2014 11:31:45
dd.MM.yy HH:mm:ss	03.06.14 11:31:45
dd/MM/yyyy HH:mm:ss	03/06/2014 11:31:45
dd/MM/yy HH:mm:ss	03/06/14 11:31:45
dd/MM/yyyy hh:mm:ss a	03/06/2014 11:31:45 AM
dd/MM/yy hh:mm:ss a	03/06/14 11:31:45 AM
dd-MM-yyyy HH:mm:ss	03-06-2014 11:31:45
dd-MM-yy HH:mm:ss	03-06-14 11:31:45
dd-MM-yyyy hh:mm:ss a	03-06-2014 11:31:45 AM
dd-MM-yy hh:mm:ss a	03-06-14 11:31:45 AM
MM/dd/yyyy hh:mm:ss a	06/03/2014 11:31:45 AM
MM/dd/yy hh:mm:ss a	06/03/14 11:31:45 AM
MM-dd-yyyy hh:mm:ss a	06-03-2014 11:31:45 AM

Format	Sample Value
MM-dd-yy hh:mm:ss a	06-03-14 11:31:45 AM
HH:mm:ss dd/MM/yyyy	11:31:45 03/06/2014
HH:mm:ss dd/MM/yy	11:31:45 03/06/14

- When you configure the Format Dates transformation, Tableau CRM detects the formats of the dates in the column. The order of the date formats listed in the Original Formats box matters. For example, if a date is 01/06/2011, the date format could be dd/MM/yyyy or MM/dd/yyyy. The Date Format transformation assumes the first matching format shown the Original Date Format box is the right one. If you want to use the other format, add it as the first date format in the box.

- In the column profile, the histogram shows a bar for each date format. If applicable, the histogram also shows one extra bar for all dates for which the format couldn't be determined. Notice that the column profile says that there are two different date formats, even though there are three bars in the histogram.

Convert Field Types in a Recipe

The type assigned to a dataset field determines how you can query that field's data. For example, you can filter and group by a dimension or date field, or perform math calculations on a measure field. When you load data into a dataset, Tableau CRM sometimes tags a dataset field with the wrong type. If needed, convert fields to the correct types.

Convert a Dimension Field Type to Date

When you load data into a dataset, Tableau CRM tags a date field as a dimension if it contains unexpected string values or dates in different formats. To use date functionality, such as grouping by month, use the To Date transformation to change the dimension field type to date.

Convert a Dimension Field Type to Measure

When you load data into a dataset, Tableau CRM tags a measure field as a dimension if it contains unexpected strings or characters. To use measure functionality, such as calculating the average, use the To Measure transformation to change the dimension field type to measure.

Convert a Measure Field Type to Dimension

When you load data into a dataset, Tableau CRM tags a dimension field as a measure if it contains only numbers. For example, Tableau CRM tags a numerical Room Number field as a measure, even though it's actually a dimension that identifies rooms. To use dimension functionality, such as grouping and filtering by dimension fields, use the To Dimension transformation to change the measure field type to dimension.

Convert a Dimension Field Type to Date

When you load data into a dataset, Tableau CRM tags a date field as a dimension if it contains unexpected string values or dates in different formats. To use date functionality, such as grouping by month, use the To Date transformation to change the dimension field type to date.

> **Note:** To convert a dimension column to date in a Data Prep recipe, see Dimension to Date Transformation: Convert the Column Type.

USER PERMISSIONS

To create a recipe:
- Edit Analytics Dataflows OR Edit Dataset Recipes

1. On the column header of the dimension field, click ⏷, and then select **To Date**.

2. In the Date Format field, select the date format to display the dates.

3. Click **Add** and save the recipe.

Run the recipe to create a new field with the date field type and convert the date values to the specified format. If the transformation can't parse a date value, it replaces the value with null.

Convert a Dimension Field Type to Measure

When you load data into a dataset, Tableau CRM tags a measure field as a dimension if it contains unexpected strings or characters. To use measure functionality, such as calculating the average, use the To Measure transformation to change the dimension field type to measure.

> ✏️ Note: To convert a dimension to a measure in a Data Prep recipe, see Dimension to Measure Transformation: Convert the Column Type. To do it in a dataflow, see dim2mea Transformation.

During the column type conversion, the Dimension to Measure transformation rounds decimals to the nearest whole number. For example, 300.2939 becomes 300.

1. On the column header of the dimension field, click [▼], and then select **To Measure**.
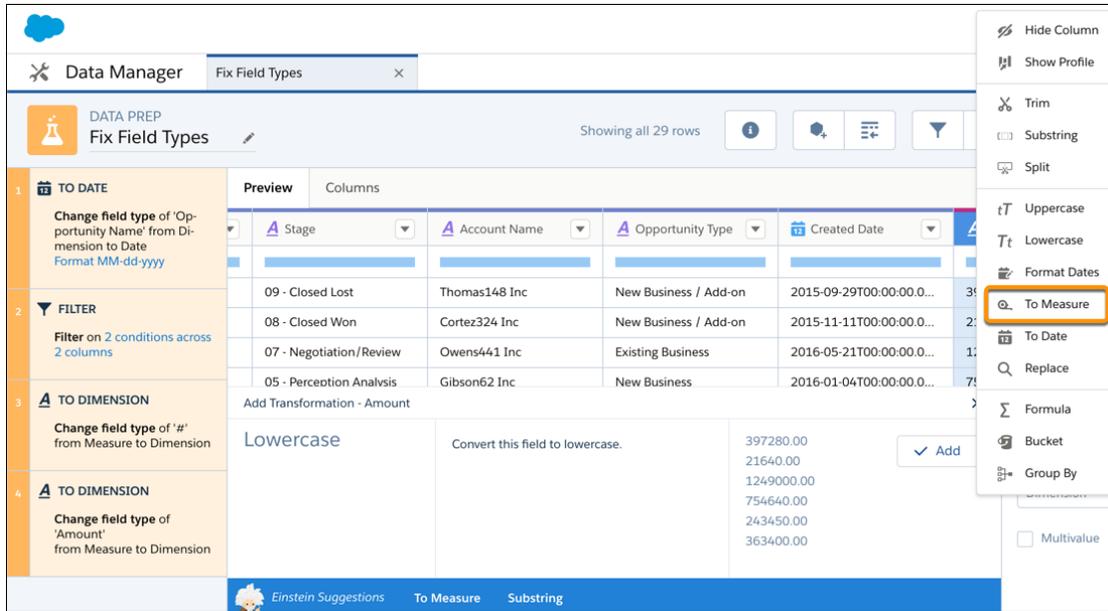
2. Click **Add** and save the recipe.

Run the recipe to create a new field with the measure field type.

Convert a Measure Field Type to Dimension

When you load data into a dataset, Tableau CRM tags a dimension field as a measure if it contains only numbers. For example, Tableau CRM tags a numerical Room Number field as a measure, even though it's actually a dimension that identifies rooms. To use dimension functionality, such as grouping and filtering by dimension fields, use the To Dimension transformation to change the measure field type to dimension.

Note: To convert a measure to a dimension in a Data Prep recipe, see Measure to Dimension Transformation: Convert the Column Type.

1. On the column header of the measure field, click [▼], and then select **To Dimension**.
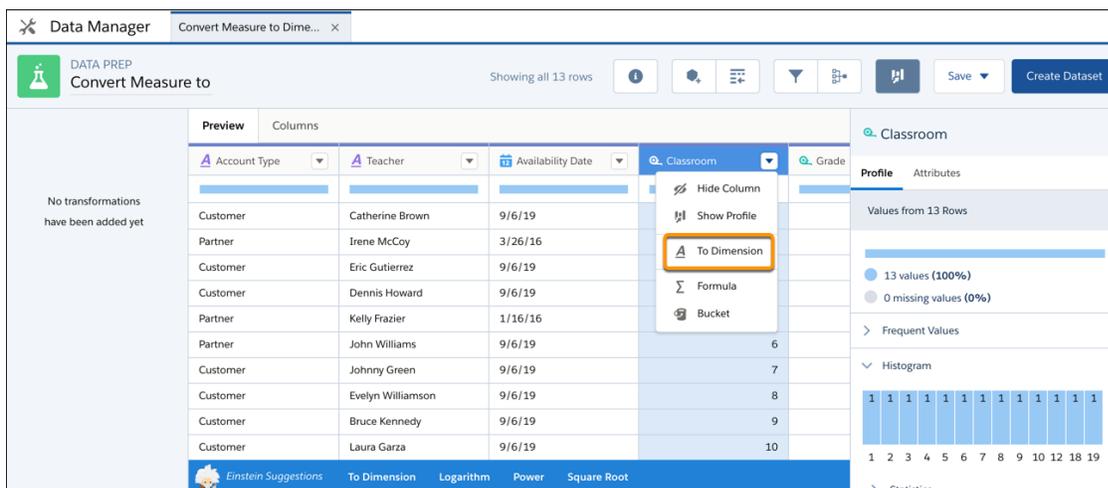
2. Click **Add** and save the recipe.

Run the recipe to create a new field with the dimension field type.

Predict Missing Values in Dimension Columns

When a dataset or connected object has missing values in a dimension column, Tableau CRM can fill in missing values to complete your data. Tableau CRM intelligently predicts values based on values in other strongly correlated columns in your data.

> 📝 **Note:** To predict missing values in a Data Prep recipe, see Predict Missing Values Transformation: Fill In Missing Values.

Consider these limitations before using this feature.

- If there aren't enough records to make accurate predictions, Tableau CRM doesn't insert predicted values .
- You can't perform column profiling or transformations on predicted columns.
- Recipes that predict values can take longer to run.

To predict missing values in a dimension column:

1. On the dataset recipe page, click the dimension column.

2. In the Einstein Suggestions bar, click **Predict Missing Values**.

3. Select up to three dimension columns to use to predict the missing values for the selected column.

Tip: To make an accurate prediction, each column must have less than 200 unique values. Also, verify that these predictive columns contain clean, quality data. For example, you have an Education predictive column that contains values such as "Bachelors Degree" and "Bachelors." Use the bucket transformation to bucket field values with the same meaning. Then use the column with the clean data as a predictive column. For more information about bucketing, see Bucket a Dimension Field in a Recipe.

4. Click **Add** to confirm.

The preview shows the original column with the missing values and the new column with "predict" at the end. The preview shows "Prediction TBD" for predicted values in the new column. The predicted values don't appear until after you run the recipe.
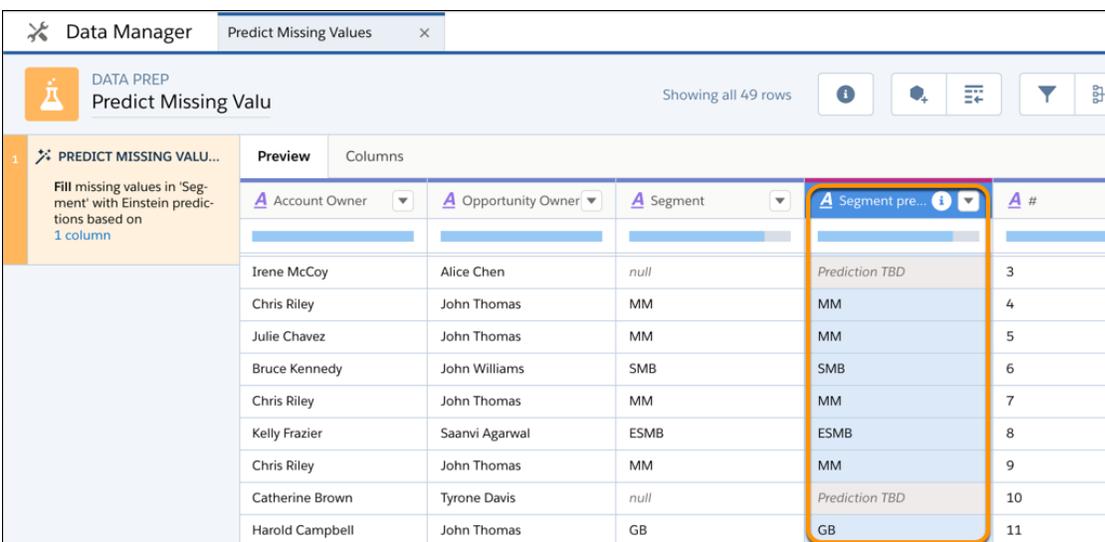
5. Click **Save** > **Save Recipe**.

When you run the recipe to create the dataset, you can include the original column and the new column with the predictions. To review the predictions, view the dataset as a values table.

Opportunity Name	Account Type	Account Owner	Segment	Segment_predict	Amount
Opportunity for Snyder164	Customer	Chris Riley	MM	MM	2,505,200
Opportunity for Nichols171	Customer	Catherine Brown	SMB	SMB	874,780
Opportunity for Cunningha...	Customer	Chris Riley	MM	MM	4,357,300
Opportunity for McLaughlin...	Customer	Irene McCoy	-	COMM	1,249,000
Opportunity for Lewis258	-	Catherine Brown	-	SMB	309,938
Opportunity for Butler275	Customer	Irene McCoy	-	COMM	209,104
Opportunity for Watson358	-	Catherine Brown	-	SMB	117,480
Opportunity for Wise361	Customer	Catherine Brown	-	SMB	1,595,250
Opportunity for Gordon370	Customer	John Williams	-	GB	179,225
Opportunity for Hawkins22	Customer	Irene McCoy	-	COMM	136,464
Opportunity for Stevens152	Partner	Harold Campbell	-	GB	11,883

Navigate Columns in a Recipe

As you add data and transform fields, the number of columns in your recipe preview increases. This makes it harder to find the columns that you want to work with. Use the column view to quickly find the columns that you need, and hide the ones you don't.

To open the column view, click the **Columns** tab at the top of the recipe editor.

Columns are color coded to distinguish columns from different sources. To quickly find a column, start typing its name in the Search columns box (1) to narrow down the list of columns. When you see the column that you want, click it to see its column profile on the right. To view or edit the column's metadata, click **Attributes** (2) in the profile pane. Here you can edit the API names and labels of new columns that you create in a recipe, and the labels of existing columns. If you decide that you don't need the column in the recipe, select **Hide Column** (3) from the column menu. Columns that you hide appear in a separate list (4). To add a hidden column to the recipe, select **Add to Recipe** from its menu.

Note: If you hide a column in the navigator, by default it is not included when the recipe runs. However, you can choose to include the column when you run the recipe.

Navigate and Edit Recipe Steps
As you prepare your data, each change you make appears as a recipe step on the left. Think of these steps as your recipe history. You can move back and forward through this history to see how the data looks at different stages of the recipe. If you don't like what you see, you can edit or remove any step.

To access recipe step actions, hover over the step and click ⏷.
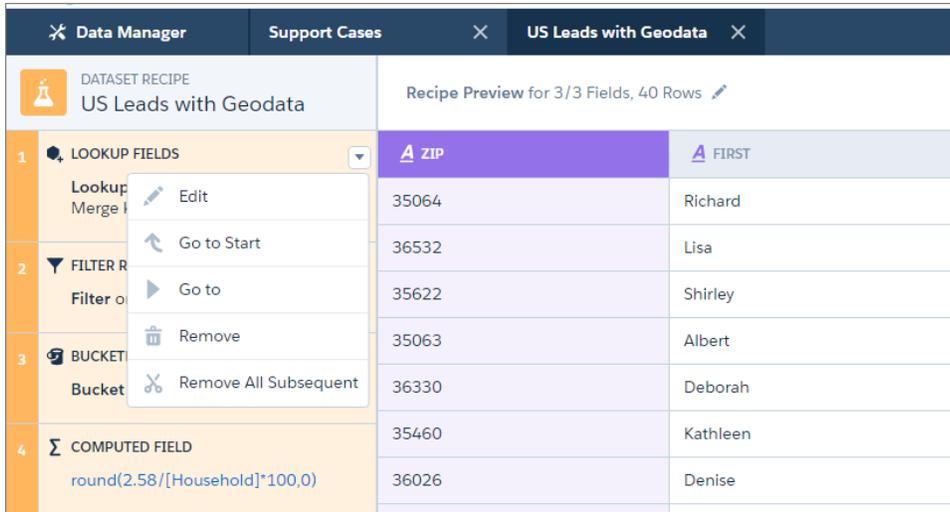
Note: The actions available for each step depend on the step's position.

Here's what the actions do.

Action	Description
Edit	Edit the current step. **Tip:** You can also edit a step by clicking any of the blue values.
Go to Start	Display the preview data as it was at the start of the recipe, before you added any steps. This action is only available for the first step.
Go to	Display the preview data as it was at the current step, before you added any subsequent steps.
Remove	Remove the current step.
Remove all Subsequent	Remove all steps after the current step.

Note: Removing a step may cause subsequent steps to stop working. For example, if you remove a formula field step that is used in a later formula field, Tableau CRM can't calculate the second formula. When this happens, the affected step appears in red, with an explanation of the error, alerting you to the need to edit the step.
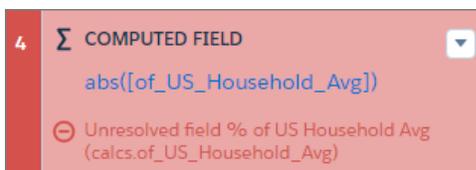
Save a Recipe

If you're not ready to create the target dataset, save the recipe and come back to it later. That saves your steps without you having to create the dataset.

To save a recipe, click **Save**.
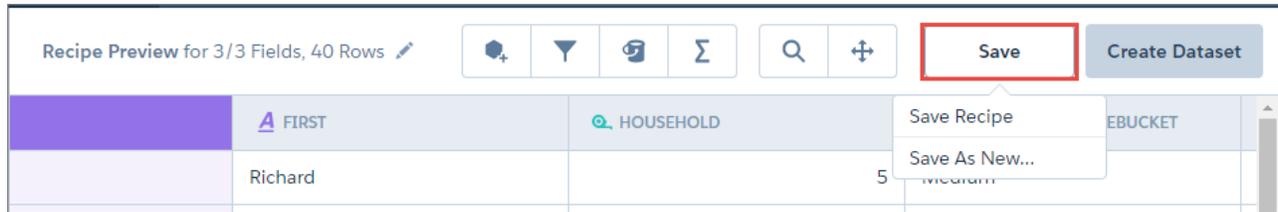
You have two options when saving a recipe.

Save Recipe

Save the recipe with its current name. The recipe does not run. Use this option to save your progress.

Save As New...

Save a copy of your recipe with a new name. The copy opens in a new tab in the data manager. Use this option to try out different transformations in a recipe, without changing your original recipe.

Open a saved recipe on the Dataflows & Recipes tab of the data manager to update it or run it. See Manage Recipes.

Manage Recipes

After you create a recipe, use the Dataflows and Recipes tab to edit and delete them.

Open and Edit a Recipe

Use Data Prep to open or edit recipes built with Data Prep or Data Prep Classic. By default, all recipes open with Data Prep. When you edit a Data Prep Classic recipe, the recipe upgrades to and opens with Data Prep.

Revert a Data Prep Recipe to Data Prep Classic

By default, when you open a Data Prep Classic recipe, it upgrades to and opens in Data Prep. If the upgraded version has any issues, you can revert it to the Data Prep Classic version. The recipe reverts to the last saved Data Prep Classic version. Tableau CRM also creates a copy of the Data Prep recipe so you don't lose changes made in Data Prep.

Delete a Recipe

Delete recipes that you no longer need. Deleting a recipe does not delete its target dataset.

Open and Edit a Recipe

Use Data Prep to open or edit recipes built with Data Prep or Data Prep Classic. By default, all recipes open with Data Prep. When you edit a Data Prep Classic recipe, the recipe upgrades to and opens with Data Prep.

1. In Tableau CRM, click **Data Manager** in the left pane.
   Data manager opens in a new browser tab.

2. In Data Manager, click **Dataflows & Recipes**.

3. In the Dataflows & Recipes tab, click **Recipes**.
   The Recipes subtab displays a list of recipes created with either Data Prep or Data Prep Classic.
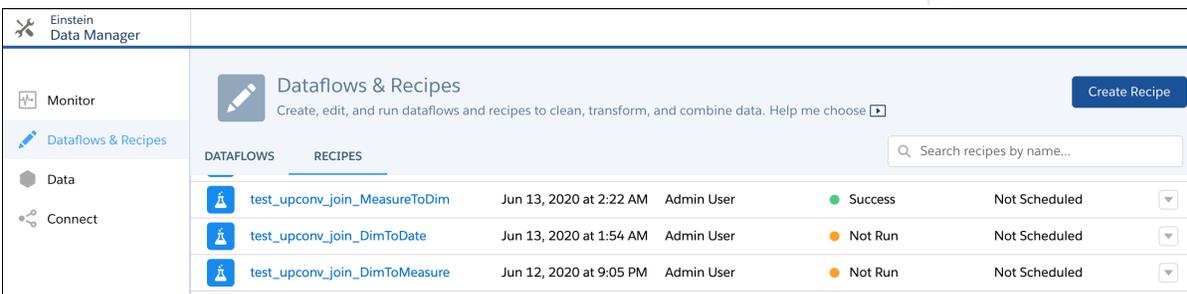
4. To edit a recipe in Data Prep, click the actions button (▼) to the right of the recipe, and then click **Edit**.

5. Edit the recipe.

6. Save the recipe.

Run the recipe to update the dataset based on your changes.

> **Note:** For information about editing recipes created with Data Prep Classic in Data Prep Classic, see Revert a Data Prep Recipe to Data Prep Classic on page 868.

Revert a Data Prep Recipe to Data Prep Classic

By default, when you open a Data Prep Classic recipe, it upgrades to and opens in Data Prep. If the upgraded version has any issues, you can revert it to the Data Prep Classic version. The recipe reverts to the last saved Data Prep Classic version. Tableau CRM also creates a copy of the Data Prep recipe so you don't lose changes made in Data Prep.

1. In Tableau CRM, click **Data Manager** in the left pane.
Data manager opens in a new browser tab.

2. In Data Manager, click **Dataflows & Recipes**.

3. In the Dataflows & Recipes tab, click **Recipes**.
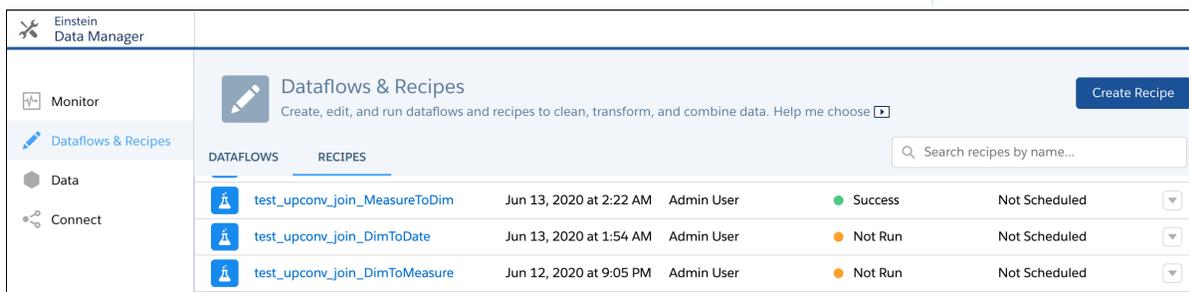The Recipes subtab displays a list of existing recipes created in either Data Prep or Data Prep Classic.

4. To revert a Data Prep recipe to Data Prep Classic, click the actions button (▼) to the right of the recipe, and then click **Revert to Data Prep Classic Recipe**.

The Revert to Data Prep Classic Recipe option appears only for recipes that were created with Data Prep Classic and upgraded to Data Prep.

The reverted recipe opens in Data Prep Classic. This recipe version is the last saved recipe in Data Prep Classic—the one that was previously upgraded to Data Prep. This version doesn't contain changes made with Data Prep.

5. When you're done, save the recipe in Data Prep Classic.
The Recipes subtab shows two versions of the recipe, one Data Prep Classic version (the reverted one) and one Data Prep version (a backup copy). Both recipes have the same name. The recipe with the Information icon ( ) is the Data Prep Classic version.

6. To distinguish the two recipes, rename the Data Prep recipe. To change the name, edit it and rename it in Data Prep.

   Both recipes also write to the same dataset. The dataset updates each time you run either recipe. Lenses and dashboards based on this dataset show the latest data based on the last run recipe.

7. To edit the Data Prep Classic recipe in Data Prep Classic, click the actions button (▼) to the right of the recipe, and then click **Open with Data Prep Classic**.

   The Open with Data Prep Classic option appears only for recipes reverted to Data Prep Classic.

8. To edit the Data Prep Classic recipe with Data Prep, click the actions button (▼) to the right of the recipe, and then click **Edit**.

   After you save the recipe in Data Prep, the recipe upgrades to Data Prep. You no longer see the Information icon because it's no longer a Data Prep Classic recipe. If you didn't rename the previous Data Prep recipe, you now have two Data Prep recipes with the same name.
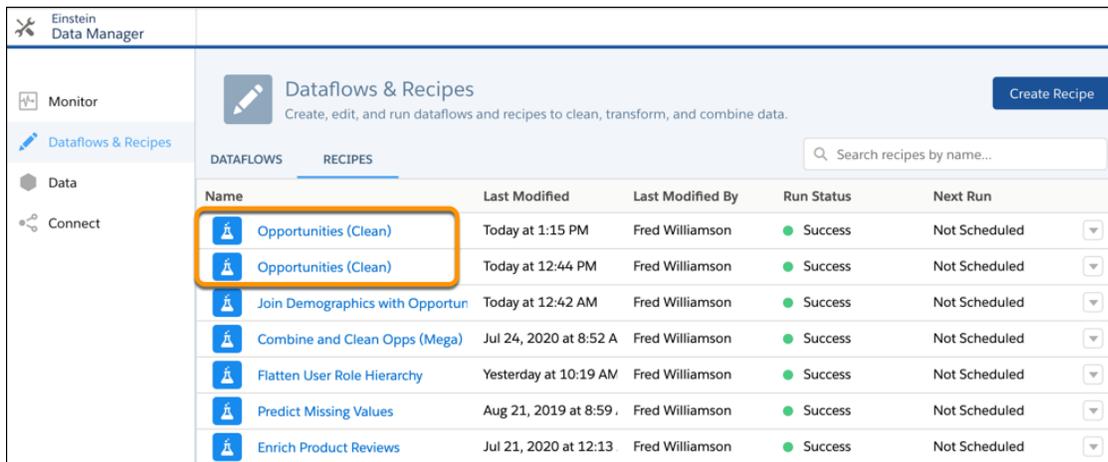
If needed, you can revert the latest Data Prep recipe to Data Prep Classic.

Delete a Recipe

Delete recipes that you no longer need. Deleting a recipe does not delete its target dataset.

1. In Tableau CRM, click **Data Manager** in the left pane.
   Data manager opens in a new browser tab.

2. In Data Manager, click **Dataflows & Recipes**.

3. In the Dataflows & Recipes tab, click **Recipes**.
   The Recipes subtab displays a list of existing recipes, those created in Data Prep Classic and Data Prep.

4. Click the actions button to the right of the recipe that you want to delete, and then click **Delete**.

Design Datasets with Dataflows and the Dataset Builder

Use a dataflow to create one or more datasets based on source data from existing datasets or synced data. A dataflow is a set of instructions that specifies what input data to include, how to transform that data, and which datasets to load the transformed data into.

To build a simple dataset from related Salesforce data, use Dataset Builder. To build more complex datasets that can include external and transformed data, use Dataflow Editor. Both tools add the logic for building the datasets to a dataflow.

After you design a dataflow with either tool, run the dataflow to create the datasets. You can schedule a dataflow to refresh the data in the datasets on a regular interval. You can create multiple dataflows if you want to run them on different intervals or break up the data integration logic used to build your datasets. Just keep in mind that there's a limit on the number of dataflows you can create in your org.

Design a Simple Dataset with Dataset Builder
Use Dataset Builder to create a single dataset based on data from one or more related Salesforce objects. Dataset Builder adds the instructions for building the dataset to the specified dataflow. The dataset is created the next time the dataflow runs, and refreshes each time the dataflow runs thereafter. To edit the dataflow, use the Dataflow Editor.

Design Complex Datasets with Dataflow Editor
Use Dataflow Editor, a point-and-click interface, to build your dataflow logic from scratch. Add transformations to determine what source data to use, how to transform that data, and which datasets to load the results into.

SEE ALSO:
Should I Use a Recipe or Dataflow?
Tableau CRM Limits

Design a Simple Dataset with Dataset Builder

Use Dataset Builder to create a single dataset based on data from one or more related Salesforce objects. Dataset Builder adds the instructions for building the dataset to the specified dataflow. The dataset is created the next time the dataflow runs, and refreshes each time the dataflow runs thereafter. To edit the dataflow, use the Dataflow Editor.

1. On the home page or on an app page, click **Create** > **Dataset**.

2. Click **Salesforce Data**.
   The New Dataset dialog opens.

New Dataset ✕

Name the dataset and select a dataflow to add it to. The dataflow creates the dataset and keeps it updated. Learn More

Dataset Name

Opportunities with Accounts and Users

● Add to existing dataflow

Sales Operations Dataflow   ✕ ▼

○ Add to new dataflow

Enter name for new dataflow...

Back   Next

3. Enter a name for the dataset.

   📝 Note: If you enter a dataset name that is already used, when you create the dataset, Dataset Builder appends a number to the dataset name. For example, if you entered MyOpportunities, Dataset Builder creates MyOpportunities1. The dataset name can't exceed 80 characters.

4. Select a dataflow to add the transformations to. You can select an existing dataflow, or a new dataflow.

   a. To add the transformations to an existing dataflow, select **Add to existing dataflow**, and then select the dataflow from the list.

   b. To add the transformations to a new dataflow, select **Add to new dataflow**, and then enter a name for the new dataflow.

   📝 Note: The option to add to a new dataflow is only available if you have enabled data sync in your org. See Enable Data Sync and Connections.

5. Click **Next**.
   Dataset Builder opens inside the Dataflow Editor.
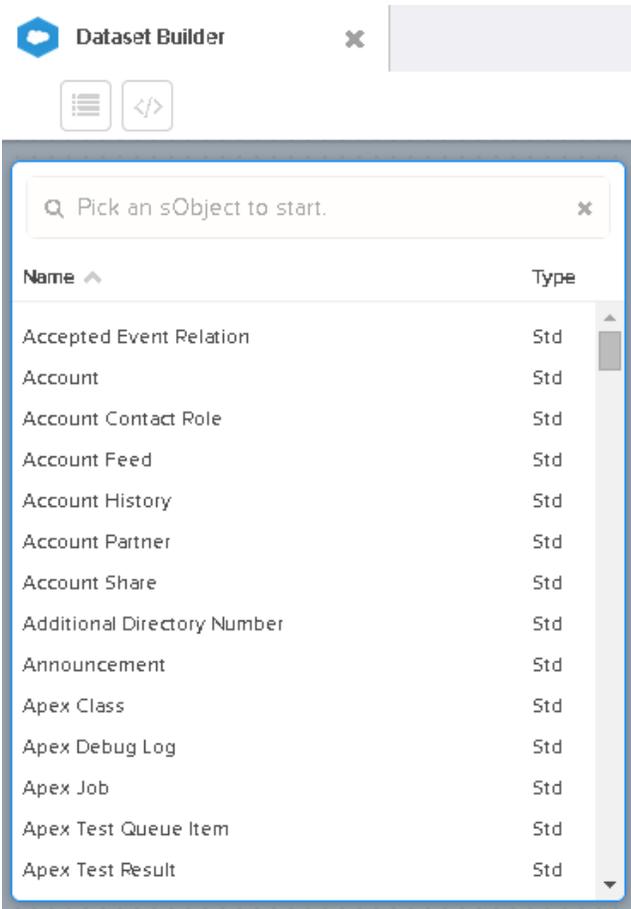
6. Select the root object.

   The root object is the lowest level child object that you can add to the canvas. After you select the root object, you can add only parent objects of the root object—you can't add its child objects. To change the root object, refresh the page and start over.

7. Hover over the root object, and then click ![plus icon].
   The Select Fields dialog box appears. By default, the Fields tab appears.
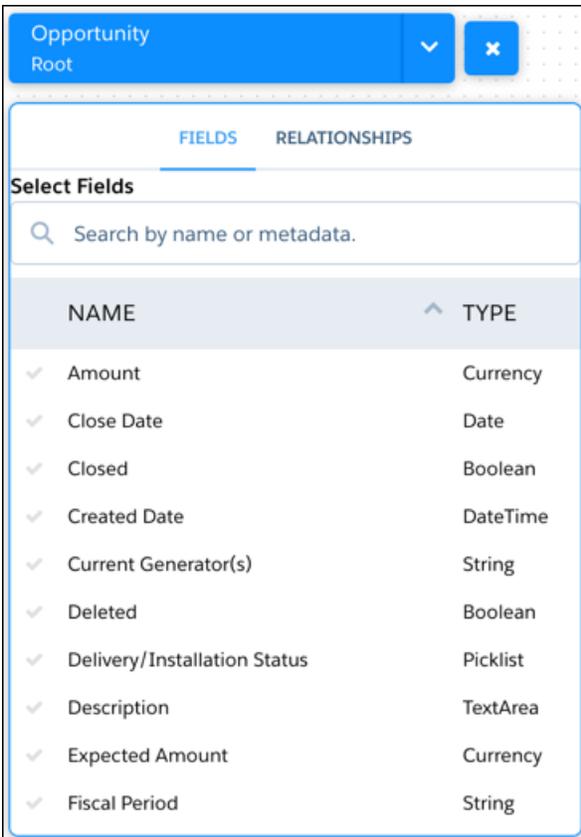
> **Note:** You can view this dialog box for any object included in the canvas.

8. In the Fields tab, select the fields from which you want to extract data.

   To locate fields more quickly, you can search for them or sort them by name or type.

   > **Important:** Select at least one field for each object that you add to the canvas. If you add an object and don't add any of its fields, the dataflow fails at run time.

9. In the Relationships tab, click **Join** to add the related objects to the canvas.
   When you add a related object, the related object appears on the canvas.
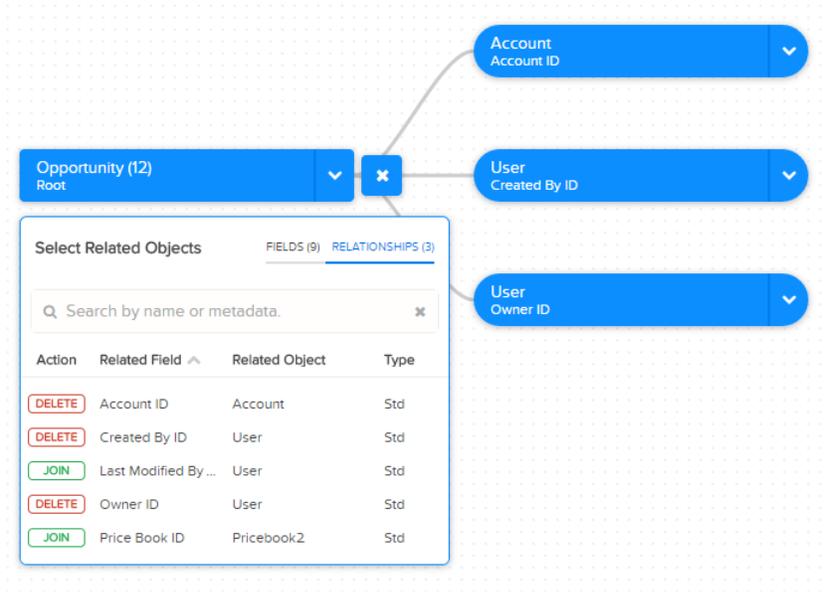
10. To remove a related object, click **Delete**.

> ⚠ Warning: When you delete a related object, you also delete all objects that descend from the related object. For example, if you delete Account shown below, you delete the branch that contains Account and User.
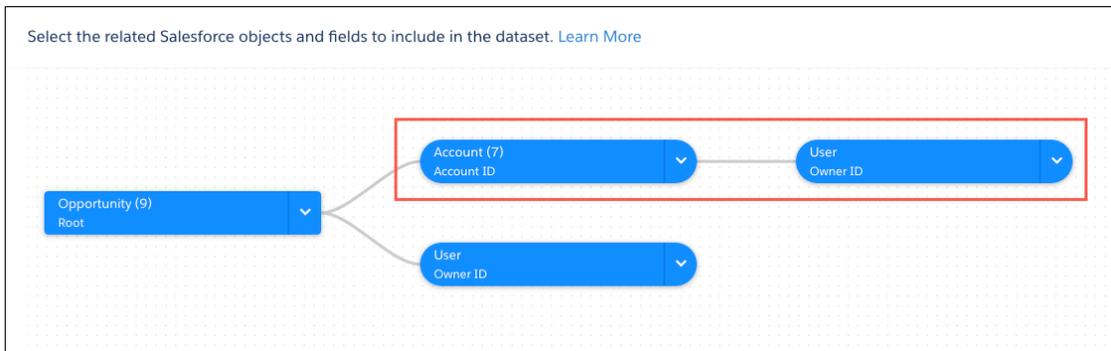
11. For each related object, select the fields from which you want to extract data.

12. To move the entire diagram, select a white space in the canvas and drag it.

If needed, move the diagram to view a different section.

13. When you finish adding objects and fields, click **Next**.

The transformations for the new dataset are added to the dataflow you selected.
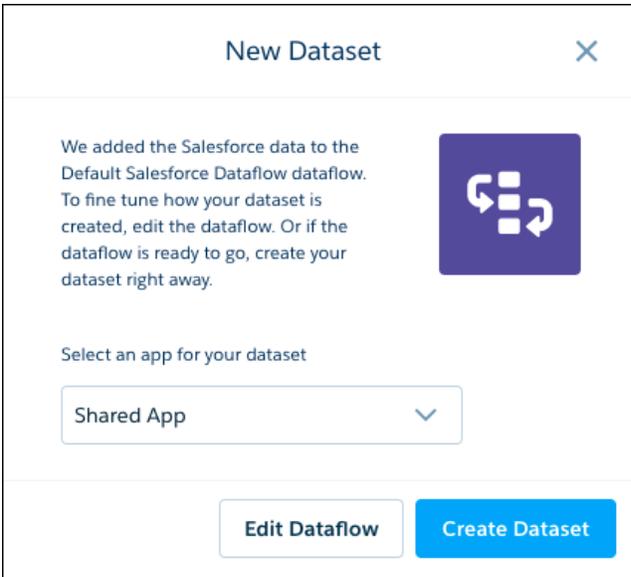
14. Select the app that will contain the dataset, if it's not already selected.

15. To view or edit the dataflow, click **Edit Dataflow**.

> 📝 Note: To edit the dataflow, you must select the Shared App for the dataset.

16. To run the dataflow and create the dataset, click **Create Dataset**.

SEE ALSO:

Run a Dataflow

Design Complex Datasets with Dataflow Editor

Use Dataflow Editor, a point-and-click interface, to build your dataflow logic from scratch. Add transformations to determine what source data to use, how to transform that data, and which datasets to load the results into.

> 💡 Tip: More advanced users can edit the underlying dataflow definition file. A dataflow definition file is a JSON file that contains transformations that represent the dataflow logic. See Edit the Dataflow JSON File.

Dataflow Editor displays the dataflow you're editing.
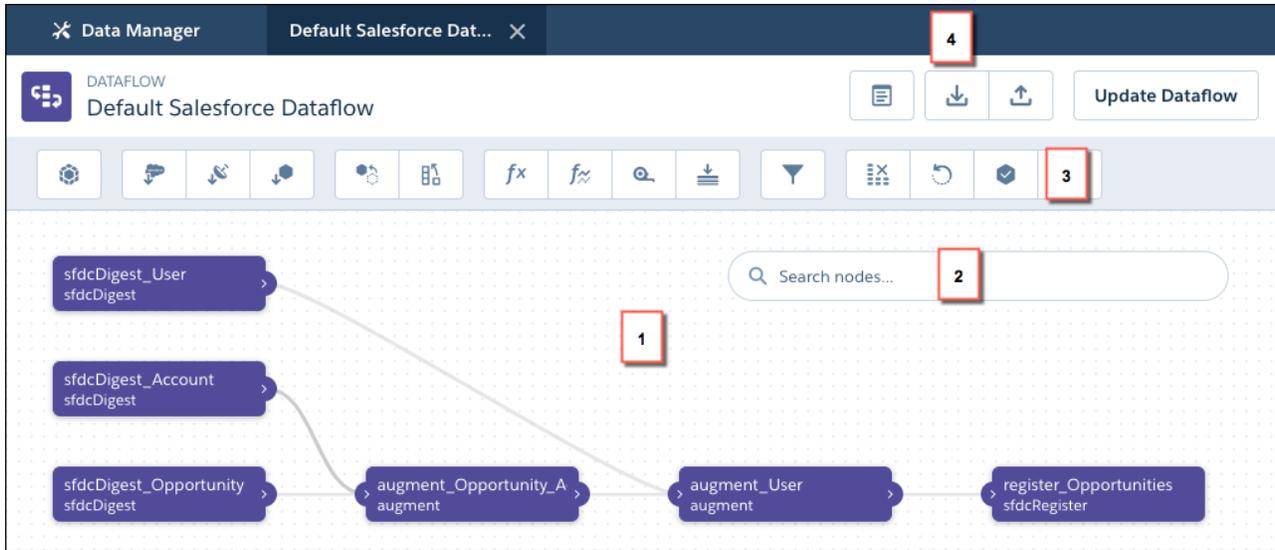
The canvas (1) shows the individual nodes in your dataflow and the links between them. If you can't find a node, enter the node name in the search box (2). The node palette (3) has buttons for each type of node. Click one to add it to the canvas. Use the buttons at the top of the editor (4) to work with the dataflow JSON. Preview and download the underlying JSON or upload an existing JSON file to work with it in the editor.

The dataflow in this example extracts data from the Salesforce Opportunity, Account, and User objects, augments it together, and creates a registered dataset. If you also want to extract cases in this dataflow, augment them with case data, and register the result as a new dataset. Let's look at how you can use the editor to modify the dataflow.

1. In Analytics Cloud, click the gear icon (⚙) and then click **Data Manager**.

2. Click the **Dataflows & Recipes** tab.

3. To create a dataflow from scratch, click **Create Dataflow**.

4. To edit an existing dataflow, click the actions button to the right of the dataflow that you want to edit, and select **Edit**.
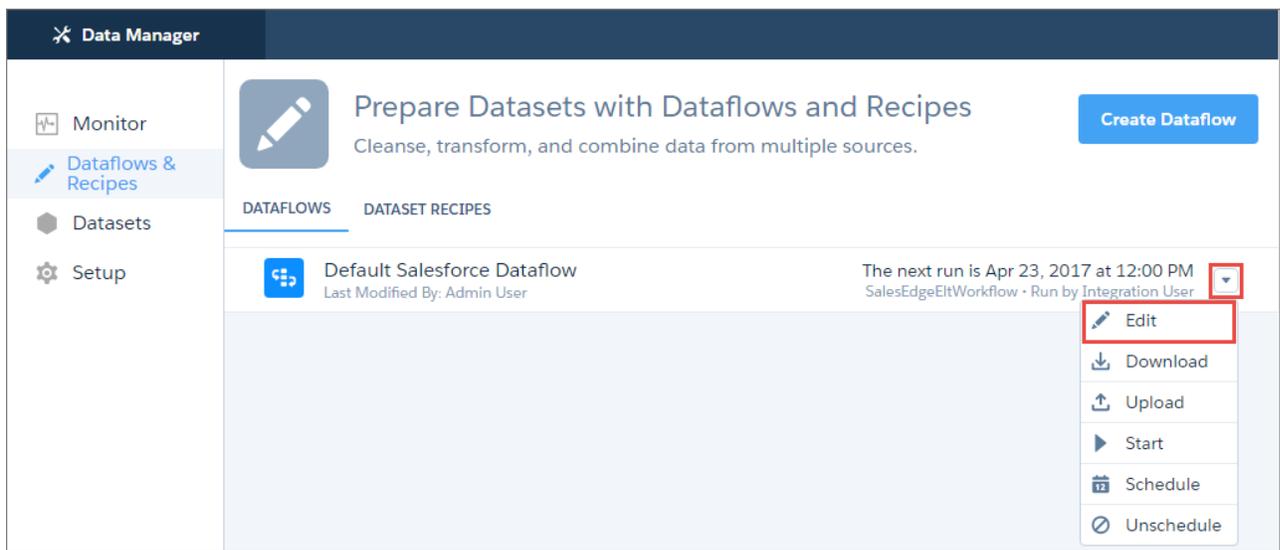
Dataflow Editor opens, displaying the nodes in the dataflow.

5. In the node palette, click the node that you want to add.

For example, to extract data from a Salesforce object, click the sfdcDigest node ( ).

6. Enter a name for the node.
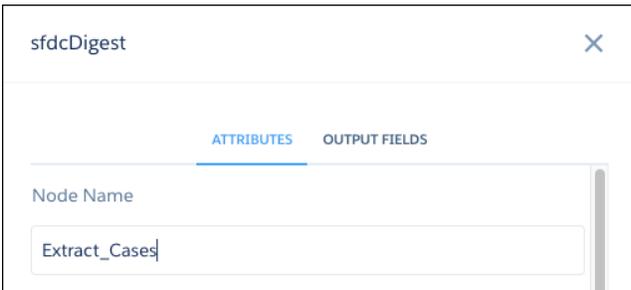
> 🖉 **Note:** Node names must be unique within the dataflow and not contain spaces.

7. Enter the node attributes. In this example, you select Case as the source object, and the fields you need. You don't need an incremental update or filters.
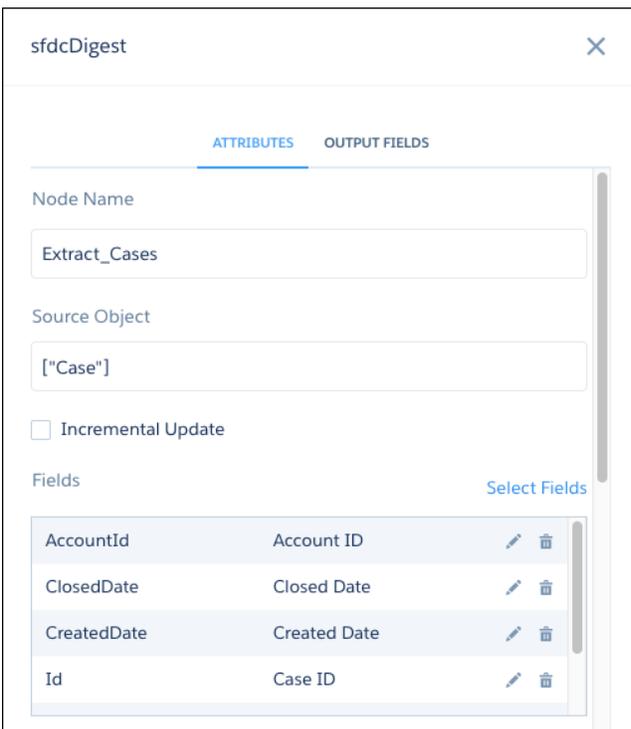
> 🖉 **Note:** When you enter attributes that reference objects, fields, or other nodes, click inside the attribute field to see a list that you can search. Then select from. To see the field list in sfdcDigest nodes, click **Select Fields**.

8. Click **Create**.

The new node appears on the canvas.

> **Tip:** To edit a node on the canvas, click it. To delete a node, hover over it and click the trash can.

9. In the node palette, click the next node you want to add. In this example, you want to augment the case data with user data, so click the augment node ( ).

10. Enter the node name and attributes. In this example, you select the `Extract Cases` node as the left source. This node is the one you want to add the user columns to. The `Extract Users` node contains the columns you want to add, so select it as the right source. The left and right keys are the fields from each node that are used to match records. In this case, select the `OwnerId` field on cases and the `Id` field on users.
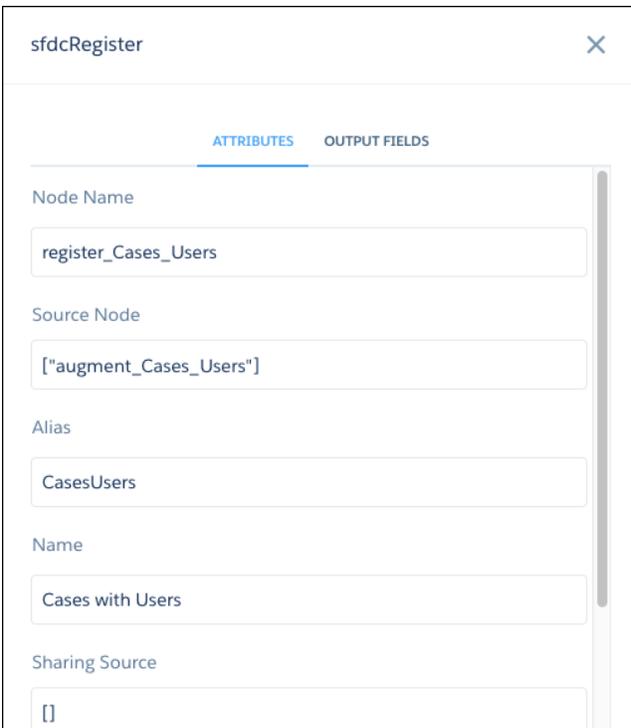

11. Click **Create**.
    The new node appears on the canvas, with lines showing the input of data from the source nodes.
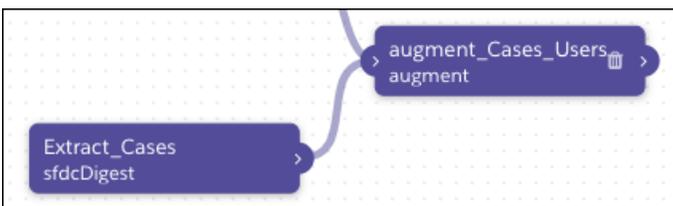
> 💡 **Tip:** If a node is not where you want it, drag it to wherever you want on the canvas. If it gets crowded, use your scroll wheel to zoom in and out if your browser supports it, or drag the canvas to move nodes into view.

12. In the node palette, click the next node you want to add. In this example, you want to make the augmented case and user data

available to use in a dataset, so click the sfdcRegister node (🛡️).

13. Enter the node name and attributes. In this example, you select the `augment_Cases_Users` node as the source. The Alias is the API name of the registered dataset, and the Name is its display name.

14. Click **Create**.

The new node appears on the canvas, with a line showing the input of data from the source node.
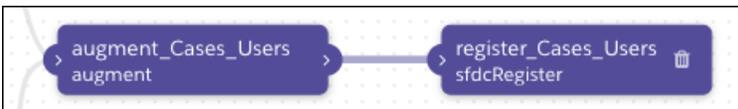
> 💡 **Tip:** You can drag the output arrow of a source node to draw a link to another node on the canvas.

15. When you finish editing, click **Update Dataflow** to update the definition file with your changes.

16. In the Update dataflow dialog, enter a description in the Version History field.

17. Click **Update Dataflow**.

Tableau CRM validates your dataflow to ensure that you provided all the required attribute values. Tableau CRM also ensures that you can't continue if there are errors. Correct the errors it finds before trying again. After Tableau CRM validates the dataflow, it saves the new version of the dataflow.

18. Click **Run Dataflow** to run the dataflow right away and create newly defined datasets. Alternatively, you can run the dataflow later from the Dataflows & Recipes tab of Data Manager.

Plan the Dataflow

Before you start creating the dataflow, think about the dataflow design. Consider what data to make available for queries, where to extract the data from, and whether you need to transform the extracted data to get the data you want.

Transformations for Tableau CRM Dataflows

A *transformation* refers to the manipulation of data. You can add transformations to a dataflow to extract data from Salesforce objects or datasets, transform datasets that contain Salesforce or external data, and register datasets.

Edit Nodes in the Dataflow

When you edit a dataflow node, for example to change its name, or add or remove fields, consider that these changes can impact downstream nodes. If you're manually editing nodes in the JSON dataflow definition file, make sure that your changes are reflected in downstream nodes. If you're working in the dataflow editor, Tableau CRM propagates many of your changes to downstream nodes for you.

Configure the Dataflow Through the Definition File

You can configure the dataflow by adding transformations directly to the dataflow definition file.

SEE ALSO:

Should I Use a Recipe or Dataflow?

Design a Simple Dataset with Dataset Builder

Transformations for Tableau CRM Dataflows

Run a Dataflow

Plan the Dataflow

Before you start creating the dataflow, think about the dataflow design. Consider what data to make available for queries, where to extract the data from, and whether you need to transform the extracted data to get the data you want.

To illustrate some key design decisions, let's consider an example. In this example, the goal is to create a dataset called "Won Opportunities." The dataset will contain opportunity details, including the account name for each opportunity.

To create this dataset, you design the following dataflow:
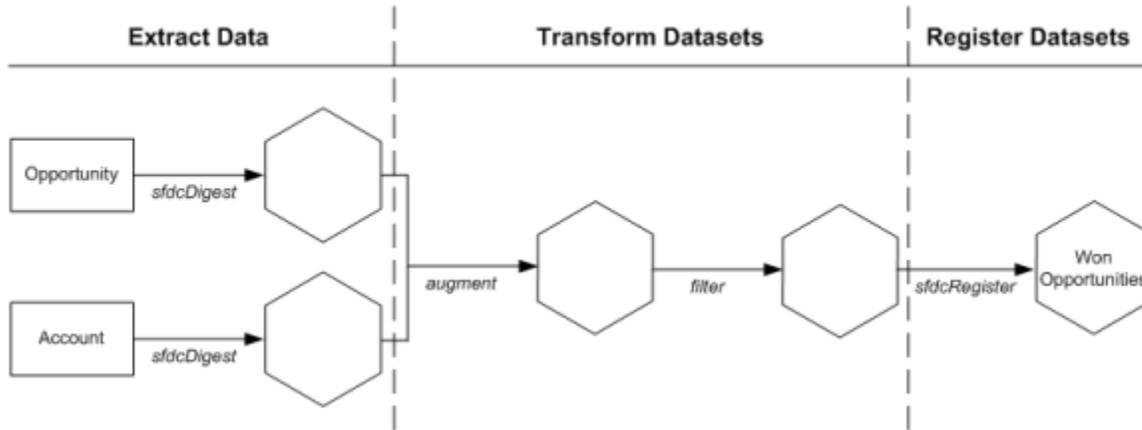
The dataflow extracts opportunity data from the Opportunity object and extracts the account name from the Account object. For each extracted object, the dataflow creates a new dataset.

The dataflow then transforms the datasets created from the extracted data. First, the dataflow joins the opportunity and account data into a new dataset. Next, the dataflow filters the records based on the opportunity stage so that the dataset contains only won opportunities. Each time the dataflow transforms a dataset, it creates a new dataset.

Finally, because you want users to be able to query won opportunities only, you configure the dataflow to register the final dataset only. However, if you wanted, you could register any dataset created by the dataflow and register as many datasets as you like.

Carefully choose which datasets to register because:

- The total number of rows in all registered datasets cannot exceed 100,000 per platform license, or 250 million per platform license *purchased before* October 20, 2015.

- Users that have access to registered datasets can query their data. Although, you can apply row-level security on a dataset to restrict access to records.

Transformations for Tableau CRM Dataflows

A *transformation* refers to the manipulation of data. You can add transformations to a dataflow to extract data from Salesforce objects or datasets, transform datasets that contain Salesforce or external data, and register datasets.

For example, you can use transformations to join data from two related datasets and then register the resulting dataset to make it available for queries.

append Transformation

The append transformation combines rows from multiple datasets into a single dataset.

augment Transformation

The augment transformation adds columns to a dataset from another related dataset. The resulting, augmented dataset enables queries across both related input datasets. For example, you can augment the Account dataset with the User dataset to enable a query to return account records and the full names of the account owners.

computeExpression Transformation

The computeExpression transformation enables you to add derived fields to a dataset. The values for derived fields aren't extracted from the input data source. Instead, Tableau CRM generates the values using a SAQL expression, which can be based on one or more fields from the input data or other derived fields. For example, you can use an expression to assign a value to a field, concatenate text fields, or perform mathematical calculations on numeric fields.

computeRelative Transformation

You can use the computeRelative transformation to analyze trends in your data by adding derived fields to a dataset based on values in other rows. For example, to analyze sales pipeline trends, create derived fields that calculate the number of days an opportunity remains in each stage. You can also calculate the changes to the opportunity amount throughout the stages of the opportunity.

delta Transformation

The delta transformation calculates changes in the value of a measure column in a dataset over a period of time. The delta transformation generates an output column in the dataset to store the delta for each record. Create deltas to make it easier for business analysts to include them in queries.

digest Transformation

The digest transformation extracts synced connected data in a dataflow. Use it to extract data synced from an external Salesforce org, or data synced through an external connection. Use the sfdcDigest transformation to extract from your local Salesforce org.

dim2mea Transformation

The dim2mea transformation creates a new measure based on a dimension. The transformation adds the new measure column to the dataset. The transformation also preserves the dimension to ensure that existing lenses and dashboards don't break if they use the dimension.

edgemart Transformation

The edgemart transformation gives the dataflow access to an existing, registered dataset, which can contain Salesforce data, external data, or a combination of the two. Use this transformation to reference a dataset so that its data can be used in subsequent transformations in the dataflow. You can use this transformation and the augment transformation together to join an existing dataset with a new dataset.

export Transformation

The export transformation creates a data file and a schema file from data in a specified source node in your dataflow. After the dataflow runs, Einstein Discovery users can access these files through the public API.

filter Transformation

The filter transformation removes records from an existing dataset. You define a filter condition that specifies which records to retain in the dataset.

flatten Transformation

The flatten transformation flattens hierarchical data. For example, you can flatten the Salesforce role hierarchy to implement row-level security on a dataset based on the role hierarchy.

prediction Transformation

The prediction transformation makes an Einstein Discovery prediction for a dataset. Einstein Discovery uses *predictive analytics*, which analyzes historical data (based on data mining, machine learning, and predictive statistical modeling) to identify patterns and predict future outcomes.

sfdcDigest Transformation

The sfdcDigest transformation generates a dataset based on data that it extracts from a Salesforce object. You specify the Salesforce object and fields from which to extract data. You might choose to exclude particular fields that contain sensitive information or that aren't relevant for analysis.

sfdcRegister Transformation

The sfdcRegister transformation registers a dataset to make it available for queries. Users cannot view or run queries against unregistered datasets.

sliceDataset Transformation

The sliceDataset transformation removes fields from a dataset in your dataflow, leaving you with a subset of fields for use in a new dataset or in other transformations. This allows you to create multiple datasets, each with different sets of fields from a single dataset.

update Transformation

The update transformation updates the specified field values in an existing dataset based on data from another dataset, which we'll call the lookup dataset. The transformation looks up the new values from corresponding fields in the lookup dataset. The transformation stores the results in a new dataset.

Overriding Metadata Generated by a Transformation

Optionally, you can override the metadata that is generated by a transformation. You can override object and field attributes. For example, you can change a field name that is extracted from a Salesforce object so that it appears differently in the dataset. To override the metadata, add the overrides to the Schema section of the transformation in the dataflow definition file.

append Transformation

The append transformation combines rows from multiple datasets into a single dataset.

📝 **Note:** To append rows in a Data Prep recipe, see Append Node: Stack Rows from Different Sets of Data. To do it in a Data Prep Classic recipe, see Add Rows in a Recipe with Append.

Consider the following rules when using this transformation.

- This transformation does not remove duplicate records.
- All input datasets must have the same structure—the corresponding columns must be in the same order and have the same name and field type.

👁 **Example:** Let's look at an example. Each month, you create a dataset that contains the month's sales targets. Now, you want a holistic view of sales targets for all months. To do that, you create the following dataflow to merge the existing datasets into a single dataset.
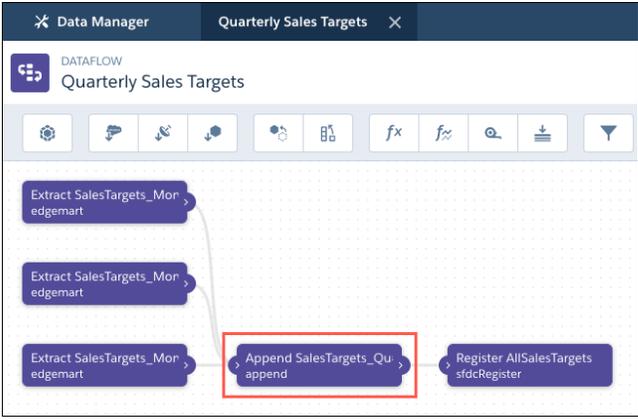
Dataflow	Append Node in Dataflow
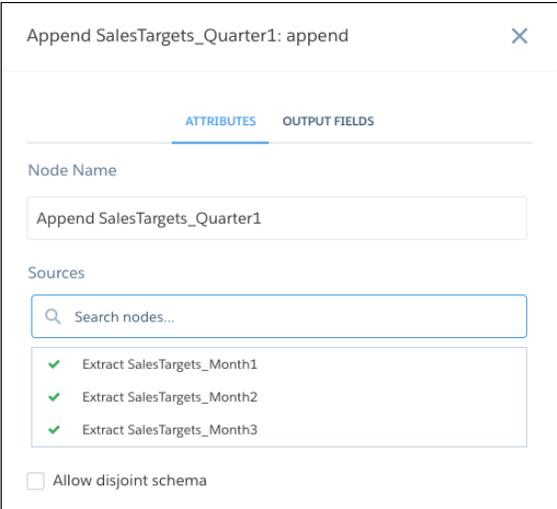
Here's the dataflow JSON.

```
{
  "Extract SalesTargets_Month3": {
    "action": "edgemart",
    "parameters": {
      "alias": "SalesTargets_Month3"
```

```
      }
    },
    "Extract SalesTargets_Month2": {
      "action": "edgemart",
      "parameters": {
        "alias": "SalesTargets_Month2"
      }
    },
    "Extract SalesTargets_Month1": {
      "action": "edgemart",
      "parameters": {
        "alias": "SalesTargets_Month1"
      }
    },
    "Append SalesTargets_Quarter1": {
      "action": "append",
      "parameters": {
        "enableDisjointedSchemaMerge": false,
        "sources": [
          "Extract SalesTargets_Month1",
          "Extract SalesTargets_Month2",
          "Extract SalesTargets_Month3"
        ]
      }
    },
    "Register AllSalesTargets": {
      "action": "sfdcRegister",
      "parameters": {
        "name": "All Sales Targets",
        "alias": "AllSalesTargets",
        "source": "Append SalesTargets_Quarter1"
      }
    }
}
```

After you create the single dataset, you can use date filters to analyze the sales targets by month, quarter, or year.

📝 **Note:** The append transformation doesn't remove duplicate rows.

Append Datasets with Different Schema

By default, input datasets must have the same structure—the corresponding columns must have the same name and field type. For example, let's say you want to append your Canada opportunities to your U.S. opportunities.

<div align="center">

US Sales

#	Name	Stage	Amount
1	US Opp1	Negotiation	100
2	US Opp2	Quote	200
3	US Opp3	Prospecting	300
4	US Opp4	Needs Analysis	200

</div>

<div align="center">

Canada Sales

#	Name ▲	Currency	Stage	Value
1	Ca Opp1	CAD	Negotiation	100
2	Ca Opp2	CAD	Quote	200
3	Ca Opp3	CAD	Prospecting	300
4	Ca Opp4	CAD	Needs Analysis	200

</div>

Here, the column names are different, and the Canada data has a Currency column. For the dataflow not to fail, select **Allow disjoint schema** in the append node in the dataflow editor. If you're working in the dataflow JSON, add the `enableDisjointedSchemaMerge` parameter and set its value to `true`.

<div align="center">

Append Node in Dataflow Editor   **Append Node in Dataflow JSON**

</div>


```
"Append US & CA Sales": {
    "action": "append",
    "parameters": {
"enableDisjointedSchemaMerge": true,
        "sources": [
        "Extract US Sales",
        "Extract CA Sales"
            ]
        }
    },
```

When you run the dataflow, the data is merged without the dataflow failing.

#	Name	Currency	Stage	Amount	Value
1	US Opp1	-	Negotiation	100	-
2	US Opp2	-	Quote	200	-
3	US Opp3	-	Prospecting	300	-
4	US Opp4	-	Needs Analysis	200	-
5	Ca Opp1	CAD	Negotiation	-	100
6	Ca Opp2	CAD	Quote	-	200
7	Ca Opp3	CAD	Prospecting	-	300
8	Ca Opp4	CAD	Needs Analysis	-	200

The append transformation adds all columns to the dataset and merges values in columns with the same name, such as Name. It also adds null values in a column for rows that didn't previously have that column. You can see this in the Currency column.

> **Note:** If null measure handling in datasets isn't enabled for your org, append adds zeros in a column for rows that didn't previously have that column.

append Parameters
When you define an append transformation in the dataflow JSON, you set the action attribute to `append` and specify the parameters.

append Parameters
When you define an append transformation in the dataflow JSON, you set the action attribute to `append` and specify the parameters.

This table describes the settings for the append node in the dataflow editor.

Setting	Required?	Value
Sources	Yes	Nodes in the dataflow that identify the datasets that you want to merge.
Allow disjoint schema	No	Select to allow appending of datasets with different schema. > **Note:** If this setting isn't selected, appending datasets with different schemas causes the dataflow to fail.

This table describes the input parameters for the append transformation in the dataflow JSON.

Parameter	Required?	Value
sources	Yes	Nodes in the dataflow definition file that identify the datasets that you want to merge.
enableDisjointedSchemaMerge	No	Indicates whether appending datasets with different schema is allowed. • To allow appending of disjoint schema, set to true.

Parameter	Required?	Value
		• To prevent appending of disjoint schema, set to false. The default is false.
		Example:
		`"enableDisjointedSchemaMerge": true`
		📝 Note: If this parameter is set to false, appending datasets with different schemas causes the dataflow to fail.

SEE ALSO:

    append Transformation

augment Transformation

The augment transformation adds columns to a dataset from another related dataset. The resulting, augmented dataset enables queries across both related input datasets. For example, you can augment the Account dataset with the User dataset to enable a query to return account records and the full names of the account owners.

📝 Note: To add columns from a related object in a Data Prep recipe, see Join Node: Add Related Columns of Data to the Recipe. To do it in a Data Prep Classic recipe, see Add More Data in a Recipe.

When you create the transformation, you identify each input dataset as the left or right dataset and specify the relationship between them. Analytics Cloud combines all the columns of the left dataset with only the specified columns from the right dataset. (Keep in mind that each dataset can't have more than 5,000 columns.) Analytics Cloud adds the relationship to column names from the right dataset, which is useful when the left and right datasets have columns with the same names.

For each record in the left dataset, the augment transformation performs a lookup to find a matching record in the right dataset. To match related records, the augment transformation uses a match condition. You specify the match condition based on a key from each dataset. A key can be a single-column key or a composite key. For a match condition based on a composite key, the keys for both datasets must have the same number of columns, specified in the same order.

💡 Tip: To augment three or moredatasets, augment two datasets at a time. For example, to augment three datasets, augment the first two datasets, and then augment the resulting dataset with the third dataset.

👁 Example: Let's look at an example of the augment transformation. In this example, you want to extract data from the Opportunity and Accounts objects, and then match the data based on the account ID field.
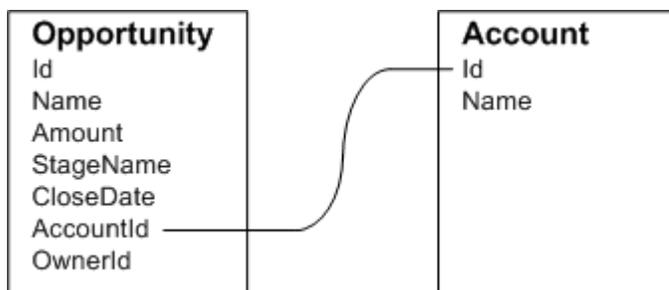
Dataflow

Augment Node in Dataflow

Here's the dataflow JSON.

```json
{
  "sfdcDigest_Account": {
    "action": "sfdcDigest",
    "parameters": {
      "fields": [
        {"name": "Id"},
        {"name": "Name"}
      ],
      "object": "Account"
    }
  },
  "sfdcDigest_Opportunity": {
```

```
        "action": "sfdcDigest",
        "parameters": {
          "fields": [
            {"name": "Id"},
            {"name": "Name"},
            {"name": "Amount"},
            {"name": "StageName"},
            {"name": "CloseDate"},
            {"name": "AccountId"},
            {"name": "OwnerId"}
          ],
          "object": "Opportunity"
        }
      },
      "augment_Oppt_Acct": {
        "action": "augment",
        "parameters": {
          "operation": "LookupSingleValue",
          "left": "sfdcDigest_Opportunity",
          "left_key": [
            "AccountId"
          ],
          "relationship": "OpptyAcct",
          "right": "sfdcDigest_Account",
          "right_key": [
            "Id"
          ],
          "right_select": [
            "Name"
          ]
        }
      },
      "Create OpptsAccts": {
        "action": "sfdcRegister",
        "parameters": {
          "alias": "Opportunity_Account",
          "source": "augment_Oppt_Acct",
          "name": "Opportunity_Account"
        }
      }
    }
}
```

After you run the dataflow, Analytics Cloud creates and registers the Opportunity_Account dataset. It also adds the relationship as a prefix to all columns from the right dataset.
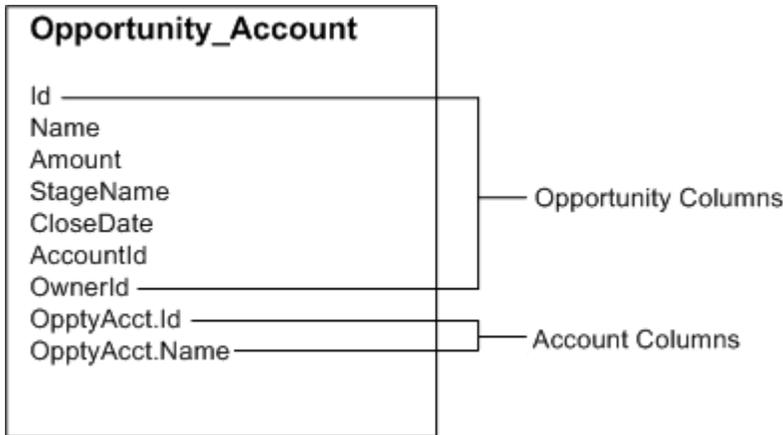
Special Cases for Matching Records with the augment Transformation

For each record in the left dataset, the augment transformation performs a lookup to find a matching record in the right dataset. However, it's critical that you understand how the augment transformation handles special cases when matching records.

augment Parameters

When you define an augment transformation, you set the action attribute to `augment` and specify the parameters.

Special Cases for Matching Records with the augment Transformation

For each record in the left dataset, the augment transformation performs a lookup to find a matching record in the right dataset. However, it's critical that you understand how the augment transformation handles special cases when matching records.

Let's look at some examples that illustrate some special cases.

Handling Null Keys

When a record in the left dataset contains a null key, Analytics Cloud doesn't perform a lookup to match the record. Instead, Analytics Cloud appends the right columns and inserts null for dimensions (including dates) and '0' for measures.

Let's look at an example. You apply the augment transformation on the following datasets, set the relationship to "Price,", and match the records based on the `Id` and `ProdId` fields.

Analytics Cloud doesn't match the last record because the product ID is null. Instead, Analytics Cloud inserts a null for the `Price.Pricebook` dimension and '0' for the `Price.UnitPrice` measure. Here's the resulting dataset after the augment.

```
{ "id":"Prod1", "Name":"Table",  "Price.Pricebook":"Standard",  "Price.UnitPrice":1000 },
{ "id":"Prod2", "Name":"Chair",  "Price.Pricebook":"Standard",  "Price.UnitPrice":450  },
{ "id":null,    "Name":"Bench",  "Price.Pricebook":null,         "Price.UnitPrice":0     }
```

Handling Empty Keys

Analytics Cloud matches empty-value left keys and empty-value right keys.

Let's look at an example. You apply the augment transformation on the following datasets, set the relationship to "Price,", and match the records based on the `Id` and `ProdId` fields.

Analytics Cloud matches the last record in the Product dataset with the third record in the Price dataset because they both have empty values (""). Here's the resulting dataset after the augment.

```
{ "id":"Prod1", "Name":"Table",  "Price.Pricebook":"Standard",  "Price.UnitPrice":1000 },
{ "id":"Prod2", "Name":"Chair",  "Price.Pricebook":"Standard",  "Price.UnitPrice":450  },
{ "id":"",      "Name":"Bench",  "Price.Pricebook":"Custom",    "Price.UnitPrice":800  }
```

Handling Non-Unique Keys

Although it's recommended, the left key doesn't have to be unique. If multiple records have the same left key, Analytics Cloud creates the same values for the appended columns.

Let's look at an example. You apply the augment transformation on the following datasets, set the relationship to "Price,", and match the records based on the `Id` and `ProdId` fields.

Analytics Cloud matches the records in the Product dataset with records in the Price dataset. Here's the resulting dataset after the augment.

```
{ "id":"Prod1", "Name":"Table",  "Price.Pricebook":"Standard",  "Price.UnitPrice":1000 },
{ "id":"Prod2", "Name":"Chair",  "Price.Pricebook":"Standard",  "Price.UnitPrice":450  },
{ "id":"Prod2", "Name":"Chair",  "Price.Pricebook":"Standard",  "Price.UnitPrice":450  }
```

Handling No Match

If the left key doesn't have a match in the right data stream, Tableau CRM appends the right columns, inserting nulls for dimensions (including dates) and sets measures based on whether null measure handling is enabled. If null measure handling is enabled, the augment transformation sets the measures to null. Otherwise, it sets the measures to '0'.

Let's look at an example where null measure handling isn't enabled. You apply the augment transformation on the following datasets, set the relationship to "Price,", and match the records based on the `Id` and `ProdId` fields.

Product			Price		
Id	**Name**		**ProdId**	**Pricebook**	**UnitPrice**
Prod1	Table		Prod4	Standard	1000
Prod2	Chair		Prod5	Standard	450
Prod3	Bench				

Because no keys match, Analytics Cloud doesn't match any records in the Product dataset with records in the Price dataset. Here's the resulting dataset after the augment.

```
{ "id":"Prod1", "Name":"Table",  "Price.Pricebook": null,  "Price.UnitPrice":0 },
{ "id":"Prod2", "Name":"Chair",  "Price.Pricebook": null,  "Price.UnitPrice":0 },
{ "id":"Prod3", "Name":"Bench",  "Price.Pricebook": null,  "Price.UnitPrice":0 }
```

If null measure handling was enabled, Price.UnitPrice would be *null*, instead of 0.

Handling Multiple Matches

If the left dataset has a one-to-many relationship with the right dataset, Analytics Cloud might find multiple matches for a left record. What Analytics Cloud does with multiple matches depends on the specified augment operation. You can specify one of the following operations to handle the multiple-match case:

LookupSingleValue

The augment transformation returns results from a single row. Analytics Cloud randomly selects one row from the list of matched rows.

📝 Note: Each time you run the dataflow, Analytics Cloud can return different results depending on the returned row.

Let's look at an example. You apply the augment transformation on the following datasets, set the relationship to "Price," set the operation to `LookupSingleValue`, and match the records based on the `Id` and `ProdId` fields.

Product			Price		
Id	**Name**		**ProdId**	**Pricebook**	**UnitPrice**
Prod1	Table		Prod1	Standard	1000
Prod2	Chair		Prod2	Standard	450
Prod3	Bench		Prod3	Custom	800
			Prod3	Standard	700

Although there are multiple rows for Prod3 in the Price dataset, Analytics Cloud randomly chooses one matching row and returns values based on that row. Here's the resulting dataset after the augment if Analytics Cloud chooses the first Prod3 row.

```
{ "id":"Prod1", "Name":"Table",  "Price.Pricebook":"Standard",  "Price.UnitPrice":1000 },
{ "id":"Prod2", "Name":"Chair",  "Price.Pricebook":"Standard",  "Price.UnitPrice":450  },
{ "id":"Prod3", "Name":"Bench", "Price.Pricebook":"Custom",    "Price.UnitPrice":800 }
```

LookupMultiValue

Analytics Cloud returns results from all matched rows.

Let's look at an example. You apply the augment transformation on the following datasets, set the relationship to "Price," set the operation to `LookupMultiValue`, and match the records based on the `Id` and `ProdId` fields.

Product	
Id	**Name**
Prod1	Table
Prod2	Chair
Prod3	Bench

Price		
ProdId	**Pricebook**	**UnitPrice**
Prod1	Standard	1000
Prod2	Standard	450
Prod3	Custom	800
Prod3	Standard	700

Because the lookup returns multiple rows for Prod3, the dimension `Price.Pricebook` field in the resulting dataset becomes a multi-value field, showing all dimension values. The measure field `Price.UnitPrice` contains 1500, which is the sum of 800 and 700. Here's the resulting dataset after the augment.

```
{ "id":"Prod1", "Name":"Table", "Price.Pricebook":"Standard", "Price.UnitPrice":1000          },
{ "id":"Prod2", "Name":"Chair", "Price.Pricebook":"Standard", "Price.UnitPrice":450           },
{ "id":"Prod3", "Name":"Bench", "Price.Pricebook":"Custom",   "Price.Pricebook":"Standard", "Price.UnitPrice":1500 }
```

📝 **Note:** If you are augmenting date fields from the right dataset, the `LookupMultiValue` operation can lead to fields containing multiple date values, which can give unexpected results. We recommend that you use the `LookupSingleValue` operation when augmenting date fields, or augment the date fields in a separate LookupSingleValue augment node.

SEE ALSO:

augment Transformation

augment Parameters

When you define an augment transformation, you set the action attribute to `augment` and specify the parameters.

The following table describes the input parameters.

Parameter	Required?	Value
operation	No	Indicates what the transformation does if it matches multiple rows in the right dataset with a row in the left. Valid values: • `LookupSingleValue`. Returns values from one of the matched rows. If you don't specify the `operation` parameter, the transformation uses this operation. • `LookupMultiValue`. Returns values from all matched rows. For more information about each operation, see Special Cases for Matching Records with the augment Transformation.
left	Yes	Node in the dataflow definition file that identifies the left dataset. This is one of two input sources for this transformation.

Parameter	Required?	Value
left_key	Yes	Key column in the left dataset used to augment the datasets. If you use a composite key, the left and right keys must have the same number of columns in the same order. For a composite key, use the following syntax: `[ "Key Column1", "Key Column2", …, "Key ColumnN" ]` 📝 **Note:** The left or right key can't be a multi-value field.
right	Yes	Node in the dataflow definition file that identifies the right dataset. This is one of two input sources for this transformation.
relationship	Yes	Relationship between the left and right datasets. The dataflow adds the relationship to the beginning of the right column names in the output dataset to make the column names unique and descriptive.
right_select	Yes	An array of column names from the right dataset that you want to include in the output dataset. The dataflow adds the relationship as a prefix to the column name to determine the name of the right column in the output dataset.
right_key	Yes	Key column in the right dataset used to augment the datasets. If you use a composite key, the left and right keys must have the same number of columns in the same order.

SEE ALSO:

[augment Transformation](#)

computeExpression Transformation

The computeExpression transformation enables you to add derived fields to a dataset. The values for derived fields aren't extracted from the input data source. Instead, Tableau CRM generates the values using a SAQL expression, which can be based on one or more fields from the input data or other derived fields. For example, you can use an expression to assign a value to a field, concatenate text fields, or perform mathematical calculations on numeric fields.

📝 **Note:** To create a calculated column based on a formula in a Data Prep recipe, see Formula Transformation: Create a Calculated Column Based on an Expression. To do it in a Data Prep Classic recipe, see Add a Formula Field in a Recipe.

The computeExpression and computeRelative transformations are similar, but they have a key difference. The computeExpression transformation performs calculations based on other fields within the same row. The computeRelative transformation performs calculations based on the previous and next values of the same field in other rows.

Consider the following guidelines when creating a computeExpression transformation:

- You can include only the following SAQL operators and functions in the expression:

  - Arithmetic operators
  - Case operator
  - String operator
  - Date functions

- Multi-value fields aren't supported. The dataflow fails if you include a multi-value field in the SAQL Expression parameter.
- The values of the derived field must match its specified type. For example, set the type of the derived field to `Text` if the values are strings.

- Tableau CRM calculates the values of derived fields in the order in which they are listed in the JSON. Thus, if you create a derived field based on other derived fields in the same computeExpression transformation, the input derived fields must be listed first. For example, Derived_A must be listed before Derived_B in the following computeExpression transformation JSON snippet:

```
"CreateDerivedFields": {
   "action": "computeExpression",
   "parameters": {
      "source": "sourceNode",
      "mergeWithSource": false,
      "computedFields": [
         {
            "name": "Derived_A",
            "type": "Text",
            "label": "Derived Field A",
            "saqlExpression": "\"hello \""},
         {
            "name": "Derived_B",
            "type": "Text",
            "label": "Derived Field B Dependent on Field A",
            "saqlExpression": "Derived_A + \"world\""}
      ]
   }
}
```

- You can choose whether the resulting dataset includes only the derived fields, or includes the input and derived fields.

- If the expression produces an overflow value for a numeric field, the transformation sets the field value to the default value specified for the column. If no default value is set and null handling is disabled, the field value is set to 0. If no default value is set and null handling is enabled, it's set to null. (See the documentation about null measure handling.) Also, if the expression defines multiple columns and the expression produces an overflow value, all generated fields are set to the default value. To isolate the issue to the field with the overflow value, create a separate computeExpression transformation for each defined column.

👁 Example: Let's look at an example. You want to create a dataset based on Salesforce opportunity data. You create a dataflow that extracts the Id and Amount fields from the Opportunity object. In addition, you also want to add the following derived fields to the dataset: ModifiedId, SalesTax, FinalPrice, and ValueCategory. For the derived fields, you will:

- Append "SFDC" to each opportunity Id to get a new modified Id.

- Calculate the sales tax based on an 8% tax rate.

- Calculate the final price by adding the amount and sales tax.

- Categorize opportunities into low-, medium-, and high-value buckets based on the calculated final price.

You create the following dataflow definition.

```
{
   "salesData": {
      "action": "sfdcDigest",
      "parameters": {
         "object": "Opportunity",
         "fields": [
            {"name": "Amount"},
            {"name": "Id"}]}},
   "Derived_Fields": {
      "action": "computeExpression",
      "parameters": {
         "source": "salesData",
```

```
            "mergeWithSource": true,
            "computedFields": [
                {
                    "name": "ModifiedId",
                    "type": "Text",
                    "saqlExpression": "\"SFDC\" + Id"},
                {
                    "name": "SalesTax",
                    "type": "Numeric",
                    "precision": 18,
                    "defaultValue": "0",
                    "scale": 5,
                    "saqlExpression": "Amount * 0.08"},
                {
                    "name": "FinalPrice",
                    "type": "Numeric",
                    "precision": 18,
                    "defaultValue": "0",
                    "scale": 5,
                    "saqlExpression": "Amount + SalesTax"},
                {
                    "name": "ValueCategory",
                    "type": "Text",
                    "saqlExpression": "case when FinalPrice < 1000 then \"Low\" when
FinalPrice >= 1000 and FinalPrice < 2000 then \"Medium\" else \"High\" end"}
            ]
        }
    },
    "Register_CategorizedSales": {
        "action": "sfdcRegister",
        "parameters": {
            "alias": "Categorized_Sales",
            "name": "Categorized_Sales",
            "source": "Derived_Fields" }
    }
}
```

computeExpression Parameters

When you define a computeExpression transformation, you set the action attribute to `computeExpression`. You also specify the parameters for the input source and the expression used to generate the values.

computeExpression Parameters
When you define a computeExpression transformation, you set the action attribute to `computeExpression`. You also specify the parameters for the input source and the expression used to generate the values.

You can specify parameters in the following sections of the computeExpression node: parameters and computedFields.

Parameters

The following table describes the parameters in the `parameters` section.

Parameter	Required?	Value
source	Yes	Node in the dataflow definition file that identifies the input source for this transformation.
mergeWithSource	No	Indicates whether the input fields are included with the derived fields in the resulting dataset. When true, the resulting dataset contains all input fields from the source and the newly generated derived fields. When false, the resulting dataset contains the derived fields only. Default is true.
computedFields	Yes	Attributes and expression used to generate derived fields in the dataset. See computedFields.

computedFields

The following table describes the attributes in the `computedFields` section. It also describes optional attributes that you can provide to override the field metadata to make the data appear differently in a dataset. For example, Tableau CRM can replace null values in a field with a default value.

Attribute	Required?	Value
name	Yes	API name of the generated field. Note: The API names must be unique. Otherwise, the dataflow fails to run.
type	Yes	Analytics Cloud field type associated with the field. Valid types are Text, Numeric, or Date. Example: `"type": "Text"`
label	No	The display name of the generated field that appears in the Tableau CRM user interface. Can be up to 255 characters. Defaults to input field name if not specified.

saqlExpression	Yes	SAQL expression used to calculate the value for the derived field. The expression can be based on input fields or other derived fields in the transformation.
		
		 Example:
		
		 `"saqlExpression":"toDate(birth_day, \"yyyy-M-d\")"`
		
		 Note: If a field name in a SAQL expression contains characters other than letters, numbers, or underscores, enclose the name in single quotes. For example, field names resulting from an augment are prefixed with the relationship name and a dot, and must be. enclosed in single quotes.
		
		 `"saqlExpression":"'AccountId.Sales_Q4__c'-'AccountId.Sales_Q3__c'"`
		
		 Note: The use of multi-value fields as input fields isn't supported.

format	Yes (for Date fields only)	Format of the derived date field. For information about formats, see the Analytics Cloud External Data Format Reference.

Attribute	Required?	Value
fiscalMonthOffset	No	For date fields only. The difference, in months, between the first month of the fiscal year and the first month of the calendar year (January). For example, if the fiscal year starts in January, the offset is 0. If the fiscal year starts in October, the offset is 9. Example: `"fiscalMonthOffset": 9` When you set `fiscalMonthOffset` to a value other than 0, Tableau CRM generates fields to show the fiscal week, month, quarter, and year that a date value falls in. You can use these fields to group and filter by fiscal period in a dataset. For more information, see Handle Date Values. ⚠ Warning: Tableau CRM doesn't support fields with different `fiscalMonthOffset` values in the same dataset. Using different `fiscalMonthOffset` values can produce unexpected results when you filter by relative fiscal date ranges. We recommend that you set the same value for all `fiscalMonthOffset` attributes in a dataset.
isYearEndFiscalYear	No	For date fields only, when `fiscalMonthOffset` is greater than 0. Indicates whether the fiscal year is the year in which the fiscal year ends or begins. Because the fiscal year can start in one calendar year and end in another, you specify which year to use for the fiscal year. • If true, then the fiscal year is the year in which the fiscal year ends. The default is true. • If false, then the fiscal year is the year in which the fiscal year begins. Example: `"isYearEndFiscalYear": true` ⚠ Warning: Tableau CRM doesn't support fields with different `isYearEndFiscalYear` values in the same dataset. Using different `isYearEndFiscalYear` values can produce unexpected results when you filter by relative fiscal date ranges. We recommend that you set the same value for all `isYearEndFiscalYear` attributes in a dataset. For more information, see Handle Date Values.
firstDayOfWeek	No	For date fields only. The first day of the week for the calendar year and, if applicable, fiscal year. Use 0 to set the first day to be Sunday, 1 to set the first day to be Monday, and so on. Use -1 to set the first day to be on January 1. The default is -1. Example: `"firstDayOfWeek": 0` ⚠ Warning: Tableau CRM doesn't support fields with different `firstDayOfWeek` values in the same dataset. Using different `firstDayOfWeek` values can produce unexpected results when you

Attribute	Required?	Value
		filter by relative week date ranges. We recommend that you set the same value for all `firstDayOfWeek` attributes in a dataset.

For more information, see Handle Date Values. |
| precision | Yes (for Numeric fields only) | The maximum number of digits in a numeric value, or the length of a text value. For numeric values: Includes all numbers to the left and to the right of the decimal point (but excludes the decimal point character). Value must be from 1 to 16. For text values: Value defaults to 255 characters, and must be from 1 to 32,000 characters.

Example:

```
"precision": 10
``` |
| scale | Yes (for Numeric fields only) | The number of digits to the right of the decimal point in a numeric value. Must be less than the precision value. Value must be from 1 to 15 characters.

Example:

```
"scale": 2
``` |
| defaultValue | No | For text and numeric fields that can be null. Default value that replaces a null value for the specified field. Enter the default value as a string.

Example:

```
"defaultValue": "0"
``` |

SEE ALSO:

    computeExpression Transformation

computeRelative Transformation

You can use the computeRelative transformation to analyze trends in your data by adding derived fields to a dataset based on values in other rows. For example, to analyze sales pipeline trends, create derived fields that calculate the number of days an opportunity remains in each stage. You can also calculate the changes to the opportunity amount throughout the stages of the opportunity.

Note: To create a calculated column across rows based on a formula in a Data Prep recipe, see Formula Transformation: Create a Calculated Column Based on an Expression.

The computeExpression and computeRelative transformations are similar, but the computeExpression transformation performs calculations based on fields within the same row. The computeRelative transformation performs calculations based on the same field in other rows—particularly the current, first, previous, or next rows.

When you define a computeRelative transformation, you specify a source transformation as the input, partition the records, and sort the records within each partition. For example, you can use sfdcDigest to extract opportunity history records, and then use computeRelative to calculate changes in each opportunity over time. You can partition opportunity history records by opportunity ID, and then chronologically sort records within each partition to correctly identify the previous and next values.

Note: Derived fields can be based on a source field or on a SAQL expression.

👁 **Example:** Let's look at an example. To perform trending analysis on the sales pipeline, create a dataflow that contains the following transformations.

sfdcDigest transformation

Extracts the following data from the OpportunityHistory object.

Opportunity ID	Created Date	Stage Name	Amount	Close Date
006R0000001rerEIAQ	2017-09-11T18:47:39.000Z	Id. Decision Makers	120,000	2015-06-24
006R0000001rerGIAQ	2017-09-11T18:48:12.000Z	Qualification	20,000	2015-10-22
006R0000001rerFIAQ	2017-09-11T18:48:39.000Z	Value Proposition	75,000	2015-09-23
006R0000001rerDIAQ	2017-09-11T18:49:11.000Z	Perception Analysis	38,000	2015-07-15
006R0000001rerHIAQ	2017-09-11T18:49:36.000Z	Negotiation/Review	120,000	2015-07-31

computeRelative transformation

Performs the following tasks:

- Partitions the extracted records by opportunity ID.

- Within each partition, sorts the extracted records by CreatedDate in ascending order. Sorting by CreatedDate ensures that the changes that occur for each opportunity are listed in chronological order.

- Adds the following derived fields to the final dataset.

  OpportunityCreatedDate

  Determines the date that the opportunity was first created. You can use this date with the actual close date to determine the number of days required to close the sale. The goal is to shorten the sales cycle to recognize revenue.

  AmountPrev

  Determines the previous amount of the opportunity. You can use this field to determine if the values of opportunities are increasing or decreasing, which can affect whether you hit your sales targets.

  CloseDatePrev

  Determines the previous expected close date for the opportunity. You can use this field to analyze how the expected close date changes over the sales cycle of the opportunity. If the expected close date keeps getting pushed out, identify the issues that are causing the longer sales cycle.

  AmountChange

  Uses a SAQL expression to calculate the percentage change of the opportunity amount from its previous amount.

  AmountChangeDirection

  Uses a SAQL expression to generate a text value to show the direction in which an opportunity amount has changed: Up, Down, or No Change.

sfdcRegister transformation

Registers the final dataset that contains the extracted fields from the sfdcDigest transformation and the derived fields from computeRelative transformation.

You create the following dataflow definition.

```
{
  "extractOppHistory": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "OpportunityHistory",
            "fields": [
                {"name": "OpportunityId"},
```

```
                            {"name": "CreatedDate"},
                            {"name": "StageName"},
                            {"name": "Amount"},
                            {"name": "CloseDate"}
                    ]
            }
    },
    "computeTrending": {
        "action": "computeRelative",
        "parameters": {
            "source": "extractOppHistory",
            "partitionBy": ["OpportunityId"],
            "orderBy": [
                {
                    "name": "CreatedDate",
                    "direction": "asc"
                }
            ],
            "computedFields": [
                {
                    "name": "OpportunityCreatedDate",
                    "label": "Opportunity Created Date",
                  "description": "Determines the date that the opportunity was first created.",

                    "expression": {
                        "sourceField": "CreatedDate",
                        "offset": "first()",
                        "default": "current()"
                    }
                },
                {
                    "name": "AmountPrev",
                    "label": "Previous Amount",
                    "description": "Determines the previous amount of the opportunity",
                    "expression": {
                        "sourceField": "Amount",
                        "offset": "previous()",
                        "default": "null"
                    }
                },
                {
                    "name": "CloseDatePrev",
                    "label": "Previous Close Date",
                    "description": "Determines the previous expected close date for the
opportunity",
                    "expression": {
                        "sourceField": "CloseDate",
                        "offset": "previous()",
                        "default": "current()"
                    }
                },
                {
                    "name": "AmountChange",
                    "label": "Amount % Change",
```

```
          "description": "Determines percentage change from previous amount",
          "expression": {
           "saqlExpression": "(current(Amount)-previous(Amount))/previous(Amount)*100",

            "type": "Numeric",
            "scale": 2,
            "default": "null"
          }
        },
        {
          "name": "AmountChangeDirection",
          "label": "Amount Change Direction",
          "description": "Determines text to indicate direction of change",
          "expression": {
           "saqlExpression": "case when current(Amount)>previous(Amount) then \"Up\"
 when current(Amount)<previous(Amount) then \"Down\" else \"No Change\" end",
            "type": "Text",
            "default": ""
          }
        }
      ]
    }
  },
  "Register_OppportunityHistory_Dataset": {
    "action": "sfdcRegister",
    "parameters": {
      "alias": "SalesPipelineTrending1",
      "name": "Sales Pipeline Trending1",
      "source": "computeTrending"
    }
  }
}
```

The dataflow runs and creates the dataset with the new derived fields.

Opportunity ID	Created Date	Amount	OpportunityCreatedDate	Previous Amount	Amount % Change	Amount Change Direction	Previous Close Date
006R0000001rerDIAQ	2017-09-11T18:49:11.000Z	38,000	2017-08-28T18:24:41.000Z	40,000	-5.0	Down	2013-05-06
006R0000001rerEIAQ	2017-08-28T18:24:41.000Z	140,000	2017-08-28T18:24:41.000Z	-	-	No Change	1970-1-1
006R0000001rerEIAQ	2017-08-28T18:24:41.000Z	140,000	2017-08-28T18:24:41.000Z	140,000	0	No Change	2013-03-07
006R0000001rerEIAQ	2017-08-28T18:24:41.000Z	140,000	2017-08-28T18:24:41.000Z	140,000	0	No Change	2013-04-05
006R0000001rerEIAQ	2017-09-11T18:47:39.000Z	120,000	2017-08-28T18:24:41.000Z	140,000	-14.0	Down	2013-04-05

Notice that Tableau CRM partitions the records by opportunity ID and then sorts the records in ascending order based on the CreatedDate field within each partition. Tableau CRM can now use the previous and next rows within each partition to determine changes in field values in the dataset.

computeRelative Parameters

When you define a computeRelative transformation, you set the action attribute to `computeRelative`. You also specify the parameters for the input source, partition-by field, sort field, and derived field definitions.

computeRelative Parameters

When you define a computeRelative transformation, you set the action attribute to `computeRelative`. You also specify the parameters for the input source, partition-by field, sort field, and derived field definitions.

You can specify parameters in the following sections of the computeRelative node.

Parameters

The following table describes the parameters in the `parameters` section.

Parameter	Required?	Value
source	Yes	Node in the dataflow definition file that identifies the input source for this transformation.
partitionBy	Yes	API name of the field used to partition the records in the dataset. Specify one partition-by field only.
orderBy	Yes	Field used to sort the records within each partition and the sort order: ascending (`asc`) or descending (`desc`). Specify one sort field only. Example: ```\n"orderBy": [\n {\n "name":"CreatedDate",\n "direction":"asc"\n }\n]\n```
computedFields	Yes	A list of definitions for derived fields. Derived fields can be based on a source field or on a SAQL expression. Example showing derived fields based on a source field and on a SAQL expression: ```\n"computedFields": [\n {\n "name": "PreviousAmount",\n "label": "Previous Amount",\n "description": "Previous amount of\nopportunity",\n "expression": {\n "sourceField": "Amount",\n "offset": "previous()",\n "default": "null"\n }\n },\n {\n "name": "AmountChange",\n "label": "Amount % Change",\n "description": "Percentage change from\nprevious amount",\n "expression": {\n "saqlExpression":\n"(current(Amount)-previous(Amount))/previous(Amount)*100",\n\n "type": "Numeric",\n "scale": 2,\n```

Parameter	Required?	Value
| | | ```
        "default": "null"
      }
    }
  ]
```  See computedFields. |

computedFields

The following table describes the attributes in the `computedFields` section.

Parameter	Required?	Value
name	Yes	API name of the derived field to add to the dataset. The name must be unique in the dataset. 📝 Note: If the name is not unique, the dataflow fails to run.
label	No	The display name of the derived field that appears in the Tableau CRM user interface. Can be up to 255 characters. Defaults to the API name if not specified.
description	No	Description of the derived field for information only.
expression	Yes	Expression attributes used to calculate the value for the derived field. The expression can be based on input fields or other derived fields in the transformation. Example: ```
"expression": {
"sourceField": "CloseDate",
"offset": "previous()",
"default": "01-01-1970"
`` See expression.

expression

The following table describes the attributes in the `expression` section when creating a derived field based on a source field.

Parameter	Required?	Value
sourceField	Yes	API name of the input field from the source node that's used in the expression.
offset	Yes	The function used in the expression. You can use the following functions: **current()** Gets the value from the current record. Example: ```
"offset": "current()"
``

Parameter	Required?	Value
		first()
		Gets the value from the first record in the partition, like the first CreateDate for an opportunity.
		Example:
		```"offset": "first()"```
		next()
		Gets the value from the next record.
		Example:
		```"offset": "next()"```
		previous()
		Gets the value from the previous record.
		Example:
		```"offset": "previous()"```
		Note: Derived fields are computed in the order that they're defined. The calculation of a derived field can be based on the value from another derived field as long as it has already been defined. For example, `next()` can't access the value of a derived field in the next row.
		Tip: To get the correct results when using the `previous()` and `next()` functions, the computeRelative transformation requires you to sort the records.
default	Yes (for numeric fields only)	The default value if one can't be calculated. For example, you can specify a default value when no previous or next value exists. You can insert a constant value or `current()` as a default value.
		Examples:
		```"default": "3000-01-01T00:00:00.000Z"```
		```"default": "current()"```

The following table describes the attributes in the `expression` section when creating a derived field based on a SAQL expression.

Parameter	Required?	Value
saqlExpression	Yes	SAQL expression used to calculate the value for the derived field. The expression can be based on input fields or other derived fields in the transformation. You can use the offset functions `current()`, `first()`, `next()`, and `previous()` in the expression.

Parameter	Required?	Value
		Example to calculate the percentage change from the previous amount to the current amount:
		```"saqlExpression": "(current(Amount)-previous(Amount))/previous(Amount)*100"```
		You can also use the SAQL case operator in the expression.
		Example to output *Up*, *Down*, or *No Change* values based on the change from the previous amount to the current amount:
		```"saqlExpression": "case when current(Amount)>previous(Amount) then \"Up\" when current(Amount)<previous(Amount) then \"Down\" else \"No Change\" end"```
type	Yes	Tableau CRM field type associated with the field. Valid types are Numeric and Text. Example: ```"type": "Text"```
scale	No	The number of digits to the right of the decimal point in a numeric value. Example: ```"scale": 2```
default	Yes (for numeric fields only)	The default value if one can't be calculated. Example: ```"default": "null"```

delta Transformation

The delta transformation calculates changes in the value of a measure column in a dataset over a period of time. The delta transformation generates an output column in the dataset to store the delta for each record. Create deltas to make it easier for business analysts to include them in queries.

> **Note:** The `delta` transformation isn't supported when null measure handling is enabled and dataflows containing delta transformations fail. Use `computeRelative` and `computeExpression` transformations instead in your dataflows, to calculate changes in measure values over time. For an example, see Enable Null Measure Handling on page 564.

The delta transformation calculates each delta value by comparing the value in each record with the value in the previous record. Because records might not be sorted, the delta transformation orders the records before computing the delta values. To do this, the transformation sorts the data by the specified dimension, and then by the specified epoch date column.

> ✏️ **Note:** When Analytics Cloud processes dates, it creates the following epoch date columns for each date processed:

Epoch Time Column	Description
<date_column_name>_sec_epoch	For example, if the date column is CloseDate, the generated epoch second column is CloseDate_sec_epoch. This column provides the number of seconds that have elapsed since January 1, 1970 (midnight UTC/GMT).
<date_column_name>_day_epoch	For example, if the date column is CloseDate, the generated epoch day column is CloseDate_day_epoch. This column provides the number of days that have elapsed since January 1, 1970 (midnight UTC/GMT).

> 👁️ **Example:** Let's look at an example. You want to create an OppHistoryDelta dataset that contains opportunity history from the OpportunityHistory object and also calculates the deltas for opportunity amounts.
>
> The OpportunityHistory object contains the following data.

OpportunityId	CloseDate	StageName	Amount
1	1/1/2014	New	100
2	1/1/2014	New	100
2	2/1/2014	ClosedWon	200
1	3/1/2014	ClosedWon	100

You create the following dataflow definition.

```
{
   "Extract_Opportunities": {
      "action": "sfdcDigest",
      "parameters": {
         "object": "OpportunityHistory",
         "fields": [
            { "name": "OpportunityId" },
            { "name": "CloseDate" },
            { "name": "StageName" },
            { "name": "Amount" }
         ]
      }
   },
   "Calculate_Delta": {
      "action": "delta",
      "parameters": {
         "dimension": "OpportunityId",
         "epoch": "CloseDate_day_epoch",
         "inputMeasure": "Amount",
         "outputMeasure": "DeltaAmount",
         "source": "Extract_Opportunities"
```

```
        }
    },
    "Register_Dataset": {
        "action": "sfdcRegister",
        "parameters": {
            "alias": "OppHistoryDelta",
            "name": "OppHistoryDelta",
            "source": "Calculate_Delta"
        }
    }
}
```

To calculate the delta values for each opportunity amount, the delta transformation sorts the records by the dimension (OpportunityId) first, and then by time (CloseDate_day_epoch) as shown here.

OpportunityID	CloseDate	StageName	Amount
1	1/1/2014	New	100
1	3/1/2014	ClosedWon	100
2	1/1/2014	New	100
2	2/1/2014	ClosedWon	200

After the records are sorted, for each dimension (OpportunityId), the transformation compares the previous value to the next value to determine the delta for each record. The transformation creates the following dataset.

OpportunityId	CloseDate	StageName	Amount	DeltaAmount
1	1/1/2014	New	100	0
1	3/1/2014	ClosedWon	100	0
2	1/1/2014	New	100	0
2	2/1/2014	ClosedWon	200	100

For the first record of each dimension, the transformation inserts '0' for the delta value.

📝 Note: If an opportunity contains multiple changes on the same day, you must sort the records on a shorter time interval. In this case, sort on CloseDate_sec_epoch column. Otherwise, records might not be sorted correctly, which means delta values will be incorrect.

delta Parameters
When you define a delta transformation, you set the action attribute to `delta` and specify the parameters.

delta Parameters
When you define a delta transformation, you set the action attribute to `delta` and specify the parameters.

The following table describes the input parameters:

Parameter	Required?	Value
dimension	Yes	Dimension column in the dataset used to sort records when calculating the delta values.
epoch	Yes	Epoch date column in the dataset used to sort records within each dimension when calculating delta values.
inputMeasure	Yes	Measure column on which you want to calculate the delta.
outputMeasure	Yes	Name of the output column that contains the delta value.
source	Yes	Node in the dataflow definition file that contains the dataset to which you want to add the delta column.

SEE ALSO:

> delta Transformation

digest Transformation

The digest transformation extracts synced connected data in a dataflow. Use it to extract data synced from an external Salesforce org, or data synced through an external connection. Use the sfdcDigest transformation to extract from your local Salesforce org.

👁 **Example:** Let's say you extract opportunity data from your company's org in Arizona using a Salesforce external connection. To use this data in your dataflow, add a digest node specifying the connection name, object, and fields.

```
"Extract Arizona Opportunities": {
    "action": "digest",
    "parameters": {
        "connectionName": "Arizona_Sales",
        "object": "Opportunity",
        "fields": [
        {"name": "CloseDate"},
        {"name": "Amount"},
        {"name": "StageName"},
        {"name": "Name"}
      ],
    }
  },
```

> digest Parameters
>
> When you define a digest transformation, you set the action attribute to `digest` and specify the parameters.

digest Parameters

When you define a digest transformation, you set the action attribute to `digest` and specify the parameters.

The following table describes the input parameters:

Parameter	Required?	Value
connectionName	Yes	Name of the connection used to sync the data.
object	Yes	The connected object that you want to extract data from.
fields	Yes	An array of names of the fields that you want to extract.

dim2mea Transformation

The dim2mea transformation creates a new measure based on a dimension. The transformation adds the new measure column to the dataset. The transformation also preserves the dimension to ensure that existing lenses and dashboards don't break if they use the dimension.

Note: To convert a dimension to a measure in a Data Prep recipe, see Dimension to Measure Transformation: Convert the Column Type. To do it in a Data Prep Classic recipe, see Convert a Dimension Field Type to Measure.

During the column type conversion, the Dimension to Measure transformation rounds decimals to the nearest whole number. For example, 300.2939 becomes 300.

If the transformation cannot create a measure from a dimension, the transformation populates the measure with the specified default value. If no default value is provided, the transformation inserts '0.'

Example: Let's look at an example. Your Opportunity object contains a custom text field called StageVal__c, which contains the opportunity amount at a particular stage. Because this is a text field, Analytics Cloud loads this data as a dimension. However, you'd like to create a measure from this dimension to enable users to perform calculations on stage amount.

You create the following dataflow definition.

```
{
    "Extract_Opportunities": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Opportunity",
            "fields": [
                { "name": "Id" },
                { "name": "Name" },
                { "name": "Amount" },
                { "name": "StageName" },
                { "name": "CloseDate" },
                { "name": "AccountId" },
                { "name": "StageVal__c" }
            ]
        }
    },
    "Create_Measure_From_Dimension": {
        "action": "dim2mea",
        "parameters": {
            "dimension": "StageVal__c",
            "measure": "StageValue",
            "measureDefault": "0",
            "measureType": "long",
            "source": "Extract_Opportunities"
```

```
            }
        },
        "Register_The_Dataset": {
            "action": "sfdcRegister",
            "parameters": {
                "alias": "OpportunitiesWithConvertedMeasure",
                "name": "OpportunitiesWithConvertedMeasure",
                "source": "Create_Measure_From_Dimension"
            }
        }
    }
}
```

[dim2mea Parameters](#)

When you define a dim2mea transformation, you set the action attribute to `dim2mea` and specify the parameters.

dim2mea Parameters

When you define a dim2mea transformation, you set the action attribute to `dim2mea` and specify the parameters.

The following table describes the input parameters:

Parameter	Required?	Value
dimension	Yes	Dimension column in the dataset from which you want to create the measure.
measure	Yes	Name of the output measure. This column name must be unique within the dataset. Do not use the same name as the dimension because the transformation preserves the dimension in the dataset. 📝 Note: The measure name is also the new field's API name, so it: • Can contain only alphanumeric and underscore characters. • Must begin with a letter. • Can't end with an underscore. • Can't contain 2 consecutive underscore characters, except when ending with "__c" (case-sensitive).
measureDefault	Yes	Default value for the measure if the transformation is unable to create a measure from a dimension.
measureType	Yes	Type of measure. Valid value: "long"

Parameter	Required?	Value
source	Yes	Node in the dataflow definition file that contains the dataset to which you want to add the measure.

SEE ALSO:

  [dim2mea Transformation](#)

edgemart Transformation

The edgemart transformation gives the dataflow access to an existing, registered dataset, which can contain Salesforce data, external data, or a combination of the two. Use this transformation to reference a dataset so that its data can be used in subsequent transformations in the dataflow. You can use this transformation and the augment transformation together to join an existing dataset with a new dataset.

📝 Note: To pull data from an existing dataset in a Data Prep recipe, see Create a Recipe with Data Prep. To do it in a Data Prep Classic recipe, see Create a Recipe with Data Prep Classic.

👁 Example: Let's look at an example. You would like to compare the final sales amount against the opportunity amount to determine if heavy discounts were offered to close deals. You previously created and registered the FinalSales dataset. The FinalSales dataset contains the final sale amount of each opportunity that was closed and won.

Table 5: FinalSales Dataset

OppID	UpdateDate	StageName	SaleAmount
1	1/1/2014	ClosedWon	100,000
2	11/1/2013	ClosedWon	150,000
3	2/1/2014	ClosedWon	200,000

You would now like to create a dataset that contains opportunity information from the Opportunity object. Then, you would like to join the data from the existing FinalSales dataset with the Opportunity dataset.

You create the following dataflow definition.

```
{
    "Extract_Opportunities": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Opportunity",
            "fields": [
                { "name": "Id" },
                { "name": "Name" },
                { "name": "Amount" }
            ]
        }
    },
    "Extract_Final_Sales_Data": {
        "action": "edgemart",
        "parameters": { "alias": "FinalSales" }
    },
    "Combine_Opportunities_FinalSales": {
```

```
            "action": "augment",
            "parameters": {
                "left": "Extract_Opportunities",
                "left_key": [ "Id" ],
                "relationship": "Opportunity",
                "right": "Extract_Final_Sales_Data",
                "right_key": [ "OppID" ],
                "right_select": [ "SaleAmount" ]
            }
        },
        "Register_Opportunity_FinalSales_Dataset": {
            "action": "sfdcRegister",
            "parameters": {
                "alias": "OpportunityVersusFinalSales",
                "name": "OpporunityVersusFinalSales",
                "source": "Combine_Opportunities_FinalSales"
            }
        }
    }
}
```

edgemart Parameters

When you define an edgemart transformation, you set the action attribute to `edgemart` and specify the parameters.

edgemart Parameters
When you define an edgemart transformation, you set the action attribute to `edgemart` and specify the parameters.

The following table describes the input parameter:

Parameter	Required?	Value
alias	Yes	API name of the dataset from which you want to extract data. To determine the API name of a dataset, edit the dataset and view the system name.

SEE ALSO:

edgemart Transformation

export Transformation
The export transformation creates a data file and a schema file from data in a specified source node in your dataflow. After the dataflow runs, Einstein Discovery users can access these files through the public API.

📝 Note: The export transformation is available in Salesforce orgs that have Einstein Discovery enabled.

When you add an export transformation to a dataflow, you specify a source and a target parameter. The source is the node in the dataflow that outputs the data you want to export, and the target is Einstein Discovery. You also specify a user, which determines who can access the data in Einstein Discovery. If the source is a dataset configured with a security predicate, Tableau CRM applies the predicate based on the specified user. For example, if the security predicate allows the user to access only dataset records that they own, then only those dataset records are exported to Einstein Discovery.

👁 **Example:** Here's an example. Let's say your dataflow contains an sfdcDigest node, `ExtractOpportunities`, which extracts all fields from the Opportunity object. To make this data available in Einstein Discovery, add an export node to the dataflow, specifying the source node and a target of `EinsteinDiscovery`.

export Node in Dataflow Editor

export Node in Dataflow JSON

```
"ExportOpportunities": {
    "action": "export",
    "parameters": {
      "source": "ExtractOpportunities",
      "target": "EinsteinDiscovery",
      "userId": "005R0000000IUNDIA4"
    }
  }
```

📝 **Note:** When you add an export node in the dataflow editor, you can search for the user by name in the User field.

📝 **Note:** Multi-value dimensions (for example, those used in multi-select picklists) are not included in the export file.

[export Parameters](#)
Add an export transformation to a dataflow either in the dataflow editor, or directly in the JSON dataflow definition file. The parameters you specify are the same for both methods.

export Parameters

Add an export transformation to a dataflow either in the dataflow editor, or directly in the JSON dataflow definition file. The parameters you specify are the same for both methods.

📝 **Note:** The export transformation is available in Salesforce orgs that have Einstein Discovery enabled.

To add an export transformation in the dataflow editor, click .

To add an export transformation in the JSON definition file, add a node and set the action attribute to `export`.

The following table describes the input parameters:

Parameter	Required?	Value
In dataflow editor: **Source Node**	Yes	Node in the dataflow that identifies the dataset that you want to export. This is the input source for this transformation.

Parameter	Required?	Value
In dataflow JSON: `source`		
In dataflow editor: **Target**	Yes	The destination for the exported data.
In dataflow JSON: `target`		In the dataflow editor, select *Einstein Discovery*.
		In the dataflow definition file, specify `EinsteinDiscovery`.
In dataflow editor: **User**	Yes	The Id of the user who can access the data in Einstein Discovery.
In dataflow JSON: `userId`		In the dataflow editor, you can search for the user by name in this field.

filter Transformation

The filter transformation removes records from an existing dataset. You define a filter condition that specifies which records to retain in the dataset.

📝 Note: To filter rows in a Data Prep recipe, see Filter Node: Filter Rows. To do it in a Data Prep Classic recipe, see Add a Filter in a Recipe.

👁 Example: Let's look at an example. You want to create a dataset that contains only opportunities that were Closed Won. First, you extract all opportunities from the Opportunity object. Next, you filter the records so that you only include opportunities with a Closed Won stage name.

You create the following dataflow.

Dataflow   **Filter Node in Dataflow**

Here's the dataflow JSON.

```
{
    "Extract_Opportunities": {
        "action": "sfdcDigest",
        "parameters": {
```

```
            "object": "Opportunity",
            "fields": [
                { "name": "Id" },
                { "name": "Name" },
                { "name": "Amount" },
                { "name": "StageName" },
                { "name": "CloseDate" },
                { "name": "AccountId" },
                { "name": "OwnerId" }
            ]
        }
    },
    "Filter_Opportunities": {
        "action": "filter",
        "parameters": {
            "filter": "StageName:EQ:Closed Won",
            "source": "Extract_Opportunities"
        }
    },
    "Register_My_Won_Oppportunities_Dataset": {
        "action": "sfdcRegister",
        "parameters": {
            "alias": "MyWonOpportunities",
            "name": "MyWonOpportunities",
            "source": "Filter_Opportunities"
        }
    }
}
```

For more complex filters, you can use SAQL in a filter node.

💡 **Tip:** Depending on your need, you can filter Salesforce object records at different levels.

Transformation level

To filter a stream of data loaded into a dataset, add a filter transformation to the dataflow. Use a filter transformation instead of an sfdcDigest transformation filter because complex filters in an sfdcDigest transformation impact global filters. To reduce the amount of data processing downstream, add the filter transformation as early as possible in the dataflow.

Connection level

To prevent Salesforce records from being loaded into any dataset, add the filter on the connected object. You can add a filter in the sync settings for the connected Salesforce object.

filter Parameters

When you define a filter transformation, you set the action attribute to `filter` and specify the parameters.

filter Expression Syntax

You create a filter expression in the filter transformation based on one or more dimensions in a dataset. You can use a standard filter expressions or a SAQL filter expression.

filter Parameters

When you define a filter transformation, you set the action attribute to `filter` and specify the parameters.

This table describes the settings for the filter node in the dataflow editor.

917

Setting	Required?	Value
Source	Yes	Node in the dataflow definition file that contains the dataset that you want to filter.
Use SAQL	No	Select whether to use a SAQL expression in the filter.
Filter	No	Standard filter expression that specifies which records to include in the new dataset. See filter Expression Syntax. 📝 Note: This field is visible when the Use SAQL option is not selected.
SAQL Filter	No	SAQL filter expression that specifies which records to include in the new dataset. See filter Expression Syntax. 📝 Note: This field is visible when the Use SAQL option is selected.

This table describes the input parameters for the filter transformation in the dataflow JSON.

Parameter	Required?	Value
filter	No	Filter expression that specifies which records to include in the new dataset. See filter Expression Syntax.
saqlFilter	No	SAQL filter expression that specifies which records to include in the new dataset. See filter Expression Syntax.
source	Yes	Node in the dataflow definition file that contains the dataset that you want to filter.

📝 Note: You can include either the `filter` parameter or the `saqlFilter` parameter in a filter node in the dataflow JSON, but not both.

SEE ALSO:

> filter Transformation

filter Expression Syntax

You create a filter expression in the filter transformation based on one or more dimensions in a dataset. You can use a standard filter expressions or a SAQL filter expression.

📝 Note:

- String comparisons in a filter expression are case-sensitive.

- When you filter on a Salesforce ID field extracted in an upstream sfdcDigest node, use 18-character ID values in your filter
  expression. sfdcDigest nodes extract 18-character IDs, and the filter transformation performs string comparisons on ID fields.

  Example: `RecordTypeId:EQ:0126A0000016QfyQAE`

To use a standard filter expression, deselect the **Use SAQL** option in the dataflow editor, or use the `filter` parameter in the dataflow
JSON. You can use the following types of filter expressions:

Filter Expression Syntax	Description
dim:EQ:value	True if the dimension and value are equal.
	Dataflow editor example: `StageName:EQ:Closed Won`
	Dataflow JSON example: `"filter": "StageName:EQ:Closed Won"`
dim:R:val0:val1	True if the left dimension falls within the specified range between val0 and val1.
	Dataflow editor example: `EmployeeId:R:100:1000`
	Dataflow JSON example: `"filter": "EmployeeId:R:100:1000"`
dim:R:val	True if the dimension is greater than or equal to the value based on binary sort order. For example, this is true when the dimension is 'City' and the value is 'Anderson' because ' 'City' > 'Anderson').
	Dataflow editor example: `LastName:R:Li`
	Dataflow JSON example: `"filter": "LastName:R:Li"`
dim:R::val	True if the dimension is less than or equal to the value based on binary sort order.
	Dataflow editor example: `LastName:R::Levy`
	Dataflow JSON example: `"filter": "LastName:R::Levy"`
dim:N:val	True if the dimension and value are not equal.
	Dataflow editor example: `RoleName:N:Manager`
	Dataflow JSON example: `"filter": "RoleName:N:Manager"`
dim:EQ:val1|val2	True if the dimension equals values val1 or val2. This filter expression uses the logical OR operator (|). You can compare the dimension value against more than 2 values. For example, to compare against 3 values, use the following syntax: dim1:EQ:val1|val2|val3.
	Dataflow editor example: `Lead Status:EQ:Open|Contacted`

Filter Expression Syntax	Description
	Dataflow JSON example: `"filter": "Lead Status:EQ:Open|Contacted"`
dim1:EQ:val1,dim2:EQ:val2	True if dimension dim1 equals value val1 and dimension dim2 equals value val2. This filter expression uses the logical AND operator (,). You can compare more than 2 dimensions. For example, to compare 3 dimensions, use the following syntax: dim1:EQ:val1,dim2:EQ:val2,dim3:EQ:val3. Dataflow editor example: `Lead Status:EQ:Qualified,Rating:EQ:Hot` Dataflow JSON example: `"filter": "Lead Status:EQ:Qualified,Rating:EQ:Hot"`

To use a SAQL filter expression, select the **Use SAQL** option in the dataflow editor, or use the `saqlFilter` parameter in the dataflow JSON. Use the SAQL filter statement in your expression without the `a = filter a by` part. See

👁 Example: This example returns rows where the Stage Name field contains `Proposal/Price Quote` or `Negotiation/Review`.

Dataflow editor: `StageName in ["Proposal/Price Quote", "Negotiation/Review"]`

Dataflow JSON: `"saqlFilter": "StageName in [\"Proposal/Price Quote\", \"Negotiation/Review\"]"`

📝 Note: In the JSON, escape double quotes in the filter with `\`.

For complete information about SAQL operators, see *Analytics Cloud SAQL Reference*.

SEE ALSO:

[filter Transformation](#)

flatten Transformation

The flatten transformation flattens hierarchical data. For example, you can flatten the Salesforce role hierarchy to implement row-level security on a dataset based on the role hierarchy.

When you configure the flatten transformation to flatten a hierarchy, you specify the field that contains every node in the hierarchy and the field that contains their corresponding parent based on the hierarchy. The flatten transformation generates one record for each hierarchy node, which we refer to as the "self ID." Each record contains two generated columns that store the hierarchy for each self ID node. One column contains a comma-separated list of all ancestors for each node in the hierarchy. The other column contains the hierarchy path.

See the Roles and RolePath columns in the following flattened dataset to see how ancestors are stored.

Role ID (Self ID)	Role Name	Parent Role ID	Roles	RolePath
1	Salesperson 1	10	10, 20, 30	\10\20\30
2	Salesperson 2	10	10, 20, 30	\10\20\30
3	Salesperson 3	11	11, 20, 30	\11\20\30

Role ID (Self ID)	Role Name	Parent Role ID	Roles	RolePath
10	Regional Manager 1	20	20, 30	\20\30
11	Regional Manager 2	20	20, 30	\20\30
20	Vice President 1	30	30	\30
21	Vice President 2	30	30	\30
30	CEO	Not applicable	Not applicable	Not applicable

You can also configure the flatten transformation to include the self ID node in the generated hierarchy columns. The following dataset shows the self ID in bold.

Role ID (Self ID)	Role Name	Parent Role ID	Roles	RolePath
1	Salesperson 1	10	**1**, 10, 20, 30	**1**\10\20\30
2	Salesperson 2	10	**2**,10, 20, 30	**2**\10\20\30
3	Salesperson 3	11	**3**,11, 20, 30	**3**\11\20\30
10	Regional Manager 1	20	**10**,20, 30	**10**\20\30
11	Regional Manager 2	20	**11**,20, 30	**11**\20\30
20	Vice President 1	30	**20**,30	**20**\30
21	Vice President 2	30	**21**,30	**21**\30
30	CEO	Not applicable	**30**	**30**

Example: Let's look at an example. You want to create a dataset that contains all opportunities. For each opportunity, you want to include user and role information about the opportunity owner. Also, to implement row-level security based on the role hierarchy, each opportunity record must also contain a list of all ancestors above each opportunity owner based on the role hierarchy. To generate the list of ancestors, use the flatten transformation to flatten the role hierarchy.

You create the following dataflow definition file:

```
{
"Extract_Opportunity": {
    "action": "sfdcDigest",
    "parameters": {
        "object": "Opportunity",
        "fields": [
            { "name": "Id" },
            { "name": "Name" },
            { "name": "Amount" },
            { "name": "StageName" },
            { "name": "AccountId" },
            { "name": "OwnerId" }
        ]
```

```
      }
   },
   "Extract_User": {
      "action": "sfdcDigest",
      "parameters": {
         "object": "User",
         "fields": [
            { "name": "Id" },
            { "name": "Name" },
            { "name": "Department" },
            { "name": "UserRoleId" }
         ]
      }
   },
   "Extract_UserRole": {
      "action": "sfdcDigest",
      "parameters": {
         "object": "UserRole",
         "fields": [
            { "name": "Id" },
            { "name": "Name" },
            { "name": "ParentRoleId" }
         ]
      }
   },
   "Flatten_UserRole": {
      "action": "flatten",
      "parameters": {
         "source": "Extract_UserRole",
         "self_field": "Id",
         "parent_field": "ParentRoleId",
         "multi_field": "Roles",
         "path_field": "RolePath",
         "include_self_id":false
      }
   },
   "Augment_User_FlattenUserRole": {
      "action": "augment",
      "parameters": {
         "left": "Extract_User",
         "left_key": [ "UserRoleId" ],
         "relationship": "Role",
         "right": "Flatten_UserRole",
         "right_key": [ "Id" ],
         "right_select": [
            "Id",
            "Name",
            "Roles",
            "RolePath"
         ]
      }
   },
   "Augment_Opportunity_UserWithRoles": {
      "action": "augment",
```

```
   "parameters": {
      "left": "Extract_Opportunity",
      "left_key": [ "OwnerId" ],
      "right": "Augment_User_FlattenUserRole",
      "relationship": "Owner",
      "right_select": [
         "Name",
         "Department",
         "Role.Id",
         "Role.Name",
         "Role.Roles",
         "Role.RolePath"
      ],
      "right_key": [ "Id" ]
   }
},
"Register_OpportunityWithRoles_Dataset": {
   "action": "sfdcRegister",
   "parameters": {
      "alias": "OppRoles",
      "name": "OppRoles",
      "source": "Augment_Opportunity_UserWithRoles",
      "rowLevelSecurityFilter": "'Owner.Role.Roles' == \"$User.UserRoleId\" || 'OwnerId'
== \"$User.Id\""
   }
}
}
}
```

To flatten the Salesforce role hierarchy, the flatten transformation uses the following input fields from the UserRole object.

Id

> Id identifies each node in the Salesforce role hierarchy.

ParentRoleId

> ParentRoleId identifies the parent as defined in the role hierarchy.

After traversing through each parent-child relationship in the UserRole object, the flatten transformation generates one record for each role ID. Each record contains all ancestor roles for each role in the hierarchy. The flatten transformation generates two output columns—Roles and RolePath—to store all ancestor roles for each role.

[flatten Parameters](#)

> When you define a flatten transformation, you set the action attribute to `flatten` and specify the parameters.

flatten Parameters

When you define a flatten transformation, you set the action attribute to `flatten` and specify the parameters.

The following table describes the input parameters:

Parameter	Required?	Value
include_self_id	No	Indicates whether to include the self ID node in the generated multi_field and path_field columns. Valid values are `false` (default) and `true`.
self_field	Yes	Name of the input field that identifies each node in the hierarchy.

Parameter	Required?	Value
parent_field	Yes	Name of the input field that identifies the direct parent of each node in the hierarchy. For example, the Regional Manager 1 role is the parent of the Salesperson 1 role in a role hierarchy.
multi_field	Yes	Name of the multi-value output field that contains a list of all ancestors in the hierarchy, in order from the lowest to the highest level. The flatten transformation creates this field and generates the list of ancestors for each node in the hierarchy. For example, for Salesperson 1 role, the hierarchy of ancestors is: `Sales Manager 1, Regional Manager 1, Vice President 1, CEO.`
path_field	Yes	A string representation of the multi-field field, separated by backslashes. This output field contains the hierarchical path of all ancestors in the hierarchy, in order from the lowest to the highest level. The flatten transformation creates this field and generates the ancestry path for each node in the hierarchy. For example, for a salesperson role in a role hierarchy, the value is: `Sales Manager 1\Regional Manager 1\Vice President 1\CEO.`
source	Yes	Node in the dataflow definition file that contains the hierarchical data that you want to flatten. This node is the input source for this transformation and it must contain the input fields mapped to self_field and parent_field.

📝 **Note:** By default, the multi_field and path_field fields are created as system fields, which aren't visible in the user interface. To make the fields appear in the user interface, add a schema section to the flatten transformation and set the `IsSystemField` metadata attribute to *false* for each field in the transformation. The schema section is shown in bold in this sample JSON.

```
"Flatten_UserRole": {
    "schema": {
      "objects": [
        {
          "label": "UserWithRoles",
          "fields": [
            {
              "name": "Roles",
              "label": "Roles",
              "isSystemField": false
            },
            {
              "name": "RolePath",
              "label": "RolePath",
              "isSystemField": false
            }
          ]
        }
```

```
      ]
    },
    "action": "flatten",
    "parameters": {
      "source": "Extract_UserRole",
      "self_field": "Id",
      "parent_field": "ParentRoleId",
      "multi_field": "Roles",
      "path_field": "RolePath",
      "include_self_id": false
    }
  },
```

For more information about overriding metadata using a schema section in a transformation, see Overriding Metadata Generated by a Transformation.

SEE ALSO:

flatten Transformation

prediction Transformation

The prediction transformation makes an Einstein Discovery prediction for a dataset. Einstein Discovery uses *predictive analytics*, which analyzes historical data (based on data mining, machine learning, and predictive statistical modeling) to identify patterns and predict future outcomes.

After you create an Einstein Discovery story, deploying its model (see Deploy a Model) produces a prediction definition. When you create the transformation, you identify the name of the prediction definition to use. A prediction definition contains a model to use for predicting an outcome. Einstein Discovery creates a special column in your dataset to contain the prediction score (the predicted value of the outcome, such as the likelihood that a customer will churn). You also specify the name and display name of the new column.

Consider the following guidelines when creating a prediction transformation:

- To see the Prediction button in the toolbar, Einstein Discovery must be enabled in your org, and your user account must have the View Einstein Discovery Recommendations permission. For instructions, see Set Up Einstein Discovery)

- Before you can add a prediction transformation, you must deploy the model. For instructions, see Deploy a Model).

- Dataset column names must match the dataset column names that were used to create the story. Do not rename column names in the dataflow, as this will result in missing column warnings when the dataflow completes. Other columns are ignored, such as fields created using computeExpression. The field mappings to a Salesforce object, specified when deploying the model (see Deploy a Model)., do not apply to predictions in dataflows.

- Use a prediction definition with one model. A prediction definition with multiple models (those deployed with filters) isn't currently supported in dataflows.

prediction Transformation Properties in the Dataflow Editor

This table describes the settings for the prediction node in the dataflow editor.

Setting	Required?	Value
Node Name	Yes	Unique name that identifies this node in the dataflow JSON. Use this name to reference this node in other nodes.

Setting	Required?	Value
Source Node	Yes	Dataflow node used as input for this transformation. The prediction transformation generates the prediction score based on the data provided by the input node.
Prediction Definition	Yes	Prediction definition to use. Click the field to select from a list of deployed prediction definitions.
Prediction Output Field Name	Yes	Name of the new dataset column to contain the score.
Prediction Output Field Labels	Yes	Label of the new dataset column to contain the score.

prediction Transformation Parameters in the Dataflow JSON

When you define a prediction transformation in the dataflow JSON, you specify the node name, action, source dataflow node, and input parameters for this transformation.

This table describes the JSON input parameters for the prediction transformation (**"action": "prediction"**) in the dataflow.

Parameter	Required?	Value
predictionDefinitionName	Yes	Name of the prediction definition to use. The Model Manager displays prediction definition Ids. See Manage Models and Predictions.
predictionColumnName	Yes	Name of the new dataset column to contain the score.
predictionColumnLabel	Yes	UI label of the new dataset column to contain the score.

👁 **Example:** Let's look at an example that predicts the likelihood that each customer will churn and then compares the prediction against the actual outcome.

```
{
  "Customers": {
    "action": "edgemart",
    "parameters": {
      "alias": "Customers"

  },
  "edScore": {
    "action": "prediction",
    "parameters": {
      "predictionDefinitionName": "How_to_min_pAw3D_7383",
      "source": "Customers",
      "predictionColumnLabel": "Likelihood to Churn",
      "predictionColumnName": "attrition_score"

  },
  "Predicted": {
    "action": "computeExpression",
    "parameters": {
      "source": "edScore",
```

```
        "computedFields": [

            "name": "PredictedChurn",
            "saqlExpression": "case when 'attrition_score' > 0.4988 then \"true\" else
\"false\" end",
            "label": "PredictedChurn",
            "type": "Text"

        ],
        "mergeWithSource": true

    },
    "Correct": {
      "action": "computeExpression",
      "parameters": {
        "source": "Predicted",
        "computedFields": [

            "name": "CorrectPrediction",
            "saqlExpression": "case when 'PredictedChurn' == 'Churn__c' then \"true\"
else \"false\" end",
            "label": "CorrectPrediction",
            "type": "Text"

        ],
        "mergeWithSource": true

    },
    "Register": {
      "action": "sfdcRegister",
      "parameters": {
        "name": "Customers with Score",
        "alias": "CustomerWithScore",
        "source": "Correct"


}
```

sfdcDigest Transformation

The sfdcDigest transformation generates a dataset based on data that it extracts from a Salesforce object. You specify the Salesforce object and fields from which to extract data. You might choose to exclude particular fields that contain sensitive information or that aren't relevant for analysis.

📝 Note: To access data in a Salesforce object in a Data Prep recipe, see Input Node: Add Data to the Recipe. To do it in a Data Prep Classic recipe, see Create a Recipe with Data Prep Classic.

When you upload the dataflow definition file, Analytics Cloud validates access to Salesforce objects and fields based on the user profile of the Integration User. If the user profile does not have read access to a field or object, the upload fails.

At run time, Analytics Cloud runs the dataflow as the Integration User. Again, Analytics Cloud validates access to the objects and fields based on the profile of the Integration User. For example, if the dataflow tries to extract data from a custom field on which the Integration User does not have read access, the dataflow job fails.

📝 Note: The Integration User is a preconfigured user that is created when Analytics Cloud is enabled in your organization. If you or the Integration User need permission on a Salesforce object or field, ask the administrator to grant access.

For more information about preconfigured users in Analytics Cloud, see the *Tableau CRM Security Implementation Guide*.

👁 **Example:** Let's look at an example. You would like to create a dataset that contains all opportunities from the Opportunity object. You create the following dataflow definition.

```
{
    "Extract_Opportunities": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Opportunity",
            "fields": [
                { "name": "Id" },
                { "name": "Name" },
                { "name": "Amount" },
                { "name": "StageName" },
                { "name": "CloseDate" },
                { "name": "AccountId" },
                { "name": "OwnerId" },
                { "name": "OpportunitySupportTeamMembers__c" }
            ]
        }
    },
    "Register_Opportunities_Dataset": {
        "action": "sfdcRegister",
        "parameters": {
            "alias": "Opportunities",
            "name": "Opportunities",
            "source": "Extract_Opportunities"
        }
    }
}
```

Considerations When Using the sfdcDigest Transformation

- Consider dataset storage limits when extracting data. For example, a dataset can contain a maximum of 5,000 fields, so be selective when choosing fields. See Tableau CRM Limits.

- The sfdcDigest transformation runs a SOQL query to extract data from a Salesforce object, and so is subject to SOQL limits. If the query exceeds any of these limits, it may return no results or cause the dataflow job to fail. For example, The length of the SOQL query cannot exceed 100,000 characters. To reduce the SOQL query length, consider breaking up the extract into two or more sfdcDigest transformations and then use the augment transformation to combine the results. For example, you might create one sfdcDigest transformation to extract half of the fields and create another sfdcDigest transformation to extract the remaining fields. See SOQL and SOSL Limits.

- The sfdcDigest transformation can extract data from Salesforce Big Objects, but incremental extract isn't supported and filtering is possible only on primary key fields.

Filtering Records Extracted from a Salesforce Object

Add a filter to the sfdcDigest transformation to extract a subset of all records from a Salesforce object. You can filter records to reduce the number of extracted and processed records, exclude records that contain irrelevant or sensitive data, and increase dataflow performance.

Overriding Salesforce Field Metadata

You can override the field metadata that the sfdcDigest transformation extracts from a Salesforce object to make the data appear differently in a dataset. For example, Analytics Cloud can add a default value to records that have missing values for a field.

The sfdcDigest transformation can extract data from standard and custom big objects in Salesforce, with or without data sync enabled. Consider these limitations for big objects before you configure extraction.

The sfdcDigest transformation and data sync can't extract data from all Salesforce objects and fields. Consider these limitations before configuring extraction and sync.

When you define an sfdcDigest transformation, you set the action attribute to `sfdcDigest` and specify the parameters for the object and fields that you want to extract. Optionally, you can also specify parameters to filter the records extracted from the Salesforce object.

Filtering Records Extracted from a Salesforce Object

Add a filter to the sfdcDigest transformation to extract a subset of all records from a Salesforce object. You can filter records to reduce the number of extracted and processed records, exclude records that contain irrelevant or sensitive data, and increase dataflow performance.

A filter consists of one or more filter conditions, where each filter condition compares a field value to a value. For example, `Amount >= 1000000`. You can also apply SOQL functions on the field value in a filter condition, like `CALENDAR_YEAR(CreatedDate) = 2011`. You can add multiple filter conditions using logical operators AND, OR, and NOT. You can also use a backslash (\) to escape double quotes included in strings.

The sfdcDigest transformation extracts all records for which the filter is true. If you configured the sfdcDigest transformation for incremental extraction, the filter applies to data extracted during the incremental run only—Analytics Cloud doesn't apply the filter to records that were previously loaded into the dataset. If you add an invalid filter, the dataflow fails at run time.

For each instance of sfdcDigest, you can use one of the following types of filters:

- Structured filter
- Advanced filter

💡 **Tip:** Are you trying to decide whether to use a filter in the sfdcDigest transformation or use a filter transformation? Use a filter transformation to filter records at any point in the dataflow. For example, you can add it after the dataflow joins two datasets. However, to reduce the number of rows processed in the dataflow and optimize dataflow performance, add the filter closest to the point at which records are extracted—when possible, add the filter in the sfdcDigest transformation.

📝 **Note:** Filtering records extracted from Salesforce Big Objects is supported only on primary key fields in the sfdcDigest transformation.

You define a structured filter using JSON syntax.

You define an advanced filter using a Salesforce Object Query Language (SOQL) WHERE clause expression. Use an advanced filter only if you are familiar with SOQL.

SEE ALSO:

Structured Filter in sfdcDigest Transformation
You define a structured filter using JSON syntax.

A structured filter uses the following JSON syntax for each filter condition.

```
{
"field": "<field name>",
"operator": "<operator>",
"value": "<value>"|"["<value 1>", "<value 2>"]",
"isQuoted": true|false}
```

The value can be a number, date, string, list of strings, or date literal. Analytics Cloud automatically quotes strings unless you set isQuoted to true, which indicates that the string is already quoted.

You can use one of the following operators in a filter condition.

Operator	Comment
=	Filter condition is true if the value in the field equals the specified value. String comparisons using the equals operator are case-insensitive. Example: ``` "filterConditions": [{ "field": "OwnerId", "operator": "=", "value": "0056A0000020jzDQAQ" }] ```
!=	Filter condition is true if the value in the field does not equal the specified value. Example (using backslashes to escape double quotes in a string value): ``` "filterConditions": [{ "field": "Nickname__c", "operator": "!=", "value": "\"Sammy\"" }] ```
>	Filter condition is true if the value in the field is greater than the specified value. Example: ``` "filterConditions": [{ "field": "Amount", "operator": ">", "value": "100000" }] ```

Operator	Comment

<	Filter condition is true if the value in the field is less than the specified value.
	
	Example (using a date literal):
	
	``
	"filterConditions": [
	    {
	    "field": "CloseDate",
	    "operator": "<",
	    "value": "THIS_MONTH",
	    "isQuoted": false
	    }
	]
	``

>=	Filter condition is true if the value in the field is greater than or equal to the specified value.
	
	Example:
	
	``
	"filterConditions": [
	    {
	    "field": "Amount",
	    "operator": ">=",
	    "value": "100000"
	    }
	]
	``

<=	Filter condition is true if the value in the field is less than or equal to the specified value.
	
	Example (using a SOQL function):
	
	``
	"filterConditions": [
	    {
	    "field": "CALENDAR_YEAR (CreatedDate)",
	    "operator": "<=",
	    "value": "2015",
	    "isQuoted": true
	    }
	]
	``

LIKE	Filter condition is true if the value in the field matches the specified value. The LIKE operator is similar to the LIKE operator in SQL; it provides a mechanism for matching partial text strings and supports wildcards. • The % and _ wildcards are supported for the LIKE operator. • The % wildcard matches zero or more characters. • The _ wildcard matches exactly 1 character. • The LIKE operator is supported for string fields only.

Operator	Comment
	• The LIKE operator performs a case-insensitive match. • The LIKE operator supports escaping of special characters % or _. Use a backslash (\) to escape special characters. Example: ```"filterConditions": [\n {\n "field": "FirstName",\n "operator": "LIKE",\n "value": "Chris%"\n }\n]```
IN	Filter condition is true if the value in the field equals any one of the values in the specified list. You can specify a quoted or non-quoted list of values. If the list is quoted, set isQuoted to true. Example: ```"filterConditions": [\n {\n "field": "StageName",\n "operator": "IN",\n "value": ["Closed Won", "Closed Lost"]\n }\n]```
NOT IN	Filter condition is true if the value in the field does not equal any of the values in the specified list. Example: ```"filterConditions": [\n {\n "field": "BillingState",\n "operator": "NOT IN",\n "value": ["California", "New York"]\n }\n]```
INCLUDES	For picklist or multi-select picklist fields only. Filter condition is true if the value in the picklist field includes the specified value. Example: ```"filterConditions": [\n {\n "field": "BillingState",\n "operator": "INCLUDES",\n "value": ["California"]```

Operator	Comment
	<pre> }]</pre>

EXCLUDES	For picklist or multi-select picklist fields only. Filter condition is true if the value in the picklist field excludes the specified value.
	
	Example:
	<pre>"filterConditions": [
	    {
	    "field": "BillingState",
	    "operator": "EXCLUDES",
	    "value": ["California", "New York"]
	    }
	]</pre>

Let's look at a few examples of structured filters.

👁 **Example:** Let's look at an example with a basic structured filter. To perform pipeline analysis on opportunities in fiscal quarter 2 of fiscal year 2015, you create this dataflow definition file to create the relevant dataset.

```
{
    "Extract_Filtered_Opportunities": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Opportunity",
            "fields": [
                { "name": "Id" },
                { "name": "Name" },
                { "name": "AccountId" },
                { "name": "Amount" },
                { "name": "StageName" },
                { "name": "CloseDate" },
                { "name": "OwnerId" },
                { "name": "FiscalYear" },
                { "name": "FiscalQuarter" },
                { "name": "SystemModstamp" }
            ],
            "filterConditions": [
                {
                    "field": "FiscalYear",
                    "operator": "=",
                    "value": "2015"
                },
                {
                    "field": "FiscalQuarter",
                    "operator": "=",
                    "value": "2"
                }
            ]
```

```
        }
    },
    "Register_Opportunities_Dataset": {
        "action": "sfdcRegister",
        "parameters": {
            "alias": "Opportunities_2015Q2",
            "name": "Opportunities_2015Q2",
            "source": "Extract_Filtered_Opportunities"
        }
    }
}
```

📝 **Note:** If you do not specify a logical operator for multiple filter conditions—as is the case in this example—Analytics Cloud applies AND between the conditions.

👁 **Example:** Let's look at an example of a structured filter with a logical operator. To help forecast expected revenue, you create this dataflow to view all opportunities that have either closed or have greater than 90% probability of closing.

```
{
    "Extract_Opportunities": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Opportunity",
            "fields": [
                { "name": "Id" },
                { "name": "Name" },
                { "name": "AccountId" },
                { "name": "Amount" },
                { "name": "StageName" },
                { "name": "CloseDate" },
                { "name": "OwnerId" },
                { "name": "Probability" },
                { "name": "FiscalYear" },
                { "name": "FiscalQuarter" }
            ],
            "filterConditions": [
                {
                    "operator": "OR",
                    "conditions": [
                        {
                            "field": "StageName",
                            "operator": "=",
                            "value": "Closed Won"
                        },
                        {
                            "field": "Probability",
                            "operator": ">=",
                            "value": "90"
                        }
                    ]
                }
            ]
        }
    },
```

```
    "Register_Opportunities_Dataset": {
        "action": "sfdcRegister",
        "parameters": {
            "alias": "OpportunitiesExpectedToWin",
            "name": "OpportunitiesExpectedToWin",
            "source": "Extract_Opportunities"
        }
    }
}
```

👁 Example: Finally, let's look at an example of a structured filter with nested logical operators. You create this dataflow to view all opportunities that closed in the current fiscal quarter and are owned by either one of your two direct reports.

```
{
    "Extract_Opportunities": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Opportunity",
            "fields": [
                { "name": "Id" },
                { "name": "Name" },
                { "name": "AccountId" },
                { "name": "Amount" },
                { "name": "StageName" },
                { "name": "CloseDate" },
                { "name": "OwnerId" },
                { "name": "FiscalYear" },
                { "name": "FiscalQuarter" }
            ],
            "filterConditions": [
                {
                    "operator": "AND",
                    "conditions": [
                        {
                            "field": "CloseDate",
                            "operator": "=",
                            "value": "THIS_FISCAL_QUARTER",
                            "isQuoted": false
                        },
                        {
                            "operator": "OR",
                            "conditions": [
                                {
                                    "field": "OwnerId",
                                    "operator": "=",
                                    "value": "0056A0000020jzDQAQ"
                                },
                                {
                                    "field": "OwnerId",
                                    "operator": "=",
                                    "value": "0056A0000020jzGQAQ"
                                }
                            ]
                        }
```

```
                    ]
                }
            ]
        }
    },
    "Register_Opportunities_Dataset": {
        "action": "sfdcRegister",
        "parameters": {
            "alias": "DirectReport_Opportunities",
            "name": "DirectReport_Opportunities",
            "source": "Extract_Opportunities"
        }
    }
}
```

Advanced Filter in sfdcDigest Transformation

You define an advanced filter using a Salesforce Object Query Language (SOQL) WHERE clause expression. Use an advanced filter only if you are familiar with SOQL.

📝 **Note:** When you add an sfdcDigest node filter with Data Sync enabled, that filter is added automatically to the SFDC_LOCAL connection for that object. If you delete a filter from an sfdcDigest node, the SFDC_LOCAL connection filter isn't changed. Manually update the SFDC_LOCAL connection filter.

👁 **Example:** Let's look at an example of an advanced filter. You want to extract only opportunity records that are owned by a specific user and that have either high value or a high probability of closing. You create this dataflow in the dataflow editor and add a filter in the Complex Filter Conditions field of the sfdcDigest node.

To add an advanced filter in the dataflow JSON, add a `complexFilterConditions` parameter in the sfdcDigest node.

```
{
    "Extract_Filtered_Opportunities": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Opportunity",
            "fields": [
                { "name": "Id" },
                { "name": "Name" },
                { "name": "AccountId" },
                { "name": "Amount" },
                { "name": "StageName" },
                { "name": "CloseDate" },
                { "name": "Probability" },
                { "name": "OwnerId" }
            ],
            "complexFilterConditions": "OwnerId = '005460000022HhMAAU' AND (Amount >
100000 OR Probability > 75)"
        }
```

```
      },
      "Register_Opportunities_Dataset": {
         "action": "sfdcRegister",
         "parameters": {
            "alias": "FilteredOpportunities",
            "name": "FilteredOpportunities",
            "source": "Extract_Filtered_Opportunities"
         }
      }
   }
}
```

Consider the following requirements for advanced filters.

- Always enclose OR conditions in parentheses in advanced filters, even if there are no other conditions. For example, to extract only closed won or closed lost opportunities, use this advanced filter: *(StageName = 'Closed Won' OR StageName = 'Closed Lost')*. Excluding parentheses causes the dataflow to fail.

- You can't use subqueries in an advanced filter if incremental sync is enabled on the Salesforce object.

Overriding Salesforce Field Metadata

You can override the field metadata that the sfdcDigest transformation extracts from a Salesforce object to make the data appear differently in a dataset. For example, Analytics Cloud can add a default value to records that have missing values for a field.

You can add the following field parameters to the sfdcDigest transformation node to override the field metadata:

- defaultValue

- type

- fiscalMonthOffset

- isYearEndFiscalYear

- firstDayOfWeek

- isMultiValue

- multiValueSeparator (not available in dataflow editor)

- precision

- scale

For a description of each of these field parameters, see Field Parameters. For information about using metadata attributes to configure dates, see Handle Date Values.

👁 Example: Let's look at an example. You would like to override metadata extracted from the Opportunity object.

To override field metadata from the Opportunity object, you add the bold text to the sfdcDigest node in the datflow definition file.

```
{
   "Extract_Opportunities": {
      "action": "sfdcDigest",
      "parameters": {
         "object": "Opportunity",
         "fields": [
            { "name": "Id" },
            { "name": "Name" },
            {
               "name": "Amount",
               "defaultValue":"0",
```

```
                    "precision":18,
                    "scale":2
                },
                { "name": "StageName" },
                {
                    "name": "CloseDate",
                    "fiscalMonthOffset":9,
                    "firstDayOfWeek":2,
                    "isYearEndFiscalYear":true
                },
                { "name":"AccountId" },
                { "name":"OwnerId" },
                {
                    "name": "OpportunitySupportTeamMembers__c",
                    "type":"Text",
                    "isMultiValue":true,
                    "multiValueSeparator":",",
                    "precision":255
                }
            ]
        }
    },
    "Register_Opportunities_Dataset":{
        "action":"sfdcRegister",
        "parameters":{
            "alias":"Opportunities",
            "name":"Opportunities",
            "source":"Extract_Opportunities"
        }
    }
}
```

If you're working in the dataflow editor, click the sfdcDigest node and then click [✏] next to the field that you want to change.

939

Update the field attributes, and then click **Save**.

> **Note:** Changing a field's type attribute to anything other than Text can result in an error in your dataflow or unexpected values in your datasets. For example, if you change a text field to a numeric field, you see an error when you try to update the dataflow.

SEE ALSO:

sfdcDigest Transformation

Salesforce Big Object Support in Tableau CRM
The sfdcDigest transformation can extract data from standard and custom big objects in Salesforce, with or without data sync enabled. Consider these limitations for big objects before you configure extraction.

- Incremental data sync isn't supported, so a full sync is performed every time data sync runs.

- Filtering is supported only on primary key fields.

- Custom big object names must end in __b

- These standard big objects are currently supported.

  - ApiEvent

  - BigObjectCounter

  - BotAnalytics

  - ChatbotAnalytics

  - EngagementHistory

  - FeedRead

  - FeedSentimentAnalysis

  - FeedSentimentFeedbacks

  - LoginEvent

- The sfdcDigest transformation uses the Bulk API to extract data from Salesforce objects, and so it's subject to Bulk API limits.

  - When a large amount of data is extracted from a big object, a maximum of 15 result files are created. The Bulk Query stops when this 15-file limit is reached, even if the extract isn't complete. Each extracted file is limited to 1 GB of data.

  - The total number of big object records that can be extracted from the Bulk API is currently limited to 70 million rows across all files in the dataset. The number of rows varies based on the number and size of columns in the big object being queried.

  - Each batch can run for up to 10 minutes or 1 GB of data, whichever limit is reached first.

For more information about the Bulk API, see Introduction to Bulk API.

Unsupported Salesforce Objects and Fields in Tableau CRM
The sfdcDigest transformation and data sync can't extract data from all Salesforce objects and fields. Consider these limitations before configuring extraction and sync.

For information about all Salesforce objects and fields, see the Object Reference for Salesforce and Lightning Platform.

Unsupported Objects

The sfdcDigest transformation and data sync can't extract data from these Salesforce objects.

- ActivityHistory

- ActivityMetric

- ApexEmailNotification

- AuthProvider

- BrandTemplate

- ChatbotMetric
- ChatterConversation
- ChatterConversationMember
- ChatterMessage
- ConnectedApplication
- ContentDocumentLink
- ContentFolderLink
- ContentFolderMember
- ContentWorkspace
- ContentWorkspaceDoc
- ContentWorkspaceMember
- ContentWorkspacePermission
- CopyExport
- CorsWhitelistEntry
- DataAssessmentFieldMetric
- DataAssessmentMetric
- DataAssessmentValueMetric
- DataHubSetupData
- DataHubSetupDefinition
- DirectMessage
- DirectMessageFeed
- DirectMessageMember
- EmailCapture
- EmailDomainKey
- EmailServicesAddress
- EmailServicesFunction
- EmailStatus
- EmailTemplate
- EnvironmentHub
- EnvironmentHubInvitation
- EnvironmentHubMemberRel
- EventType
- EventTypeParameter
- ExternalString
- FeedLike
- FeedPollChoice
- FeedPollVote
- FieldDefinition
- KnowledgeArticle
- KnowledgeArticleVersion

- KnowledgeArticleVersionHistory
- KnowledgeArticleViewStat
- KnowledgeArticleVoteStat
- LeadChangeEvent
- ListViewEvent
- LoginGeo
- LoginHistory
- NetworkActivityAudit
- NetworkModeration
- OpenActivity
- OrganizationProperty
- OrgWideEmailAddress
- PackageLicense
- PartnerNetworkSyncLog
- ReputationLevel
- ReputationLevelLocalization
- ReputationPointsRule
- S2XEventMap
- SalesforceIqUser
- SampledEntity
- SandOmInfo
- SandOmInfoDetail
- SandOmObserver
- ScoreIntelligence
- SearchPromotionRule
- SecurityCustomBaseline
- SelfServiceUser
- SsoUserMapping
- TenantSecret
- TwoFactorInfo
- TwoFactorTempCode
- UserPackageLicense
- UserProvAccount
- UserProvAccountStaging
- UserProvisioningConfig
- UserProvisioningLog
- UserProvisioningRequest
- UserProvisioningRequestOwnerSharingRule
- UserProvisioningRequestShare
- UserProvMockTarget

- UserRecordAccess
- VerificationHistory
- VoiceUserLine
- VoiceUserLineOwnerSharingRule
- VoiceUserLineShare
- VoiceVendorLine
- VoiceVendorLineOwnerSharingRule
- VoiceVendorLineShare
- WebLink
- WebLinkLocalization

The sfdcDigest transformation cannot extract data from external objects created in Salesforce. External objects are similar to custom objects, except that they map to data located outside Salesforce.

If you include an unsupported or inaccessible object in the sfdcDigest transformation, the dataflow fails at run time with an error message.

> 📝 Note: If an object isn't listed but you still can't use it in data sync or sfdcDigest, consider these factors.
>
> - Supported objects must be visible to the Integration User to digest its fields. If you can't manage a field's visibility in Setup, you can't grant the Integration User access to it.
> - Objects that require query clauses outside those used in this example likely aren't supported.
>
>   ```
>   SELECT fields
>   FROM object
>   WHERE Id > '0xx00000000xxxx'
>   AND Id <= '0xx00000000yyyy'
>   ORDER BY Id
>   ASC LIMIT 1
>   OFFSET 249999;
>   ```
>
>   SELECT and FROM describe the fields and object that are being synced. The rest of the query controls batching. Bulk API processes batches of 10,000 records, determined by WHERE filtering and ORDER sorting by the record Ids. Additional clauses that limit the ability to use this structure of query, such as constraints on ORDER, WHERE, or others, will prevent Analytics from querying the object for data sync or sfdcDigest. To learn more about the SELECT clause, see this help page.

Unsupported Fields

The sfdcDigest transformation can't extract data from these fields.

Object	Unsupported Fields
Account	CleanStatus
ActionPlanItem	ItemId
AuthSession	- LoginGeoId - LoginHistoryId
CaseArticle	KnowledgeArticleId
Contact	- CanAllowPortalSelfReg - CleanStatus

Object	Unsupported Fields
ContentDocument	ParentId
CustomPerson__p	Title
DocumentAttachmentMap	ParentId
EmailMessage	- ActivityId - EmailTemplateId
EmailRoutingAddress	EmailServicesAddressId
EnvironmentHubMember	EnvironmentHubId
ExternalEventMapping	EventId
InstalledMobileApp	ConnectedApplicationId
Lead	CleanStatus
KnowledgeArticle	MasterLanguage
Network	- CaseCommentEmailTemplateId - ChangePasswordEmailTemplateId - ForgotPasswordEmailTemplateId - WelcomeEmailTemplateId
Organization	- SelfServiceEmailUserOnCaseCreationTemplateId - SelfServiceNewCommentTemplateId - SelfServiceNewPassTemplateId - SelfServiceNewUserTemplateId - WebToCaseAssignedEmailTemplateId - WebToCaseCreatedEmailTemplateId - WebToCaseEmailTemplateId - WebToLeadEmailTemplateId
PermissionSet	- PermissionsEditEvent - PermissionsEditTask
PermissionSetLicense	- MaximumPermissionsEditEvent - MaximumPermissionsEditTask
Profile	- PermissionsEditEvent - PermissionsEditTask
ThirdPartyAccountLink	SsoProviderId

Object	Unsupported Fields
User	- LastPasswordChangeDate - ProfilePhotoId - UserPreferencesEnableVoicePilot
WorkBadge	RewardId
WorkBadgeDefinition	RewardFundId

If you include a field with an unsupported field in the sfdcDigest transformation, the dataflow ignores the field.

In addition, Salesforce recommends that you do not extract data from the MayEdit field of the Account object. Extracting data from this field significantly decreases performance and can cause the dataflow to fail.

Unsupported Field Types

The sfdcDigest transformation can't extract data from fields with these field types.

- base64
- composite (like address and location)
- data category group reference
- encrypted string

If you include a field with an unsupported field type in the sfdcDigest transformation, the dataflow ignores the field.

SEE ALSO:

    sfdcDigest Transformation

sfdcDigest Parameters

When you define an sfdcDigest transformation, you set the action attribute to `sfdcDigest` and specify the parameters for the object and fields that you want to extract. Optionally, you can also specify parameters to filter the records extracted from the Salesforce object.

You can specify parameters in the following sections of the sfdcDigest node: parameters, fields, and filterConditions.

Parameters

The following table describes the parameters in the `parameters` section.

Parameter	Required?	Value
object	Yes	API name of the Salesforce object from which you want to extract data. This object is the input source for this transformation. The sfdcDigest transformation doesn't support extraction from all Salesforce objects.
incremental	No	Performs an incremental sync, which extracts only changes to the Salesforce object since the last dataflow run. Valid values: true or false. 📝 **Note:** Incremental sync: - Is available if you have enabled data sync. - Isn't supported for Salesforce big objects.

Parameter	Required?	Value
fullRefreshToken	No	Performs a one-time full extraction to synchronize the data in the dataset with data in the Salesforce object. Specify any value for this parameter.
		After the full extraction, the dataflow performs an incremental extraction each time thereafter even though the `fullRefreshToken` parameter is included in the dataflow definition. To run a full extraction again, change the value of the `fullRefreshToken` parameter to a different value.
		Note: Incremental sync is available if you have enabled data sync.
fields	Yes	An array of names of all fields from which you want to extract data from the specified Salesforce object. The sfdcDigest transformation doesn't support extraction from all field types.
		See Field Attributes.
filterConditions	No	A filter that restricts the records extracted from the specified Salesforce object. The sfdcDigest transformation extracts all records from the Salesforce object for which the filter is true. You can specify a structured or advanced filter.
		See Filter Conditions Parameters.
complexFilterConditions	No	For advanced filters only. A SOQL WHERE clause used to filter records extracted from the specified Salesforce object.

Field Attributes

The following table describes the attributes in the `fields` section. It also describes optional attributes that you can provide to override the field metadata. You can override the metadata that the sfdcDigest transformation extracts from a Salesforce object to make the data appear differently in a dataset. For example, Analytics Cloud can add a default value to records that have missing values for a field. If you don't override the values, Analytics Cloud gets the values from Salesforce.

Attribute	Required?	Value
name	Yes	API name of the field in the Salesforce object that you want to include in the dataset. You can specify multiple fields.
defaultValue	No	For text and numeric fields that can be null. Default value that replaces a null value for the specified field. Enter a string value.
		Example:
		`"defaultValue": "0"`
type	No	Analytics Cloud field type associated with the specified field. Valid types are Text, Numeric, or Date. Any value, including numeric values, can be Text. For example, by default, fiscal quarter from Salesforce objects is Number. However, you can change it to Text. Specify a type to override the type determined by Analytics Cloud.
		Example:
		`"type": "Text"`

Attribute	Required?	Value
		📝 **Note:** You can't change a field's type to Numeric or Date.
fiscalMonthOffset	No	For date fields only. The difference, in months, between the first month of the fiscal year and the first month of the calendar year (January). For example, if the fiscal year starts in January, the offset is 0. If the fiscal year starts in October, the offset is 9.
		Example:
		`"fiscalMonthOffset": 9`
		📝 **Note:** This attribute also controls whether Analytics Cloud generates fiscal date fields. To generate fiscal date fields, set `fiscalMonthOffset` to a value other than 0.
		⚠ **Warning:** Tableau CRM doesn't support fields with different `fiscalMonthOffset` values in the same dataset. Using different `fiscalMonthOffset` values can produce unexpected results when you filter by relative fiscal date ranges. We recommend that you set the same value for all `fiscalMonthOffset` attributes in a dataset.
		For more information, see Handle Date Values.
isYearEndFiscalYear	No	For date fields only. Indicates whether the fiscal year is the year in which the fiscal year ends or begins. Because the fiscal year can start in one calendar year and end in another, you must specify which year to use for the fiscal year.
		• If true, then the fiscal year is the year in which the fiscal year ends. The default is true.
		• If false, then the fiscal year is the year in which the fiscal year begins.
		Example:
		`"isYearEndFiscalYear": true`
		This field is relevant only when `fiscalMonthOffset` is greater than 0.
		⚠ **Warning:** Tableau CRM doesn't support fields with different `isYearEndFiscalYear` values in the same dataset. Using different `isYearEndFiscalYear` values can produce unexpected results when you filter by relative fiscal date ranges. We recommend that you set the same value for all `isYearEndFiscalYear` attributes in a dataset.
		For more information, see Handle Date Values.
firstDayOfWeek	No	For date fields only. The first day of the week for the calendar year and, if applicable, fiscal year. Use 0 to set the first day to be Sunday, 1 to set the first day to be Monday, and so on. Use -1 to set the first day to be on January 1. The default is -1.
		Example:
		`"firstDayOfWeek": 0`
		⚠ **Warning:** Tableau CRM doesn't support fields with different `firstDayOfWeek` values in the same dataset. Using different

Attribute	Required?	Value
		`firstDayOfWeek` values can produce unexpected results when you filter by relative week date ranges. We recommend that you set the same value for all `firstDayOfWeek` attributes in a dataset. For more information, see Handle Date Values.
isMultiValue	No	For text fields only. Indicates whether the specified field has multiple values. Example: `"isMultiValue": false`
multiValueSeparator	No	For text fields only. Single character used to separate multiple values in the specified field when isMultiValue equals true. This value defaults to a semicolon (;) if you do not specify a value and isMultiValue equals true. Set to null when isMultiValue equals false. Example: `"multiValueSeparator": ";"`
precision	No	The maximum number of digits in a numeric value, or the length of a text value. For numeric values: Includes all numbers to the left and to the right of the decimal point (but excludes the decimal point character). Value must be from 1 through 18, inclusive. For text values: Value defaults to 255 characters, and must be from 1 through 32,000 characters, inclusive. Example: `"precision": 10`
scale	No	The number of digits to the right of the decimal point in a numeric value. Must be less than the precision value. Value must be from 0 through 17 characters, inclusive. Example: `"scale": 2`

Filter Conditions Parameters

The following table describes the structured filter parameters in the `filterConditions` section. These parameters do not apply to advanced filters.

Parameter	Required?	Value
field	No	The field in the Salesforce object on which you want to apply a filter condition. Each filter condition in a structured filter uses the following syntax: ```{ "field": "<field name>", "operator": "<operator>", "value": "<value>", "isQuoted": true|false}```

Parameter	Required?	Value
operator	No	The purpose depends on the context.
		• operator can be used as a comparison operator–like =, <, and IN–that compares the field value against a constant value.
		• operator can also be used as a logical operator (AND, OR, or NOT) that links multiple filter conditions together.
		In the example below, the bold operator is the logical operator. The other instances of operator are comparison operators.

```
"filterConditions": [
    {
        "operator": "OR",
        "conditions": [
            {
                "field": "StageName",
                "operator": "=",
                "value": "Closed Won"
            },
            {
                "field": "Probability",
                "operator": ">=",
                "value": "90"
            }
        ]
    }
]
```

Parameter	Required?	Value
value	No	The value used in a filter condition.
isQuoted	No	Indicates whether you quoted the string value in a filter condition.
		Example with quoted values:

```
"filterConditions": [
    {
    "field": "StageName",
    "operator": "IN",
    "value": "('Closed Won', 'Closed Lost')",
    "isQuoted": true
    }
]
```

Example with non-quoted values:

```
"filterConditions": [
    {
    "field": "StageName",
    "operator": "IN",
    "value": ["Closed Won", "Closed Lost"],
    "isQuoted": false
    }
]
```

Parameter	Required?	Value
		If you don't include isQuoted for a filter on a string value, Analytics Cloud assumes that the string value is not quoted and adds the quotes for you.
conditions	No	Use to specify a logical operator to link multiple filter conditions together.

SEE ALSO:

sfdcRegister Transformation

The sfdcRegister transformation registers a dataset to make it available for queries. Users cannot view or run queries against unregistered datasets.

The sfdcRegister transformation overwrites the current version of the dataset if it already exists.

You don't need to register all datasets. For example, you don't need to register an intermediate dataset that is used to build another dataset and does not need to be queried. In addition, you don't need to register datasets that are created when you upload external data because Analytics Cloud automatically registers these datasets for you.

Carefully choose which datasets to register because:

- The total number of rows in all registered datasets cannot exceed 100,000 per platform license, or 250 million per platform license *purchased before* October 20, 2015.

- Users that have access to registered datasets can query their data. Although, you can apply row-level security on a dataset to restrict access to records.

👁 **Example:** Let's look at an example. You create a dataflow that extracts opportunities from the Opportunity object. To register the dataset, name it "Opportunities," and apply row-level security on it, you add the sfdcRegister transformation as shown in the following dataflow definition file.

```
{
   "Extract_Opportunities": {
      "action": "sfdcDigest",
      "parameters": {
         "object": "Opportunity",
         "fields": [
            { "name": "Id" },
            { "name": "Name" },
            { "name": "Amount" },
            { "name": "StageName" },
            { "name": "CloseDate" },
            { "name": "AccountId" },
            { "name": "OwnerId" }
         ]
      }
   },
   "Register_Oppportunities_Dataset": {
      "action": "sfdcRegister",
      "parameters": {
         "alias": "Opportunities",
         "name": "Opportunities",
         "source": "Extract_Opportunities",
```

```
            "rowLevelSecurityFilter": "'OwnerId' == \"$User.Id\""
        }
    }
}
```

sfdcRegister Parameters

When you define an sfdcRegister transformation, you set the action attribute to `sfdcRegister` and specify the parameters.

sfdcRegister Parameters

When you define an sfdcRegister transformation, you set the action attribute to `sfdcRegister` and specify the parameters.

The following table describes the input parameters:

Parameter	Required?	Value
alias	Yes	API name of the registered dataset. This name can contain only underscores and alphanumeric characters, and must be unique among other dataset aliases in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. It also cannot exceed 80 characters. 📝 Note: When the alias is unique, the sfdcRegister transformation registers a new dataset with the alias. When an existing dataset has the alias, the transformation overwrites the current version of that dataset.
name	Yes	Display name of the registered dataset. The name cannot exceed 80 characters. 📝 Note: To change the name after you create the dataset, you must edit the dataset.
source	Yes	Node in the dataflow definition file that identifies the dataset that you want to register. This node is the input source for this transformation.
rowLevelSharingSource	No	The API name of the object from which to inherit sharing. Used when applying row-level security on the dataset when the dataset is first created. Example: "rowLevelSharingSource": "Opportunity" 📝 Note: To change the sharing source after you create the dataset, you must edit the dataset. If they don't match, you see a warning that says, "The sharing source and security predicate in this dataset version must be the same as in the dataflow." For more information, refer to Add Row-Level Security by Inheriting Sharing Rules.
rowLevelSecurityFilter	No	The predicate used to apply row-level security on the dataset when the dataset is first created. Example: "rowLevelSecurityFilter": "'OwnerId' == "$User.Id""

Parameter	Required?	Value
		Note: To change the predicate after you create the dataset, you must edit the dataset. When entering the predicate in the Register transformation of the dataflow JSON, you must escape the double quotes around string values. After the dataset is created, Analytics Cloud ignores its security predicate setting in the dataflow. To change the security predicate for an existing dataset, edit the dataset in the user interface.

SEE ALSO:

    sfdcRegister Transformation

sliceDataset Transformation

The sliceDataset transformation removes fields from a dataset in your dataflow, leaving you with a subset of fields for use in a new dataset or in other transformations. This allows you to create multiple datasets, each with different sets of fields from a single dataset.

 Note: To drop columns in a Data Prep recipe, see Drop Columns Transformation: Drop Columns from the Recipe.

The transformation includes the `mode` parameter, which lets you choose between two modes: `drop` or `select`. Set it to `drop` to *remove* the fields you specify. Set it to `select` to *keep* the fields you specify.

 Example: Here's an example. Let's say your dataflow contains an sfdcDigest node, `ExtractOpportunities`, that extracts all fields from the Opportunity object. However, some of your users don't want Id fields in their Opportunities dataset. You resolve this by adding a sliceDataset node to your dataflow, specifying the source node, the fields you want to remove from that node, and `drop` mode.

sliceDataset Node in Dataflow Editor

sliceDataset Node in Dataflow JSON

```json
{
  "RemoveFieldsFromExtractOpportunities":
  {
    "action": "sliceDataset",
    "parameters": {
      "source": "ExtractOpportunities",
      "mode": "drop",
      "fields": [
        {
          "name": "Id"
        },
        {
          "name": "Pricebook2Id"
        }
      ]
    }
  }
}
```
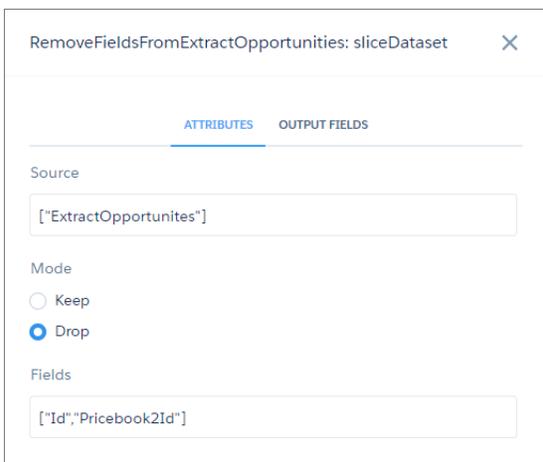
sliceDataset Parameters

Add a sliceDataset transformation to a dataflow either in the dataflow editor, or directly in the JSON dataflow definition file. The parameters you specify are the same for both methods.

sliceDataset Parameters

Add a sliceDataset transformation to a dataflow either in the dataflow editor, or directly in the JSON dataflow definition file. The parameters you specify are the same for both methods.

To add a sliceDataset transformation in the dataflow editor, click [icon].

To add a sliceDataset transformation in the JSON definition file, add a node and set the action attribute to `sliceDataset`.

The following table describes the input parameters:

Parameter	Required?	Value
source	Yes	Node in the dataflow definition file that identifies the dataset that you want to slice. This is the input source for this transformation.
mode	Yes	The action you want to take on the specified fields. Set it to `drop` to *remove* the fields. Set it to `select` to *keep* the fields.
		Note: A dataset must have at least 1 dimension field. If a sliceDataset node removes all dimension fields, the dataflow fails.
fields	Yes	An array of names of the fields that you want to drop or select from the specified source node.

update Transformation

The update transformation updates the specified field values in an existing dataset based on data from another dataset, which we'll call the lookup dataset. The transformation looks up the new values from corresponding fields in the lookup dataset. The transformation stores the results in a new dataset.

When you create the transformation, you specify the keys that are used to match records between the two datasets. To dictate which field in the lookup dataset updates the field in the source dataset, you also map the corresponding fields from both datasets.

👁 **Example:** Let's look at an example. You have an existing Accounts dataset that contains account information—Id, Name, and AnnualRevenue. Unfortunately, some of the account names in the dataset are now incorrect because of a series of mergers and acquisitions. To quickly update the account names in the dataset, you perform the following tasks.

1. Create a .csv file that contains the new account names and associated account IDs for accounts that have name changes.

2. Upload the .csv file to create a dataset called UpdatedAccountNames.

3. Create a dataflow definition file to update account names in the Accounts dataset by looking up the new account names in the UpdatedAccountNames dataset.
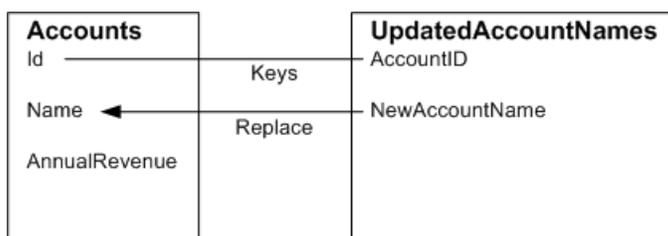
You create the following dataflow definition file.

```
{
   "Extract_AccountDetails": {
      "action": "sfdcDigest",
      "parameters": {
         "object": "Account",
         "fields": [
            { "name": "Id" },
            { "name": "Name" },
            { "name": "AnnualRevenue" }
         ]
      }
   },
   "Extract_UpdatedAccountNames": {
      "action": "edgemart",
      "parameters": { "alias": "UpdatedAccountNames" }
   },
   "Update_AccountRecords": {
      "action": "update",
      "parameters": {
         "left": "Extract_AccountDetails",
         "right": "Extract_UpdatedAccountNames",
         "left_key": [ "Id" ],
         "right_key": [ "AccountID" ],
         "update_columns": { "Name": "NewAccountName" }
      }
   },
   "Register_UpdatedAccountRecords": {
      "action": "sfdcRegister",
      "parameters": {
         "alias": "Accounts",
         "name": "Accounts",
         "source": "Update_AccountRecords"
      }
   }
}
```

👁 **Example:** Let's look at another example, where a composite key is used to match records between both datasets. In this case, you match records using the account ID and account name fields.

You create the following dataflow definition file.

```
{
   "Extract_AccountDetails": {
      "action": "sfdcDigest",
      "parameters": {
         "object": "Account",
         "fields": [
            { "name": "Id" },
            { "name": "Name" },
            { "name": "AnnualRevenue" }
         ]
```

```
        }
    },
    "Extract_UpdatedAccountNames": {
        "action": "edgemart",
        "parameters": { "alias": "UpdatedAccountNames" }
    },
    "Update_AccountRecords": {
        "action": "update",
        "parameters": {
            "left": "Extract_AccountDetails",
            "right": "Extract_UpdatedAccountNames",
            "left_key": ["Id","Name"],
            "right_key": ["AccountId","NewAccountName"],
            "update_columns": {
                "Name": "NewAccountName",
                "CreatedDate":"NewCreatedDate",
                "AnnualRevenue":"NewAnnualRevenue"
            }
        }
    },
    "Register_UpdatedAccountRecords": {
        "action": "sfdcRegister",
        "parameters": {
            "alias": "Accounts",
            "name": "Accounts",
            "source": "Update_AccountRecords"
        }
    }
}
```

[update Parameters](#)

When you define an update transformation, you set the action attribute to `update` and specify the parameters.

update Parameters

When you define an update transformation, you set the action attribute to `update` and specify the parameters.

The following table describes the input parameters.

Parameter	Required?	Value
left	Yes	Node in the dataflow definition file that identifies the dataset that contains the records that you want to update.
right	Yes	Node in the dataflow definition file that identifies the lookup dataset that contains the new values.
left_key	Yes	Key column in the left dataset used to match records in the other dataset. If you use a composite key, the left and right keys must have the same number of columns in the same order. For an example, see update Transformation.

Parameter	Required?	Value
right_key	Yes	Key column in the right dataset used to match records in the other dataset. If you use a composite key, the left and right keys must have the same number of columns in the same order.
update_columns	No	An array of corresponding columns between the left and right datasets. Use the following syntax: `"update_columns": { "LeftColumn1": "RightColumn1", "LeftColumn2": "RightColumn2",... "LeftColumnN": "RightColumnN" }`. The value from right column replaces the value from the corresponding left column. The field types of the left and right column must match. 📝 Note: If you specify a column name that does not exist, the dataflow fails. If you do not specify this parameter, the transformation updates the left dataset by matching all columns in the right dataset with those in the left. In this case, the right column names must match exactly with the left column names. Otherwise, an error might occur.

SEE ALSO:

Overriding Metadata Generated by a Transformation

Optionally, you can override the metadata that is generated by a transformation. You can override object and field attributes. For example, you can change a field name that is extracted from a Salesforce object so that it appears differently in the dataset. To override the metadata, add the overrides to the Schema section of the transformation in the dataflow definition file.

In the Schema section, you can override the metadata attributes for one object only.

The Schema section in this sample sfdcDigest transformation contains metadata overrides:

```
"Extract_Opportunities": {
    "action": "sfdcDigest",
```

```
    "parameters": {
      "object": "Opportunity",
      "fields": [
          { "name": "Name" },
          { "name": "Amount" }
      ]
    },
    "schema": {
      "objects": [
          {
              "label":"Sales Opportunities",
              "fields": [
                  {
                      "name": "Amount",
                      "label": "Opportunity Amount"
                  }
              ]
          }
      ]
    }
}
```

Object Attributes

You can override the following object attributes.

Object Attribute	Type	Description
label	String	The display name for the object. Can be up to 40 characters. Example: `"label": "Sales Data"`
description	String	The description of the object. Must be less than 1,000 characters. Add a description to annotate an object in the dataflow definition file. This description is not visible to users in the Analytics Cloud user interface. Example: `"description": "The SalesData object tracks basic sales data."`
fields	Array	The array of fields for this object.

Field Attributes

You can override attributes of each specified dataset field.

Field Attribute	Type	Description
name	String	Name of the field in the dataset. Identifies the field that you want to override. Examples:

Field Attribute	Type	Description
		`"name": "Amount"`
		`"name": "Role.Name"`
label	String	The display name for the field. Can be up to 255 characters. Example: `"label": "Opportunity Amount"`
description	String	The description of the field. Must be less than 1,000 characters. Add a description to annotate a field in the dataflow definition file. This description is not visible to users in the Analytics Cloud user interface. Example: `"description": "The Amount field contains the opportunity amount."`
isSystemField	Boolean	Indicates whether this field is a system field to be excluded from query results. Example: `"isSystemField": false`
format	String	The display format of the numeric value. Example: `"format": "$#,##0.00"` (Numeric) For more information about valid formats, see Numeric Formats.

Numeric Formats

An example of a typical numeric value is $1,000,000.99, which is represented as $#,##0.00. You are required to specify the precision and scale of the number. The format is specified by using the following symbols:

Symbol	Meaning
0	One digit. Use to add leading or trailing 0s, like `#,###.00` for $56,375.00.
#	Adds zero or one digit
.	Default symbol used as the decimal separator. Use the `decimalSeparator` field to set the decimal separator to a different symbol.
-	Minus sign
,	Grouping separator
$	Currency sign

> 📝 **Note:** The format for numeric values when displayed in the UI defaults to No Format. Existing formatting is removed. Also, you can't override date formats.

> 👁 **Example:** Let's consider an example where you want to override the following object and field attributes that the sfdcDigest transformation extracts from the Opportunity object.

Object/Field	Attribute Changes
Opportunity object	• Change the object label to "Sales Opportunities" • Add an object description
Id field	• Change the field label to "Opportunity Id" • Hide the field from queries
Amount field	• Change the field label to "Opportunity Amount" • Change the format to $#,##0.00
CloseDate field	• Change the field label to "Closing Date"

To override the attributes, you add the Schema section with the override values to sfdcDigest in the dataflow definition file.

```
{
"Extract_Opportunities": {
    "action": "sfdcDigest",
    "parameters": {
        "object": "Opportunity",
        "fields": [
            { "name": "Id" },
            { "name": "Name" },
            { "name": "Amount" },
            { "name": "StageName" },
            { "name": "CloseDate" },
            { "name": "AccountId" },
            { "name": "OwnerId" }
        ]
    },
    "schema": {
        "objects": [
            {
                "label":"Sales Opportunities",
                "description": "These are all sales opportunities.",
                "fields": [
                    {
                        "name": "Id",
                        "label": "Opportunity Id",
                        "isSystemField": true
                    },
                    {
                        "name": "Amount",
                        "label": "Opportunity Amount",
```

```
                "format": "$#,##0.00"
            },
            {
                "name": "CloseDate",
                "label": "Closing Date"
            }
        ]
    }
    ]
    }
},
    "Register_Dataset_Opportunities": {
        "action": "sfdcRegister",
        "parameters": {
            "source": "Extract_Opportunities",
            "alias": "Opportunities",
            "name": "Opportunities"
        }
    }
}
```

> **!** **Important:** Tableau CRM removes Schema sections from all transformations except sfdcDigest when you add a dataflow to a package. If you intend to package a dataflow, we recommend that you specify field attributes in the transformation itself, instead of in a schema section. For example, this computeRelative transformation uses a label attribute to change a field's label.

```
    "CalcAmountAfterdiscount": {
        "action": "computeExpression",
        "parameters": {
            "mergeWithSource": true,
            "source": "getOpps",
            "computedFields": [
                {
                    "name": "DiscountedAmount",
                    "label": "Discounted Amount",
                    "type": "Numeric",
                    "saqlExpression": "Amount * Discount_Percentage__c",
                    "precision": 10,
                    "scale": 2
                }
            ]
        }
    },
```

Edit Nodes in the Dataflow

When you edit a dataflow node, for example to change its name, or add or remove fields, consider that these changes can impact downstream nodes. If you're manually editing nodes in the JSON dataflow definition file, make sure that your changes are reflected in downstream nodes. If you're working in the dataflow editor, Tableau CRM propagates many of your changes to downstream nodes for you.

Renaming Node Names

When you rename a node in the dataflow JSON, you need to manually check that the name is unique and change the name wherever the node is referenced in downstream nodes.

In the dataflow editor, Tableau CRM validates that the new name is unique, and changes the name in downstream nodes for you.

Adding Fields in a Node

When you add a field to a node in the dataflow JSON, consider that you might also need to add the field in these downstream nodes:

- augment: If you want to augment the new field on other data in the dataflow, add the field to the `right_select` parameter in the corresponding augment node.

- sliceDataset: If there is a downstream sliceDataset node with the `mode` parameter set to `select`, add the new field to the `fields` parameter to make sure it isn't dropped.

When you add a field to a node in the dataflow editor, Tableau CRM offers to propagate the field in downstream augment and sliceDataset nodes.
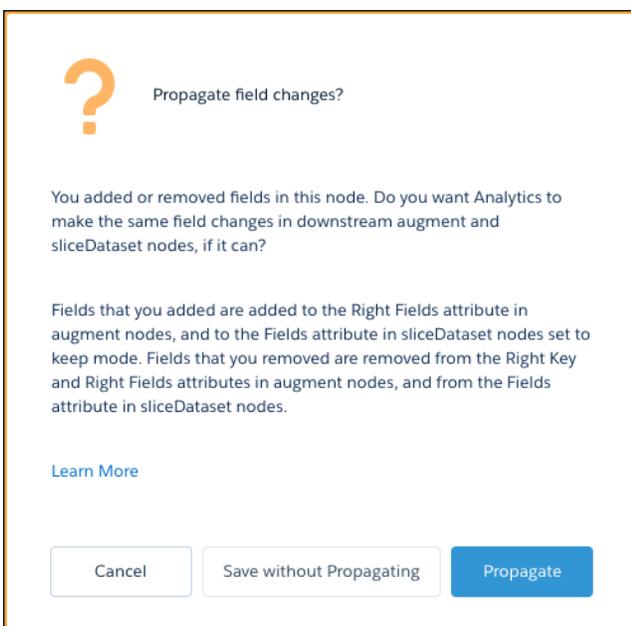
If you choose to propagate your changes, Tableau CRM adds the new field to:

- The Right Fields attribute in downstream augment nodes.
- The Fields attribute in downstream sliceDataset nodes where the Mode attribute is set to Keep.

Removing Fields in a Node

When you remove a field in a node in the dataflow JSON, the dataflow can fail if you don't also remove the field in these places:

- left_key, right_key, and right_select parameters in downstream augment nodes.
- The `fields` parameter in downstream sliceDataset nodes.
- Downstream computeExpression or computeRelative nodes that reference the field.

When you remove a field in a node in the dataflow editor, Tableau CRM offers to propagate the field in downstream augment and sliceDataset nodes. If you choose to propagate your changes, Tableau CRM removes the field from:

- The Left Key, Right Key, and Right Fields attributes in downstream augment nodes.
- The Fields attribute in downstream sliceDataset nodes.

> 📝 **Note:** The propagate option doesn't remove fields from downstream computeExpression or computeRelative nodes that reference it.

You see the propagate option whenever you add or remove fields from these types of nodes:

- sfdcDigest
- digest
- augment
- computeExpression
- computeRelative
- sliceDataset

Configure the Dataflow Through the Definition File

You can configure the dataflow by adding transformations directly to the dataflow definition file.

A *dataflow definition file* is a JSON file that contains transformations that represent the dataflow logic. The dataflow definition file must be saved with UTF-8 encoding.

Before you can configure a dataflow to process external data, you must upload the external data to Tableau CRM.

1. In Tableau CRM, click the gear icon (⚙) and then click **Data Manager**.

2. Click the Dataflows & Recipes tab.

3. Download the existing dataflow definition file by clicking **Download** in the actions menu.
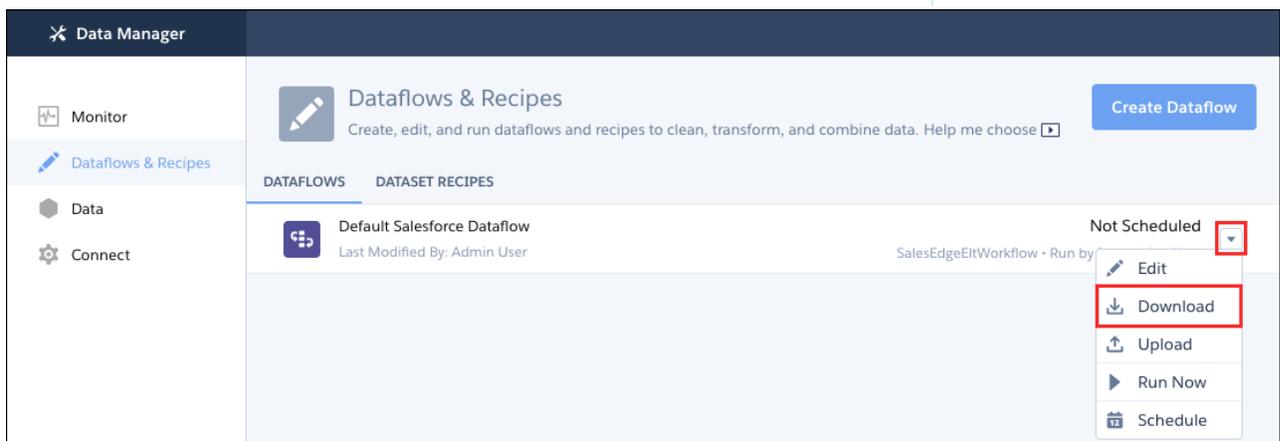
4. Make a backup copy of the existing dataflow definition file before you modify it.

Tableau CRM doesn't retain previous versions of the file. If you make a mistake, you can upload the previous version to roll back your changes.

5. Add each transformation as a node in the dataflow definition file, using a JSON editor.

For example, based on the design in the , you can add the following transformation nodes:

```json
{
    "Extract_Opportunities": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Opportunity",
            "fields": [
                { "name": "Id" },
                { "name": "Name" },
                { "name": "Amount" },
                { "name": "StageName" },
                { "name": "CloseDate" },
                { "name": "AccountId" },
                { "name": "OwnerId" }
            ]
        }
    },
    "Extract_AccountDetails": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Account",
            "fields": [
                { "name": "Id" },
                { "name": "Name" }
            ]
        }
    },
    "Transform_Augment_OpportunitiesWithAccountDetails": {
        "action": "augment",
        "parameters": {
            "left": "Extract_Opportunities",
            "left_key": [ "AccountId" ],
            "relationship": "OpptyAcct",
            "right": "Extract_AccountDetails",
            "right_key": [ "Id" ],
            "right_select": [
                "Name"
            ]
        }
    },
    "Transform_Filter_Opportunities": {
        "action": "filter",
        "parameters": {
            "filter": "StageName:EQ:Closed Won",
            "source": "Transform_Augment_OpportunitiesWithAccountDetails"
        }
    },
    "Register_Dataset_WonOpportunities": {
        "action": "sfdcRegister",
```

964

```
        "parameters": {
            "alias": "WonOpportunities",
            "name": "WonOpportunities",
            "source": "Transform_Filter_Opportunities"
        }
    }
}
```

See Transformations for Tableau CRM Dataflows for more about each transformation and its JSON.

📝 **Note:** The JSON keys and values are case-sensitive. Each bolded key in the example JSON is the node name for a transformation. Each node contains an action value, which identifies the transformation type. The order in which you add the transformations to the dataflow definition file doesn't matter. Tableau CRM determines the order in which to process the transformations by traversing the dataflow to determine the dependencies among them.

🔴 **Important:** Node names must be unique in a dataflow definition file, and can't contain space or tab characters. Consider that node names are **not** treated as case sensitive, so names such as "Extract_Opportunities" and "extract_opportunities" aren't unique in the same definition file.

6. Before you save the dataflow definition file, use a JSON validation tool to verify that the JSON is valid.

An error occurs if you try to upload the dataflow definition file with invalid JSON. You can find JSON validation tools on the internet.

7. Save the dataflow definition file with UTF-8 encoding, and then close the file.

8. In the Dataflow view of the Monitor tab, click **Upload** from the action menu to upload the updated dataflow definition file.

📝 **Note:**
- Uploading the dataflow definition file doesn't affect any running dataflow jobs and doesn't automatically start the dataflow job.
- The maximum length of a dataflow definition file is 1,000,000 characters.

You can now start the dataflow on demand or wait for it to run on the schedule. Users can't query the registered datasets until the dataflow runs.

To create dataflows, you must have Data Sync enabled. Data Sync is enabled by default if you turned on Tableau CRM after the Winter '20 release. If your org's first license was provisioned before the Winter '20 release, you can manually enable Data Sync.

Run Data Sync, Recipes, and Dataflows to Create and Refresh Datasets

Whether you use local Salesforce data or pull data from an external source, you must set up Tableau CRM to load the data, make it available to Tableau CRM, and keep it up to date.

To get the latest data, always run a data sync job to synchronize your data with the source before you run your recipe or dataflow to load data into the dataset. To run these jobs on an ongoing basis, schedule syncs, recipes, and dataflows to run automatically and as frequently as needed. As you plan your strategy for loading and keeping data up to date, consider the limits Salesforce places on your Tableau CRM datasets and jobs.

Run Data Sync to Synchronize Source Data in Tableau CRM

Make sure that the data available to Tableau CRM matches the data in your source, whether its Salesforce or an external data source. You can manually run or schedule each sync.

Run a Recipe

Run a recipe for the first time to create the dataset it defines. Run a recipe again to update the dataset with the latest synced data. You can run a recipe manually or on a schedule.

Run a Dataflow

Like recipes, you can run dataflows manually or schedule them to run automatically at regular intervals to create your datasets and keep the data up to date. You can also stop a dataflow while it's running.

Schedule Smarter with Priority Scheduling

Priority scheduling for recipes and dataflows automatically manages your run queue. It prioritizes smaller and faster runs while ensuring that larger and longer runs are completed on time. Priority is automatically calculated based on factors such as historic runtime, dataset input size, and CSV file size. Priority scheduling is most helpful to smooth out occasional queue-time spikes. If you never or frequently see long queue times, then priority scheduling isn't as helpful. Activate the feature in advance to manage your queue, not during a problem when your queue is already overloaded. This feature doesn't increase your maximum number of concurrent runs.

SEE ALSO:

[Connect and Sync Your Data to Tableau CRM](#)

[Prepare and Load Data into Datasets with Recipes and Dataflows](#)

[Tableau CRM Limits](#)

Run Data Sync to Synchronize Source Data in Tableau CRM

Make sure that the data available to Tableau CRM matches the data in your source, whether its Salesforce or an external data source. You can manually run or schedule each sync.

Run Data Sync Manually

Run data sync manually the first time to make the data available to build recipes and dataflows. Schedule subsequent syncs to regularly update the data used by recipes and dataflows.

Schedule Data Sync to Run Automatically

Schedule data syncs to run regularly. You schedule data sync for each connection, where all objects under the connection sync at the specified time, and not individual objects. To sync objects from the same data source on different intervals, create multiple connections to the data source, and set a unique schedule for each connection. To ensure that the latest source data is loaded into datasets, schedule data syncs to run before the corresponding recipes and dataflows.

Monitor a Data Sync Job

Monitor the progress of syncs in Data Manager.

SEE ALSO:

[Connect and Sync Your Data to Tableau CRM](#)

Run Data Sync Manually

Run data sync manually the first time to make the data available to build recipes and dataflows. Schedule subsequent syncs to regularly update the data used by recipes and dataflows.

📝 **Note:** Data sync jobs don't count towards your daily dataflow and recipe run limit.

1. In Tableau CRM, click **Data Manager** in the left pane to open Data Manager.

2. To display the list of objects enabled for sync, group by connection, click the **Connect** tab.

3. To show and hide the list of objects, click the arrow to the left of the connection name.

   If you can't see the Connect tab, you must enable data sync in your org. See Enable Data Sync and Connections.

4. To run sync for all objects in a connection, click ⊡ to the right of the connection name, and select **Run Now**. The connection is queued to sync.

5. To run sync for a single remote object, click ⊡ to the right of the object name, and select **Run Data Sync**.

6. To run sync for a single local Salesforce object, first select the connection mode you want to use. Click ⊡ to the right of the local object name, and select **Edit Connection Mode**.

7. Choose from the following:

   a. **Incremental Sync** updates only rows that changed since the last sync. It's the fastest option.

   b. **Periodic Full Sync** updates rows incrementally and periodically overwrites all rows with records in the Salesforce object.

   c. **Full Sync** updates all rows with records in the Salesforce object.

8. Click **Save**.

9. Click ⊡ again, and select **Run Data Sync**. The Monitor tab of Data Manager opens so you can see the status of your sync.

SEE ALSO:

Monitor a Data Sync Job

Tableau CRM Limits

Schedule Data Sync to Run Automatically

Schedule data syncs to run regularly. You schedule data sync for each connection, where all objects under the connection sync at the specified time, and not individual objects. To sync objects from the same data source on different intervals, create multiple connections to the data source, and set a unique schedule for each connection. To ensure that the latest source data is loaded into datasets, schedule data syncs to run before the corresponding recipes and dataflows.

📝 **Note:** Data sync jobs don't count towards your daily dataflow and recipe run limit.

1. In Tableau CRM, click **Data Manager** in the left pane to open Data Manager.

2. Click the **Connect** tab. The Connect tab displays a list of objects enabled for sync, grouped by connection.

   If you can't see the Connect tab, you must enable data sync in your org. See Enable Data Sync and Connections.

3. Click ▼ to the right of the connection that you want to schedule, and select **Schedule**. The scheduler appears.

4. Select the time to run the recipe. You can schedule it to run by minute, hour, week, or month. Tableau CRM runs the data sync according to the time zone of the user who set the schedule.

   💡 **Tip:** If you don't have a Tableau CRM Plus license but want to schedule a run every 15, 20, or 30 minutes, contact Salesforce Customer Support to request subhour scheduling. This feature isn't available in sandbox orgs.

5. If you schedule the recipe to run by minute or hour, select **Stop queuing at a specific time** to stop the recipe from running after a certain time. For example, to restrict runs to office hours, set a job to start at 8:00 am, run every hour, and stop at 6:00 pm.

6. Click **Save**. Schedule information for the connection displays in the connection header.

💡 **Tip:** To ensure that sync completes before the recipe or dataflow starts, use event-based scheduling for the recipe or dataflow.

📝 **Note:** Sandbox and developer edition org schedules for data sync, dataflow, and recipes are removed 30 days after the last save. Users subscribed to its notifications receive an email notifying them when a schedule is removed. Set the schedule again anytime.

SEE ALSO:

Monitor a Data Sync Job

Tableau CRM Limits

Monitor a Data Sync Job

Monitor the progress of syncs in Data Manager.

To monitor syncs, perform the following steps.

1. In Tableau CRM, click **Data Manager** in the left pane to open Data Manager.
    Data Manager opens to the Monitor tab with the Jobs view selected.

2. In the data manager, click the **Monitor** tab.
    The Jobs subtab displays information about your sync jobs.

SEE ALSO:

Troubleshoot Canceled Data Sync, Recipe, and Dataflow Runs Due to Overlapping Schedules

Run a Recipe

Run a recipe for the first time to create the dataset it defines. Run a recipe again to update the dataset with the latest synced data. You can run a recipe manually or on a schedule.

Confirm that sync jobs for connections and/or objects used to create the recipe are complete before the recipe runs. That step ensures that the recipe uses up-to-date data. You can view the status of a recipe job in the Monitor tab in Data Manager.

For a production org with the Tableau CRM Plus platform license, Tableau CRM runs up to two recipes concurrently when multiple jobs overlap. If more than two jobs overlap, Tableau CRM puts the remaining jobs in queue. Tableau CRM runs one job at a time in production orgs with the Tableau CRM Growth license and sandbox orgs.

Run a Recipe Manually

Run a recipe for the first time to create the datasets it defines. Run it again to update the dataset with the latest synced data.

Schedule a Recipe to Run Automatically

You can schedule a recipe to run after an event or at a specific time. For example, you can use event-based scheduling to run a recipe after the sync runs to ensure that datasets include up-to-the-minute data. Use time-based scheduling to ensure that fresh data is available by a particular time or to run the job during non-business hours. You can schedule a recipe to run every 15 minutes, hourly, weekly, monthly, or on specific days of the week or dates.

Monitor a Recipe Job

The Jobs subtab on the Monitor tab shows the status of data sync, dataflow, and recipe jobs. You can view error messages about a job, view the run-time details about every transformation that is processed, and monitor the number of jobs remaining from your org's 24-hour limit.

Set Recipe Notifications

Set recipe notifications to receive an email notification when a recipe job finishes. You can receive notifications for warnings, failures, and completions. You can also set an elapsed time notification to notify you when a recipe is still running after a specified length of time.

Determine Whether a Target Dataset in a Recipe Is Created and Up to Date

After you save a recipe, you must run it to create or update the target dataset based on the latest logic. To help you monitor which recipes have been updated and run to update their target datasets, review their statuses in the Data Manager

Run a Recipe Manually

Run a recipe for the first time to create the datasets it defines. Run it again to update the dataset with the latest synced data.

1. In Tableau CRM, click **Data Manager** in the left pane to open Data Manager.
   Data Manager opens to the Monitor tab with the Jobs view selected.

2. Click the **Dataflows & Recipes** tab.

3. Select **Recipes** to view all your recipes.

4. Click the triangle ▽ to the right of the recipe name you want to run, and select **Run Now** or **Run Last Updated**.

Note: To secure access to sensitive data in datasets and connected objects, you also need the Edit Analytics Dataflows user permission when working with recipes. With that permission, you can:

- Use connected objects in a recipe.

- Edit security predicates in existing recipes, which allows you to grant others, including yourself, access to more rows in the dataset.

- Run or schedule recipes that pull data from existing datasets with security predicates.

- Run or schedule recipes that pull data from existing datasets with security predicates.

Users with the Edit Dataset Recipes permission but without the Edit Analytics Dataflows permission can:

- Create a recipe that outputs datasets to their private app only.

- Run and schedule recipes that write to datasets that don't have security predicates.

- Use a recipe to extract data from a dataset with a security predicate, but can't change the predicate on the resulting dataset.

SEE ALSO:

Monitor a Recipe Job

Tableau CRM Limits

Schedule a Recipe to Run Automatically

You can schedule a recipe to run after an event or at a specific time. For example, you can use event-based scheduling to run a recipe after the sync runs to ensure that datasets include up-to-the-minute data. Use time-based scheduling to ensure that fresh data is available by a particular time or to run the job during non-business hours. You can schedule a recipe to run every 15 minutes, hourly, weekly, monthly, or on specific days of the week or dates.

1. In Tableau CRM, click **Data Manager** in the left pane to open Data Manager.
   Data Manager opens to the Monitor tab with the Jobs view selected.

2. Click the **Dataflows & Recipes** tab.

3. To view all your recipes, select **Recipes**.

4. Click the triangle ▾ to the right of the recipe name you want to run, and select **Schedule**.
   The scheduler appears.

5. To schedule the recipe to run after the data sync finishes, choose **Event-based** under Schedule Mode. For example, if the recipe pulls data from the Salesforce Accounts and Opportunities objects, the recipe runs after both objects sync.

   📝 Note: Event-based schedules are triggered after syncs on objects from the local Salesforce org, not remote Salesforce orgs. When you use data from multiple local connections in a recipe or dataflow, event-based scheduling will run only after all local connections are synced. For example, if one local connection is scheduled to sync hourly and another syncs daily, an event-based scheduled recipe using data from both runs one time daily.

6. To run the recipe at a specific time, perform the following tasks.

   a. Under Schedule Mode, select **Time-based**.

   b. Select the time to run the recipe. You can schedule it to run by minute, hour, week, or month.
      Tableau CRM runs the recipe according to the time zone of the user who set the schedule.

   c. If you schedule the recipe to run by minute or hour, select **Stop queuing at a specific time** to stop the recipe from running after a certain time. For example, to restrict runs to office hours, set a job to start at 8:00 am, run every hour, and stop at 6:00 pm.

7. Click **Save**.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To run a recipe:
- Edit Dataset Recipes OR Edit Analytics Dataflows

📝 **Note:** Sandbox and developer edition org schedules for data sync, dataflow, and recipes are removed 30 days after the last save. Users subscribed to its notifications receive an email notifying them when a schedule is removed. Set the schedule again anytime.

📝 **Note:** To secure access to sensitive data in datasets and connected objects, you also need the Edit Analytics Dataflows user permission when working with recipes. With that permission, you can:

- Use connected objects in a recipe.
- Edit security predicates in existing recipes, which allows you to grant others, including yourself, access to more rows in the dataset.
- Run or schedule recipes that pull data from existing datasets with security predicates.
- Run or schedule recipes that pull data from existing datasets with security predicates.

Users with the Edit Dataset Recipes permission but without the Edit Analytics Dataflows permission can:

- Create a recipe that outputs datasets to their private app only.
- Run and schedule recipes that write to datasets that don't have security predicates.
- Use a recipe to extract data from a dataset with a security predicate, but can't change the predicate on the resulting dataset.

SEE ALSO:

Monitor a Recipe Job

The Jobs subtab on the Monitor tab shows the status of data sync, dataflow, and recipe jobs. You can view error messages about a job, view the run-time details about every transformation that is processed, and monitor the number of jobs remaining from your org's 24-hour limit.

1. In Tableau CRM, click **Data Manager** in the left pane to open Data Manager.
   Data Manager opens to the Monitor tab with the Jobs view selected.

2. In the data manager, click the **Monitor** tab.
   The Jobs subtab displays information about all jobs.

3. Monitor the flow indicator to see how many dataflow and recipe jobs can be run in a rolling 24-hour period.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer Edition**.

USER PERMISSIONS

To monitor a recipe:
- Edit Analytics Dataflows OR Edit Dataset Recipes

> ✎ **Note:** Runs less than 2 minutes in duration, and data sync, don't count toward this limit. To request a higher 24-hour limit, contact Salesforce Support.

4.
Click 🔄 to see the latest status of a job.

Each job can have one of these statuses.

Status	Description
Running	The job is running
Failed	The job failed.
Successful	The job completed successfully.
Warning	The job completed successfully, but some rows failed.

5. If the job fails or completes with warning, expand the job node to get more details.

6. If there's a problem with the recipe logic, edit the recipe and then run it again.

SEE ALSO:

Set Recipe Notifications

Set Recipe Notifications

Set recipe notifications to receive an email notification when a recipe job finishes. You can receive notifications for warnings, failures, and completions. You can also set an elapsed time notification to notify you when a recipe is still running after a specified length of time.

1. In Tableau CRM, click **Data Manager** in the left pane to open Data Manager.
   Data Manager opens to the Monitor tab with the Jobs view selected.

2. Click the **Dataflows & Recipes** tab.

3. Select **Recipes** to view all your recipes.

4. Click the triangle ▾ to the right of the recipe name, and select **Notifications**.
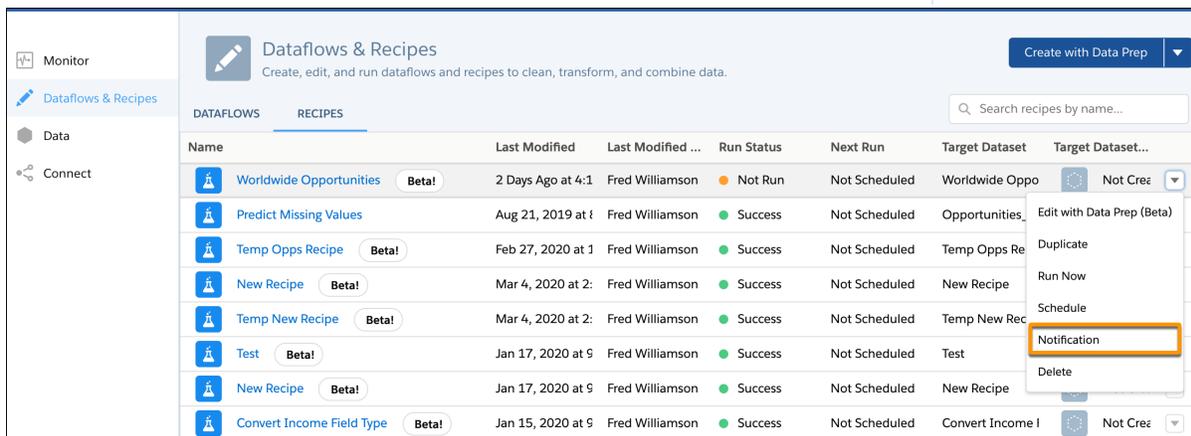
5. In the Notify Me About picklist, select what you want to be notified about. You can choose warnings, failures, or all events.

6. To receive an elapsed time notification, select **Elapsed time notification**, and then select the number of hours and minutes.

> **Note:** By default, email notifications are sent only to the person setting the notifications. Other users with the Edit Analytics Dataflows or Edit Dataset Recipes permission can set up their own notifications.

Determine Whether a Target Dataset in a Recipe Is Created and Up to Date

After you save a recipe, you must run it to create or update the target dataset based on the latest logic. To help you monitor which recipes have been updated and run to update their target datasets, review their statuses in the Data Manager

If you save a recipe, you can view the status of the recipe and its target datasets on the Recipes subtab in the Dataflows & Recipes tab of Data Manager.

A recipe's icon and status indicate its current state.

Icon	Status	Explanation
	Pushed	New or updated recipe that has been run.
	Saved	New or updated recipe that has been saved but not run.

A target dataset's icon and status indicate its current state.

Icon	Status	Explanation
	Up to Date	Dataset has been created and is up to date.
	Not Up to Date	Dataset has been created, but is not currently up to date. Its recipe has been updated but not run, or is queued to run.
	Not Created	Dataset hasn't been created yet. Its recipe has been created but never run, or is queued to run for the first time.

SEE ALSO:

Run a Recipe Manually

Monitor a Recipe Job

Run a Dataflow

Like recipes, you can run dataflows manually or schedule them to run automatically at regular intervals to create your datasets and keep the data up to date. You can also stop a dataflow while it's running.

Run a Dataflow Manually

Run a dataflow for the first time to create the datasets it defines. Run it again to update the dataset with the latest synced data.

Schedule a Dataflow to Run Automatically

You can schedule a dataflow to run after an event, such as when a sync job completes. Or you can schedule it to run to up to every 15 minutes, hourly, weekly, monthly, or on specific days of the week or dates.

Monitor a Dataflow Job

The Dataflows subtab on the Monitor tab shows status, start time, and duration of the last 10 dataflow jobs. It retains the last 7 days of job history. You can view error messages about each job, view the run-time details about every transformation that is processed, monitor the number of jobs remaining from your org's 24-hour limit, and download error logs.

Restore a Previous Version of a Dataflow

Tableau CRM uses version history to create copies of dataflows when you edit them so you can restore a previous version.

Set Dataflow Notifications

Set dataflow notifications to receive an email notification when a dataflow job finishes. You can be notified only when there are warnings, only when the dataflow fails, or every time the dataflow finishes. You can also set an elapsed time notification to notify you when a dataflow is still running after a specified length of time.

Run a Dataflow Manually

Run a dataflow for the first time to create the datasets it defines. Run it again to update the dataset with the latest synced data.

1. In Tableau CRM, click **Data Manager** in the left pane to open Data Manager.
   Data Manager opens to the Monitor tab with the Jobs view selected.

2. Click the **Dataflows & Recipes** tab.
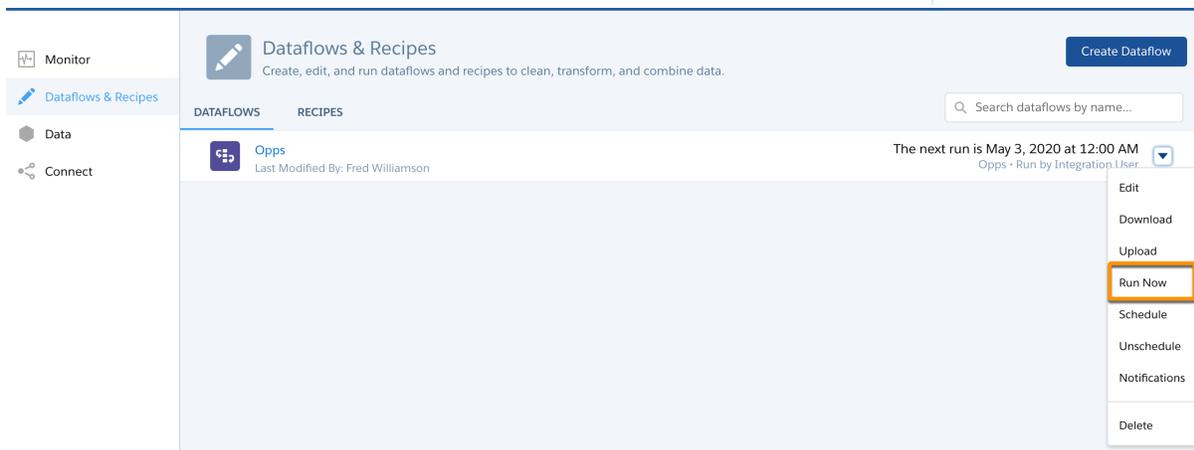
3. Click the triangle ▾ to the right of the dataflow you want to run, and select **Run Now**.
   The dataflow job is added to the job queue. The **Start** action is grayed out while the dataflow job runs.

4. After the job completes, Analytics Cloud sends an email notification to the user who created the dataflow.

   The email notification indicates whether the job completed successfully. It also shows job details like start time, end time, duration, and number of processed rows. If the job failed, the notification shows the reason for the failure.

   📝 Note: If the dataflow creator is not an active user, the notification is sent to the user who last modified the dataflow schedule or definition file.

5. To stop a dataflow job that is currently running, click ✖ next to the job status.
   If you click **Start** to restart a stopped dataflow, the job starts over—the dataflow job does not resume from the point at which it was stopped.

> **Note:** A dataflow automatically restarts if it is forcibly terminated by an external process like patches to the OS or Maestro, specifically bifrost.

You can monitor the dataflow job on the Monitor tab to determine when the dataflow completes. After the dataflow completes successfully, refresh the Tableau CRM home page to view the registered datasets.

SEE ALSO:

Monitor a Dataflow Job

Tableau CRM Limits

Schedule a Dataflow to Run Automatically

You can schedule a dataflow to run after an event, such as when a sync job completes. Or you can schedule it to run to up to every 15 minutes, hourly, weekly, monthly, or on specific days of the week or dates.

Use event-based scheduling to run a dataflow after the sync runs to ensure that datasets include up-to-the-minute data. Use time-based scheduling to ensure that fresh data is available by a particular time, or to run the job during non-business hours.

1. In Analytics, click **Data Manager** in the left pane to open Data Manager.
   Data Manager opens to the Monitor tab with the Jobs view selected.

2. Click the **Dataflows & Recipes** tab.

3. Click the triangle ▾ to the right of the dataflow you want to run, and select **Schedule**.
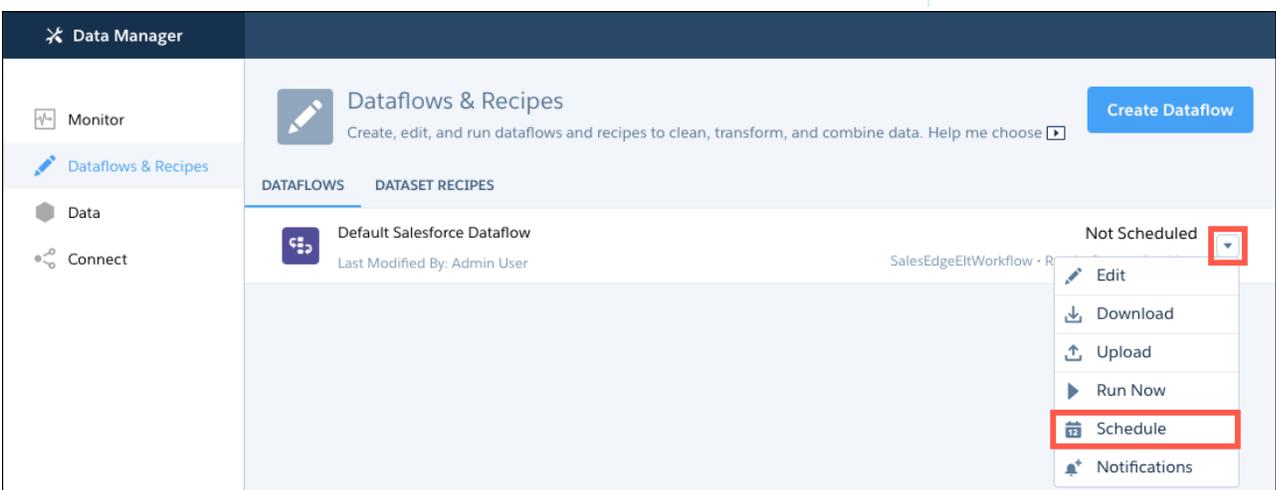
The scheduler appears.

4. Under Schedule Mode, select **Time-based** or **Event-based**.

   To schedule the dataflow to run at a specific time, choose **Time-based**. If you select this option, skip to Step 5 and set the time.

To schedule the dataflow to run after the data sync finishes, choose **Event-based**. For example, if the dataflow pulls data from the Salesforce Accounts and Opportunities objects, the dataflow runs after both objects sync. After you choose **Event-based**, click **Save**.

> 📝 Note: Event-based schedules are triggered only by completed data sync of local objects in the dataflow's sfdcDigest nodes through a scheduled run, manual run, or another successful dataflow run. When you use data from multiple local connections in a recipe or dataflow, event-based scheduling will run only after all local connections are synced. For example, if one local connection is scheduled to sync hourly and another syncs daily, an event-based scheduled recipe using data from both runs one time daily.

5. If you select time-based scheduling, select the time to run the dataflow. You can schedule the dataflow to run by minute, hour, week, or month. Tableau CRM runs the dataflow according to the time zone of the user who set the schedule.

> 💡 Tip: If you don't have a Tableau CRM Plus license but want to schedule a run every 15, 20, or 30 minutes, contact Salesforce Customer Support to request subhour scheduling. This feature isn't available in sandbox orgs.

6. If you schedule the dataflow to run by minute or hour, select **Stop queuing at a specific time** to stop the dataflow from running after a certain time to restrict runs to office hours. For example, set a job to start at 8:00 am, run every hour, and stop at 6:00 pm.

7. Click **Save**.

> 📝 Note: Sandbox and developer edition org schedules for data sync, dataflow, and recipes are removed 30 days after the last save. Users subscribed to its notifications receive an email notifying them when a schedule is removed. Set the schedule again anytime.

To cancel the scheduled dataflow, select **Unschedule** from the dataflow's menu.
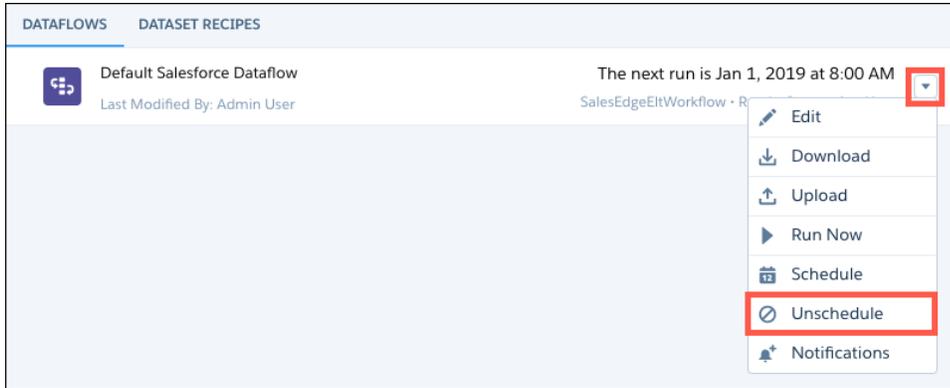
SEE ALSO:

Schedule, Run, and Monitor Data Sync

Monitor a Dataflow Job

Tableau CRM Limits

Monitor a Dataflow Job

The Dataflows subtab on the Monitor tab shows status, start time, and duration of the last 10 dataflow jobs. It retains the last 7 days of job history. You can view error messages about each job, view the run-time details about every transformation that is processed, monitor the number of jobs remaining from your org's 24-hour limit, and download error logs.

1. On the Monitor tab of Data Manager, click **Dataflows** (1).

> **Note:** Duration is calculated as the sum of the job queue time and job run time.

2. Monitor the flow indicator to see how many dataflow and recipe jobs can be run in a rolling 24-hour period.

> **Note:** Runs less than 2 minutes in duration, and data sync, don't count toward this limit. To request a higher 24-hour limit, contact Salesforce Support.

3. Click [icon] to see the latest status of a job.

   Each job can have one of these statuses.

Status	Description
Running	The job is running.
Failed	The job failed.
Successful	The job completed successfully.
Warning	The job completed successfully, but some rows failed.

4. If the dataflow job fails, expand the job node (2) to view the run-time details for every transformation that was processed.

5. If an error log is available for a node, click the download log button (3) to download a CSV file containing the failed rows.

   > **Note:** Error logs display the data from rows that have failed to load. To maintain data security and prevent unauthorized access to this data, only the dataflow owner or users with the View All Data permission can download an error log.

6. If there's a problem with the dataflow logic, edit the dataflow and then run it again.

SEE ALSO:

Set Dataflow Notifications

Troubleshoot Canceled Data Sync, Recipe, and Dataflow Runs Due to Overlapping Schedules

Restore a Previous Version of a Dataflow

Tableau CRM uses version history to create copies of dataflows when you edit them so you can restore a previous version.

1. In Tableau CRM, click **Data Manager** in the left pane. Data Manager opens to the Monitor tab with the Jobs view selected.

2. Click the **Dataflows & Recipes** tab.

3.
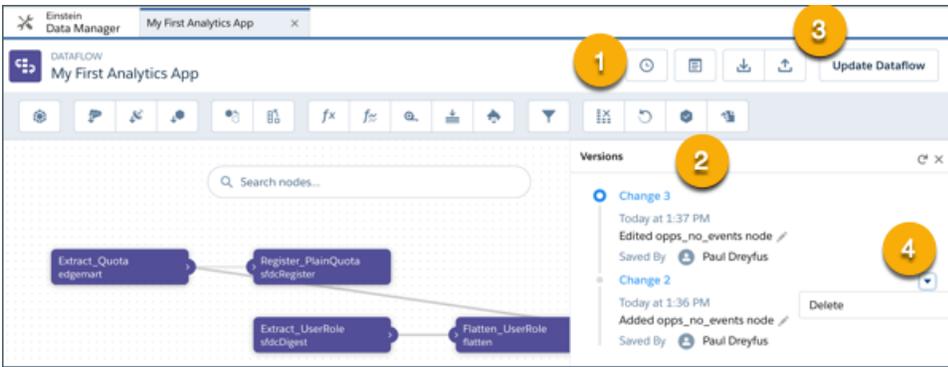
Open a dataflow, then click the history icon (1). All versions of the dataflow appear in the **Versions** panel (2), with the live version on top.

4. Revert to a previous version by clicking its name, such as *Change 2*, then click **Update Dataflow** (3).

5. To delete a previous version, click the dropdown (4) and click **Delete**.

For details, see Backup and Restore Previous Versions of Analytics Assets with History API.

Set Dataflow Notifications

Set dataflow notifications to receive an email notification when a dataflow job finishes. You can be notified only when there are warnings, only when the dataflow fails, or every time the dataflow finishes. You can also set an elapsed time notification to notify you when a dataflow is still running after a specified length of time.

1. In Analytics Cloud, click the gear icon (⚙) and then click **Data Manager**.
The data manager opens on the Monitor tab, with the Jobs tab open by default.

2. Click the **Dataflows & Recipes** tab.

3. On the right of the dataflow, click ▼ and select **Notifications**.
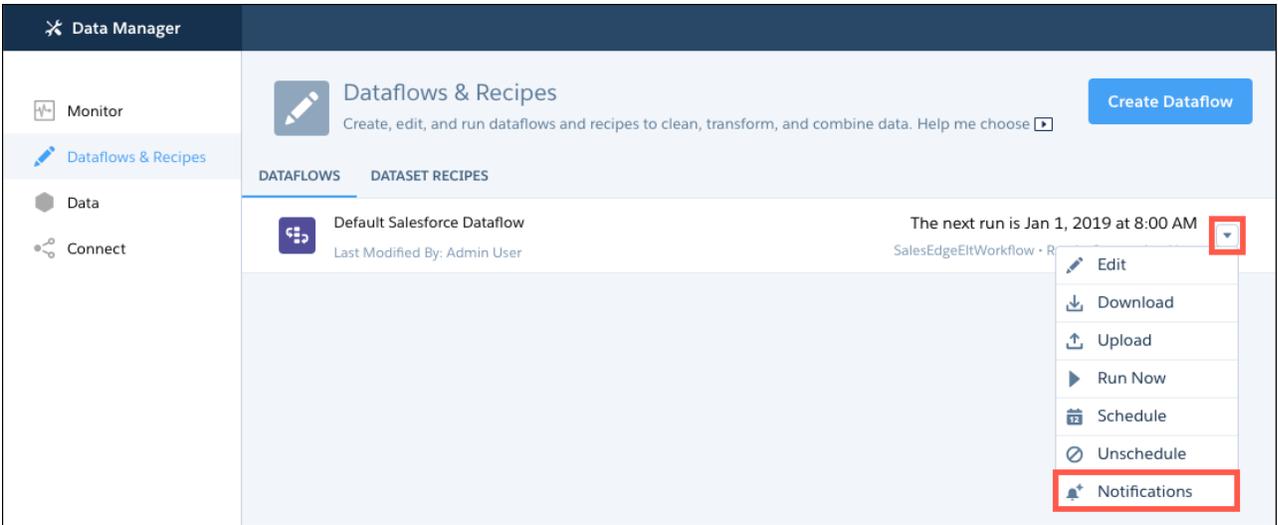
4. From the Notify Me About picklist, select what you want to be notified about.
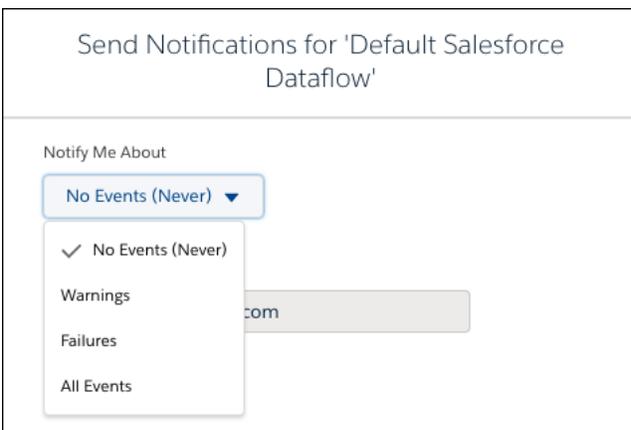
5. To receive an elapsed time notification, select **All Events** from the Notify Me About options. Then select **Elapsed time notification** and select the number of hours and minutes.
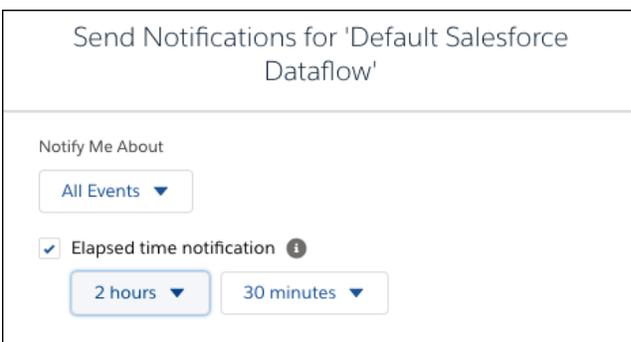
> 📝 **Note:** By default, email notifications are sent only to the person setting the notifications. Other people with the Edit Analytics Dataflows permission can subscribe to a dataflow by setting notifications.

Schedule Smarter with Priority Scheduling

Priority scheduling for recipes and dataflows automatically manages your run queue. It prioritizes smaller and faster runs while ensuring that larger and longer runs are completed on time. Priority is automatically calculated based on factors such as historic runtime, dataset input size, and CSV file size. Priority scheduling is most helpful to smooth out occasional queue-time spikes. If you never or frequently see long queue times, then priority scheduling isn't as helpful. Activate the feature in advance to manage your queue, not during a problem when your queue is already overloaded. This feature doesn't increase your maximum number of concurrent runs.

For example, you schedule three recipes to run at the same time. One is larger with more rows and runs for 30 minutes. The other two are smaller with fewer rows and run for three minutes. The larger recipe's run request arrives a millisecond before the shorter requests. Without priority scheduling, requested runs are queued first in, first out, so the smaller runs wait 30 minutes until the larger run finishes. With priority scheduling, the smaller recipes execute first because of their shorter run time, followed by the larger run. If higher-priority runs continue to queue in front of the longer run, the queue temporarily switches to first in, first out an hour after the scheduled request.

1. From Setup, enter `Analytics` in the Quick Find box, then select **Settings**.

2. Select **Use priority scheduling for recipe and dataflow requests**.

3. Click **Save**.

You can disable priority scheduling at any time to return to a first in, first out queue.

SEE ALSO:

[Run a Recipe Manually](#)

[Schedule a Recipe to Run Automatically](#)

[Run a Dataflow Manually](#)

[Schedule a Dataflow to Run Automatically](#)

Other Ways to Integrate Data

Dataflows and recipes can integrate most data from internal to external data sources. However, Tableau CRM provides additional ways to integrate data that are more effective for specific use cases. For example, if you don't have to transform data before loading it into a dataset, you can import data directly from CSV or Excel files. If you need to transform it, you can upload the file to create an intermediate dataset, and then use a recipe or dataflow to change that data before loading the results into another dataset. Or, to analyze Salesforce report data over time, you can create a dataset and dashboard that trends the report with just a few clicks. With trending, Tableau CRM creates a dataset based on snapshots of your Salesforce report data.

[Create Datasets from Uploaded CSV Files](#)
Use the upload user interface to create a single dataset based on external .csv data. To refresh the data, you can overwrite the data in the dataset by uploading a new .csv file.

Create Datasets Programmatically from CSV Files Using the External Data API

You can use the External Data API to create a single dataset based on external data in a .csv file. You can also use the API to edit the dataset by uploading a new .csv file. When you edit the dataset, you can choose to overwrite all records, append records, update records, or delete records.

Create Datasets Directly from Microsoft Excel Spreadsheets

The Salesforce Tableau CRM Connector for Excel makes it easy to import data from Microsoft Excel 2013 to Analytics Cloud.

Create Datasets and Trending Dashboards from Salesforce Reports

By creating trending Tableau CRM dashboards that track key metrics over time, you can make sense of your growing, ever-changing datasets. Trending data in Tableau CRM helps you discover hidden insights and allows you to share them with colleagues. When you create a trending dashboard based on a Salesforce report, Tableau CRM generates a dataset and a dashboard based on the underlying Salesforce data.

SEE ALSO:

Considerations Before Integrating Data into Datasets

Create Datasets from Uploaded CSV Files

Use the upload user interface to create a single dataset based on external .csv data. To refresh the data, you can overwrite the data in the dataset by uploading a new .csv file.

When Analytics Cloud loads any data into a dataset, it also adds metadata about each column of data. For example, metadata can include the field type, precision, scale, and default value.

When you upload a .csv file, Analytics Cloud infers metadata about each column in the file. Tableau CRM previews all inferred settings. If needed, you can change the metadata in the user interface before uploading the data.

> 📝 **Note:** You can also specify the metadata in a metadata file. A *metadata file* is a JSON file that describes the structure of an external data file. After you create a dataset based on an external data file, you can edit the dataset to apply a new metadata file. This enables you to change the metadata attributes of each column. For more information about the metadata file, see the *Analytics External Data API Developer Guide*.

Tableau CRM temporarily stores the uploaded CSV and metadata files for processing only. After a dataset is created, Tableau CRM purges the files.

Upload a CSV File to Create a Dataset

You can upload external data in a .csv format through the user interface. When you upload a .csv file, Tableau CRM infers the metadata about each column in the .csv file. Metadata describes the structure of the data in the file, like the data type, precision, and scale. If you upload a .csv from the user interface, Analytics Cloud automatically generates the metadata, which you can preview and change.

Rules for Automatic Generation of a Metadata File

When you upload a CSV file from the user interface, Analytics Cloud automatically generates the metadata file as long as the CSV file meets certain requirements.

Monitor an External Data Upload

When you upload an external data file, Analytics Cloud kicks off a job that uploads the data into the specified dataset. You can use the Monitor tab of the data manager to monitor and troubleshoot the upload job.

When you upload .csv data to create or update a dataset, you can select the time zone of the data if it's not GMT.

SEE ALSO:

Upload a CSV File to Create a Dataset

You can upload external data in a .csv format through the user interface. When you upload a .csv file, Tableau CRM infers the metadata about each column in the .csv file. Metadata describes the structure of the data in the file, like the data type, precision, and scale. If you upload a .csv from the user interface, Analytics Cloud automatically generates the metadata, which you can preview and change.

Before uploading a .csv file:

- Review the format requirements, like date formats, in the Analytics Cloud External Data Format Reference.

- Ensure that the column names in the external data file don't conflict with the generated date column names. For example, if you load a CSV with column Create_Date, Tableau CRM generates the Create_Date_Year column in the dataset. If the CSV also has a column named Create_Date_Year, Tableau CRM throws an error because the names conflict.

When you upload the file, Tableau CRM temporarily stores it for processing only. After the dataset is created, Tableau CRM purges the file. If you want to use the file again later, keep a copy.

1. On the Tableau CRM Studio home tab or an app page, click **Create** > **Dataset**, and select **CSV File**.

2. Click **Select a file or drag it here**, then select the file and click **Open**.

3. Click **Next**.

4. In the Dataset Name field, enter a name for the dataset.

   By default, Tableau CRM uses the file name as the dataset name. The name can't exceed 80 characters.

5. Select the app where the dataset will be created.

   By default, Tableau CRM selects your My Private App. To change an app, click the cross on it and select a different one.

6. In the File Properties Detected box, check that Tableau CRM has correctly identified the properties of your file.

   Usually, Tableau CRM correctly identifies your file properties. If it doesn't, your data may not load correctly and you will see unexpected results when you preview the data on the next screen. To edit the file properties, click [▼] and select **Edit**.

   📝 Note: Tableau CRM also generates a metadata file for your data. This file contains properties such as field labels, field types, and settings. You can view and change these schema properties when you preview the data on the next screen. However, if you want to download the file first, or replace it with your own file, click [▼] in the Data Schema File field.

7. Click **Next**.

   The Edit Field Attributes screen appears. Here, you can preview the data, and view or edit the attributes for each field.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To upload external data:
- Upload External Data to Analytics

8. To view or change a field's attributes, either click the field in the list on the left, or click the field's column.

   Field attributes appear in a panel on the right. The field attributes that you see are determined by the field type.

   🛑 **Important:** Tableau CRM detects the format for date fields based on a sample of values. If the sample contains values with unsupported formats or a mixture of formats, Tableau CRM sets the field type to Text. If you change the date format that Tableau CRM detects, rows with a different format will fail.

   Consider this example data.

Row	SIC Code	SIC Description	Last Updated
1	1110	Barley growing	1/10/17
2	1120	Rice growing	11/14/17
3	1130	Alliaceous vegetable growing	1/1/17

   Tableau CRM detects the date format for the Last Updated field as $M/d/yy$. This format displays months and days below 10 without leading zeros, and years as 2 digits, as in 1/1/17. If you change the format to $MM/dd/yy$, rows 1 and 3 will fail because Tableau CRM expects the month and day parts of the date values to have 2 digits.

9. When you finish reviewing or editing field attributes, click **Upload File**.

   Tableau CRM uploads the data, prepares and creates the dataset, and shows you progress as it happens.
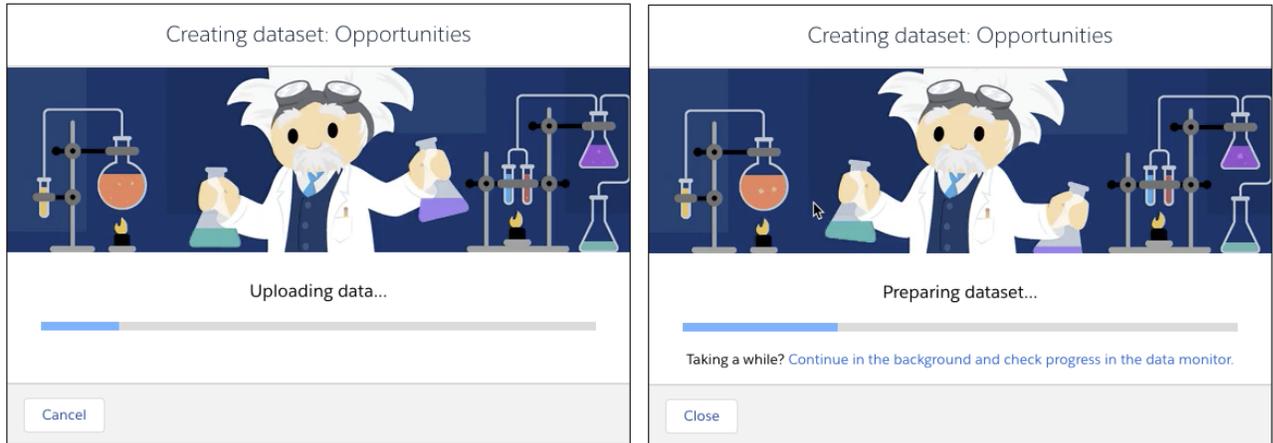
10. Choose one of these options while Tableau CRM creates the dataset.

   a. To cancel the process and stop dataset creation, click **Cancel** on the progress dialog.

   This option is available only when the data is uploading.

   b. To close the progress dialog and leave the process running in the background, click **Close**.

   c. To close the progress dialog but continue monitoring progress on the Monitor tab of Data Manager, click the **Continue in the background and check progress in the data monitor** link.

   d. Do nothing. When the dataset is created, you're taken to the dataset's edit page, where you can explore the data in a lens, prepare it in a recipe, or create a story in Einstein Discovery.
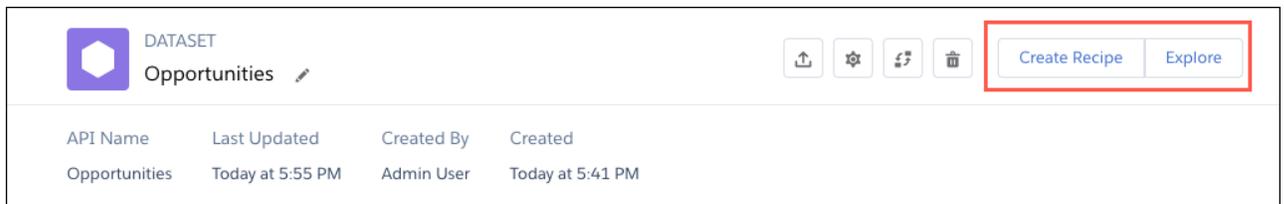
Note: You can also use the External Data API to programmatically upload .csv files. Use the API to take advantage of more features, like performing incremental extracts and performing append, delete, and upsert operations.

SEE ALSO:
   Handle Date Values

Rules for Automatic Generation of a Metadata File

When you upload a CSV file from the user interface, Analytics Cloud automatically generates the metadata file as long as the CSV file meets certain requirements.

To enable Analytics Cloud to generate the metadata file, a CSV file must meet the following requirements.

- The file type must be .csv, not .gz or .zip.
- The file must contain one row for the column header and at least one record.

- The CSV file must meet all Analytics Cloud requirements as mentioned in the Analytics Cloud External Data Format Reference.

Analytics Cloud generates the metadata attributes for each CSV column based on the first 100 rows in the CSV file. Analytics Cloud uses the following rules to convert the CSV column names to field labels.

- Replaces special characters and spaces with underscores. For example, "Stage Name" becomes "Stage_Name."
- Replaces consecutive underscores with one underscore, except when column name ends with "__c." For example, "stage*&name" becomes "stage_name."
- Prefixes the field label with "X" when the first character of the column name is numeric. For example, "30Day" becomes "X30Day."
- Replaces the field name with "Column" + column number when all characters in the column name are not alphanumeric. For example, the fourth column name "*&^*(&*(%" becomes "Column4."
- Deletes underscores at the beginning and end of the field label to ensure that it doesn't start or end with an underscore.
- Increments the derived field label if the label is the same as an existing label. For example, if "X2" already exists, uses "X21," "X22," "X23."

💡 **Tip:** You can download the generated metadata file to change the metadata settings, and then upload it to apply the changes. You can download the metadata file when you create or edit a dataset.

Monitor an External Data Upload

When you upload an external data file, Analytics Cloud kicks off a job that uploads the data into the specified dataset. You can use the Monitor tab of the data manager to monitor and troubleshoot the upload job.

The Jobs subtab (1) of the Monitor tab shows the status, start time, and duration of each dataflow, data sync, recipe, and external data upload job. It shows jobs for the last 7 days.

📝 **Note:** Duration is calculated as the sum of the job queue time and job run time.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To access the Monitor tab of the data manager:
- Edit Analytics Dataflows, Upload External Data to Analytics, or Manage Analytics
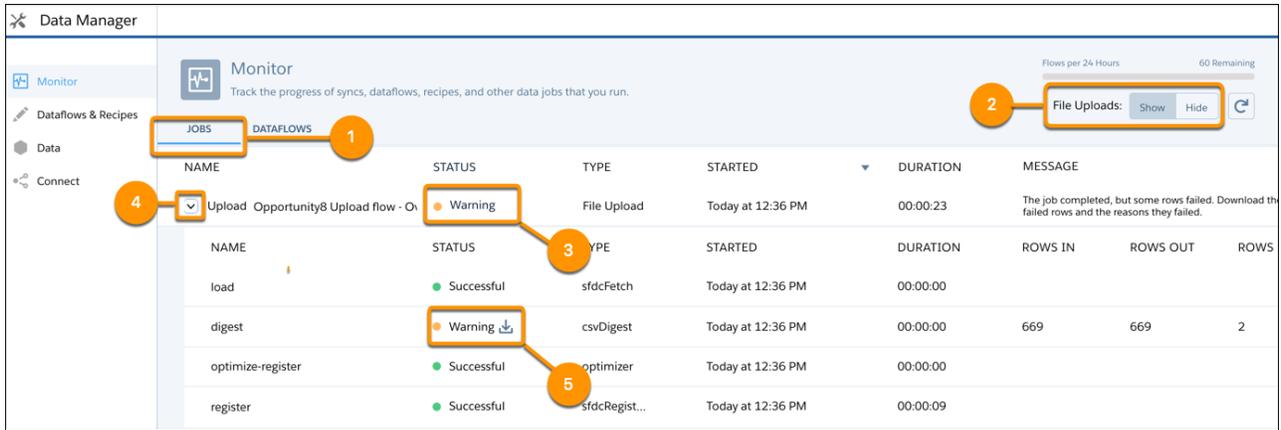
1.
In Analytics Cloud, click the gear icon (⚙) and then click **Data Manager**.
The data manager opens on the Monitor tab, with the Jobs subtab selected by default. It displays each upload job name as
`<dataset_name upload flow>`. You can hover a job to view the entire name.

📝 Note: To view external data upload jobs in the Jobs view, make sure that the File Uploads selector (2) is set to **Show**. Show is the default selection.

2.
To see the latest status of a job, click the Refresh Jobs button (🔄).

Each job can have one of these statuses.

Status	Description
Queued	The job is in queue to start.
Running	The job is running.
Failed	The job failed.
Successful	The job completed successfully.
Warning	The job completed successfully, but some rows failed.

A message is displayed next to jobs with a status of Warning (3) or Failed . If you can't see the complete message, click the message to view all of it.

3. To view the run-time details for a job, expand the job node (4).
The run-time details display under the job. In the run-time details section, scroll to the right to view information about the rows that were processed.

4. To troubleshoot a job that has failed rows, view the error message. If an error log is available, click the status text or download button (5) to download the log.

📝 Note: Only the user who uploaded the external data file can download the error log.

The error log contains a list of failed rows.

5. To troubleshoot a failed job, view the error message and the run-time details.

Set a Source Time Zone for .CSV Data

When you upload .csv data to create or update a dataset, you can select the time zone of the data if it's not GMT.

1. Upload your .CSV file to create a dataset.

2. At the page where you create a name and select an app for the dataset, specify the source and target time zone.

From the Source File Time Zone picklist (1), select the time zone of the data. The Target Time Zone (2) is the time zone selected for your org in settings. Tableau CRM converts date-time values in the file from the source time-zone to the target time zone in the resulting dataset.

Create Datasets Programmatically from CSV Files Using the External Data API

You can use the External Data API to create a single dataset based on external data in a .csv file. You can also use the API to edit the dataset by uploading a new .csv file. When you edit the dataset, you can choose to overwrite all records, append records, update records, or delete records.

When you upload a .csv file with the External Data API, you have to explicitly set the metadata attributes for each column in the .csv file. For example, you can set the field type and default value for each column. You set the metadata attributes for a .csv file in its own metadata file. A *metadata file* is a JSON file that describes the structure of an external data file. If no metadata file is provided when you upload external data using the API, Analytics Cloud treats every column as a dimension and sets the field type to 'Text.' The field type determines the types of queries that can be placed on the column. For example, you can't perform mathematical calculations on dataset columns with a Text field type. You can perform mathematical calculations only on dataset columns with a Numeric field type.

For more information about the External Data API, see the *Analytics External Data API Developer Guide*.

Create Datasets Directly from Microsoft Excel Spreadsheets

The Salesforce Tableau CRM Connector for Excel makes it easy to import data from Microsoft Excel 2013 to Analytics Cloud.

The Tableau CRM Connector for Excel is available as an add-in for Excel 2013 on the desktop and Excel Online in Office 365. The Connector is available from the Office Add-In Store or your organization's private add-in catalog. After you install the Connector just point and click to import data from Excel to Tableau CRM.

Considerations When Using theTableau CRM Connector for Excel

- The Tableau CRM Connector for Excel doesn't support loading data to Salesforce orgs that use custom domains. To load Excel data to a custom domain org, save the data locally in .csv format, and then use the Tableau CRM .csv upload tool to load the data.

- Null measure handling isn't supported when you load data using the Tableau CRM Connector for Excel. Null measure values are replaced with zeros in the resulting dataset, even if null measure handling is enabled in your org.

Install the Tableau CRM Connector for Excel
The Salesforce Tableau CRM Connector for Excel gives you a fast, easy way to import data from Excel 2013 into Analytics Cloud.

Install the Tableau CRM Connector for Excel

The Salesforce Tableau CRM Connector for Excel gives you a fast, easy way to import data from Excel 2013 into Analytics Cloud.

If you use Excel 2013 on the desktop or Office 365, the Office Online version of Excel, the Tableau CRM Connector for Excel gives you a great way to get your data into Analytics Cloud. After installing the Connector, you just select data from Excel, click **Submit Data**, and the Connector does the work for you, importing the data to Analytics Cloud and creating a dataset.

Here's how to install the Connector:

1. Open Excel, either on your desktop or in Office Online.

2. Click the **Insert** tab.

3. Click **Store**.

USER PERMISSIONS
To import data from Excel 2013 to Analytics Cloud: • Upload External Data to Analytics Cloud

4. Search for the Tableau CRM Connector for Excel, and click **Add** to install it.

5. Click **Log in to Salesforce**, and enter your Salesforce credentials to open the Connector.

Once you've installed the Connector, follow the instructions in the Connector window to create datasets from your Excel data. Opening the Connector automatically logs you in to Analytics Cloud. Click the Connector Help icon for complete information about using the app.

Create Datasets and Trending Dashboards from Salesforce Reports

By creating trending Tableau CRM dashboards that track key metrics over time, you can make sense of your growing, ever-changing datasets. Trending data in Tableau CRM helps you discover hidden insights and allows you to share them with colleagues. When you create a trending dashboard based on a Salesforce report, Tableau CRM generates a dataset and a dashboard based on the underlying Salesforce data.

Watch a Demo: ▶ Trend Report Data in Analytics (English Only)

Trending refers to data that is displayed over a timeline, allowing trends and patterns to emerge which can be difficult to recognize in a sea of information. If the relevant data is plotted over time, questions on sales cycles, performance, comparisons, seasonality, and more can be answered at a glance. Any data that can be visualized in time, such as opportunities or cases, can be trended. So can geographical data, such as quarterly sales in specific regions. In fact, any regularly updated dataset which has dates associated with it (a date or timestamp attribute for each row) can be trended.

Trend Salesforce Reports

To trend data in Tableau CRM, you trend a report that's collecting that data.

Stop Trending Salesforce Reports

The number of reports you can trend is limited. To remain under the limit so you can trend other reports, stop trending reports that are no longer needed.

Share Trending Dashboards

You can share a trended dashboard just as you can with any Tableau CRM dashboard.

Limits and Limitations for Trending Data in Tableau CRM

Trended datasets count toward the overall Tableau CRM platform limits, including total number of rows. To prevent the performance of your org from degrading, certain limitations are placed on snapshot size and the number of trended datasets each user can create.

Trend Salesforce Reports

To trend data in Tableau CRM, you trend a report that's collecting that data.

Watch a Demo: ▶ Trend in Tableau CRM

1. Create a dataset.
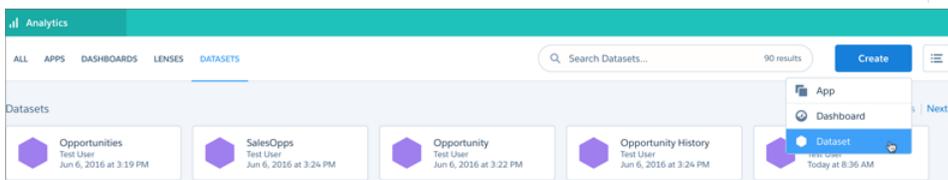
2. Choose **A Salesforce Report** from the available data sources.

3. Select a report.

If the report you want to trend does not exist, you must create it first.

> **Note:** Only the saved version of a report is trended. If you've made unsaved changes to a report, then trend the report, the changes are not reflected in the trending dashboard.

4. Enter a title for the trending dashboard and the new trending dataset.

5. Set a schedule for updating the dashboard. This sets the frequency of how often snapshots of report data are taken.

> **Note:**
> - Once you set the schedule, it can't be changed.
> - If you choose a schedule that updates on the 29th, 30th, or 31st of the month, no updates will occur on months that have too few days to meet that schedule.
> - To avoid unwanted behavior during changes to daylight savings, don't choose 2 A.M. as an update time.

6. Choose a different app if you do not want to save the trending dataset in **My Private App**.
   If you save the dataset to a shared app, all users who can view that app can view the dataset.

7. Click **Start Trending**.
   When the new trending dashboard is ready, you'll receive an email with a link. Once the dashboard is ready, you can also begin to explore the associated dataset, which appears in the Tableau CRM datasets list.

8. Connect to the new trending dashboard using the link in the notification email, or by going to the app in which you saved it (after receiving the notification email).

> **Note:** At first, the trending dashboard displays only 1 data point. As snapshots are taken and added to the new Tableau CRM trending dataset underlying the dashboard, more data points are added.

You can also trend any report directly by clicking **Trend** on that report.

In Lightning Experience:

In classic Salesforce:
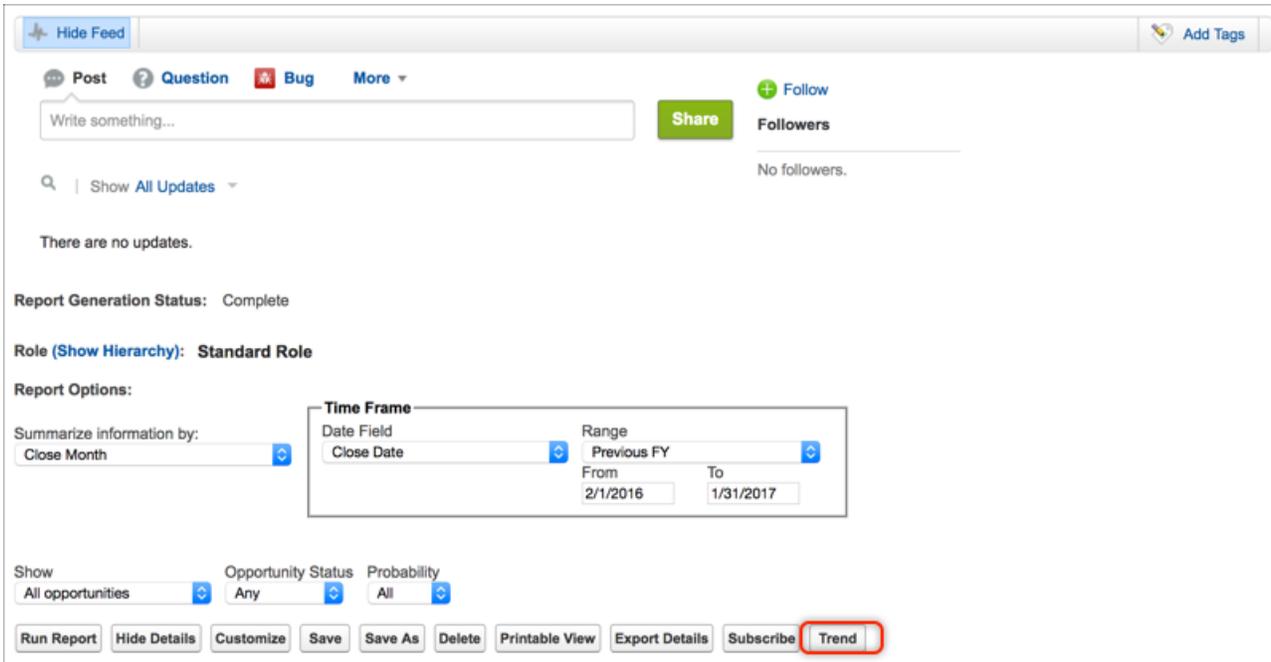
Once you start trending, the **Trend** button on the report becomes **View Trend**. If you click that button in Lightning Experience, an embedded dashboard appears.
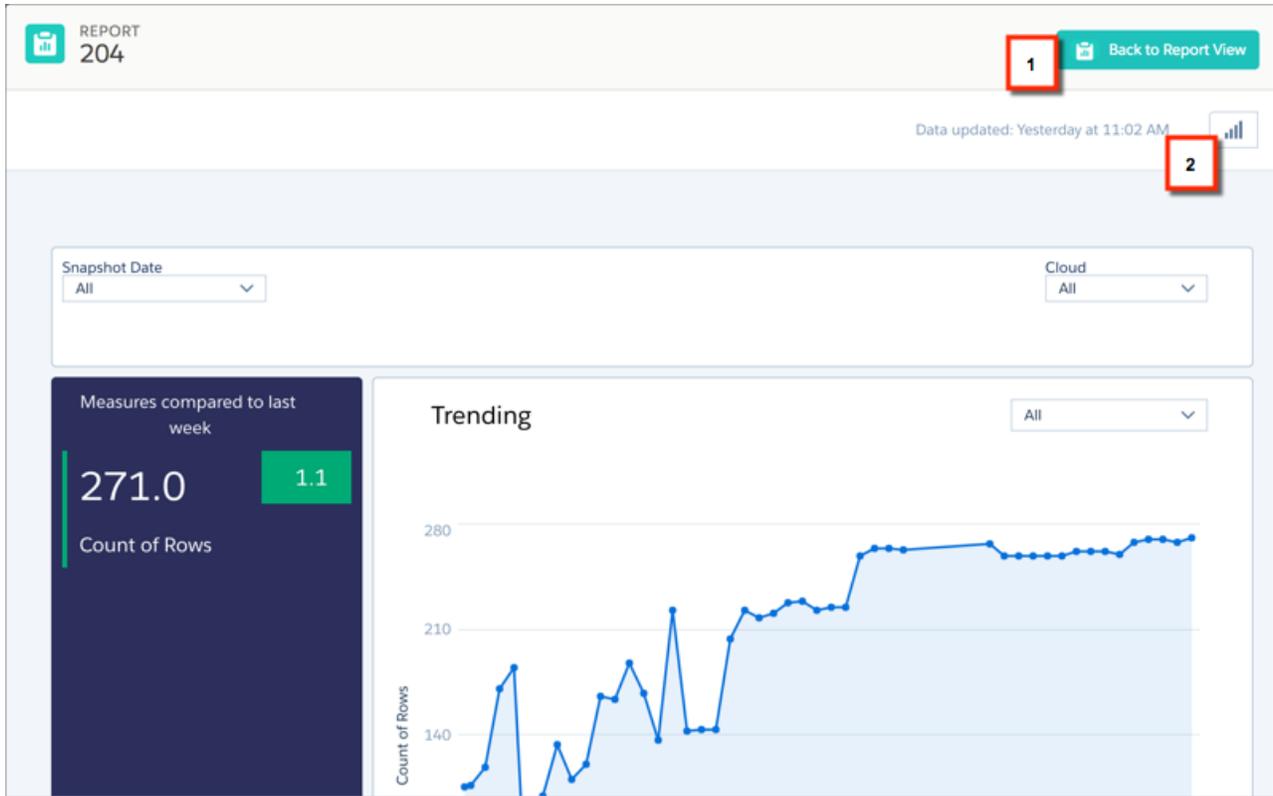
To return to the report, click **Back to Report View** (1). To view the report in Tableau CRM, click the Analytics button (2).

For best results with trending, follow these guidelines:

- Do not attempt to trend matrix reports. These reports do not have the required structure to be trended.
- Choose reports that have changes on a frequency matching the trending schedule you set up.
- Always clone the source reports into your personal folders. After you start trending, don't edit the reports. Keeping the report unchanged ensures a consistent view. If you plan to share trending dashboards, save them to view-only apps. This keeps the data consistent and prevents anomalies from appearing on the trending Tableau CRM dashboard.
- To get focused, easily observable, and actionable insights, tightly scope source reports. For example, trend only new business deals closing in the current quarter rather than all opportunities. A broad scope makes it more difficult to spot trends.
- Group and summarize your reports in the report builder with key information before you start trending them.
- Do not create joined reports just for trending. Trending dashboards can't be created from joined reports.

Stop Trending Salesforce Reports

The number of reports you can trend is limited. To remain under the limit so you can trend other reports, stop trending reports that are no longer needed.

To stop trending from Tableau CRM, open the trending dataset by choosing to edit it, then click **Stop Trending**.

To stop trending from Lightning Experience, go to the trending report and open the menu next to **View Trend** and choose **Stop Trending**.
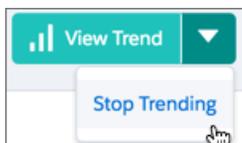
To stop trending from classic Salesforce, go to the trending report and click **View Trend** on the report to open a dialog, then click **Stop Trending**.

> **Note:** If you stop trending, you cannot restart where the trending left off (there is no pause button). Trending the same report again creates a new dataset and a new trending dashboard.

Share Trending Dashboards

You can share a trended dashboard just as you can with any Tableau CRM dashboard.

You can share a trending dashboard just as you share any dashboard. You can post an image of it to a feed, or download and share an image. You can also embed your trending Tableau CRM dashboard as you would any other dashboard.

To share the dashboard directly using its URL, first share the trending dashboard's underlying snapshots dataset by saving it to a Tableau CRM app that your target audience can access.

⊘ **Important:** Sharing the trending Tableau CRM dashboard or the underlying dataset could expose data to users who normally would not have access to that data.

It is possible to share trending Tableau CRM dashboards with users who would normally be prevented from seeing data based on security settings. If you save a trending dashboard to any app that is not My Private App, then anyone with access to that app (and the underlying dataset) can view all data in that Tableau CRM dashboard regardless of any sharing restrictions or field-level security that has been set.

Limits and Limitations for Trending Data in Tableau CRM

Trended datasets count toward the overall Tableau CRM platform limits, including total number of rows. To prevent the performance of your org from degrading, certain limitations are placed on snapshot size and the number of trended datasets each user can create.

Check out the following limits on trending data.

Limit	Value
Maximum number of trended datasets per user	5
Maximum number of rows per snapshot	100,000
Maximum number of rows in the report to be trended	500,000 for admins, 100,000 for non-admins
Maximum total number of rows in a trended dataset	5,000,000
Maximum monthly number of rows for all snapshots per org	40 million

To control the impact that trended datasets have on general org-wide limits, configure these limits, available on the Analytics Settings page:

- **Max number of rows per snapshot** is an option to control the maximum number of rows that a trended report can have when a snapshot is taken. If the report exceeds the limit, the snapshot is not taken and an error email is sent to the user.
- **Max total number of rows in a trended dataset** is an option to limit the maximum number of rows that the Tableau CRM dataset containing the snapshots can have. If the dataset exceeds the limit, the oldest snapshot is deleted from the dataset.
- **Enable automatic deletion of inactive datasets** is an option that is on by default. Trending dashboards that aren't being updated or accessed for the number of days configured in **Number of days to keep inactive datasets** are deleted. The default number of days a trending dashboard can remain inactive is 100. The dashboard owner receives an email warning a week before deletion. When the dashboard and dataset are deleted, the owner receives an email confirmation.

Roll out the capability gradually to your user base and observe data usage and performance. Stop trending datasets that are no longer needed for active use and delete the underlying trended datasets to recover storage limits.

The best candidates for trending are reports that can be visualized in time and have some frequently changing values. Examples: open opportunities pipeline or escalated cases.

You can't use trended datasets as sources in recipes. To prepare data from trended datasets, use a dataflow.

Some situations can prevent a report that is trended in Tableau CRM from successfully capturing the data for a particular run. These situations include, but are not limited to: Jobs involving the User object, site switching, and instance maintenance. If a trended report attempts to run under these conditions, it fails and the dataset does not have records added for that run. You cannot reschedule or run a trending job again; however, future runs that are not subject to these conditions process as normal.

Users whose profiles have View Encrypted Data enabled can't create trended datasets because Tableau CRM does not support encrypted data.

Only the user who starts trending the report can stop trending it. The user can share the trending dashboard with other users. If another user wants to trend the same report, the user must create a trend, which affects other trends for the report.

Once trending begins, do not change access rights to the underlying report or add or delete dimensions and measures. Changing the report or the access rights can cause an error or inaccuracies at the time a snapshot is taken.

If you modify a field in a report that is trended, the change is reflected in the associated dashboard from the time of the change onward. If you change other aspects of the report, such as filters, those changes are not reflected in the associated dashboards.

Snapshots are taken every Sunday at midnight (local time), unless you set a different schedule when trending a report. Processing and surfacing the data requires more time, which can delay snapshots.

If trending is stopped and then restarted, a new snapshot dataset is created, with a new trending dashboard. The previously trended dataset is not included in the new dashboard. Data can be moved into the new dataset using the Salesforce data APIs.

Trended datasets are created in their own private folders. You can use the **Update Dataset** button to move the dataset from a private to a public folder.

The dataflow that creates a trended dataset is auto generated and cannot be viewed or modified. This restriction is required because the underlying trending logic can be sensitive to changes in the dataflow.

You cannot customize the XMD of trended datasets. Customize Actions functionality aren't available for trended datasets.

The dashboard designer is enabled by default for most orgs. Make sure that it remains enabled for your org. To render, trending dashboards rely on the dashboard designer being enabled.

Boolean fields in a trended report are treated both as a measure (for calculations) and a dimension (for sorting).

If you modify the dashboard for a trended report, the changes are reflected in the next trend update.

Manage Datasets

Edit a dataset to update its data, change its extended metadata, apply row-level security, or restore the dataset to a previous version.

You can manage your datasets on the Datasets tab of Data Manager.

Use the search box (1) to find datasets. Click a column header (2) to sort by that column. If you have the Manage Analytics permission, datasets in users' private apps are visible here.

Set Up Dataset Security to Control Access to Rows

If a Tableau CRM user has access to a dataset, the user has access to all records in the dataset by default. To restrict access to records, you can implement row-level security on a dataset when you use sharing inheritance and security predicates. Sharing inheritance automatically applies a Salesforce object's sharing logic to the dataset's rows. A security predicate is a manually assigned filter condition that defines dataset row access.

Edit a Dataset

Edit a dataset to change its name, app, security, or extended metadata (XMD). You can also replace data in a dataset, restore it to a previous version, or delete it. The dataset edit page also provides key information about when the dataset was created and last updated, and where it is used.

Delete a Dataset

Delete unnecessary datasets to reduce app clutter and avoid reaching your org's total row limit for all registered datasets. You can delete datasets from shared apps on which you have at least Editor access, your My Private App, and, with a special user permission, another user's My Private App.

Set Up Dataset Security to Control Access to Rows

If a Tableau CRM user has access to a dataset, the user has access to all records in the dataset by default. To restrict access to records, you can implement row-level security on a dataset when you use sharing inheritance and security predicates. Sharing inheritance automatically applies a Salesforce object's sharing logic to the dataset's rows. A security predicate is a manually assigned filter condition that defines dataset row access.

To implement effective dataset row-level security, most Salesforce orgs can use a combination of sharing inheritance and a backup security predicate. Sharing inheritance provides the correct record access to your users who do not have many employees or shared records. For users with access to many of their own or shared records, like a CEO or dashboard builder, a security predicate is set as backup to sharing inheritance.

To get started, learn more about sharing inheritance and security predicates. Then turn on sharing inheritance and evaluate how well sharing inheritance covers your users' dataset access needs. Finally, set the dataset's security predicate if needed and test.

Add Row-Level Security with a Security Predicate

Applying a predicate to a dataset is more than just defining the predicate expression. You also need to consider how the predicate is dependent on the information in the dataset and where to define the predicate expression.

Add Row-Level Security by Inheriting Sharing Rules

Use sharing inheritance to let Tableau CRM apply the same sharing setup for your datasets as Salesforce uses for your objects. Sharing inheritance increases access accuracy and reduces the need for complicated security predicates for most objects and situations. The tradeoff for applying sharing inheritance is an increase in the time to complete data syncs, dataflow and recipe jobs, and queries. The more complicated the sharing settings, the more impact there is.

SEE ALSO:

sfdcRegister Transformation

sfdcRegister Parameters

Add Row-Level Security with a Security Predicate

Applying a predicate to a dataset is more than just defining the predicate expression. You also need to consider how the predicate is dependent on the information in the dataset and where to define the predicate expression.

Define a predicate for each dataset on which you want to restrict access to records. A *security predicate* is a filter condition that defines row-level access to records in a dataset.

When a user submits a query against a dataset that has a predicate, Analytics Cloud checks the predicate to determine which records the user has access to. If the user doesn't have access to a record, Analytics Cloud does not return that record.

📝 Note:

- Changes to security settings (rowLevelSharingSource or rowLevelSecurityFilter) in a dataflow have no effect on datasets that already exist. You must change those settings on the edit dataset page.
- When sharing inheritance is enabled, you can set the security predicate to 'false' to block all users not covered by sharing. In fact, this predicate is the default when sharing is enabled on existing datasets.

The predicate is flexible and can model different types of security policies. For example, you can create predicates based on:

- Record ownership. Enables each user to view only records that they own.
- Management visibility. Enables each user to view records owned or shared by their subordinates based on a role hierarchy.
- Team or account collaboration. Enables all members of a team, like an opportunity team, to view records shared with the team.

- Combination of different security requirements. For example, you might need to define a predicate based on the Salesforce role hierarchy, teams, and record ownership.

The type of security policy you implement depends on how you want to restrict access to records in the dataset.

⚠️ **Warning**: If row-level security isn't applied to a dataset, any user that has access to the dataset can view all records in the dataset.

You can create a predicate expression based on information in the dataset. For example, to enable each user to view only dataset records that they own, you can create a predicate based on a dataset column that contains the owner for each record. If needed, you can load additional data into a dataset required by the predicate.

The location where you define the predicate varies.

- To apply a predicate on a dataset created from a dataflow, add the predicate in the **rowLevelSecurityFilter** field of the Register transformation. The next time the dataflow runs, Analytics Cloud will apply the predicate.

- To apply a predicate on a dataset created from an external data file, define the predicate in the **rowLevelSecurityFilter** field in the metadata file associated with the external data file. Analytics Cloud applies the predicate when you upload the metadata file and external data file. If you already created the dataset from a external data file, you can edit the dataset to apply or change the predicate.

Row-Level Security Example based on Record Ownership

Let's look at an example where you create a dataset based on a CSV file and then implement row-level security based on record ownership. In this example, you will create a dataset that contains sales targets for account owners. To restrict access on each record in the dataset, you will create a security policy where each user can view only sales targets for accounts that they own. This process requires multiple steps that are described in the sections that follow.

Row-Level Security Example based on Opportunity Teams

Let's look at an example where you create a dataset based on Salesforce data and then implement row-level security based on an opportunity team. In this example, you will create a dataset that contains only opportunities associated with an opportunity team. To restrict access on each record in the dataset, you will create a security policy where only opportunity members can view their opportunity. This process requires multiple steps that are described in the sections that follow.

Row-Level Security Example based on Role Hierarchy and Record Ownership

Let's look at an example where you create a dataset based on Salesforce data and then implement row-level security based on the Salesforce role hierarchy and record ownership. In this example, you will create a dataset that contains all opportunities. To restrict access on each record in the dataset, you will create a security policy where each user can view only opportunities that they own or that are owned by their subordinates based on the Salesforce role hierarchy. This process requires multiple steps that are described in the sections that follow.

Predicate Expression Syntax for Datasets

You must use valid syntax when defining the predicate expression.

Row-Level Security Example based on Record Ownership

Let's look at an example where you create a dataset based on a CSV file and then implement row-level security based on record ownership. In this example, you will create a dataset that contains sales targets for account owners. To restrict access on each record in the dataset, you will create a security policy where each user can view only sales targets for accounts that they own. This process requires multiple steps that are described in the sections that follow.

📝 **Note**: Although this example is about applying a predicate to a dataset created from a CSV file, this procedure can also be applied to a dataset that is created from Salesforce data.

1. Determine Which Data to Include in the Dataset

    First, determine what data you want to include in the dataset. For this example, you will create a Targets dataset that contains all
    sales targets.

2. Determine Row-Level Security for Dataset

    Now it's time to think about row-level security. How will you restrict access to each record in this dataset?

3. Add the Predicate to the Metadata File

    For a dataset created from a CSV file, you can specify the predicate in the metadata file associated with the CSV file or when you edit
    the dataset.

4. Create the Dataset

    Now that you updated the metadata file with the predicate, you can create the dataset.

5. Test Row-Level Security for the Dataset

    You must verify that the predicate is applied properly and that each user can see their own sales targets.

Determine Which Data to Include in the Dataset

First, determine what data you want to include in the dataset. For this example, you will create a Targets dataset that contains all sales
targets.

You will obtain sales targets from the CSV file shown below.

AccountOwner	Region	Target	TargetDate
Tony Santos	Midwest	10000	1/1/2011
Lucy Timmer	Northeast	50000	1/1/2011
Lucy Timmer	Northeast	0	12/1/2013
Bill Rolley	Midwest	15000	1/1/2011
Keith Laz	Southwest	35000	1/1/2011
Lucy Timmer	Southeast	40000	1/1/2011

If you were to create the dataset without implementing row-level security, any user that had access to the dataset would be able to see
the sales targets for all account owners. For example, as shown below, Keith would be able to view the sales targets for all account
owners.
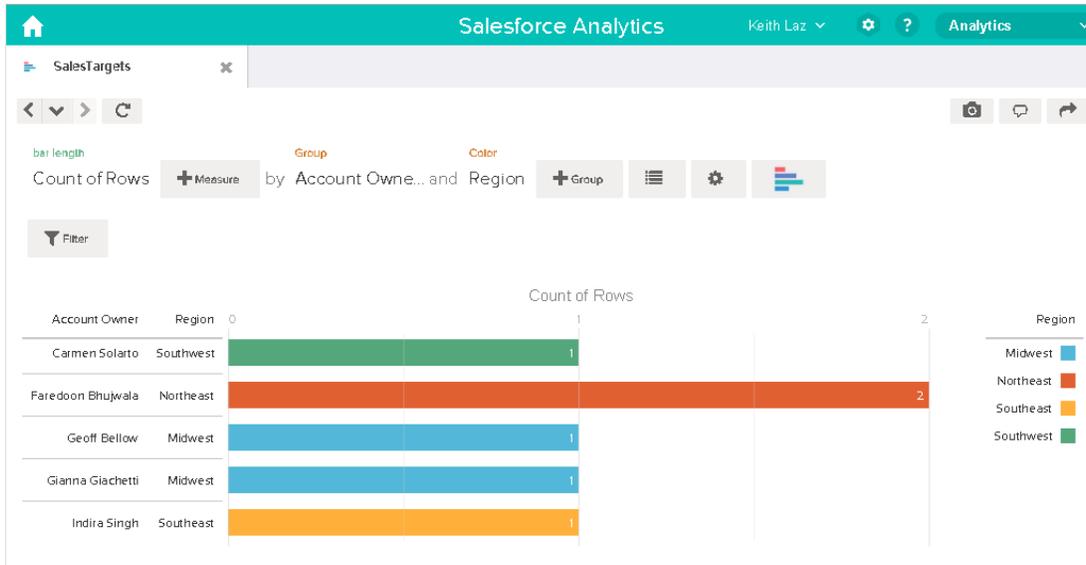
You need to apply row-level security to restrict access to records in this dataset.

Determine Row-Level Security for Dataset

Now it's time to think about row-level security. How will you restrict access to each record in this dataset?

You decide to implement the following predicate on the dataset.

```
'AccountOwner' == "$User.Name"
```

> **Note:** All predicate examples in this document escape the double quotes because it's required when you enter the predicate in the Register transformation or metadata file.This predicate implements row-level security based on record ownership. Based on the predicate, Analytics Cloud returns a sales target record when the user who submits the query on the dataset is the account owner.

Let's take a deeper look into the predicate expression:

- AccountOwner refers to the dataset column that stores the full name of the account owner for each sales target.
- $User.Name refers to the Name column of the User object that stores the full name of each user. Analytics Cloud performs a lookup to get the full name of the user who submits each query.

> **Note:** The lookup returns a match when the names in AccountOwner and $User.Name match exactly—they must have the same case.

Add the Predicate to the Metadata File

For a dataset created from a CSV file, you can specify the predicate in the metadata file associated with the CSV file or when you edit the dataset.

You must escape the double quotes around string values when entering a predicate in the metadata file.

In this example, you add the predicate to the metadata file shown below.

```
{
    "fileFormat": {
    "charsetName": "UTF-8",
    "fieldsDelimitedBy": ",",
```

```
    "fieldsEnclosedBy": "\"",
    "numberOfLinesToIgnore": 1 },
    "objects": [
        {
            "name": "Targets",
            "fullyQualifiedName": "Targets",
            "label": "Targets",
            "rowLevelSecurityFilter": "'AccountOwner' == \"$User.Name\"",
            "fields": [
        {
            "name": "AccountOwner",
            "fullyQualifiedName": "Targets.AccountOwner",
            "label": "Account Owner",
            "type": "Text"
        },
        {
            "name": "Region",
            "fullyQualifiedName": "Targets.Region",
            "label": "Region",
            "type": "Text"
        },
        {
            "name": "Target",
            "fullyQualifiedName": "Targets.Target",
            "label": "Target",
            "type": "Numeric",
            "precision": 16,
            "scale": 0,
            "defaultValue": "0",
            "format": null
        },
        {
            "name": "TargetDate",
            "fullyQualifiedName": "Targets.TargetDate",
            "label": "TargetDate",
            "description": "",
            "type": "Date",
            "format": "dd/MM/yy HH:mm:ss",
            "isSystemField": false,
            "fiscalMonthOffset": 0
        }
        ]
        }
    ]
}
```

Create the Dataset

Now that you updated the metadata file with the predicate, you can create the dataset.

> ⚠ Warning: If you wish to perform the steps in this sample implementation, perform the steps in a non-production environment. Ensure that these changes do not impact other datasets that you already created.

To create the dataset, perform the following steps.

1. In Analytics Cloud, go to the home page.

2. Click **Create** > **Dataset**

3. Click **CSV**.

   The following screen appears.
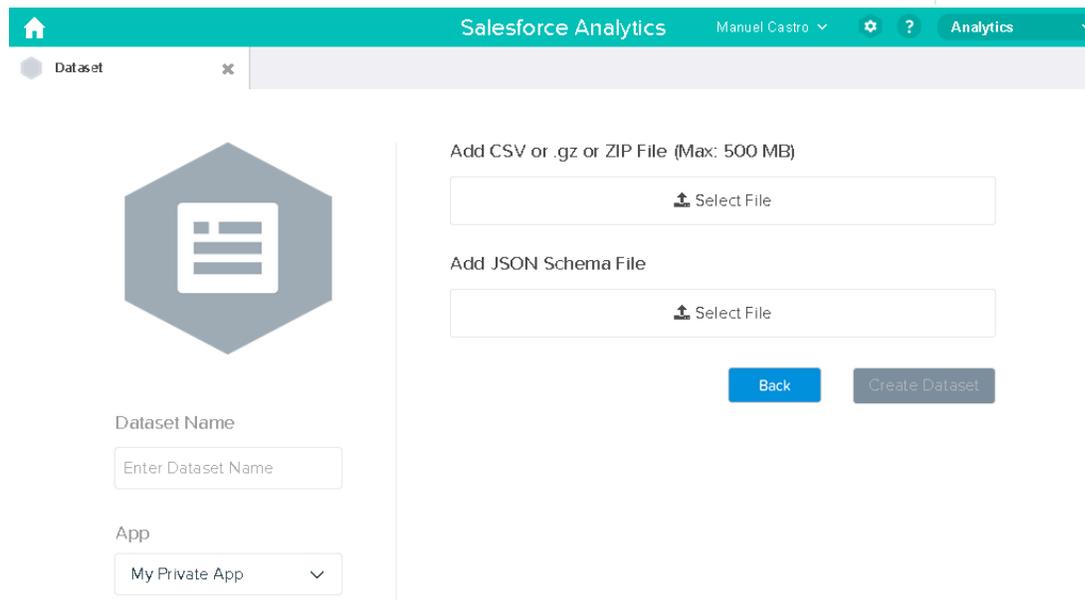
4. Select the CSV file and metadata (schema) file.

5. In the **Dataset Name** field, enter "SalesTarget" as the name of the dataset.

6. Optionally, choose a different app where you want to store the dataset.

7. Click **Create Dataset**.

   Analytics Cloud confirms that the upload is successful and then creates a job to create the dataset. You can view the SalesTarget dataset after the job completes successfully.

8. To verify that the job completes successfully, perform the following steps:

   a. Click the gear icon (⚙) and then select **Data Monitor** to open the data monitor.

   By default, the Jobs View of the data monitor appears. It shows the statuses of dataflow and external data upload jobs.

   b. Click the Refresh Jobs button (↻) to view the latest statuses of the jobs.

Test Row-Level Security for the Dataset

You must verify that the predicate is applied properly and that each user can see their own sales targets.

1. Log in to Analytics Cloud as Keith.

2. Open the SalesTargets dataset.
   As shown in the following lens, notice that Keith can see only his sales target.

Row-Level Security Example based on Opportunity Teams

Let's look at an example where you create a dataset based on Salesforce data and then implement row-level security based on an opportunity team. In this example, you will create a dataset that contains only opportunities associated with an opportunity team. To restrict access on each record in the dataset, you will create a security policy where only opportunity members can view their opportunity. This process requires multiple steps that are described in the sections that follow.

1. Determine Which Data to Include in the Dataset
   First, determine what data you want to include in the dataset. For this example, you will create an OppTeamMember dataset that contains only opportunities associated with an opportunity team.

2. Design the Dataflow to Load the Data

   Now it's time to figure out how the dataflow will extract the Salesforce data and load it into a dataset. You start by creating this high-level design for the dataflow.

3. Determine Row-Level Security for the Dataset

   Now it's time to think about row-level security. How will you restrict access to each record in this dataset?

4. Modify the Dataflow Based on Row-Level Security

   It's now time to add the predicate in the dataflow definition file.

5. Create the Dataset

   Now that you have the final dataflow definition file, you can create the dataset.

6. Test Row-Level Security for the Dataset

   You must verify that the predicate is applied properly and that each user can see the appropriate opportunities.

Determine Which Data to Include in the Dataset

First, determine what data you want to include in the dataset. For this example, you will create an OppTeamMember dataset that contains only opportunities associated with an opportunity team.

You will obtain opportunities from the Opportunity object and the opportunity teams from the OpportunityTeamMember object. Both are Salesforce objects.

In this example, your Salesforce organization has the following opportunity team and users.
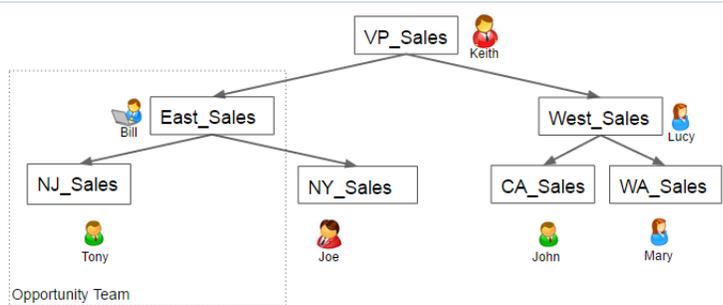
Your organization also contains the following opportunities, most of which are owned by Keith.

Acc - 1000 Widgets is the only opportunity shared by an opportunity team. Bill is the Sales Manager for this opportunity. Tony is the opportunity owner.

Design the Dataflow to Load the Data

Now it's time to figure out how the dataflow will extract the Salesforce data and load it into a dataset. You start by creating this high-level design for the dataflow.
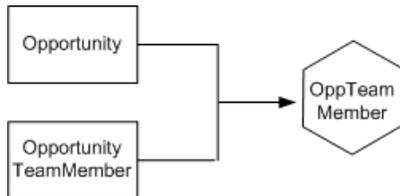
The dataflow will extract data from the Opportunity and OpportunityTeamMember objects, join the data, and then load it into the OppTeamMember dataset.

Now let's implement that design in JSON, which is the format of the dataflow definition file. A dataflow definition file contains transformations that extract, transform, and load data into a dataset.

Based on the design, you create the JSON shown below.

```
{
    "Extract_OpportunityTeamMember": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "OpportunityTeamMember",
            "fields": [
                { "name": "Name" },
                { "name": "OpportunityId" },
                { "name": "UserId" }
            ]
        }
    },
    "Extract_Opportunity": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Opportunity",
            "fields": [
                { "name": "Id" },
                { "name": "Name" },
                { "name": "Amount" },
                { "name": "StageName" },
                { "name": "AccountId" },
                { "name": "OwnerId" }
            ]
        }
    },
    "Augment_OpportunityTeamMember_Opportunity": {
        "action": "augment",
        "parameters": {
            "left": "Extract_OpportunityTeamMember",
            "left_key": [
```

```
            "OpportunityId"
        ],
        "relationship": "TeamMember",
        "right": "Extract_Opportunity",
        "right_key": [
            "Id"
        ],
        "right_select": [
            "Name","Amount"
        ]
    }
},
"Register_Dataset": {
    "action": "sfdcRegister",
    "parameters": {
        "alias": "OppTeamMember",
        "name": "OppTeamMember",
        "source": "Augment_OpportunityTeamMember_Opportunity",
        "rowLevelSecurityFilter": ""
    }
  }
 }
}
```

If you were to run this dataflow, Analytics Cloud would generate a dataset with no row-level security. As a result, any user that has access to the dataset would be able to see the opportunity shared by the opportunity team.

For example, as shown below, Lucy would be able to view the opportunity that belongs to an opportunity team of which she is not a member.
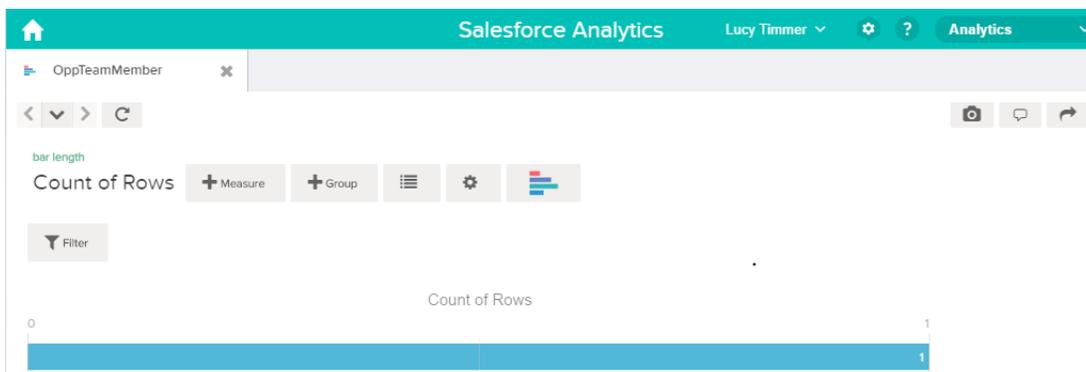
You need to apply row-level security to restrict access to records in this dataset.

Determine Row-Level Security for the Dataset

Now it's time to think about row-level security. How will you restrict access to each record in this dataset?

You decide to implement the following predicate on the dataset.

```
'UserId' == "$User.Id"
```

This predicate compares the UserId column in the dataset against the ID of the user running a query against the dataset. The UserId column in the dataset contains the user ID of the team member associated with each opportunity. To determine the ID of the user running the query, Analytics Cloud looks up the ID of the user making the query in the User object.

For each match, Analytics Cloud returns the record to the user.

Modify the Dataflow Based on Row-Level Security

It's now time to add the predicate in the dataflow definition file.

You add the predicate to the Register transformation that registers the OppTeamMember dataset as shown below.

```
{
   "Extract_OpportunityTeamMember": {
      "action": "sfdcDigest",
      "parameters": {
         "object": "OpportunityTeamMember",
         "fields": [
            { "name": "Name" },
            { "name": "OpportunityId" },
            { "name": "UserId" }
         ]
      }
   },
```

```
    "Extract_Opportunity": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Opportunity",
            "fields": [
                { "name": "Id" },
                { "name": "Name" },
                { "name": "Amount" },
                { "name": "StageName" },
                { "name": "AccountId" },
                { "name": "OwnerId" }
            ]
        }
    },
    "Augment_OpportunityTeamMember_Opportunity": {
        "action": "augment",
        "parameters": {
            "left": "Extract_OpportunityTeamMember",
            "left_key": [
                "OpportunityId"
            ],
            "relationship": "TeamMember",
            "right": "Extract_Opportunity",
            "right_key": [
                "Id"
            ],
            "right_select": [
                "Name","Amount"
            ]
        }
    },
    "Register_Dataset": {
        "action": "sfdcRegister",
        "parameters": {
            "alias": "OppTeamMember",
            "name": "OppTeamMember",
            "source": "105_Augment_OpportunityTeamMember_Opportunity",
            "rowLevelSecurityFilter": "'UserId' == \"$User.Id\""
        }
    }
}
```

Create the Dataset

Now that you have the final dataflow definition file, you can create the dataset.

⚠️ **Warning:** If you wish to perform the steps in this sample implementation, verify that you have all required Salesforce objects and fields, and perform the steps in a non-production environment. Ensure that these changes do not impact other datasets that you already created. Also, always make a backup of the existing dataflow definition file before you make changes because you cannot retrieve old versions of the file.

To create the dataset, perform the following steps.

1. In Analytics Cloud, click the gear icon (⚙️) and then select **Monitor** to open the monitor. The Jobs view of the monitor appears by default.

2. Select **Dataflow View**.

3. Click the actions list (1) for the dataflow and then select **Download** to download the existing dataflow definition file.

4. Open the dataflow definition file in a JSON or text editor.

5. Add the JSON determined in the previous step.

6. Before you save the dataflow definition file, use a JSON validation tool to verify that the JSON is valid.

   An error occurs if you try to upload the dataflow definition file with invalid JSON. You can find JSON validation tool on the internet.

7. Save and close the dataflow definition file.

8. In the Dataflow View of the monitor, click the actions list for the dataflow and then select **Upload**.

9. Select the updated dataflow definition file and click **Upload**.

10. In the Dataflow View of the monitor, click the actions list for the dataflow and then select **Run** to run the dataflow job.

11. Click the **Refresh Jobs** button (🔄) to view the latest status of the dataflow job.

    You can view the OppTeamMember dataset after the dataflow job completes successfully.

> **Note:** If you are adding a predicate to a dataset that was previously created, each user must log out and log back in for the predicate to take effect.

Test Row-Level Security for the Dataset

You must verify that the predicate is applied properly and that each user can see the appropriate opportunities.

1. Log in to Analytics Cloud as Lucy.

2. Open the OppTeamMember opportunity.
   Notice that Lucy can't view the opportunity associated with the opportunity team anymore because she is not a member of the team.
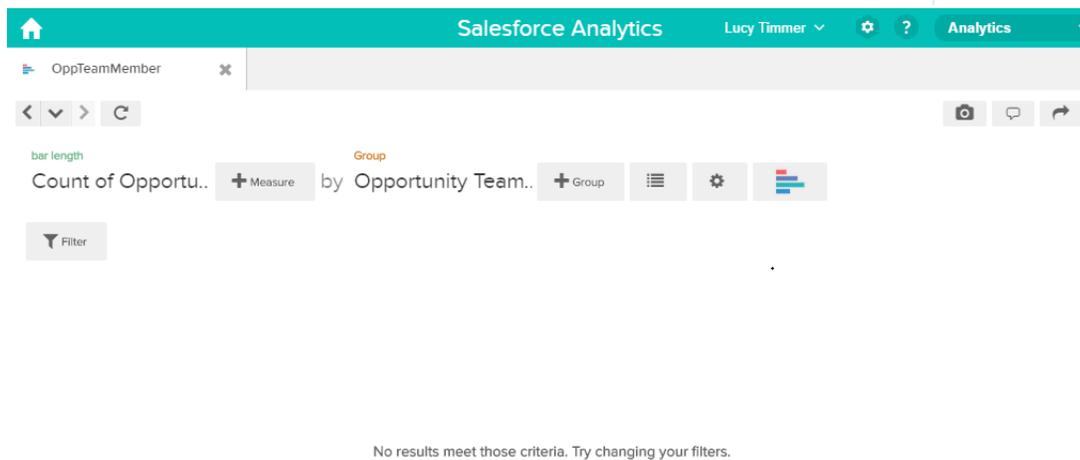
3. Log out and now log in as Bill.
   Bill can view the opportunity that is shared by the opportunity team of which he is a member.

Row-Level Security Example based on Role Hierarchy and Record Ownership

Let's look at an example where you create a dataset based on Salesforce data and then implement row-level security based on the Salesforce role hierarchy and record ownership. In this example, you will create a dataset that contains all opportunities. To restrict access on each record in the dataset, you will create a security policy where each user can view only opportunities that they own or that are owned by their subordinates based on the Salesforce role hierarchy. This process requires multiple steps that are described in the sections that follow.

1. Determine Which Data to Include in the Dataset

    First, determine what data you want to include in the dataset. For this example, you will create the OppRoles dataset that contains all opportunities as well as user details about each opportunity owner, such as their full name, division, and title.

2. Design the Dataflow to Load the Data

    Now it's time to figure out how the dataflow will extract the data and load it into a dataset. You start by creating this high-level design for the dataflow.

3. Determine Row-Level Security for the Dataset

    Now it's time to think about row-level security. How will you restrict access to each record in this dataset?

4. Modify the Dataflow Based on Row-Level Security

    Now it's time to modify the dataflow definition file to account for the predicate.

5. Create the Dataset

    Now that you have the final dataflow definition file, you can create the dataset.

6. Test Row-Level Security for the Dataset

    You must verify that the predicate is applied properly and that each user can see the appropriate opportunities.

SEE ALSO:

flatten Parameters

Determine Which Data to Include in the Dataset

First, determine what data you want to include in the dataset. For this example, you will create the OppRoles dataset that contains all opportunities as well as user details about each opportunity owner, such as their full name, division, and title.

You will obtain opportunities from the Opportunity object and user details from the User object. Both are objects in Salesforce.

In this example, your Salesforce organization has the following role hierarchy and users.
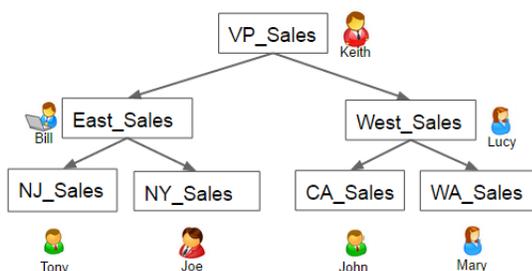
<div style="float:right">

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

</div>

Also, your organization contains the following opportunities, most of which are owned by Keith.

Design the Dataflow to Load the Data

Now it's time to figure out how the dataflow will extract the data and load it into a dataset. You start by creating this high-level design for the dataflow.

The dataflow will extract data from the Opportunity and User objects, join the data, and then load it into the OppRoles dataset.

Now let's implement that design in JSON, which is the format of the dataflow definition file. A dataflow definition file contains transformations that extract, transform, and load data into a dataset.

Based on the design, you create the JSON shown below.

```
{
    "Extract_Opportunity": {
        "action": "sfdcDigest",
        "parameters": {
        "object": "Opportunity",
        "fields": [
            { "name": "Id" },
            { "name": "Name" },
            { "name": "Amount" },
            { "name": "StageName" },
            { "name": "AccountId" },
            { "name": "OwnerId" }
          ]
        }
    },
    "Extract_User": {
        "action": "sfdcDigest",
```

```
            "parameters": {
            "object": "User",
            "fields": [
                { "name": "Id" },
                { "name": "Username" },
                { "name": "LastName" },
                { "name": "FirstName" },
                { "name": "Name" },
                { "name": "CompanyName" },
                { "name": "Division" },
                { "name": "Department" },
                { "name": "Title" },
                { "name": "Alias" },
                { "name": "CommunityNickname" },
                { "name": "UserType" },
                { "name": "UserRoleId" }
                ]
            }
        },
    "Augment_Opportunity_User": {
        "action": "augment",
        "parameters": {
            "left": "Extract_Opportunity",
            "left_key": [
                "OwnerId"
            ],
            "right": "Extract_User",
            "relationship": "Owner",
            "right_select": [
                "Name"
            ],
            "right_key": [
                "Id"
            ]
        }
    },
    "Register": {
        "action": "sfdcRegister",
        "parameters": {
            "alias": "OppRoles",
            "name": "OppRoles",
            "source": "Augment_Opportunity_User",
            "rowLevelSecurityFilter": ""
        }
    }
}
```

If you were to run this dataflow, Analytics Cloud would generate a dataset with no row-level security. As a result, any user that has access to the dataset would be able to view all opportunities. For example, as shown below, Bill would be able to view all opportunities, including those owned by his manager Keith.
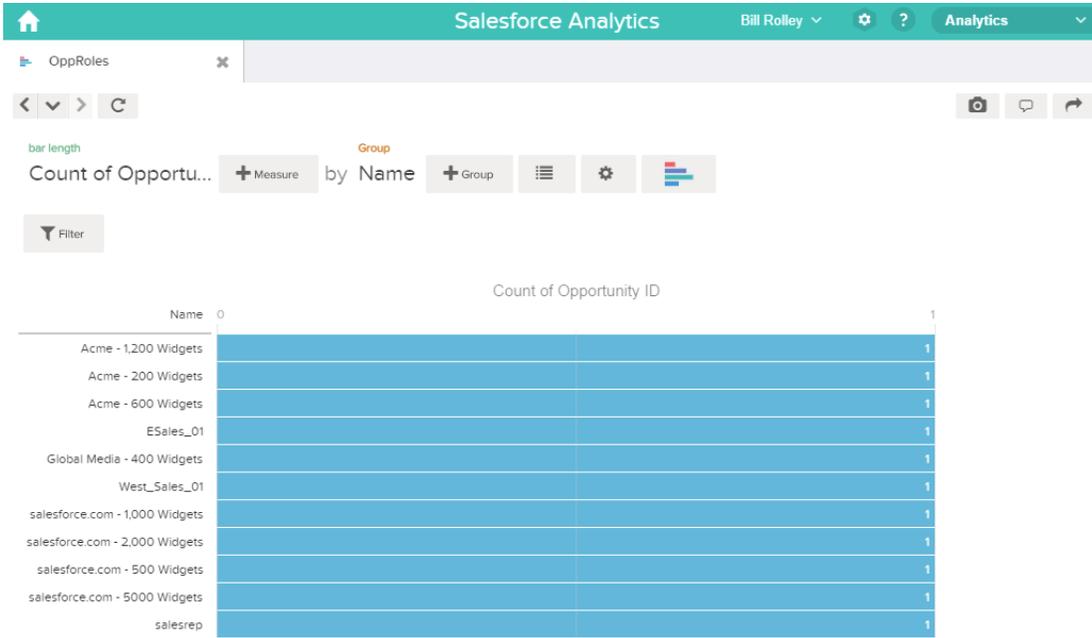
You need to apply row-level security to restrict access to records in this dataset.

Determine Row-Level Security for the Dataset

Now it's time to think about row-level security. How will you restrict access to each record in this dataset?

You decide to implement the following predicate on the dataset.

```
'ParentRoleIDs' == "$User.UserRoleId" || 'OwnerId' == "$User.Id"
```

📝 Note: The current dataflow doesn't contain logic to create a dataset column named "ParentRoleIDs." ParentRoleIDs is a placeholder for the name of a column that will contain this information. In the next step, you will modify the dataflow to add this column to the dataset. This column name will change based on how you configure the dataflow.

Based on the predicate, Analytics Cloud returns an opportunity record if:

- The user who submits the query is a parent of the opportunity owner based on the Salesforce role hierarchy. Analytics Cloud determines this based on their role IDs and the role hierarchy.

- Or, the user who submits the query on the dataset is the opportunity owner.

Let's examine both parts of this predicate.

Predicate Part	Description
'ParentRoleIDs' == "$User.UserRoleId"	• ParentRoleIDs refers to a dataset column that contains a comma-separated list of role IDs of all users above the opportunity owner based on the role hierarchy. You will create this dataset column in the next section. • $User.UserRoleId refers to the UserRoleId column of the User object. Analytics Cloud looks up the user role ID of the user who submits the query from the User object.
'OwnerId' == "$User.Id"	• OwnerId refers to the dataset column that contains the user ID of the owner of each opportunity. • $User.Id refers to the Id column of the User object. Analytics Cloud looks up the user ID of the user who submits the query from the User object.

Modify the Dataflow Based on Row-Level Security

Now it's time to modify the dataflow definition file to account for the predicate.

In this scenario, you have to make changes to the dataflow based on the predicate.

• Add a column in the dataset that stores a comma-separated list of the role IDs of all parents for each opportunity owner. When you defined the predicate in the previous step, you temporarily referred to this column as "ParentRoleIDs." To add the column, you redesign the dataflow as shown in the following diagram:
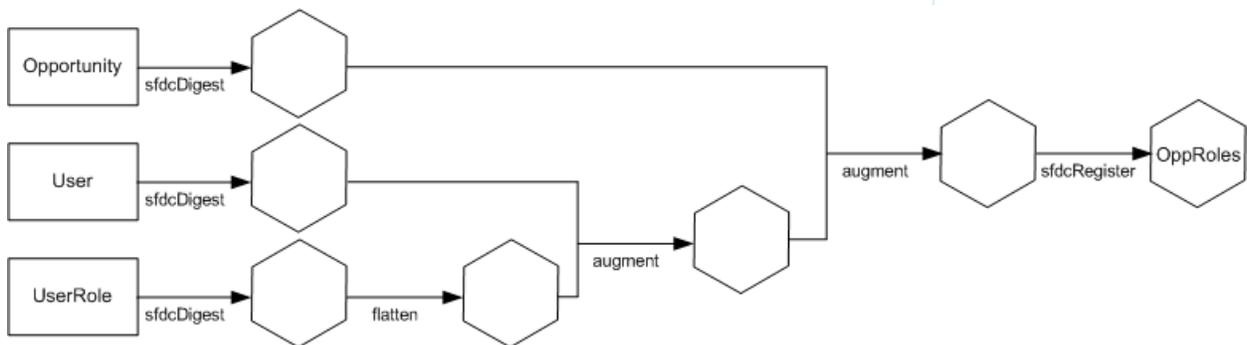
The new dataflow design contains the following changes:

– Extracts the role IDs from the UserRole object.

– Uses the Flatten transformation to generate a column that stores a comma-separated list of the role IDs of all parents of each user. When you determined the predicate in the previous step, you temporarily referred to this column as "ParentRoleIDs."

- – Link the new column to the OppRoles dataset.

- Add the predicate to the Register transformation that registers the OppRoles dataset.

You modify the dataflow as shown below.

```
{
    "Extract_Opportunity": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "Opportunity",
            "fields": [
                { "name": "Id" },
                { "name": "Name" },
                { "name": "Amount" },
                { "name": "StageName" },
                { "name": "AccountId" },
                { "name": "OwnerId" }
            ]
        }
    },
    "Extract_User": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "User",
            "fields": [
                { "name": "Id" },
                { "name": "Username" },
                { "name": "LastName" },
                { "name": "FirstName" },
                { "name": "Name" },
                { "name": "CompanyName" },
                { "name": "Division" },
                { "name": "Department" },
                { "name": "Title" },
                { "name": "Alias" },
                { "name": "CommunityNickname" },
                { "name": "UserType" },
                { "name": "UserRoleId" }
            ]
        }
    },
    "Extract_UserRole": {
        "action": "sfdcDigest",
        "parameters": {
            "object": "UserRole",
            "fields": [
                { "name": "Id" },
                { "name": "ParentRoleId" },
                { "name": "RollupDescription" },
                { "name": "OpportunityAccessForAccountOwner" },
                { "name": "CaseAccessForAccountOwner" },
                { "name": "ContactAccessForAccountOwner" },
                { "name": "ForecastUserId" },
                { "name": "MayForecastManagerShare" },
                { "name": "LastModifiedDate" },
```

```
                 { "name": "LastModifiedById" },
                 { "name": "SystemModstamp" },
                 { "name": "DeveloperName" },
                 { "name": "PortalAccountId" },
                 { "name": "PortalType" },
                 { "name": "PortalAccountOwnerId" }
             ]
         }
    },
    "Flatten_UserRole": {
         "action": "flatten",
         "parameters": {
             "multi_field": "Roles",
             "parent_field": "ParentRoleId",
             "path_field": "RolePath",
             "self_field": "Id",
             "source": "Extract_UserRole"
         }
    },
    "Augment_User_FlattenUserRole": {
         "action": "augment",
         "parameters": {
             "left": "Extract_User",
             "left_key": [
                 "UserRoleId"
             ],
             "relationship": "Role",
             "right": "Flatten_UserRole",
             "right_key": [
                 "Id"
             ],
             "right_select": [
                 "Roles",
                 "RolePath"
             ]
         }
    },
    "Augment_Opportunity_UserWithRoles": {
         "action": "augment",
         "parameters": {
             "left": "Extract_Opportunity",
             "left_key": [
                 "OwnerId"
             ],
             "right": "Augment_User_FlattenUserRole",
             "relationship": "Owner",
             "right_select": [
                 "Name",
                 "Role.Roles",
                 "Role.RolePath"
             ],
             "right_key": [
                 "Id"
             ]
```

```
        }
    },
    "Register": {
        "action": "sfdcRegister",
        "parameters": {
            "alias": "OppRoles",
            "name": "OppRoles",
            "source": "Augment_Opportunity_UserWithRoles",
            "rowLevelSecurityFilter": "'Owner.Role.Roles' == \"$User.UserRoleId\" || 'OwnerId'
 == \"$User.Id\""
        }
    }
}
```

📝 **Note:** In this example, the dataset has columns Owner.Role.Roles and OwnerId. A user can view the values of these columns for each record to which they have access.

Create the Dataset

Now that you have the final dataflow definition file, you can create the dataset.

⚠️ **Warning:** If you wish to perform the steps in this sample implementation, verify that you have all required Salesforce objects and fields, and perform the steps in a non-production environment. Ensure that these changes do not impact other datasets that you already created. Also, always make a backup of the existing dataflow definition file before you make changes because you cannot retrieve old versions of the file.

To create the dataset, perform the following steps.

1. In Analytics Cloud, click the gear icon (⚙️) and then select **Data Monitor** to open the data monitor.
The Jobs View of the data monitor appears by default.

2. Select **Dataflow View**.

3. Click the actions list (1) for the dataflow and then select **Download** to download the existing dataflow definition file.

4. Open the dataflow definition file in a JSON or text editor.

5. Add the JSON determined in the previous step.

6. Before you save the dataflow definition file, use a JSON validation tool to verify that the JSON is valid.

   An error occurs if you try to upload the dataflow definition file with invalid JSON. You can find JSON validation tool on the internet.

7. Save and close the dataflow definition file.

8. In the Dataflow View of the data monitor, click the actions list for the dataflow and then select **Upload**.

9. Select the updated dataflow definition file and click **Upload**.

10. In the Dataflow View of the data monitor, click the actions list for the dataflow and then select **Run** to run the dataflow job.

11. Click the **Refresh Jobs** button ( ) to view the latest status of the dataflow job.

   You can view the OppRoles dataset after the dataflow job completes successfully.

📝 Note: If you are adding a predicate to a dataset that was previously created, each user must log out and log back in for the predicate to take effect.

Test Row-Level Security for the Dataset

You must verify that the predicate is applied properly and that each user can see the appropriate opportunities.

1. Log in to Analytics Cloud as Bill.

2. Open the OppRoles opportunity.
   Notice that Bill can't see his manager Keith's opportunities anymore. Now, he can see only his opportunity and his subordinate Tony's opportunity.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

-
-
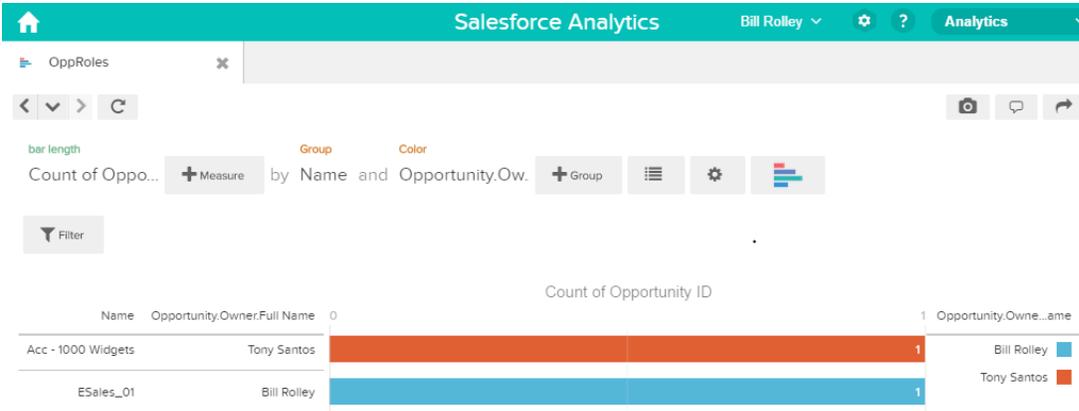
3. Log out and now log in as Keith.

   As expected, Keith can still see all opportunities.
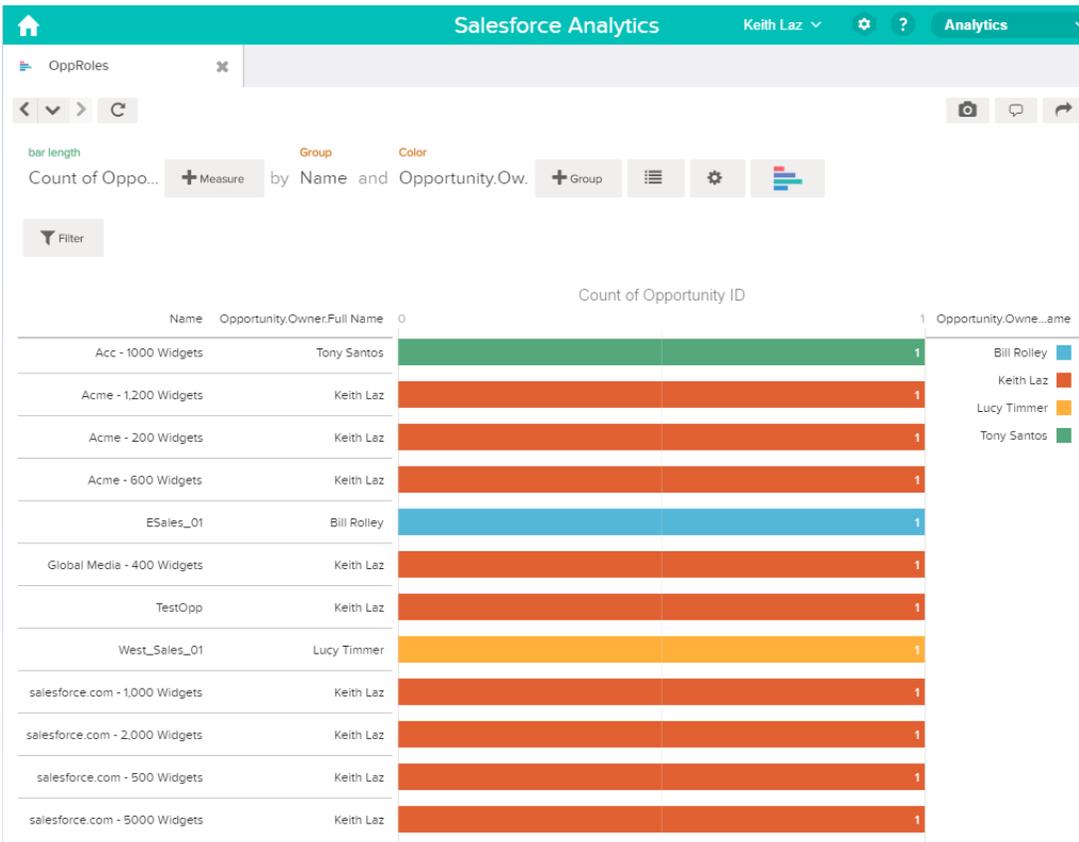
Predicate Expression Syntax for Datasets

You must use valid syntax when defining the predicate expression.

The predicate expression must have the following syntax:

```
<dataset column> <operator> <value>
```

For example, you can define the following predicate expression for a dataset:

```
'UserId' == "$User.Id"
```

You can create more complex predicate expressions such as:

```
('Expected_Revenue' > 4000 || 'Stage Name' == "Closed Won") && 'isDeleted' != "False"
```

Consider the following requirements for the predicate expression:

- The expression is case-sensitive.
- The expression cannot exceed 5,000 characters.
- There must be at least one space between the dataset column and the operator, between the operator and the value, and before and after logical operators. This expression is not valid: `'Revenue'>100`. It must have spaces like this: `'Revenue' > 100`.

If you try to apply a predicate to a dataset and the predicate is not valid, an error appears when any user tries to query the dataset.

Dataset Columns in a Predicate Expression
You include at least one dataset column as part of the predicate expression.

Values in a Predicate Expression
The value in the predicate expression can be a string literal or number literal. It can also be a field value from the User object in Salesforce.

Escape Sequences
You can use the backslash character (\) to escape characters in column names and string values in a predicate expression.

Character Set Support
Analytics Cloud supports UTF-8 characters in dataset column names and values in a predicate expression. Analytics Cloud replaces

non-UTF-8 characters with the UTF-8 symbol (⬥). If Analytics Cloud has to replace a non-UTF-8 character in a predicate expression, users may experience unexpected query results.

Special Characters
Certain characters have a special meaning in Analytics Cloud.

Operators
You can use comparison operators and logical operators in predicate expressions.

Sample Predicate Expressions for Datasets
Review the samples to see how to structure a predicate expression.

Dataset Columns in a Predicate Expression

You include at least one dataset column as part of the predicate expression.

Consider the following requirements for dataset columns in a predicate expression:

- Column names are case-sensitive.
- Column names must be enclosed in single quotes ('). For example, `'Region' == "South"`

  📝 **Note:** A set of characters in double quotes is treated as a string rather than a column name.

- Single quotes in column names must be escaped. For example, `'Team\'s Name' == "West Region Accounts"`

Values in a Predicate Expression

The value in the predicate expression can be a string literal or number literal. It can also be a field value from the User object in Salesforce.

Consider the following requirements for each value type.

Value Type	Requirements	Predicate Expression Examples
string literal	Enclose in double quotes and escape the double quotes.	`'Owner' == "Amber"` `'Stage Name' == "Closed Won"`
number literal	Can be a float or long datatype. Do not enclose in quotes.	`'Expected_Revenue' >= 2000.00` `'NetLoss' < -10000`
field value	When referencing a field from the User object, use the $User.[field] syntax. Use the API name for the field. You can specify standard or custom fields of type string, number, or multi-value picklist. When you define a predicate for a dataset, you must have read access on all User object fields used to create the predicate expression. However, when a user queries a dataset that has a predicate based on the User object, Analytics Cloud uses the access permissions of the Insights Security User to evaluate the predicate expression based on the User object. ⬛ Note: By default, the Security User does not have access permission on custom fields of the User object. To grant the Security User read access on a field, set field-level security on the field in the user profile of the Security User.	`'Owner.Role' == "$User.UserRoleId"` `'GroupID' == "$User.UserGroupId__c"` ⬛ Note: Supported User object field value types are string, number, and multi-value picklist. Other types (for example, boolean) are not supported.

Escape Sequences

You can use the backslash character (\) to escape characters in column names and string values in a predicate expression.

You can use the \' escape sequence to escape a single quote in a column name. For example:

```
'Team\'s Name' == "West Region Accounts"
```

You can use the following escape sequences for special characters in string values.

Sequence	Meaning
\b	One backspace character
\n	New line
\r	Carriage return
\t	Tab
\Z	CTRL+Z (ASCII 26)
\"	One double-quote character
\\	One backslash character
\0	One ASCII null character

Character Set Support

Analytics Cloud supports UTF-8 characters in dataset column names and values in a predicate expression. Analytics Cloud replaces

non-UTF-8 characters with the UTF-8 symbol ( ). If Analytics Cloud has to replace a non-UTF-8 character in a predicate expression, users may experience unexpected query results.

Special Characters

Certain characters have a special meaning in Analytics Cloud.

Character	Name	Description
'	Single quote	Encloses a dataset column name in a predicate expression. Example predicate expression: `'Expected_Revenue' >= 2000.00`
"	Double quote	Encloses a string value or field value in a predicate expression. Example predicate expression: `'OpportunityOwner' == "Michael Vesti"`
( )	Parentheses	Enforces the order in which to evaluate a predicate expression. Example predicate expression: `('Expected_Revenue' > 4000 || 'Stage Name' == "Closed Won") && 'isDeleted' != "False"`

Character	Name	Description
$	Dollar sign	Identifies the Salesforce object in a predicate expression. 📝 Note: You can only use the User object in a predicate expression. Example predicate expression: `'Owner.Role' == "$User.UserRoleId"`
.	Period	Separates the object name and field name in a predicate expression. Example predicate expression: `'Owner' == "$User.UserId"`

Operators

You can use comparison operators and logical operators in predicate expressions.

Comparison Operators

Comparison operators return true or false.

Logical Operators

Logical operators return true or false.

Comparison Operators

Comparison operators return true or false.

Analytics Cloud supports the following comparison operators.

Operator	Name	Description
==	Equals	True if the operands are equal. String comparisons that use the equals operator are case-sensitive. Example predicate expressions: `'Stage Name' == "Closed Won"`
!=	Not equals	True if the operands are not equal. String comparisons that use the not equals operator are case-sensitive. Example predicate expression: `'isDeleted' != "False"`
<	Less than	True if the left operand is less than the right operand. Example predicate expression: `'Revenue' < 100`

Operator	Name	Description
<=	Less or equal	True if the left operand is less than or equal to the right operand.
>	Greater than	True if the left operand is greater than the right operand.
>=	Greater or equal	True if the left operand is greater than or equal to the right operand.
in	Multi-value list filter	True if the left operand exists in the list of strings substituted for a multi-value picklist (field value). Example predicate expression: `'Demog' in ["$User.Demographic__c"]` In this example, `Demographic__c` is of type `multiPicklistField`. During evaluation, the multi-value picklist field is substituted by a list of strings, with 1 string per user-selected item. **Note:** Comma-separated lists are not supported within the square-bracket construct.

You can use the <, <=, >, and >= operators with measure columns only.

Logical Operators
Logical operators return true or false.

Analytics Cloud supports the following logical operators.

Operator	Name	Description
&&	Logical AND	True if both operands are true. Example predicate expression: 'Stage Name' == "Closed Won" && 'isDeleted' != "False"
||	Logical OR	True if either operand is true. Example predicate expression: 'Expected_Revenue' > 4000 || 'Stage Name' == "Closed Won"

Sample Predicate Expressions for Datasets
Review the samples to see how to structure a predicate expression.

The samples are based on the following Opportunity dataset.

Opportunity	Expected_Rev	Owner	OwnerRoleID	Stage_Name	IsDeleted
OppA	2000.00	Bill	20	Prospecting	True
OppB	3000.00	Joe	22	Closed Won	False
OppC	1000.00	可爱的花	36	Closed Won	False

Opportunity	Expected_Rev	Owner	OwnerRoleID	Stage_Name	IsDeleted
OppD	5000.00	O'Fallon	18	Prospecting	True
OppE		Joe	22	Closed Won	True

Let's take a look at some examples to understand how to construct a predicate expression.

Predicate Expression	Details
`'OwnerRoleID' == "$User.UserRoleId"`	Checks column values in the User object.
`'Expected_Rev' > 1000 && 'Expected_Rev' <= 3000`	
`'Owner' = "Joe" || 'Owner' = "Bill"`	
`('Expected_Rev' > 4000 || 'Stage Name' == "Closed Won") && 'isDeleted' != "False"`	Parentheses specify the order of operations.
`'Stage Name' == "Closed Won" && 'Expected_Rev' > 70000`	
`'Owner' == "可爱的花"`	String contains Unicode characters.
`'Owner' == "O\'Fallon"`	Single quote in a string requires the escape character.
`'Stage Name' == ""`	Checks for an empty string.

Add Row-Level Security by Inheriting Sharing Rules

Use sharing inheritance to let Tableau CRM apply the same sharing setup for your datasets as Salesforce uses for your objects. Sharing inheritance increases access accuracy and reduces the need for complicated security predicates for most objects and situations. The tradeoff for applying sharing inheritance is an increase in the time to complete data syncs, dataflow and recipe jobs, and queries. The more complicated the sharing settings, the more impact there is.

As an admin for your Salesforce org, you likely use a combination of sharing settings to provide users access to Salesforce data appropriate to their roles. Sharing settings include manual and rule-based sharing as well as role hierarchy, role, group, apex-managed, and team-based sharing. For more information, see Sharing Settings.

For supported objects, you can enable sharing inheritance in Tableau CRM to use the Salesforce sharing settings in Tableau CRM. When you create or edit datasets, specify the objects to inherit sharing from.

Enable Sharing Inheritance
Turn on sharing inheritance and select the objects to use as a sharing source. Sharing inheritance is on by default in new Salesforce orgs.

Determine If Sharing Inheritance Will Work for You
To understand how well sharing inheritance can work for your dataset row-level sharing needs, review these supported objects, maximum number of sharing descriptors, and considerations. Use the Sharing Inheritance Coverage Assessment tool to evaluate and understand your object and user eligibility.

Set Sharing Inheritance for a Data Prep Recipe

Specify the sharing inheritance settings when generating a dataset from a Data Prep recipe, then set a default security predicate.

Set Sharing Inheritance for a Dataflow

Specify which datasets inherit sharing settings and set a default security predicate.

Sharing Inheritance Limits and Considerations

Here are some things to consider when you work with sharing inheritance.

Enable Sharing Inheritance

Turn on sharing inheritance and select the objects to use as a sharing source. Sharing inheritance is on by default in new Salesforce orgs.

> 📝 **Note:** We recommend that you test in a sandbox environment before rolling out sharing inheritance to production. Test your particular use cases against your org's security model and data to make sure that sharing inheritance works for you.

Turn On Sharing Inheritance

1. From Setup, in the Quick Find box, enter `Analytics`, and then click **Settings**.

2. Select **Inherit sharing from Salesforce**, and click **Save**.

Enable Sharing Inheritance for Synced Objects

If your org has Data Sync enabled, enable sharing inheritance for each object that you want to use as a sharing source.

1. In Tableau CRM Studio, click **Data Manager**.

2. In Data Manager, click **Connect**.

3. On the right end of the row for the object you want to enable, click the dropdown list.

4. Click **Row Level Sharing**.

5. Click **Sharing inheritance on**.

6. Click **Save**.

Determine If Sharing Inheritance Will Work for You

To understand how well sharing inheritance can work for your dataset row-level sharing needs, review these supported objects, maximum number of sharing descriptors, and considerations. Use the Sharing Inheritance Coverage Assessment tool to evaluate and understand your object and user eligibility.

Supported Objects for Sharing Inheritance

Which Salesforce object does your dataset inherit sharing from? Each dataset can inherit sharing settings from one object, regardless of how many source objects are used to create it. Supported objects for sharing inheritance are:

- Account
- Case
- Contact
- Lead
- Opportunity

Sharing Settings Change Frequency

How often do the sharing settings change for the source object (object selected to inherit sharing from), and how long can you wait for sharing inheritance to catch the change? Each full data sync captures sharing setting changes, so evaluate your sync settings. For more information, see Security Metadata Drift.

Run the Sharing Inheritance Coverage Assessment Report

Does your object have any records or users with more than your org's maximum sharing descriptors? And what is your org's maximum sharing descriptors?

> **Note:** A sharing descriptor is the ID of any user or group that has access to a record. Sharing descriptors, tracked by Salesforce in the Object Sharing Table, ensure that only the listed users and groups can access the correct records. IDs are added to the sharing table for many reasons including record ownership, sharing rules, and manual sharing.

After you turn on sharing inheritance, run the Sharing Inheritance Coverage Assessment report on an object, then the users for that object, to investigate how well sharing inheritance will work for you.

1. In the Salesforce Setup menu, under Analytics, select **Sharing Inheritance Coverage Assessment**.
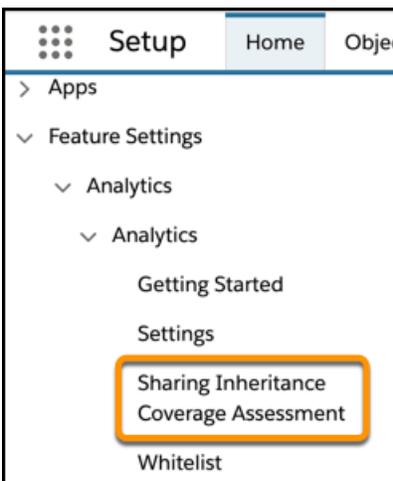
2. In the Salesforce Objects tab, choose the object to evaluate (1).
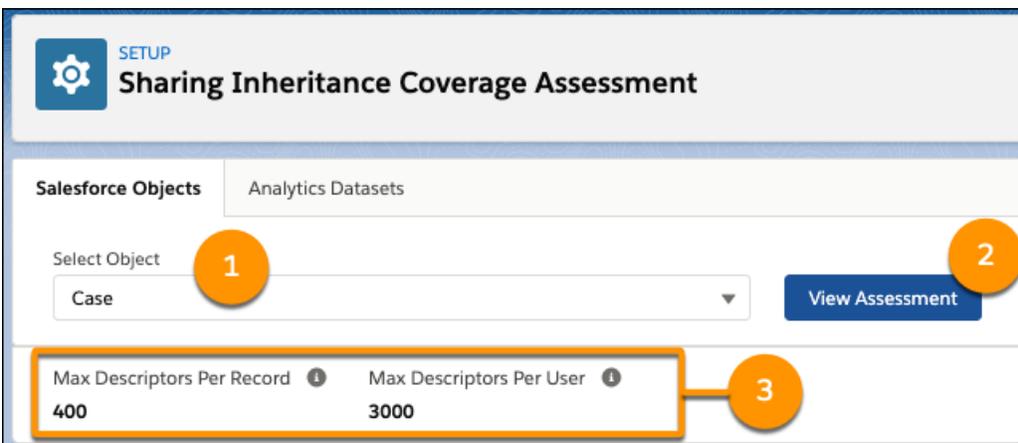
3. Select **View Assessment** (2). Records from the evaluated object with more than your org's maximum descriptors (3) are found. The top records with too many descriptors are listed along with their descriptor count.

   If records exceed the limit, depending on your org's sharing strategy flexibility, choose to reduce the number of descriptors per record or contact Salesforce Support to evaluate whether higher limits are an option.

4. Select **View User Coverage** to see which users have more than the max descriptors on the selected object. The users not covered by sharing inheritance, who need a security predicate to manage their row-level access, are listed.

Set Sharing Inheritance for a Data Prep Recipe

Specify the sharing inheritance settings when generating a dataset from a Data Prep recipe, then set a default security predicate.

> ✏️ **Note:** We recommend that you test in a sandbox environment before rolling out sharing inheritance to production. Test your particular use cases against your Salesforce org's security model and data to make sure that sharing inheritance works for you.

Configure Data Prep Recipe

Before a dataset can inherit sharing, you must configure its recipe in Data Prep. Specify which source object to inherit sharing from and the backup security predicate.

1. Enable sharing inheritance for your org and synced objects if you haven't already. For more information, see Enable Sharing Inheritance.

2. Open the recipe in Data Prep. Sharing inheritance is not available for recipes built in Data Prep classic.

3. In the output node, configure the sharing inheritance settings. Set which object is the source of sharing rules (1).

4. Optionally, add a security predicate in the Security Predicate field (2). Leaving this field empty is the same as "False," so users not covered by sharing inheritance have no record visibility. For syntax and more information, see Add Row-Level Security with a Security Predicate on page 1003.

5. Click **Apply**.

Configure Datasets

Update the sharing inheritance settings for datasets on the edit dataset page.

> 📝 **Note:** The settings in the dataset and recipe must match. If they don't, you receive the warning, "The sharing source and security predicate in this dataset version must be the same as in the recipe."

1. Edit the dataset. For more information, see Edit a Dataset.

2. Click the pencil under Sharing Source.

3. Select the object to inherit sharing settings from. Only valid objects are displayed in the list. For example, the primary key of the object must be a field in the dataset.

4. Optionally, click the pencil under Security Predicate. The security predicate is the row visibility fallback when you exceed sharing inheritance limits, described here.

   a. Add your security predicate in the syntax described in Add Row-Level Security with a Security Predicate. The default on existing datasets is `false`, so users not covered by sharing inheritance have no record visibility.

Set Sharing Inheritance for a Dataflow

Specify which datasets inherit sharing settings and set a default security predicate.

> 📝 **Note:** We recommend that you test in a sandbox environment before rolling out sharing inheritance to production. Test your particular use cases against your Salesforce org's security model and data to make sure that sharing inheritance works for you.

Configure Dataflows

Update the dataflow that generates the datasets that you want to inherit sharing. Specify which source object to inherit sharing from and the backup security predicate.

1. Enable sharing inheritance for your org and synced objects if you haven't already. For more information, see Enable Sharing Inheritance.

2. If using the dataflow editor to specify the source object and backup security predicate:

   a. In Tableau CRM, click the gear icon (⚙️) and then click **Data Manager**.

   b. Click the **Dataflows & Recipe** tab.

   c. Click the dataflow that you want to edit, or click the actions button and select **Edit**.

   d. Click the sfdcRegister node.

   e. Select the object to inherit sharing from in the Sharing Source field.

   f. Optionally, add a security predicate in the Security Predicate field. No value is "False," so users not covered by sharing inheritance have no record visibility. For syntax and more information, see Add Row-Level Security with a Security Predicate on page 1003.

   g. Click **Save**.

3. If configuring the dataflow through the definition file:

   a. Add the `rowLevelSharingSource` parameter to the `sfdcRegister` node parameters for the dataset. For more information, see sfdcRegister. The `rowLevelSharingSource` parameter takes a string, which is the API name for the

object to inherit sharing from. In this example, the parameter specifies to inherit the Salesforce sharing settings on the Opportunity object.

```
"reg": {
  "action": "sfdcRegister",
  "parameters": {
    "source": "Opportunity_final",
    "name": "Opportunity w/ Account",
    "alias": "Oppty_w_Acct",
    "rowLevelSharingSource": "Opportunity",
    "rowLevelSecurityFilter": "'OwnerId' == \"$User.Id\""
  }
},
```

📝 **Note:** Only inherit sharing settings from objects extracted using the sfdcDigest transformation.

b. Specify the security predicate in the `rowLevelSecurityFilter` parameter of the `rowLevelSharingSource` parameter. In the example, when sharing limits are exceeded, users see only the opportunities that they own. Set the security predicate to `false` to block all users not covered by sharing.

Configure Datasets

Update the sharing inheritance settings for datasets on the edit dataset page.

📝 **Note:** The settings in the dataset and dataflow must match. If they don't, you receive the warning, "The sharing source and security predicate in this dataset version must be the same as in the dataflow."

1. Edit the dataset. For more information, see Edit a Dataset.

2. Click the pencil under Sharing Source.

3. Select the object to inherit sharing settings from. Only valid objects are displayed in the list. For example, the primary key of the object must be a field in the dataset.

4. Optionally, click the pencil under Security Predicate. The security predicate is the row visibility fallback when you exceed sharing inheritance limits, described here.

a. Add your security predicate in the syntax described in Add Row-Level Security with a Security Predicate. The default on existing datasets is `false`, so users not covered by sharing inheritance have no record visibility.

Sharing Inheritance Limits and Considerations

Here are some things to consider when you work with sharing inheritance.

Sharing Inheritance Limits

- Sharing inheritance can be applied from a supported object if all object records have fewer than 400 sharing descriptors each. Supported objects for sharing inheritance are:
  - Account
  - Case
  - Contact
  - Lead
  - Opportunity

> **Note:** Objects must be local and extracted using the sfdcDigest transformation. Enable sharing inheritance for each object as described in Enable Sharing Inheritance on page 1033. All users on an unsupported object list as *UNSUPPORTED* on the Sharing Inheritance Coverage Assessment Report, described here.

- Sharing inheritance covers a user if they have "View All Data" permission or their record access is granted by fewer than 3,000 sharing descriptors. The backup security predicate takes effect for users with more than this number of sharing descriptors without the "View All Data" permission. A sharing descriptor is record access granted through several methods, including:
  - Owning the record
  - Role hierarchy
  - Sharing Rules
  - Manual sharing
  - Apex managed sharing

> **Note:** You can't easily count how many sharing descriptors are associated with a user or record without a developer's help. Instead, fetch the list of records or users not covered by sharing inheritance with the Sharing Inheritance Coverage Assessment Report. Users with more than 3,000 sharing descriptors have the uncovered reason *HIGH_VISIBILITY*.

General Considerations

- It is best practice to have a defined security predicate for datasets using inherited sharing. Without a security predicate, users not covered by sharing inheritance see no data in the dataset because they have no dataset row-level access.
- Sharing isn't automatically applied to datasets. You apply sharing to each dataset manually.
- Sharing inheritance can affect the performance of queries, dataflows, and Data Prep recipes. If your requirements include best-possible performance, use security predicates instead of sharing inheritance. If not, enjoy the convenience of sharing inheritance.
- Changes to the rowLevelSharingSource or rowLevelSecurityFilter security settings in a dataflow only affect datasets created after you save the change. Similarly, changes to a Data Prep recipe output node's Sharing Source and Security Predicate fields only affect datasets created after you save the change. Update those settings for existing datasets on the edit dataset page.
- For an object to appear in the security-sharing source list, the primary key of the custom object must be a field in the dataset. A foreign key doesn't satisfy this requirement. For example, if you have Opportunity.AccountId in your dataset but not Account.Id, you can't inherit sharing from the Account object.
- Sharing inheritance uses your Salesforce org sharing settings. If you don't want to apply incomplete Salesforce org sharing settings to Tableau CRM, do not use sharing inheritance.
- Sharing inheritance is not available for Data Prep Classic recipes.

Information Leak Considerations

Consider these points to avoid data leaks to users who shouldn't have access when they use sharing inheritance.

- A dataset can inherit sharing settings from only one object, regardless of how many source objects are used to create the dataset. Because many objects comprise the dataset, each object can use a different security model.

- The computeRelative and delta dataflow transformations can merge information from records with different security.

- Calculated fields are treated as normal fields. Row-level security applied during the calculation in Salesforce is ignored.

🛑 **Important:** If your dataflow doesn't do a full extraction each time it runs, be sure to evaluate whether security drift is a risk for the datasets you bring into Tableau CRM. Consider whether to use periodic full sync. For more information, see Security Metadata Drift.

Security Metadata Drift

The data you use in Tableau CRM can come from Salesforce objects and fields. A dataflow job runs, and then you can analyze the resulting dataset. In an ideal world, each object in your dataset would stay in perfect sync with its source object. In the real world, the correctness of an object is only as good as the last update. The longer the time between updates, the greater the likelihood of *drift*. The security metadata (predicates and descriptors) of a Salesforce object is subject to the same risk of drift.

Security Metadata Drift

The data you use in Tableau CRM can come from Salesforce objects and fields. A dataflow job runs, and then you can analyze the resulting dataset. In an ideal world, each object in your dataset would stay in perfect sync with its source object. In the real world, the correctness of an object is only as good as the last update. The longer the time between updates, the greater the likelihood of *drift*. The security metadata (predicates and descriptors) of a Salesforce object is subject to the same risk of drift.

Data drift is inevitable in any system that uses batch updates. Whether security drift affects your dataset depends on many factors: how often security permissions change, how often your dataflow runs, how sharing is configured, and whether your users have a range of security permissions.

For example, if you replicate an Opportunity record in Tableau CRM and then remove a sharing permission for a user on that record in Salesforce, your change doesn't affect the copy in Tableau CRM until the next full sync. An incremental sync isn't sufficient because changing only a sharing permission isn't considered to be a change to the record.

(However, if you remove the same user from a group that controls the same sharing permission, your change is effective immediately — another good argument for using groups to define security.)

A user who has "View All Data" permission (or a user who has access to only a few thousand records) likely experiences little to no drift. But just because an administrator can see a given record (thanks to "View All Data") doesn't mean that a user who experiences drift can see the same record.

The question that you must answer for each of your datasets is whether drift is a reasonable tradeoff for being able to inherit security metadata.

The only way to be certain that security metadata is up to date is to run full extracts as often as possible. Consider enabling periodic full sync.

Edit a Dataset

To view a dataset edit page:	Use Analytics Cloud AND Editor access to the dataset's app
To update a dataset name, app, and extended metadata:	Use Analytics Cloud AND Editor access to the dataset's app
To upload and preview data:	Upload External Data to Analytics AND Editor access to the dataset's app
To edit a dataset security:	Edit Analytics Dataflows
To restore a dataset:	Edit Analytics Dataflows

Edit a dataset to change its name, app, security, or extended metadata (XMD). You can also replace data in a dataset, restore it to a previous version, or delete it. The dataset edit page also provides key information about when the dataset was created and last updated, and where it is used.

You can edit a dataset from Tableau CRM home or from the data manager.

- To edit a dataset from Tableau CRM home, follow these steps.

    1. Click **All Items** in the left panel.

    2. Click **Datasets**.

    3. On the right of the dataset, click ▼ .

    4. Click **Edit**.

- To edit a dataset from the data manager, follow these steps.

    1. In the data manager, click the **Data** tab.

    2. Click the **Datasets** subtab.

    3. On the right of the dataset, click ▼ .

    4. Click **Edit Dataset**.

- To replace the data in the dataset with CSV data, click ⬆ .

- Edit the following settings, as needed.

Option	Description
Dataset name	Enter a new name if you'd like to change the name of the dataset. The name cannot exceed 80 characters.
App	Select a new app if you'd like to move the dataset to a different app.
Extended Metadata File	Specify an extended metadata file if you'd like to customize the formatting of dashboards associated with the dataset.
	Refer to *Extended Metadata (XMD) Reference* for information about extended metadata files.

Option	Description
Sharing Source	If you have enabled sharing inheritance, specify the object from which you want to inherit sharing for this dataset. You can't specify a sharing source for datasets created from CSV files. When you select a sharing source, you must also add a security predicate. Tableau CRM populates the Security Predicate field with the value *false*. See Salesforce Sharing Inheritance for Datasets.
Security Predicate	Add a security predicate if you'd like to apply row-level security on the dataset. For information about predicates, see Set Up Dataset Security to Control Access to Rows.

Apply Extended Metadata to Change the Formatting of a Dataset

A dataset's extended metadata (XMD) file allows you to customize the formatting of dashboards associated with the dataset. You can update the XMD file through the dataset's edit page.

Customize the Labels and Colors of Dataset Columns and Values

Choose what shows up in lenses and dashboards. Rename and hide dataset columns. Set the default columns that appear in tables. Set the labels and default colors for dimension values.

Restore a Dataset Version

Restoring a dataset rolls back its data to a previous version. Restore a dataset to reverse a recent update or to undo the results of a recipe or dataflow change.

Apply Extended Metadata to Change the Formatting of a Dataset

A dataset's extended metadata (XMD) file allows you to customize the formatting of dashboards associated with the dataset. You can update the XMD file through the dataset's edit page.

> ⛔ **Important:** An XMD file must use XMD 2.0 syntax and be no greater than 5 MB. Refer to *Extended Metadata (XMD) Reference* for information about extended metadata files and XMD 2.0 syntax.

1. On the Tableau CRM page, click the **Datasets** tab. If you're in the data manager, click the **Datasets** tab there.

2. Hover over the dataset that you want to update, click ▼ , and then click **Edit**. If you're in the data manager, click **Edit Dataset**.
   The dataset edit page opens.

3. Under Extended Metadata File, click ▼ , and then select **Download**. Download the current XMD file as a backup.

4. Under Extended Metadata File, click ▼ , and then select **Replace**. Select your new XMD file and click **Open**.
   If the XMD file is not valid, you see an error message. Correct the syntax of the file and upload it again.

> 📝 **Note:** If the XMD file displays an error message when you open the dataset edit page, the current file is not valid.

<div style="border:1px solid #ccc; padding:8px;">

formula_base.xmd.json   ⬇

⊖ The selected file is a an older XMD syntax that is no longer compatible with Wave. Please upload an extended metadata file that uses XMD 2.0 syntax.

</div>
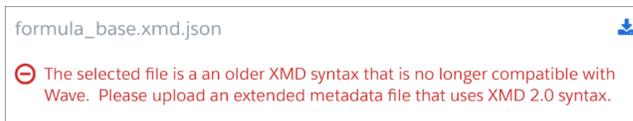
Perform step 3 to download the file, then convert it to XMD 2.0. Then perform step 4 to reupload the converted file and update the dataset.

Refer to *Extended Metadata (XMD) Reference* for information about converting extended metadata files to XMD 2.0 syntax.

Customize the Labels and Colors of Dataset Columns and Values

Choose what shows up in lenses and dashboards. Rename and hide dataset columns. Set the default columns that appear in tables. Set the labels and default colors for dimension values.

View and Configure Dataset Columns

You can view, hide, and rename dataset columns. Users with view-only access to the dataset can view the names and API names of columns. You can also change the display format of measure columns.

Choose the Default Dataset Columns That Appear in a Table

Make tables your own by choosing the default columns. For example, show the most commonly used columns in a particular order. If you don't set the defaults, the first five measures and five dimensions are selected in alphabetical order.

Change the display labels and default colors of dimension values to make them easier to understand and spot. The modifications don't alter the underlying data stored in the dataset or the API names of the columns. The changes affect only the appearance in the user interface.

View and Configure Dataset Columns

You can view, hide, and rename dataset columns. Users with view-only access to the dataset can view the names and API names of columns. You can also change the display format of measure columns.

1. To configure the dataset columns, click **Fields** while exploring the dataset.
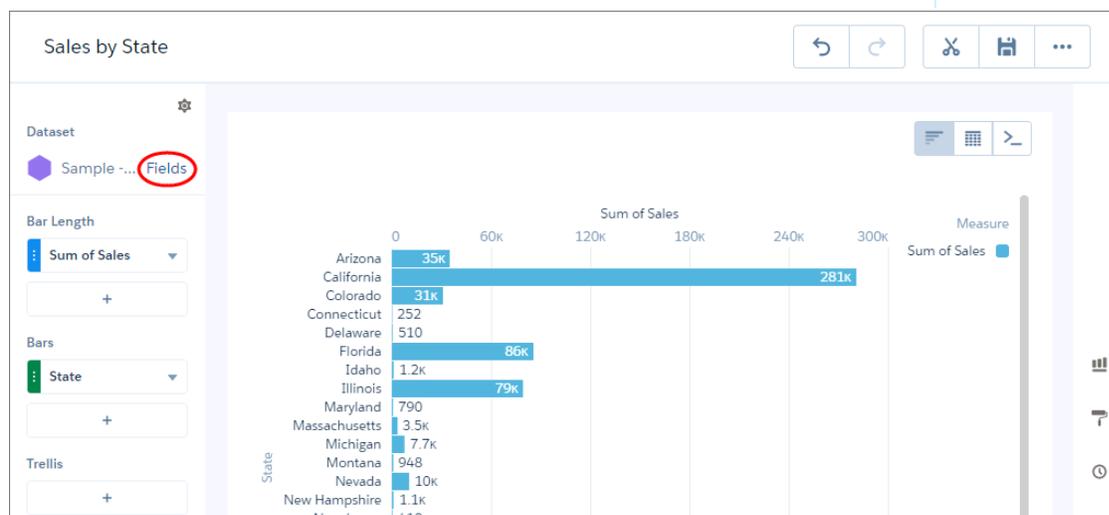
2. To reveal a column's context menu, hover over the right edge of the column.
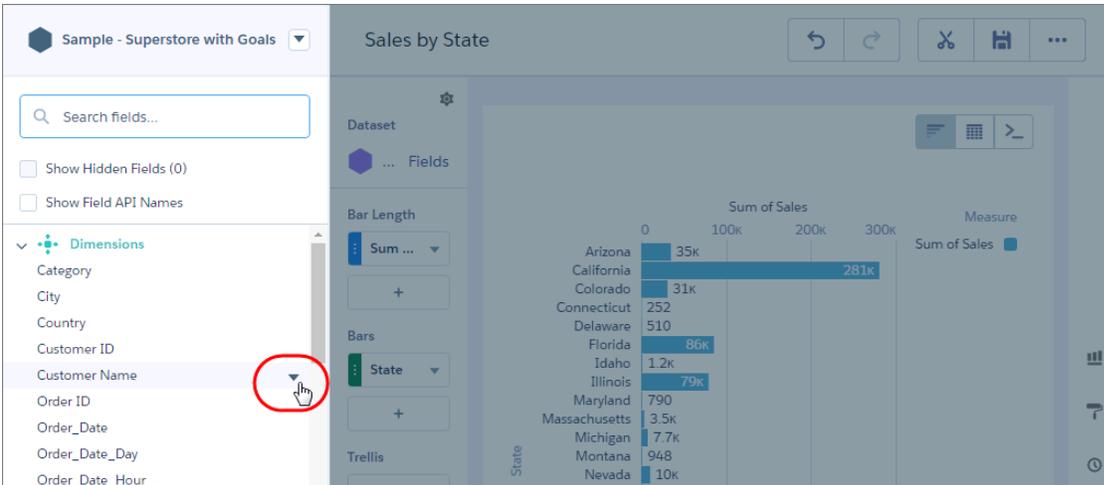
3. Click the arrow to reveal the context menu.
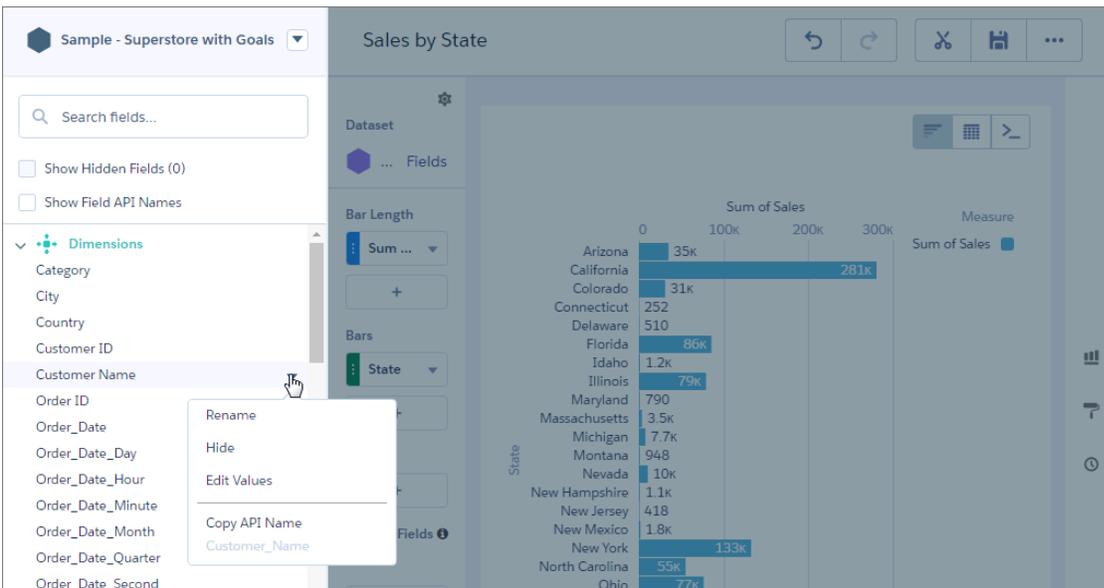
4. Select **Rename** to rename the column's visible name.

Instead of using the menu, you can click any column's name to edit it. You can copy the column's API name, but you cannot change it. To display every column's API name in the Dataset Fields list of columns, click **Show Field API Names**.

5. To hide a column, select **Hide** from its context menu.

The column is hidden from the Dataset Fields list of columns, and hidden from users who explore the dataset, and hidden from all lenses or dashboards that currently use the dataset. However, the column is still accessible via lens and dashboard JSON, Analytics REST API, and SAQL queries. To view hidden columns in the Dataset Fields list, click **Show Hidden Fields**.
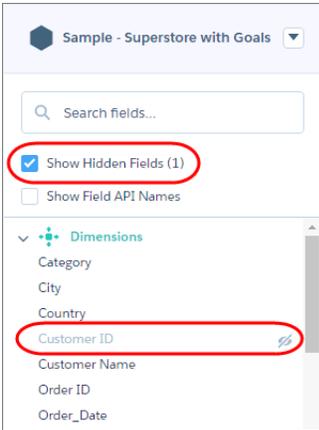
6. For measures, you can choose to apply a number format.
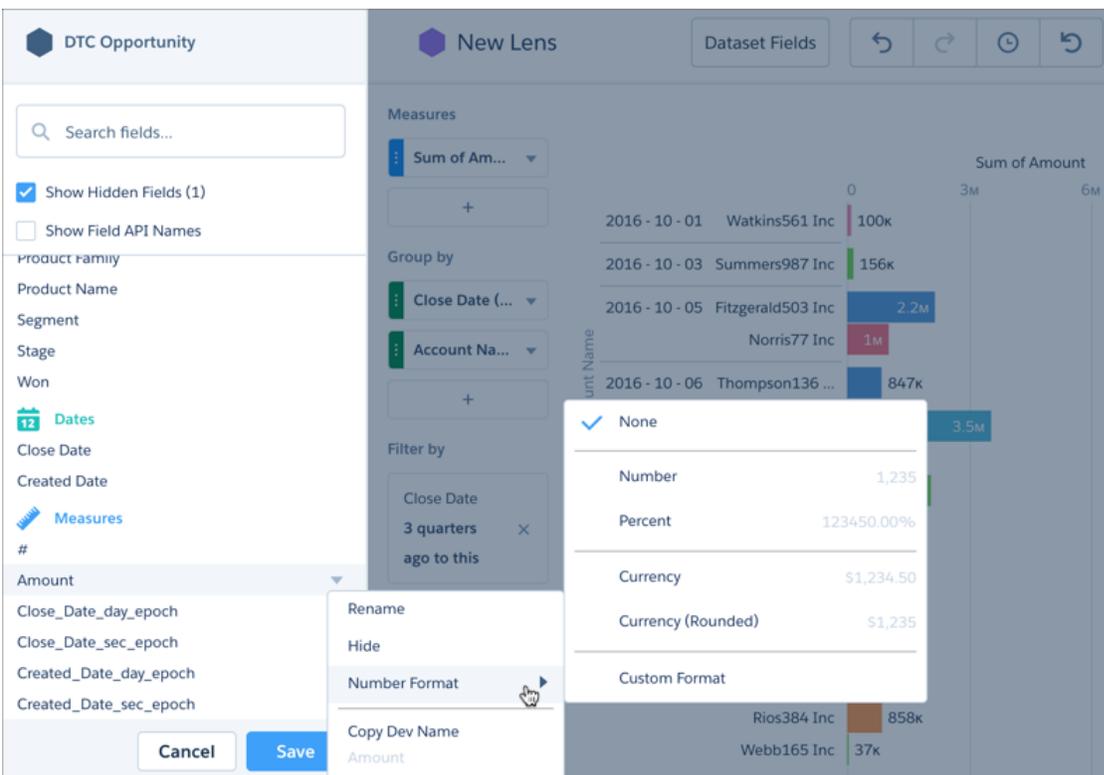
7. Click **Save** to save the changes.

Choose the Default Dataset Columns That Appear in a Table

Make tables your own by choosing the default columns. For example, show the most commonly used columns in a particular order. If you don't set the defaults, the first five measures and five dimensions are selected in alphabetical order.

1. To configure the dataset columns, click **Fields** while exploring the dataset.
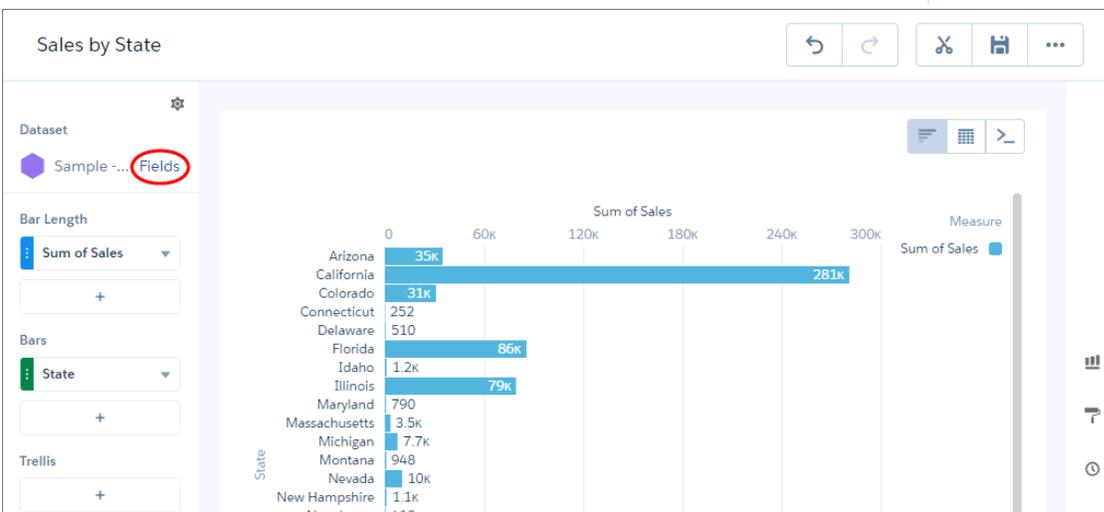
2. Click the dropdown arrow next to the dataset name, and click **Default Fields**.
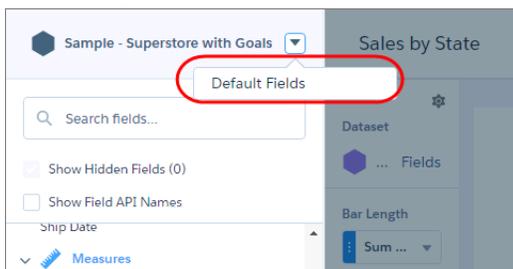
3. Select the columns to show by default, use the up and down arrows to reorder them, and click **Update**.
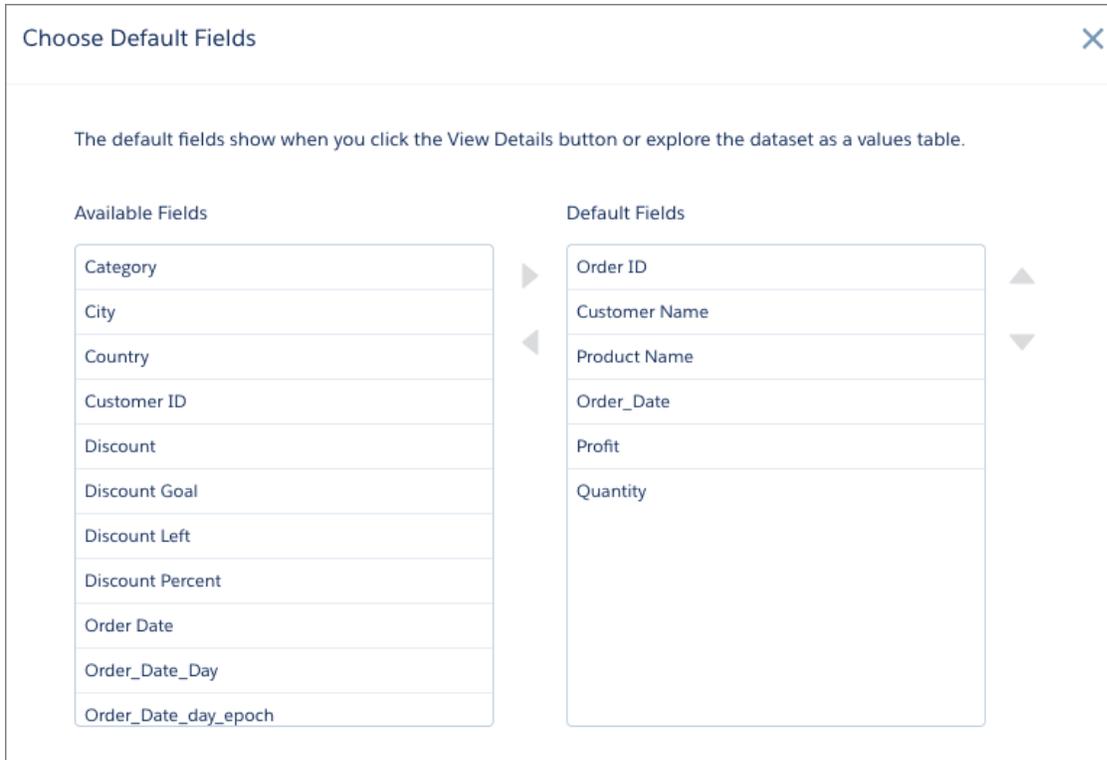
When you explore the dataset as a values table, the selected columns appear in the specified order.
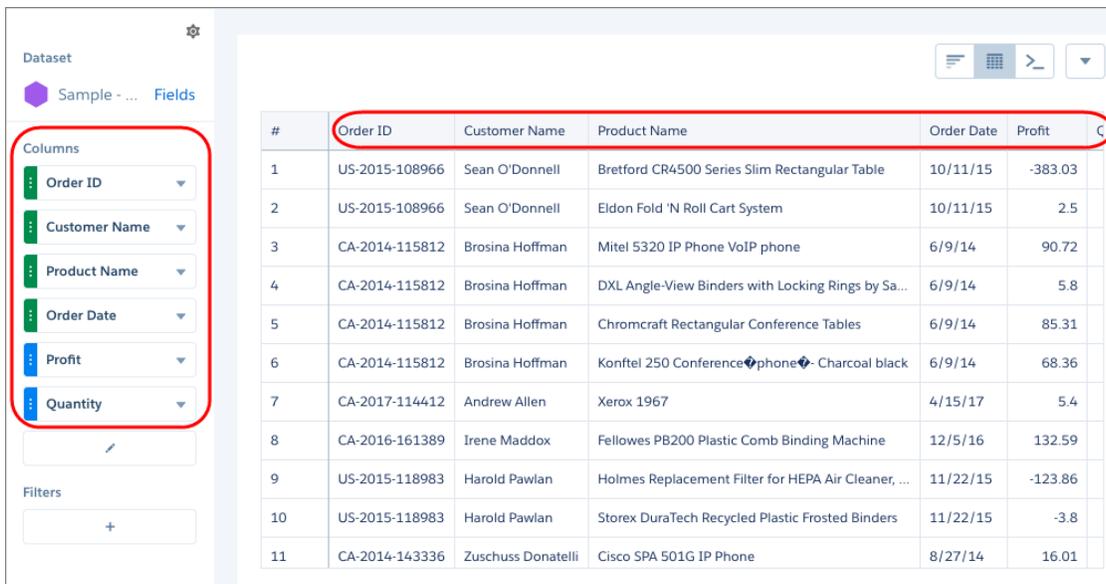
You see the same table columns and order when you click **View Data Table** to view the data in a tabular format.
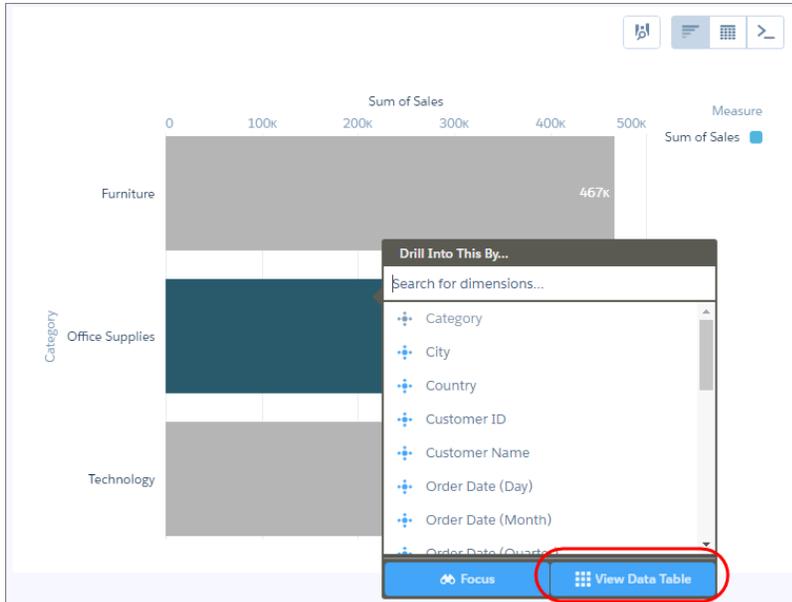
Change the Labels and Colors of Dataset Dimension Values

Change the display labels and default colors of dimension values to make them easier to understand and spot. The modifications don't alter the underlying data stored in the dataset or the API names of the columns. The changes affect only the appearance in the user interface.

Because these changes are done at the dataset level, the changes affect all charts and tables that use this dimension.

> ✏️ **Note:** You can override the default colors for a specific chart or table by setting conditional formatting in the widget properties.

1. To change colors or labels for all charts that use a selected dimension, click **Fields** while exploring a dataset as a lens.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.
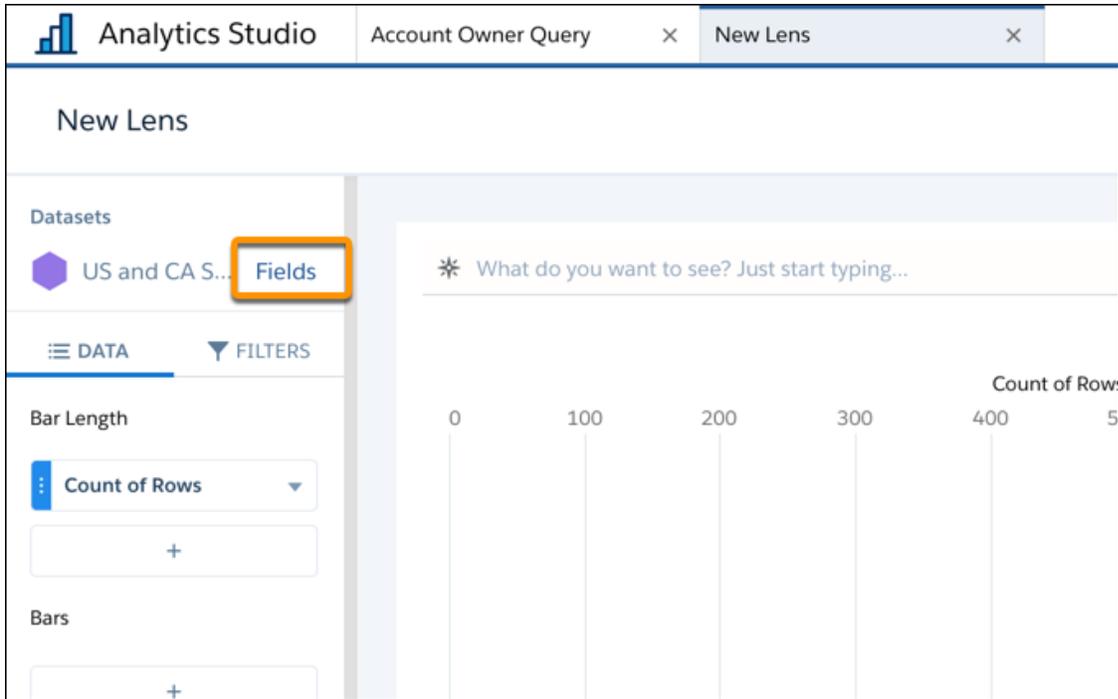
2. Click the dropdown arrow next to the dimension, and select **Edit Values**.
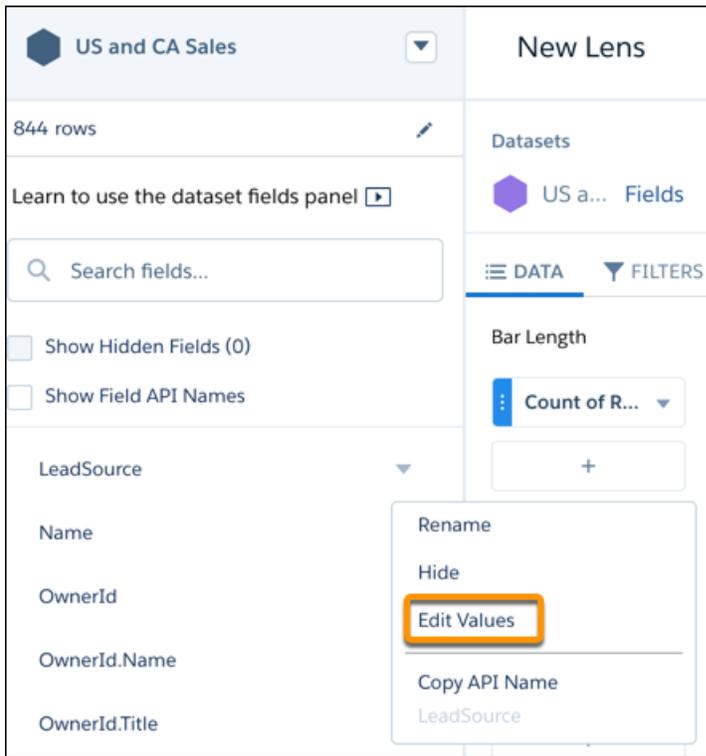
3. To change the color, click the square icon on the right (1) and select the color (2).
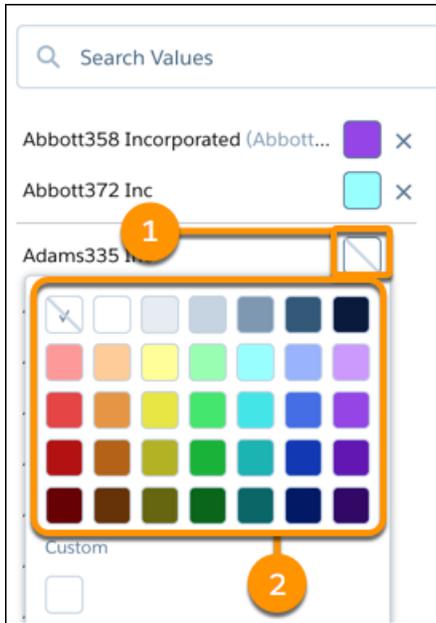
4. To change the label, click the text box and enter a new label.

If you change the colors of dimension values, the color doesn't appear in a chart until you set up conditional formatting on the column in the chart's widget properties. For more information, see Automatically Highlight Data with Conditional Formatting.

Restore a Dataset Version

Restoring a dataset rolls back its data to a previous version. Restore a dataset to reverse a recent update or to undo the results of a recipe or dataflow change.

🛑 **Important:** Restoring a dataset can impact the dashboards, lenses, recipes, and dataflows that use it. Refer to the usage information on the dataset's edit page to review the impacted assets before you restore.

1. From the Data Manager, click the **Data** tab.

2. Hover over the dataset that you want to restore, click ▼ , and then click **Edit Dataset**.

3. Click the Restore Dataset button (⟲).

4. Click the version you want to restore.

📝 **Note:** The prior two versions are stored, with any older versions deleted after 24 hours. In limited situations, a deleted version can be restored by Salesforce. Contact Salesforce Customer Support for recovery evaluation.

5. Click **Restore**.

6. Click **Restore** again to confirm.

7. To view the restore data job, click the **data monitor** link.

You see the restore data job on the Monitor tab of the data manager. If Tableau CRM can't restore the dataset, the job fails. Try restoring a later version of the dataset.

> **⊘ Important:** Restoring a dataset has no effect on associated dataflows or recipes. It's possible that when an associated dataflow or recipe next runs, it can undo the results of a restore.

Delete a Dataset

Delete unnecessary datasets to reduce app clutter and avoid reaching your org's total row limit for all registered datasets. You can delete datasets from shared apps on which you have at least Editor access, your My Private App, and, with a special user permission, another user's My Private App.

Before you delete a dataset, consider the following guidelines.

- You can't recover a deleted dataset.

- Use the data manager to delete datasets from another user's My Private App. You can't see or delete other users' private datasets from the Tableau CRM home or app tabs. For security reasons, you also can't view the data in other users' private datasets.

- You also can't delete a dataset that's used in a dashboard, lens, or dataflow. Before you delete a dataset, first remove references to it from dashboards or dataflow transformations, and delete associated lenses.

  > **💡 Tip:** To see where a dataset is used, review the usage information on the dataset's edit page.

  Usage

  This dataset is used in these places

Dashboards **Lenses** **Dataflows**	
Dashboard	Last Accessed
Opportunities	Aug 4, 2017 at 10:19 AM

- Analytics doesn't check or show if a dataset is used in recipes. Be sure to remove dataset references from recipes as well. If you delete a dataset that's used in a recipe, the recipe fails the next time it runs.

1. To locate the Delete Dataset button on the Tableau CRM home or an app page, follow these steps.

   a. Under the Datasets section, click [▼] next to the dataset that you want to delete.

   b. Click **Edit**.

2. To locate the Delete Dataset button in the data manager, follow these steps.

   a. In the data manager, click the **Data** tab.

   b. On the Datasets subtab, click [▼] next to the dataset that you want to delete.

   c. Click **Edit Dataset**.

3. On the dataset edit page, click the Delete Dataset button ([🗑]).
   If the dataset is in use, Analytics Cloud stops the deletion and shows a list of assets that reference the dataset. Click **Got It** and resolve these references before trying again.

4. Click **Delete Permanently**.

Datasets

A dataset is a collection of related data that is stored in a denormalized, yet highly compressed form. There are multiple ways to create datasets. You can also edit datasets after they've been created.

💡 **Tip:** To ensure that you don't reach the row limit for all registered datasets, delete unused datasets.

SEE ALSO:

Upload a CSV File to Create a Dataset

Edit a Dataset

Delete a Dataset

Monitor

The monitor provides options for you to manage jobs.

SEE ALSO:

Run a Dataflow Manually

Schedule a Dataflow to Run Automatically

Monitor a Dataflow Job

Monitor an External Data Upload

Data Manager

The data manager is where you monitor your data jobs, prepare datasets with recipes and dataflows, and connect to data.

To open the data manager in Analytics Cloud, click the gear icon (⚙) and then click **Data Manager**.

You land on the Monitor tab, where you can see the progress of your recipe, data sync, and other dataflow jobs. The Dataflows & Recipes tab gives you access to your two key data preparation tools. Go to the Data tab to view your datasets and connected data, and use them

to create recipes. If you have data sync enabled, the Connect tab is where you can view and update settings for your synced Salesforce objects, and set up connections to external data.

External Data

You can integrate external data into Analytics Cloud to make the data available for queries from explorer and designer.
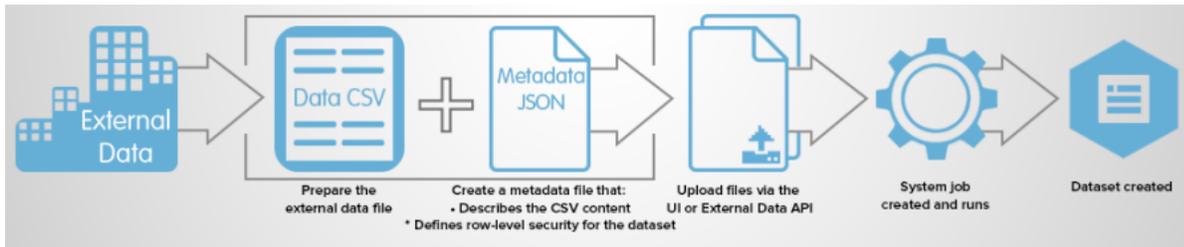
The External Data API enables you to upload external data files to Analytics Cloud. The External Data API can upload .csv files, and you can optionally specify the structure of your data by defining metadata in JSON format.

The External Data API is available in API version 31 and later.

The high-level steps for uploading external data by using the API are:

1. Prepare your data in CSV format, and then create a metadata file to specify the structure of the data.

2. Connect programmatically to your Salesforce organization.

3. Configure the upload by inserting a row into the InsightsExternalData object, and then set input values such as the name of the dataset, the format of the data, and the operation to perform on the data.

4. Split your data into 10-MB chunks, and then upload the chunks to InsightsExternalDataPart objects.

5. Start the upload by updating the `Action` field in the InsightsExternalData object.

6. Monitor the InsightsExternalData object for status updates, and then verify that the file upload was successful.

Gain Insights on Data Directly in Snowflake

You can explore your data in a Snowflake warehouse without loading data into Tableau CRM or preparing it in datasets. Tableau CRM Direct Data for Snowflake lets you set up a live connection to the Snowflake schema you want to explore. You can run queries on the Snowflake schema associated with the connection in real time. You can also build dashboard charts and tables based on the queries you run on Snowflake data.

When to Use Tableau CRM Direct Data for Snowflake

To run analysis on your Snowflake data, you can use Tableau CRM Direct Data for Snowflake or Snowflake input connectors or a combination of the two.

Tableau CRM Direct Data for Snowflake Tips and Considerations

It helps to be familiar with these tips and considerations when working with Direct Data for Snowflake.

Create a live connection to Snowflake warehouse to explore your data stored there.

After you complete setup, Tableau CRM Direct Data for Snowflake lets you run queries on the dataset associated with the connection in real time. You can also build dashboard charts and tables based on the queries you run on Snowflake data.

When to Use Tableau CRM Direct Data for Snowflake

To run analysis on your Snowflake data, you can use Tableau CRM Direct Data for Snowflake or Snowflake input connectors or a combination of the two.

Refer to these guidelines to determine whether Direct Data for Snowflake meets your business needs.

With Direct Data for Snowflake, you can:

- Save time on setup and eliminate the need for data syncs, dataflows, or recipes to keep queries based on Snowflake data up to date.

- Eliminate the need to load and prepare large volumes of Snowflake data into Tableau CRM and manage its dataflows.

- Free up data pipelines for the resources and processes that require them.

- Get real-time metrics in dashboards and lenses with queries based on Snowflake data.

- Meet data residency requirements on Snowflake data but still analyze it to make business decisions.

- Maintain on-demand access to less frequently used data, such as order details, without having to load it into Tableau CRM.

- Incorporate data from Snowflake Data Marketplace for deeper insights on your Snowflake data.

- Take advantage of Snowflake's native support for analytics on data in semi-structured data formats, such as JSON, Parquet, or XML.

- Quickly create and test which data models to use with your Snowflake data before connecting it to Tableau CRM. You can even collaborate easily with other users during this process.

SEE ALSO:

Tableau CRM Direct Data for Snowflake Tips and Considerations

It helps to be familiar with these tips and considerations when working with Direct Data for Snowflake.

Tips for Working with Direct Data for Snowflake

To improve the performance of dashboards based on Snowflake queries, it's important to optimize their query times. Queries that run faster reduce the possibility of reaching query concurrency limits. Applying sound dashboard design principles is also helpful in ensuring that Snowflake-related dashboards run smoothly.

For optimized query times, keep these tips in mind:

- Work with your Snowflake administrator to optimize Snowflake warehouse and data for analytic queries.

- To make it easier for users to creating queries, consider creating SQL views of the Snowflake data for your users.

- Consider creating materialized views in Snowflake for most frequently run queries.

- For larger tables, create filtered views that focus on subsets of data.

- Run the dashboard inspector to monitor and track query times.

- To optimize query times:

- Select only the columns needed for your visualization.

- Apply filters to limit the number of rows.

- Avoid grouping on high-cardinality fields.

- Avoid using the *contains* operator. Instead, use the *equals* operator.

Apply these design principles to dashboards based on Snowflake queries:

- Limit the number of queries on a single dashboard page. Instead, spread queries across multiple pages.

- To minimize the number of queries being executed, reuse the same query in multiple widgets whenever possible.

Direct Data for Snowflake Considerations

Keep these behaviors in mind when working with Tableau CRM Direct Data for Snowflake.

- Row-level security isn't applied on Snowflake data. So, users with access to a live dataset can view and explore all data rows in the associated Snowflake table or view.

- Dashboard widgets based on a live dataset appear blank to users without access to that dataset.

- The following features aren't available with Direct Data for Snowflake:

  - Drill, Focus, and View Data Table actions on lenses. As a workaround, add filters manually.

  - Compare tables and formulas

  - Custom SQL queries

  - Data-blended queries

  - Downloads as CSV or Excel files

  - Global filters. As a workaround, use list, range, and date widgets in dashboards to filter data.

  - Actions, notifications, subscriptions, and Einstein Analytics Watchlists.

- You can't use live datasets as sources in dataflows or recipes.

- You can't create stories on live datasets.

- You can't run join operations on live datasets.

Create a Live Connection to Snowflake

Create a live connection to Snowflake warehouse to explore your data stored there.

Create the Live Connection

1. In Tableau CRM, click the gear icon (⚙) and select **Data Manager**.

2. In the data manager, click the **Connect** tab.

3. Click **Connect to Data**.

4. Click **Live Connections**.

5. Click **Add Connection**.

6. Click **Snowflake Direct Connector** and enter its settings, as described in the Connection Settings section.

7. When done, click **Save** or **Save and Create Dataset**.

Create Live Datasets for a Snowflake Source Table

After creating the live connection, create a live dataset that let users explore Snowflake tables or views available with the connection.

📝 **Note:** Live datasets are similar to Tableau CRM datasets, except that the data remains in an external data source.

After creating the live connection, enable a live dataset for the connection.

1. In the data manager, click the **Data** tab.

2. On the Live Datasets tab, click **Create Dataset**. Select the connection and enable a Snowflake source table that is available with the connection.

3. Name the live dataset and assign it to a Tableau CRM app. Use the app to control which Salesforce users, roles, and groups have access to explore Snowflake data with the connection.

4. When done, click **Create Live Dataset**.

To begin exploring, click the live dataset from the Datasets tab in Tableau CRM Studio.

Snowflake Live Connections Settings

All settings require a value, unless otherwise indicated.

Setting	Description
Connection Name	Identifies the live connection. Use a convention that lets you easily distinguish between different connections.
Developer Name	API name for the connection. This name must be unique, begin with a letter, and can only contain underscores and alphanumeric characters. This name can't include spaces, end with an underscore, or contain two consecutive underscores. You can't change the developer name after you create the connection.
Description	Description of the connection for internal use.
Schema	Snowflake schema whose data you want to explore with the connection. You can either add all the tables you want to explore to a single Snowflake schema or create a separate connection for each schema. 📝 **Note:** To keep data secure, we recommend creating a separate Snowflake schema specifically for Direct Data for Snowflake.
Password	Password for the Snowflake user specified in Username.
Role	Snowflake role assigned to the user you're using to connect. 📝 **Note:** To keep data secure, we recommend connecting with a Snowflake user granted a read-only role.
Warehouse	Snowflake warehouse name.
Username	Snowflake username to connect to the schema.
Account	Name of your Snowflake account.

Direct Data for Snowflake Limits

Query Limits

Limit	Value
Maximum concurrent queries per organization	25
Maximum concurrent queries per user	5
Maximum number of rows returned per query	5,000
Query timeout	2 minutes

API Call Limits

Limit	Value
Maximum concurrent API calls per org	100
Maximum API calls per user per hour	10,000

SEE ALSO:

Snowflake Connection

Explore Data Directly in Snowflake

After you complete setup, Tableau CRM Direct Data for Snowflake lets you run queries on the dataset associated with the connection in real time. You can also build dashboard charts and tables based on the queries you run on Snowflake data.

To begin exploring, click the live dataset from the Datasets tab in Tableau CRM Studio.

USER PERMISSIONS

To create live connections to Snowflake:
- Enable Tableau CRM Direct Data for external data sources

To explore data directly in Snowflake:
- Explore External Data Directly

Data stored in the Snowflake object opens in a lens, which you can interactively explore and visualize.
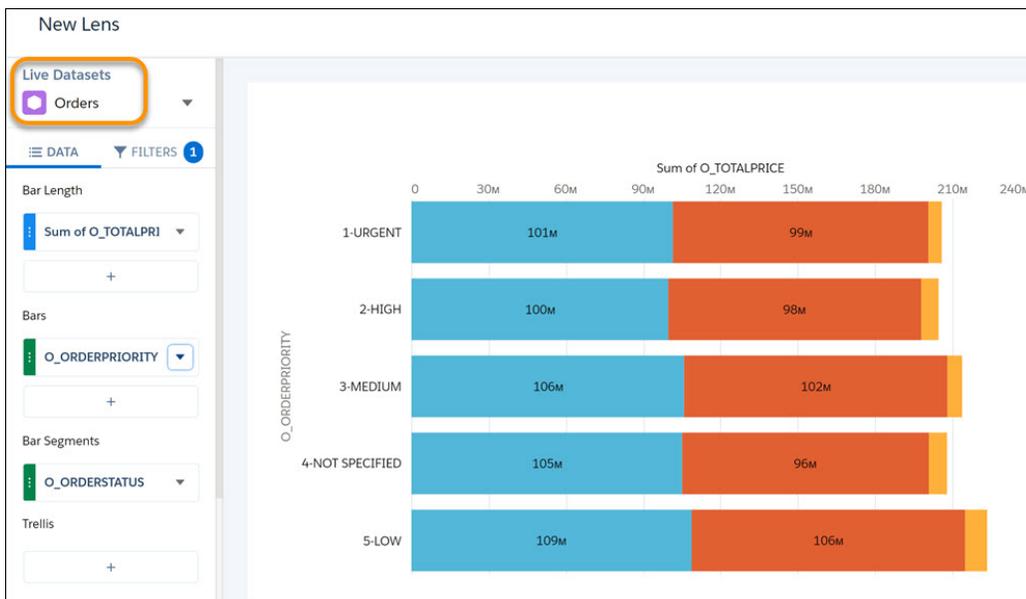
SEE ALSO:

[Snowflake Connection](#)

Explore and Visualize Your Data in Tableau CRM

When sifting through an overwhelming amount of business data, sometimes you know exactly what you're looking for. Other times you don't, but you recognize it when you see it. Salesforce Analytics Cloud gives you a fast, fluid way to discover the compelling stories within your data and to create the right visuals to tell your story.

After your data is uploaded, explore freely. Drill through the data by using interactive visualizations. Surface unexpected insights that can transform your business. Then let Analytics Cloud do the work of producing dynamic charts that look sharp and communicate volumes.

Learn to Use Explorer

See what unexpected insights you can surface by interactively exploring and visualizing your data, using explorer tools.

Explore Data with Tables

You can use tables to get a view of the data that is close to the underlying dataset, and you can use tables to manipulate and extend the data to expose fresh insights. With values, compare, and pivot tables, Tableau CRM explorer gives you options for both of those goals.

Visualize Data With Charts

If your goal is to understand vast amounts of business data, and to communicate that understanding with coworkers, partners, and customers, being able to visualize your data is critical. Tableau CRM provides a chart for every need, each a means for illustrating key aspects of your business in just the right way.

Clone a Lens

Build upon a visualization by cloning its containing lens to a new tab where you can continue exploring. You can save the original in its own tab.

Save a Visualization

Save your visualization as a lens.

Share a Visualization

Share a visualization with your colleagues by posting to Chatter, getting its unique URL, downloading a screenshot of it, or downloading its filtered data. A Chatter post provides an image and a link to the asset—lens, dashboard, or app—in Tableau CRM. Colleagues with the link and access to the asset can drill down and explore the information that's presented. To share without giving access to the asset, use the download options.

SEE ALSO:

Converse with Your Data

Learn Tableau CRM with In-App Examples

Learn to Use Explorer

See what unexpected insights you can surface by interactively exploring and visualizing your data, using explorer tools.

View Your Data in a Lens

When you look at data in an exploratory mode, you do so in a *lens*. A lens has several aspects: it's how you view data in a dataset, it's the place where you explore the data graphically, and it's the basis for building any dashboard.

Group Your Data into Categories

Group data to organize it so that the data is displayed in useful categories.

Filter Your Data

Apply filters in a lens to unclutter your chart and focus on the subset of data that's most relevant for your business. You can create partial filters on measures and dimensions. You can also create global lens filters on measures, dimensions, or dates and apply them to the compare table in a single step. You can even apply filter logic to a lens or a dashboard step.

Change the Sort Order

Choose whether data in your chart visualization is sorted ascending, sorted descending, or unsorted.

Change the Chart Type

Switch between chart types to see which visualization options tell the most compelling story for your data.

Change Measures

Your choice of measure defines your entire visualization and determines the scope of your exploration. For example, change the measure and the aggregation method from "Count of Opportunities" to "Sum of Amount." Then add a second measure to see 2 visualizations side by side.

Format Numbers

Choose either preset or custom number formats. Specify custom formats for positive, negative, and zero values.

Drill Deeper Into Data

You can drill deeper into your data to explore the underlying information and surface hidden insights.

Focus on Selected Data

You can focus your exploration by changing the view to include only specific data from an existing table or chart.

Explore Multiple Datasets with a Single Query

Answers to business questions sometimes require context beyond the current dataset. Or, your dataset structure may not match the way you want to explore your data. To surface insights that require data from disparate datasets, blend the data from multiple datasets in the explorer. Then query the blended data just as you would with a single dataset.

Change the Chart Scale

With some data, a chart displays values that range widely on an axis and make the visualization more difficult to understand. You can constrict the range's scale by applying a logarithmic scale. Scaling is available for certain charts only.

Return to a Previous View by Using History

Exploring your data can lead you down many paths. Not all of them are fruitful, and that's OK. Explorer keeps a full history of your activity in a lens. If you've changed your lens in undesired ways, or just want to see those nifty animations again, use history to backtrack quickly to a previous state in your visualization.

Limit the Size of Your Query Results

Interested in reviewing just the top 10 accounts in your product pipeline—without excessive scrolling? To fast-track to meaningful insights and for improved performance on your dashboards, you can limit the size of query results in a lens.

View the Query Behind Your Lens

Tableau CRM uses Salesforce Analytics Query Language (SAQL) or Salesforce Object Query Language (SOQL) behind the scenes in lenses and dashboards to gather data for visualizations. You can view and edit the underlying code that's written as you explore, or copy it for use elsewhere.

Clip a Lens to a Dashboard

To include the lens in a dashboard, clip it. When you clip the lens, Tableau CRM adds a query to the most recently used dashboard. If a dashboard isn't open, Tableau CRM adds the query to a new dashboard in dashboard designer. After you add the query to the designer, you can apply it to a widget in the dashboard. The widget overrides the visualization settings of the lens—the widget determines how to display the results of the query.

View Your Data in a Lens

When you look at data in an exploratory mode, you do so in a *lens*. A lens has several aspects: it's how you view data in a dataset, it's the place where you explore the data graphically, and it's the basis for building any dashboard.

From the home page, click a dataset or a lens. A lens opens in a new tab.
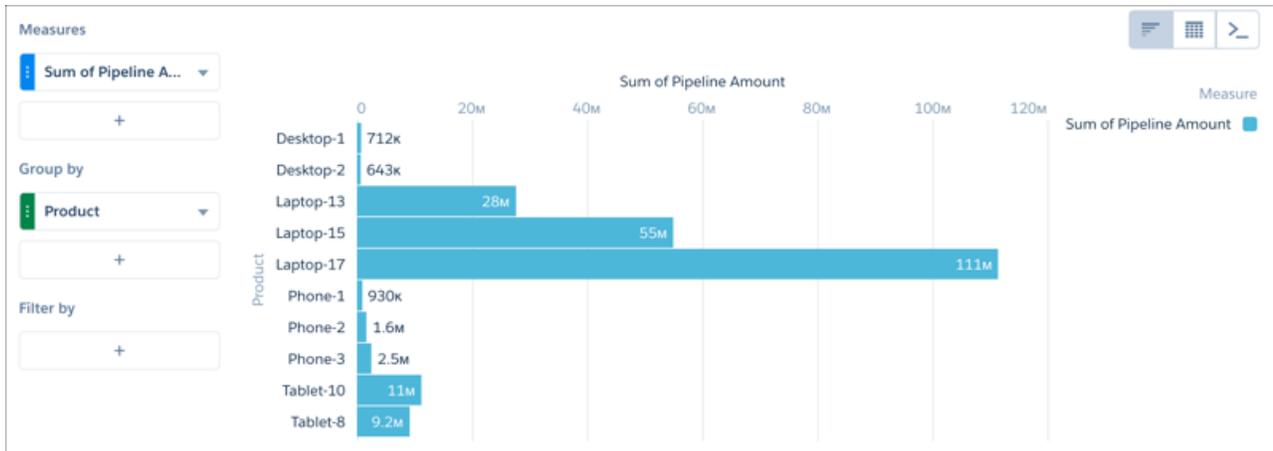
Group Your Data into Categories

Group data to organize it so that the data is displayed in useful categories.

For example, to look for seasonal patterns in your sales cycle, group opportunity data by the month when deals close.

1. Explore a dataset containing opportunity stages, pipeline amounts, and dates, setting the measure to pipeline amount.

2. Click the **Vertical Axis** plus button, click the opportunity stage date, and then select month as the dimension that you want to group by.

    The dimensions in your data determine which categories you can group by.

USER PERMISSIONS

To view visualizations:
*   Use Analytics
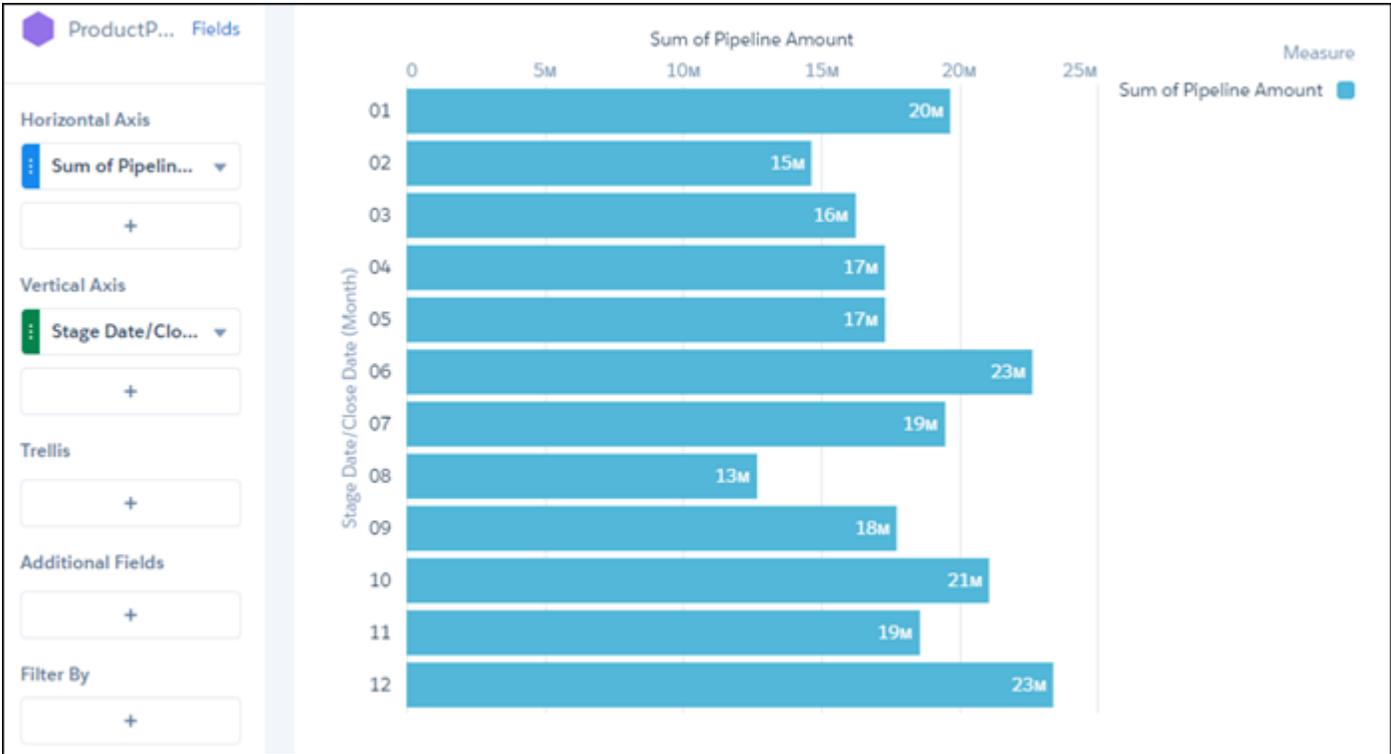
3. To find the second dimension that you want to group by, click the **Vertical Axis** plus button, then enter `stage` in the search field.
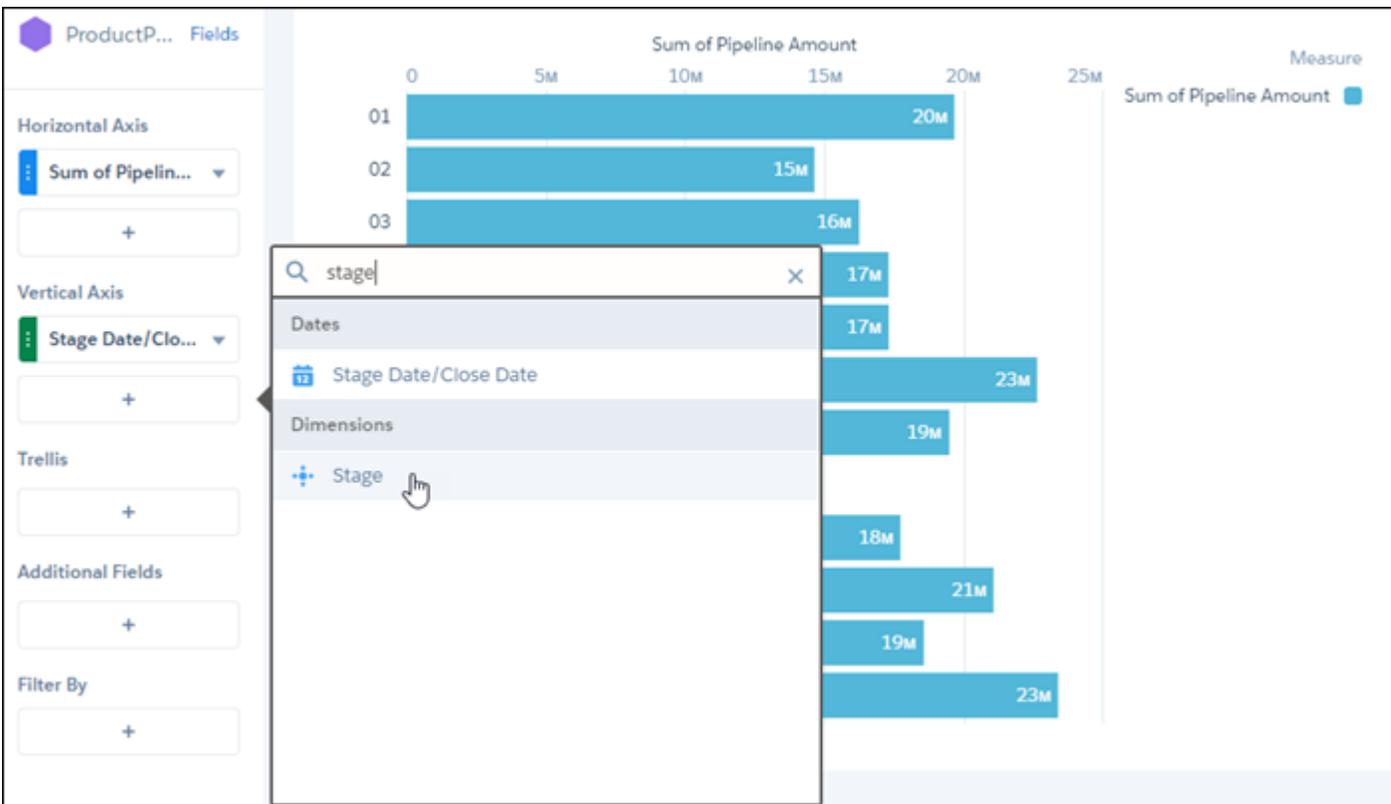
4. Click **Stage**, the second dimension you want to group by.

Pipeline amounts are shown grouped by stages, and stages are shown grouped by month. In this dataset, closed amounts are shown as negative numbers because those amounts fall out of the pipeline.
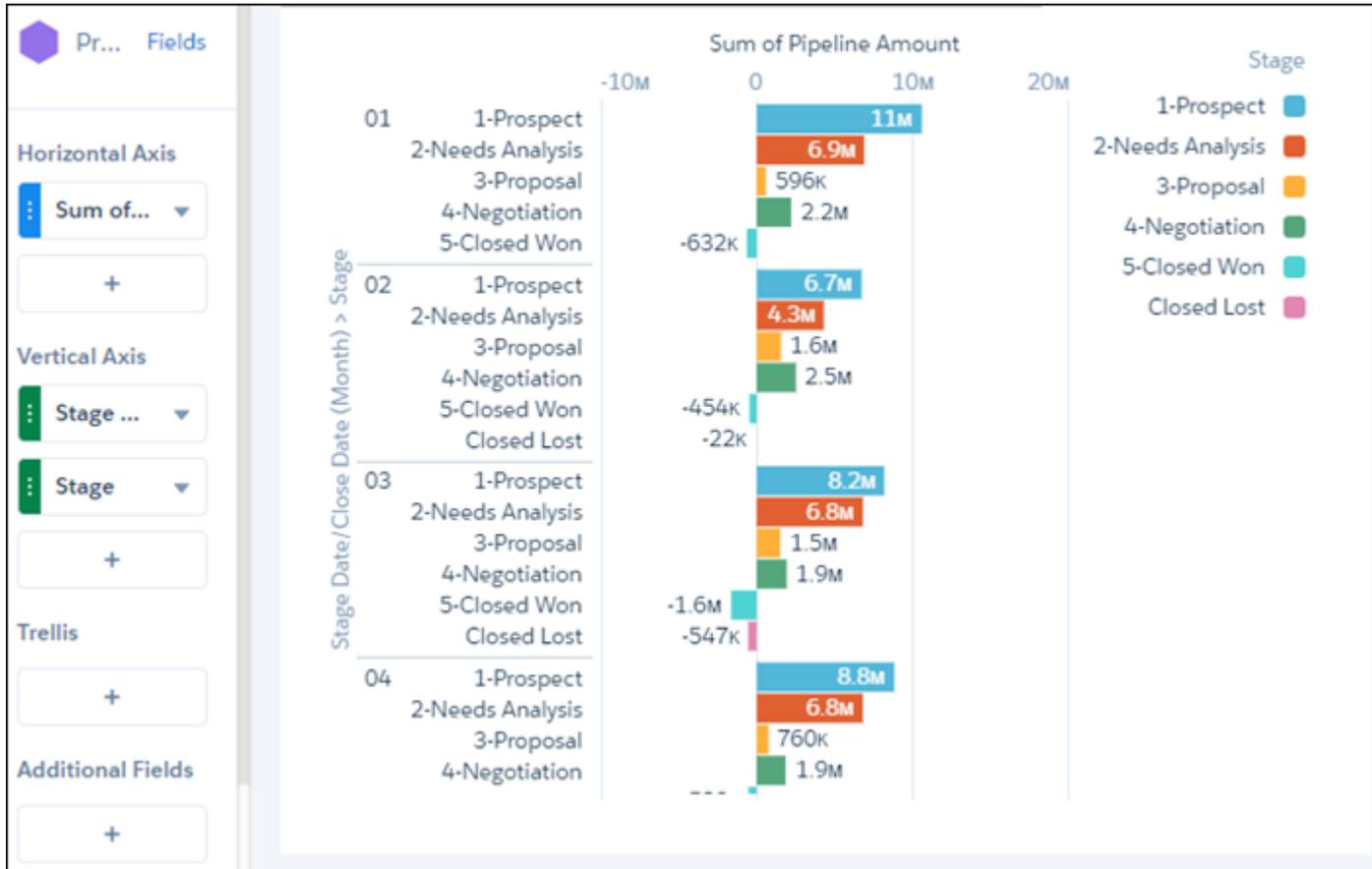
5. Click the filters plus button, then click the stage dimension.

6. Set the filter to equal the closed stages only.
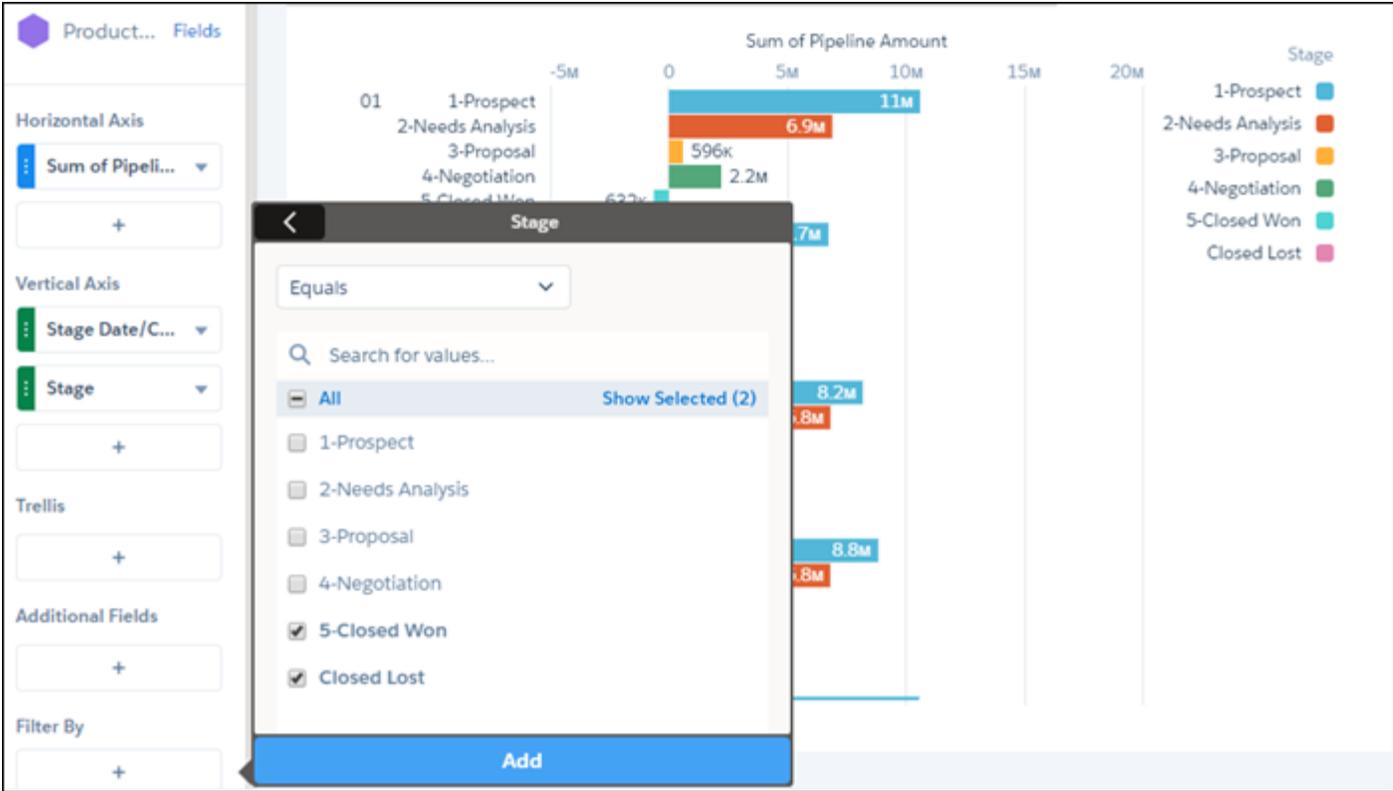
The resulting bar chart shows the information you're after, but it's not easy to interpret.
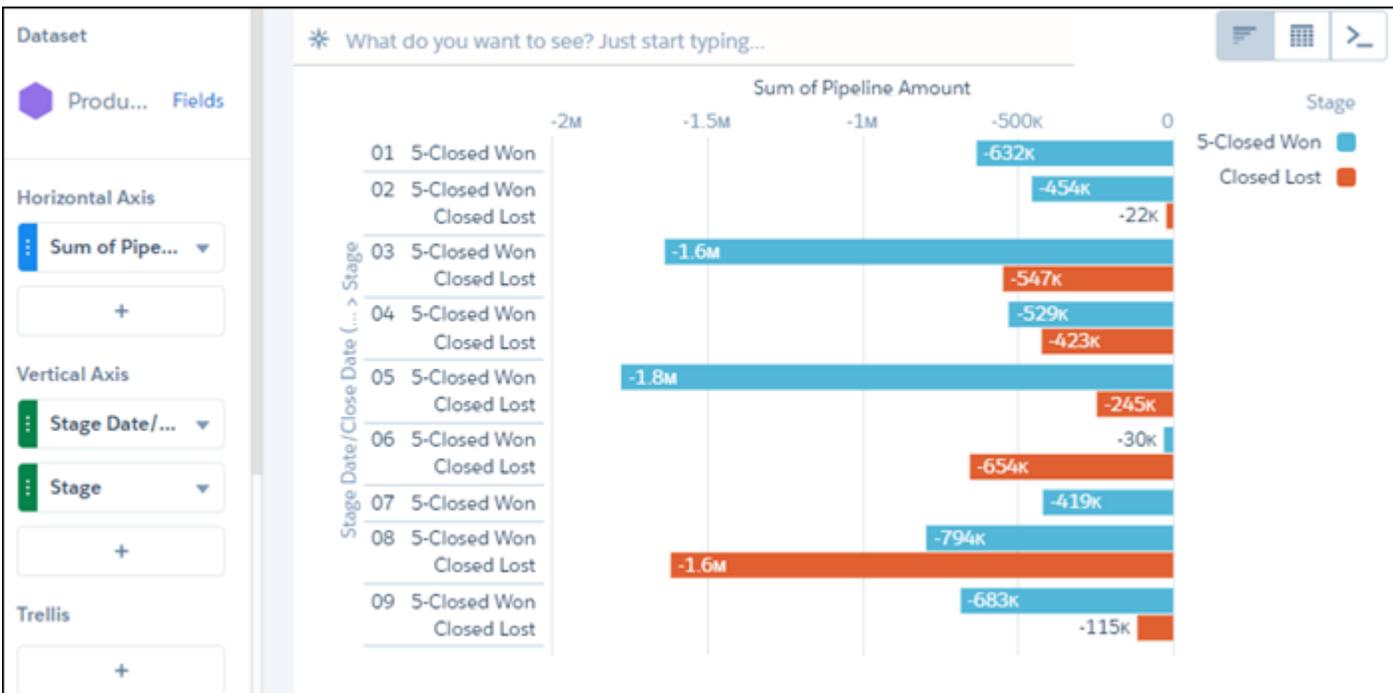
7. Open the chart menu and select stacked waterfall.

The bar chart changes from a single bar to a waterfall chart that show the data grouped separately by month. It also provides an overall total for the year. You can now easily see how much each month has contributed in both lost and won closed deals.
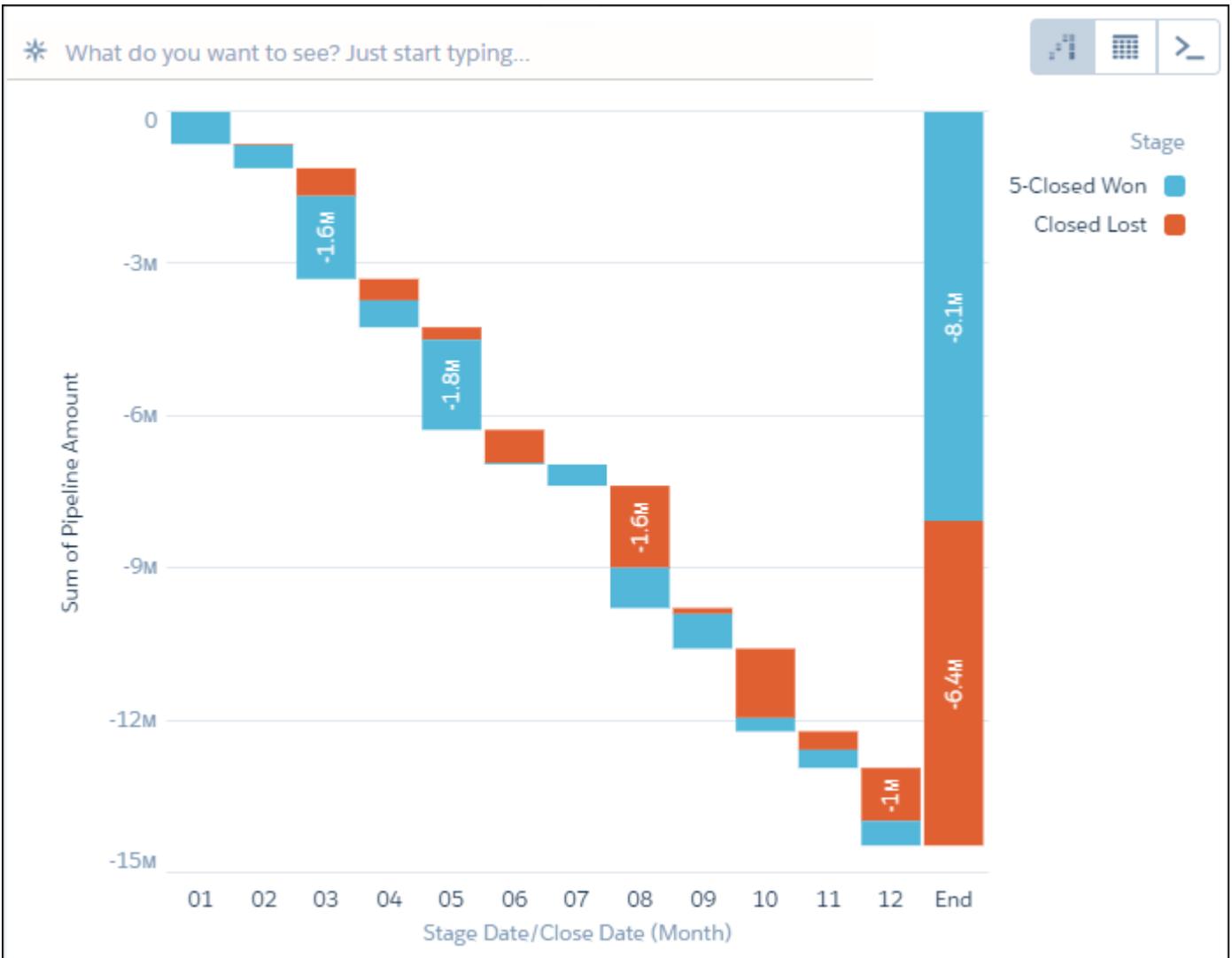
📝 **Note:** When a dimension is grouped, that dimension's null records are excluded from results, which affects the count of rows in a lens. To prevent the exclusion of null dimension records, assign a default value to those records.

Filter Your Data

Apply filters in a lens to unclutter your chart and focus on the subset of data that's most relevant for your business. You can create partial filters on measures and dimensions. You can also create global lens filters on measures, dimensions, or dates and apply them to the compare table in a single step. You can even apply filter logic to a lens or a dashboard step.

> 💡 **Tip:** Don't want filters to be applied to the dataset as you drill down? Then, define filters on individual measures, which don't persist. On the other hand, global lens filters are preserved when you switch to **View Details** or the values table.

Here's an example to limit the view of opportunities to the ones with a substantial amount in agriculture or healthcare.

1. On the Filters tab, click the filters plus button.

   In the filters dialog box, aggregated measures and measures are listed at the top, followed by dates and dimensions.

2. In the filters dialog box, add the following filters:

   - **Industry** equals **Agriculture**
   - **Amount** greater than *100,000*
   - **Industry** equals **Healthcare**
   - **Amount** greater than *200,000.*

USER PERMISSIONS

To view visualizations:
- Use Analytics
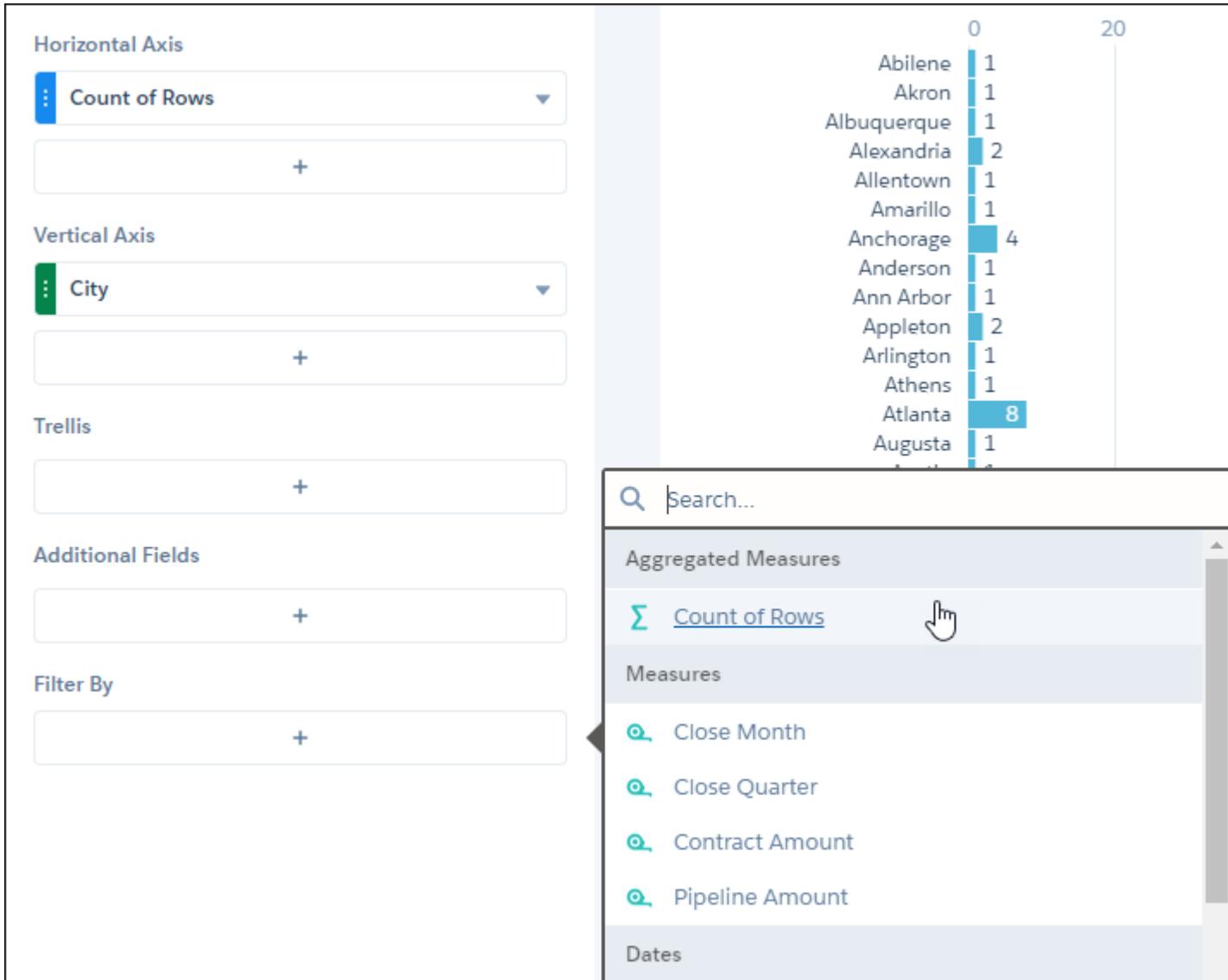
3.
To add filter logic, click [▼] in the Filters tab.

4. Select **Add Filter Logic**.

5. In Edit Filter logic, build the filter logic as follows: *(1 AND 2) OR (3 AND 4)* and click **Apply**.
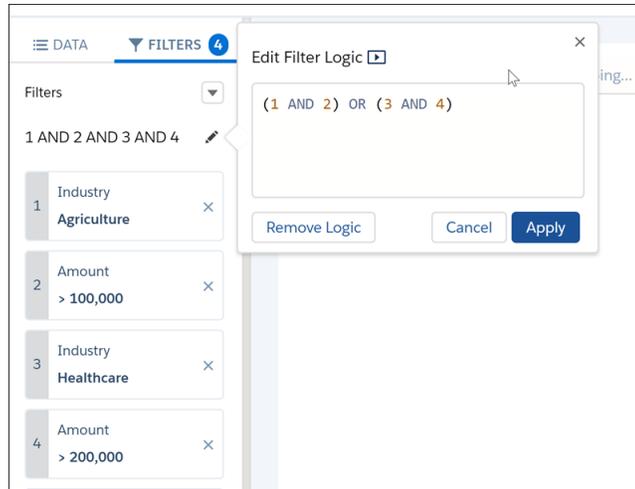
Normally, measure filters in multiple-column steps remain in the filter panel. However, in table mode, filters move to the measures to which they apply.

If a calculated column is created in table mode, then filters remain with their measures even if the lens is set to chart mode.
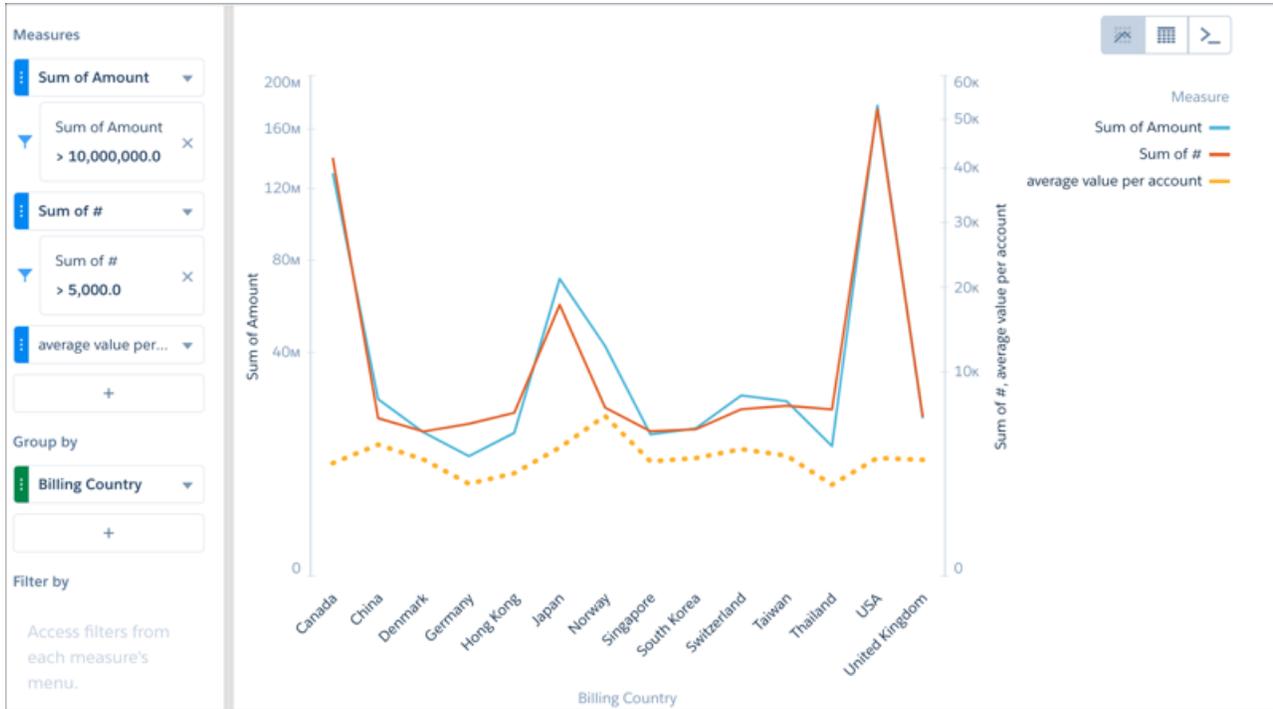
You can apply filters to new measures using their context menus.

Change the Sort Order

Choose whether data in your chart visualization is sorted ascending, sorted descending, or unsorted.

1. Click the actions menu for the measure you want to sort. For dimensions, click the actions menu in chart, compare table, or pivot table mode.
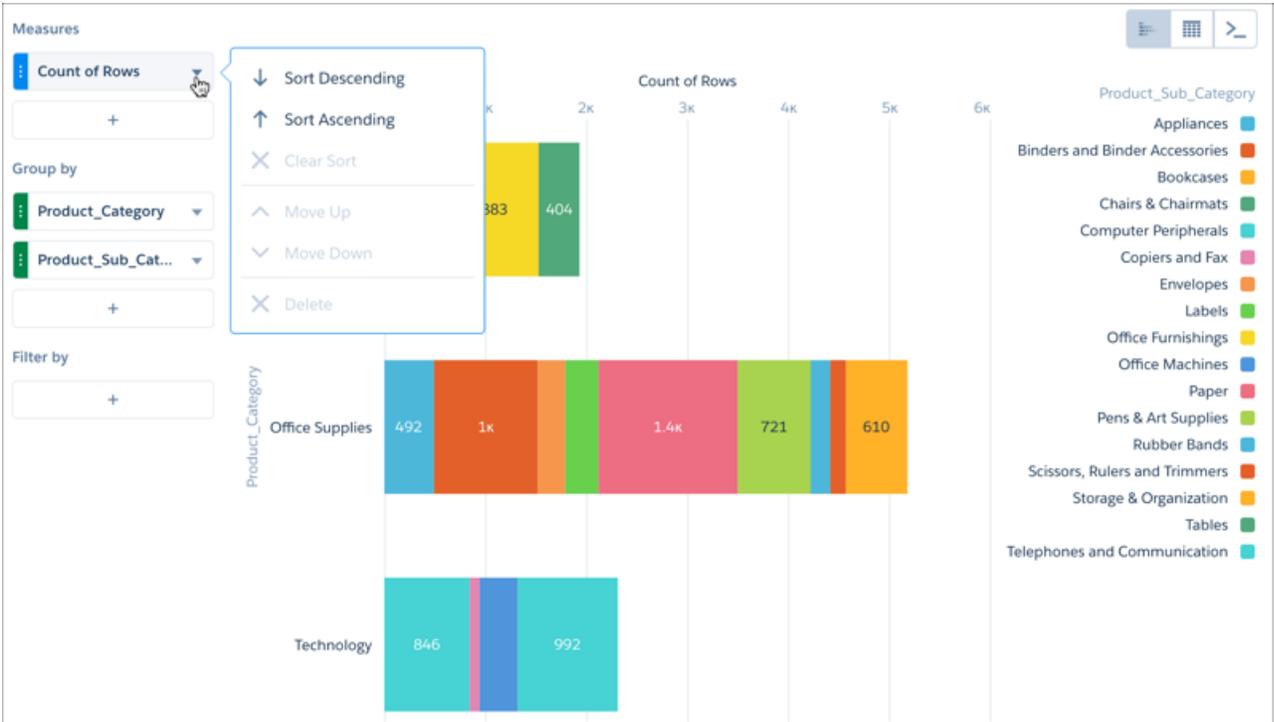
2. Select **Sort descending** or **Sort ascending**.

   If the measure or dimension is already sorted, you can unsort it by selecting **Clear sort**.

   Note: To show consistency across multiple bars in stacked charts, the stacked segments are always sorted alphabetically within each bar. Stacked charts include stacked bar, stacked column, stacked pyramid, and stacked waterfall. To help you locate a segment in a bar, the legend uses the same sort order as the bar segments.
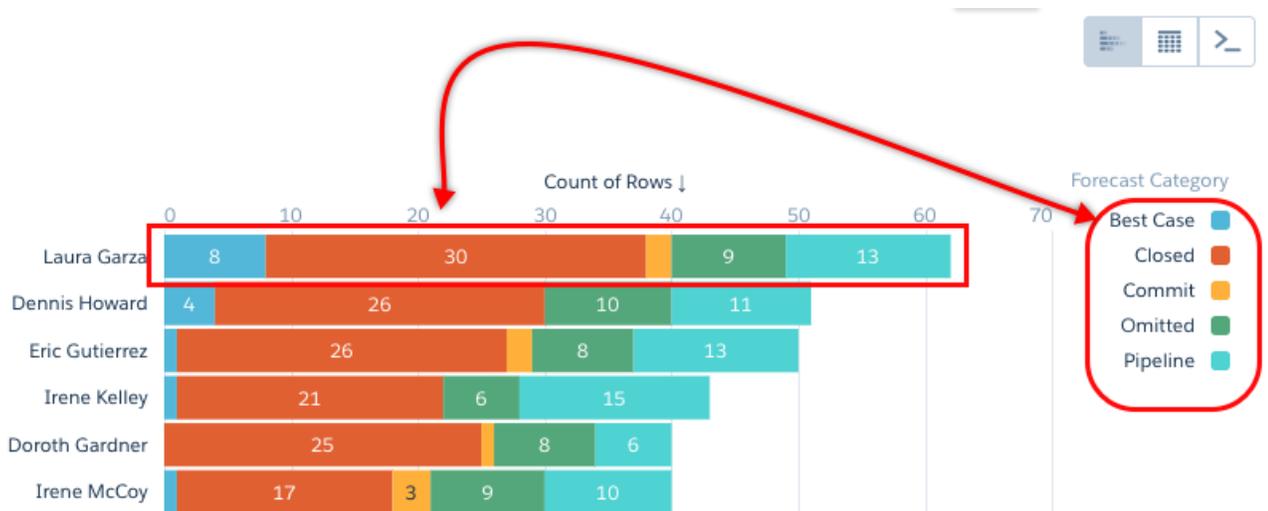
> **Note:** Sorting isn't supported on measures in pivot tables.

SEE ALSO:

  Tips for Working with SAQL Queries in the Query Editor

Change the Chart Type

Switch between chart types to see which visualization options tell the most compelling story for your data.

1.
  Click the **Charts** icon ([image]) in the quick access menu.

A gallery of charts that are available for this lens is displayed.
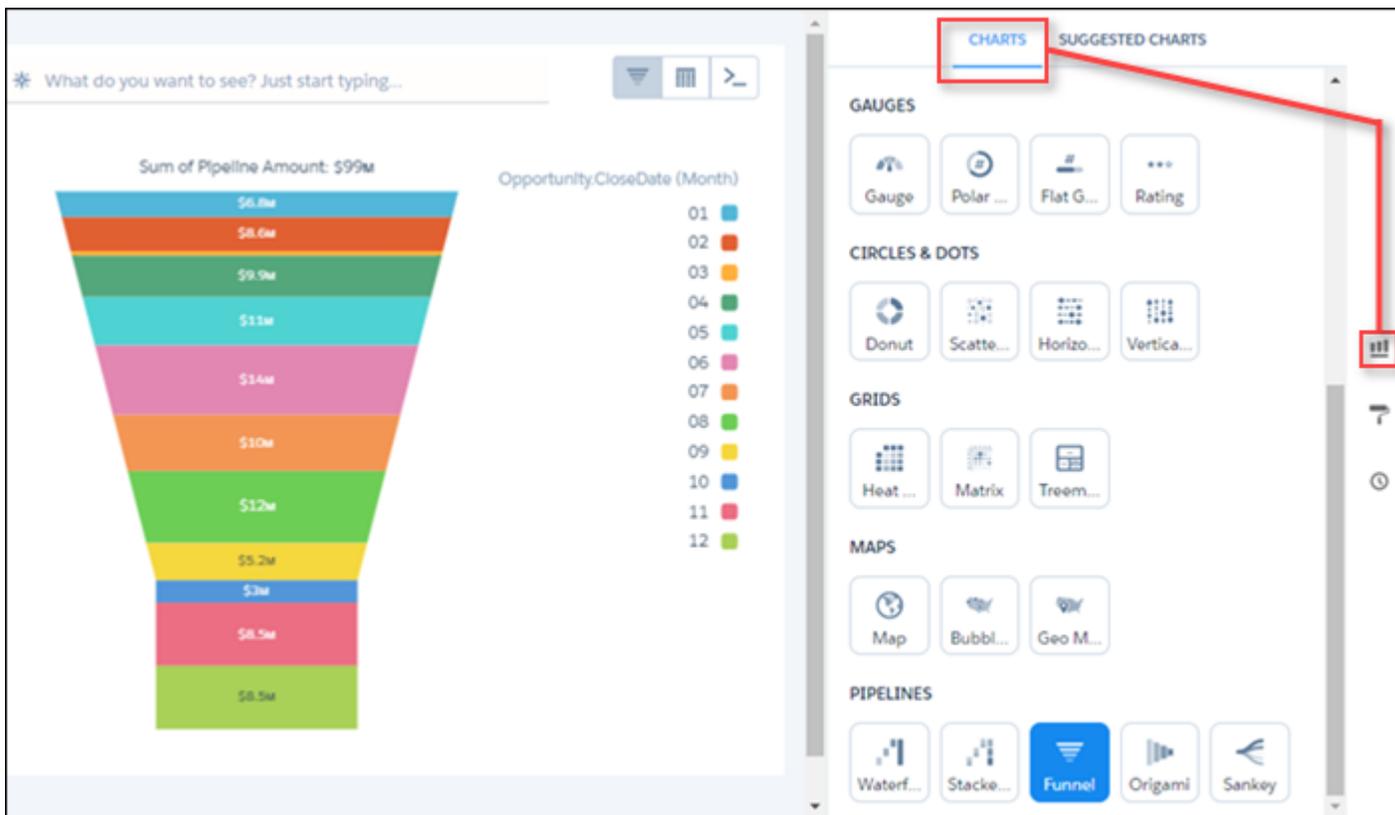
2. Hover over a chart type to see how many measures and dimensions that type of visualization requires.

   For example, a donut chart can have one measure and one or two dimensions.

3. Click a chart type, such as Stacked Bar.
   The chart visualization changes.

Change Measures

Your choice of measure defines your entire visualization and determines the scope of your exploration. For example, change the measure and the aggregation method from "Count of Opportunities" to "Sum of Amount." Then add a second measure to see 2 visualizations side by side.

1. To explore it in a lens, click a Tableau CRM dataset.

   By default, the initial measure is Count of Rows.

2. To open the Measures selection menu, click **Count of Rows**.
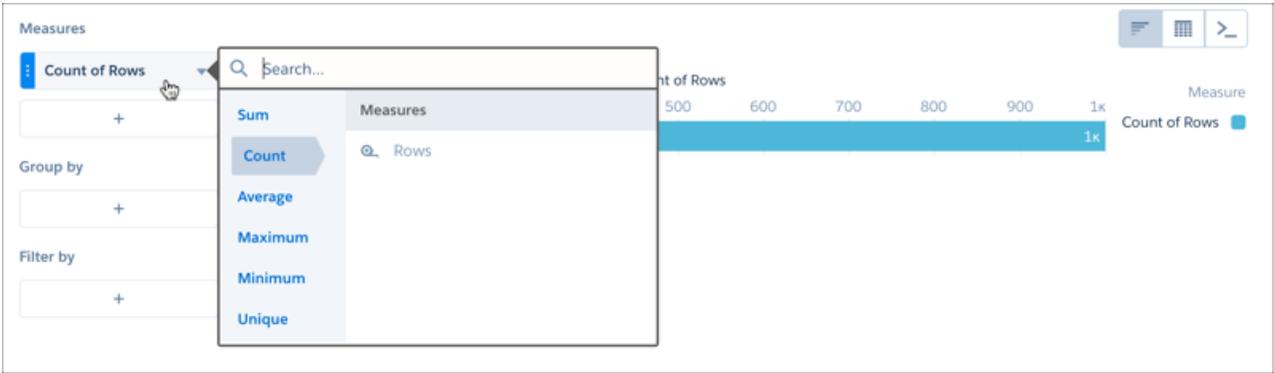
3. Click **Sum**, and click the measure you want to visualize.
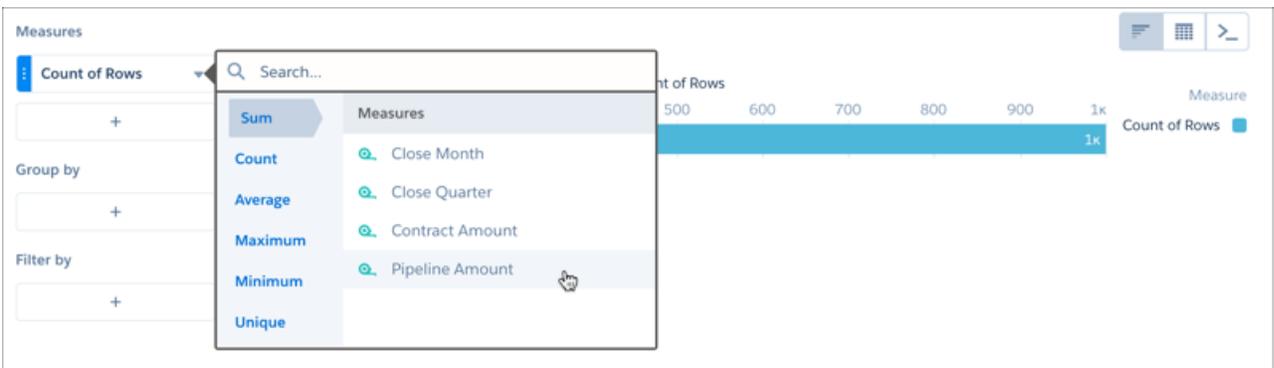
📝 Note: To count the number of unique occurrences of a measure or a dimension, click **Unique**, then select the measure or dimension you want to count.

After you group the data, the chart displays the chosen measure by group value.
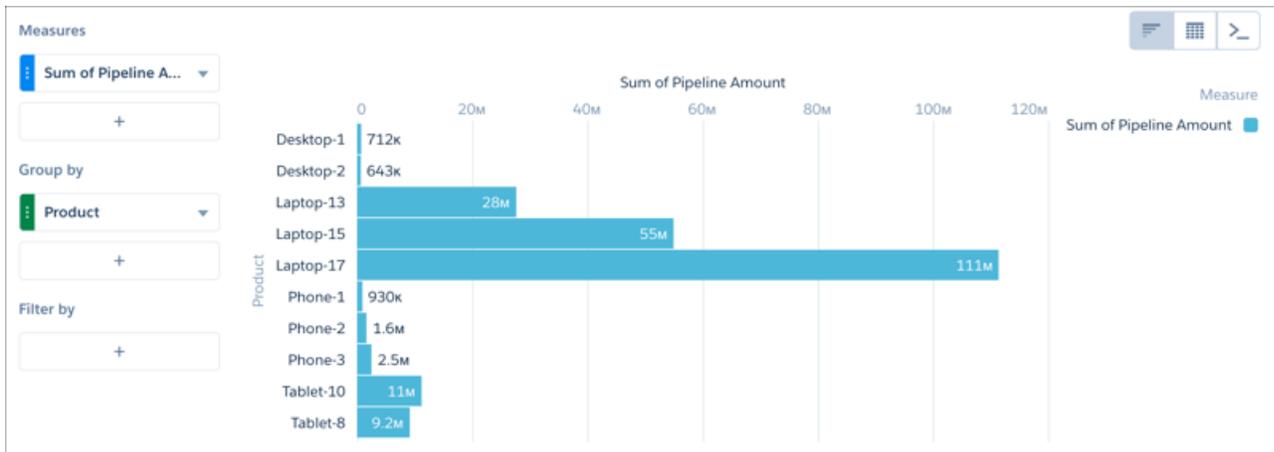
Format Numbers

Choose either preset or custom number formats. Specify custom formats for positive, negative, and zero values.

1. Click [▼] on any measure field.

2. Choose **Format Numbers** and choose a preset format.

a. Click **Custom…** for more presets or to specify your own custom format.

b. Use a preset or enter a custom string in the **Format** string.

For multiple formats, use this syntax: *<POSITIVE_FORMAT>;<NEGATIVE_FORMAT>;<ZERO_FORMAT>*.

The **Format** string always uses a comma (**,**) for the thousands separator and a period (**.**) for the decimal separator. Change the separator in the **Thousands** or **Decimal** string to override the separator in the **Format** string. CSV downloads ignore the specified separators and always use the default values of comma (**,**) for thousands and period (**.**) for decimal.

3. Click **Done** to save the format.

> Note: In the Spring '20 release we unified number behavior in charts, tables, and number widgets.

- 0 values follow formula logic. For example, a format string $#.## with a 0 value returns $0.00.
- Default behavior never returns a negative 0.
- Short numbers like 1.0x always return 1.0x, not 1x.
- Negative and zero formats are optional.

Drill Deeper Into Data

You can drill deeper into your data to explore the underlying information and surface hidden insights.

1. Select the data you want to drill into by clicking datapoints in a chart or rows in a table.

   For multiple selections, continue clicking, or click and drag to use a selection box. To deselect datapoints or rows, click them. To undo all selections, click the space just outside the datapoints or table.

   📝 **Note:** Drill selections are not available in values tables.

2.
   To open the Drill Into This By... menu, right-click the data or click 📊 .
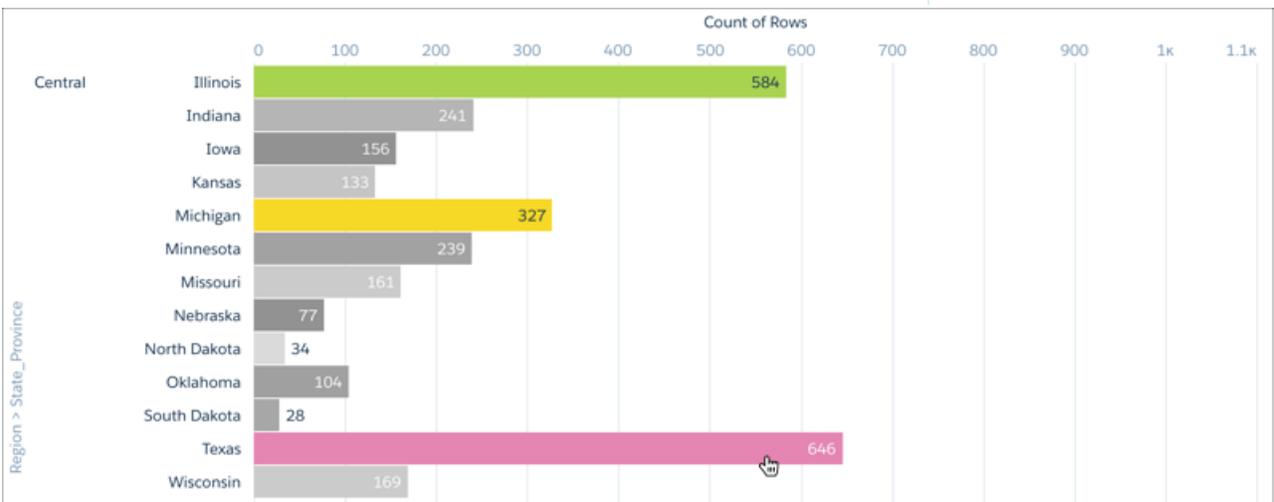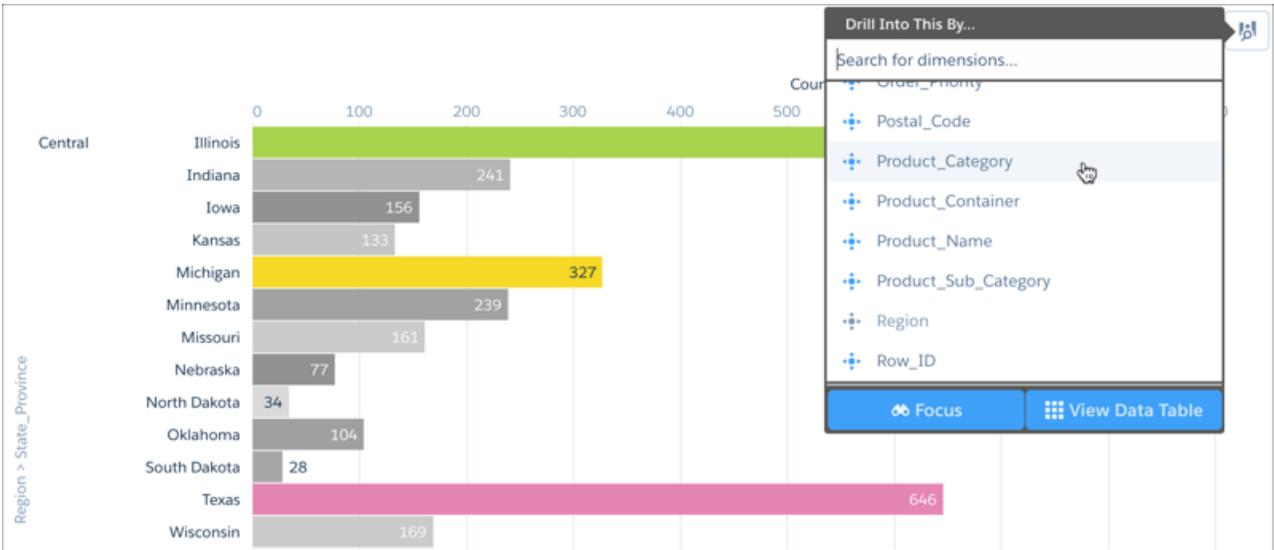
3. To view the underlying data for a dimension, select it from the Drill Into This By... menu.

A new visualization grouped by the selected dimension and filtered by the selected data appears in place of the previous visualization.
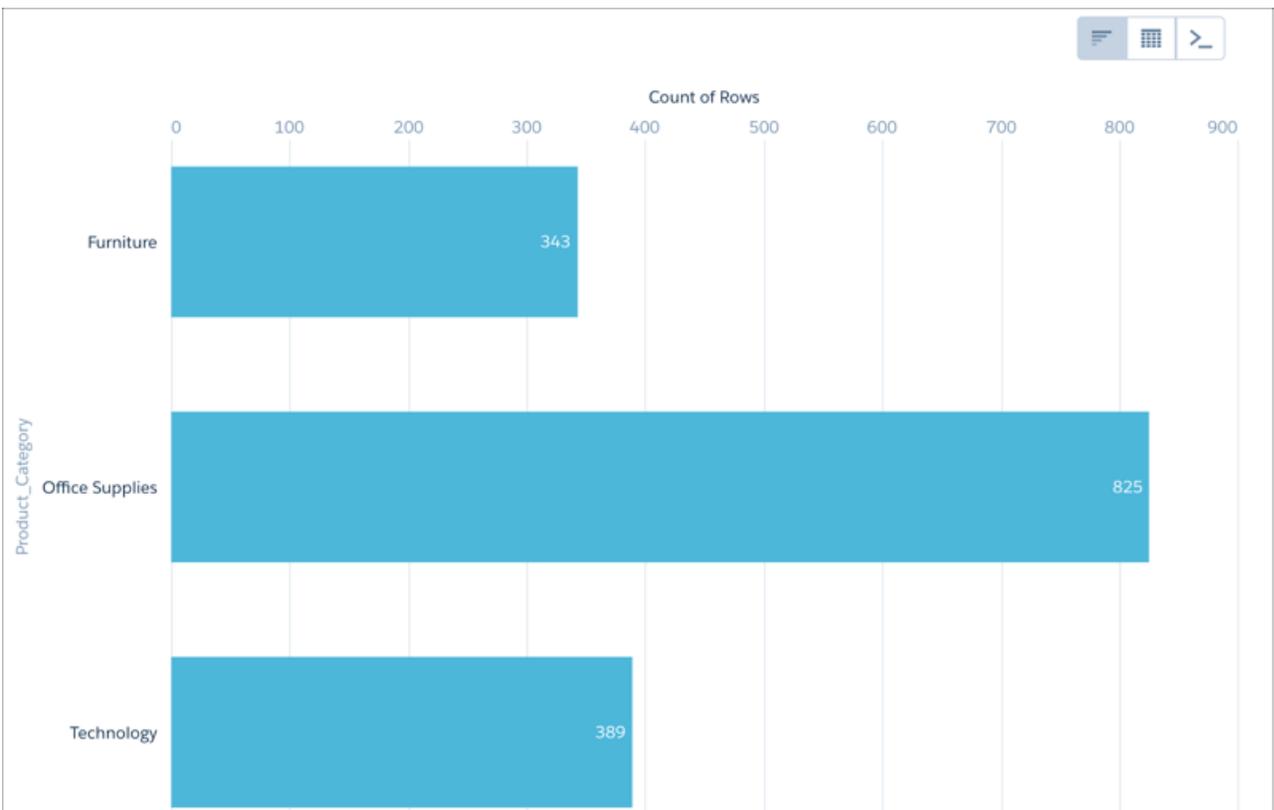
To return to the previous table or visualization, click [⤺].

Focus on Selected Data

You can focus your exploration by changing the view to include only specific data from an existing table or chart.

1. Select the data you want to focus on by clicking datapoints in a chart or rows in a table.

   For multiple selections, continue clicking, or click and drag to use a selection box. To deselect datapoints or rows, click them. To undo all selections, click the space just outside the datapoints or table.

   📝 **Note:** Focus selections are not available in values tables.
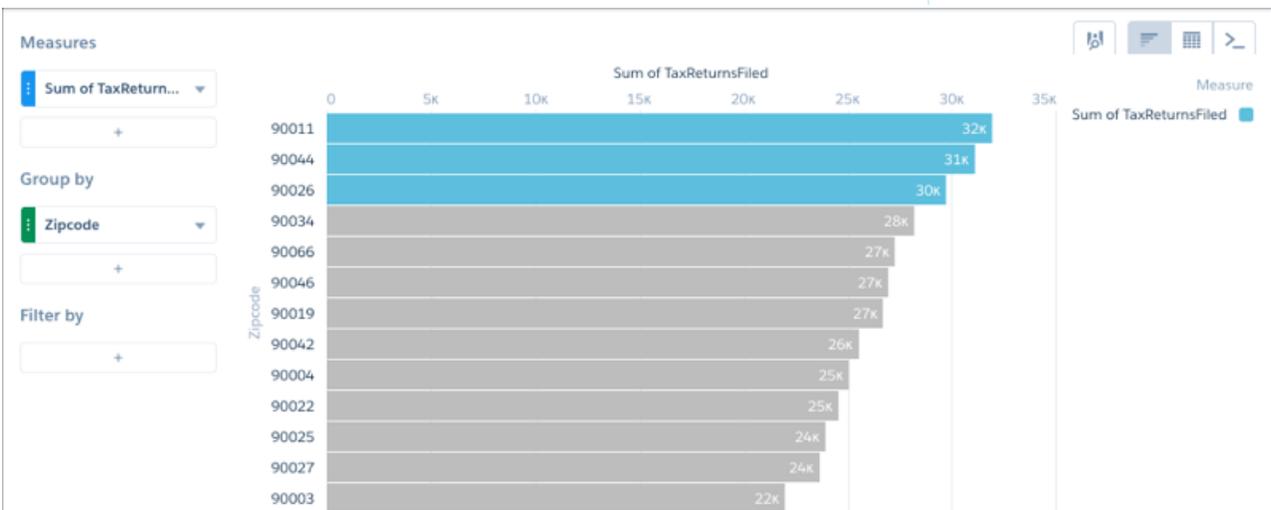
2. To open the Drill Into This By... menu, right-click the data or click .
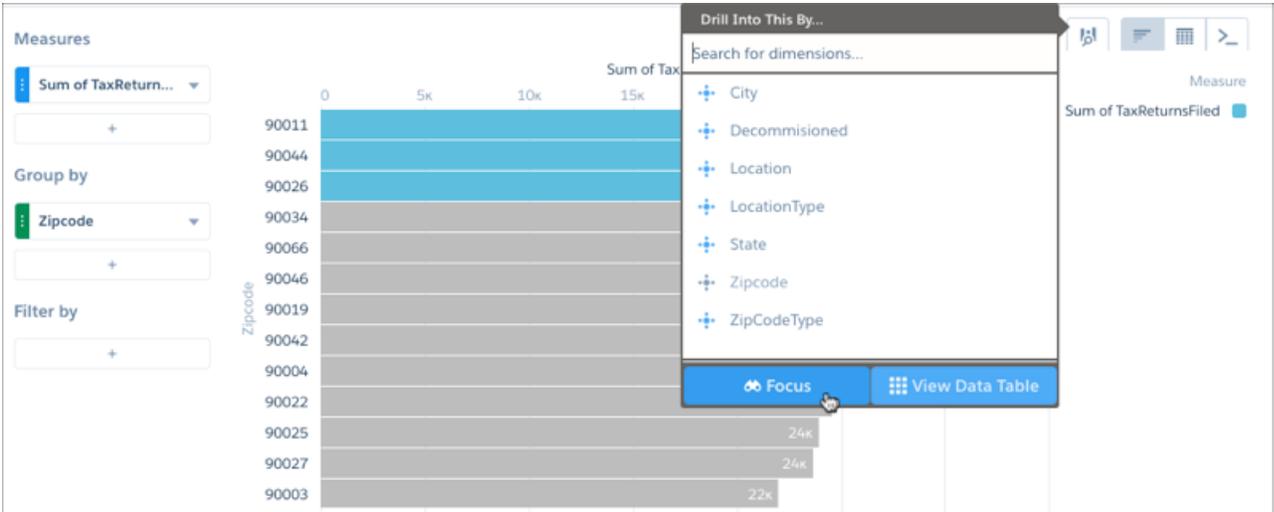
3. To view only the selected data, click **Focus**.

A new table or visualization based on the selected data appears in place of the previous table or visualization.

To return to the previous table or visualization, click .

Explore Multiple Datasets with a Single Query

Answers to business questions sometimes require context beyond the current dataset. Or, your dataset structure may not match the way you want to explore your data. To surface insights that require data from disparate datasets, blend the data from multiple datasets in the explorer. Then query the blended data just as you would with a single dataset.

Data blending differs from a join operation in that data blending affects only the data visualization. With data blending, you can reveal relationships across multiple datasets without altering the datasets.

For example, suppose that you want to see how sales for different product subcategories are doing relative to their targets. Here's how you create a side-by-side comparison of product subcategory totals from the SuperStoreSales and SuperStoreTarget datasets.

1. In the explorer, open a lens for the primary SuperStoreSales dataset.

2. Click **Add Dataset**.
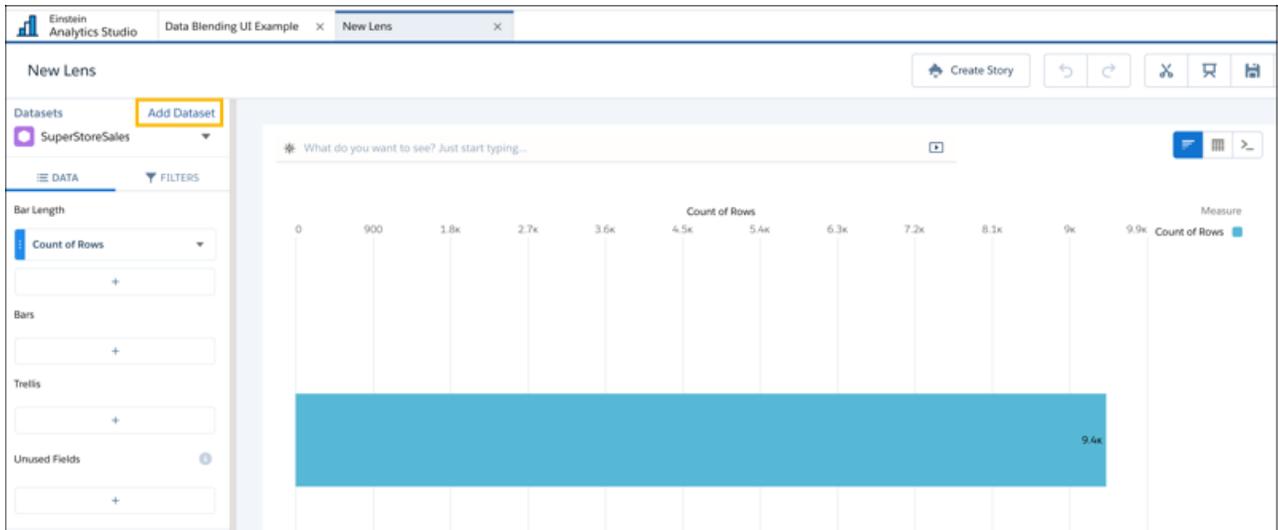
3. Add SuperStoreTarget as the secondary dataset.

   The lens now includes data from both datasets. By default, the data is combined using a left blend. All data from the primary dataset is included along with the data from the secondary dataset that matches the primary dataset.

   To extend your exploration further, add more datasets. You can query up to four datasets in a single exploration.

4. To change the blend type:

   a. Click **Edit** in the Datasets area.

   b. Select the blend type.

   c. Click **Done**.
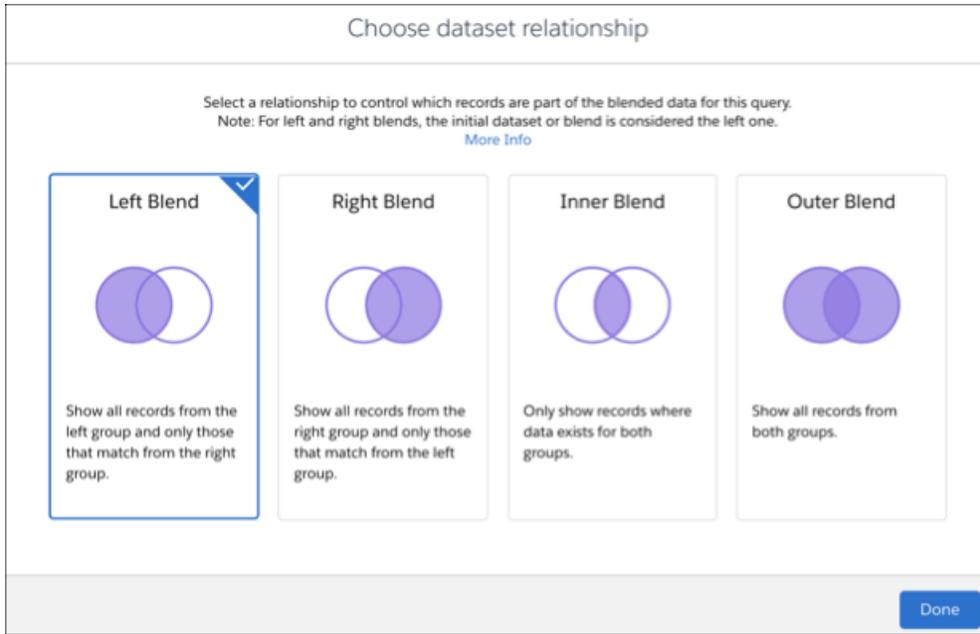
📝 **Note:** When a blend has more than 2 datasets, the same blend type is applied to all the datasets.

5. To create a meaningful side-by-side comparison, select measures and the field to group by from each dataset.

When defining groups, measures, and filters, select fields from any of the blended datasets. This example includes Sum of Sales for the SuperStoreSales dataset and Sum of Target for the SuperStoreTarget dataset. To group by product subcategories, select Product_Sub_Category for each of the datasets.
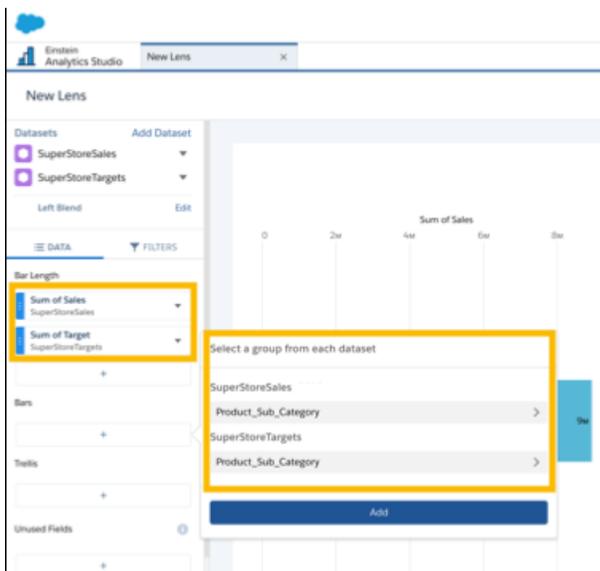
The query result shows the relationship between Sum of Sales and Sum of Target for each product subcategory.
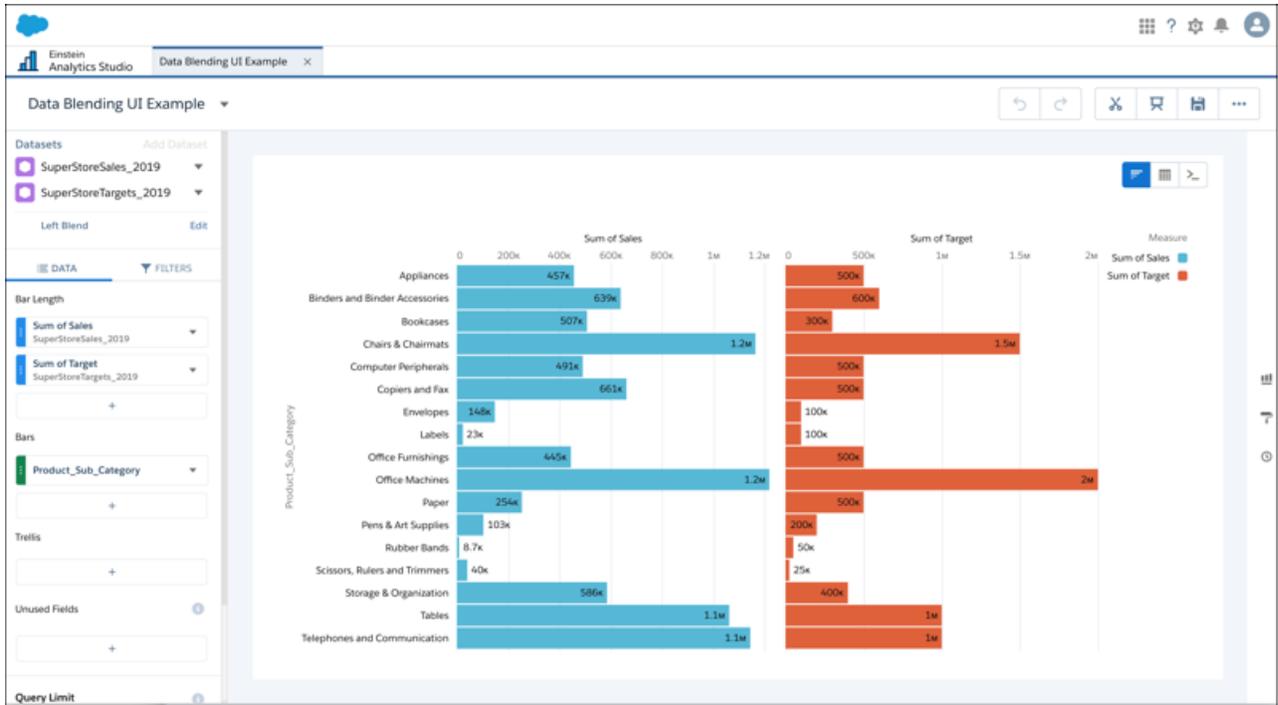
Limitations:

- You can blend up to four datasets.
- Add all datasets before beginning to explore them. The **Add Dataset** control is disabled when you add groupings, measures, or filters.
- You can add a dataset only once.
- These features aren't available: drill and focus, values table, windowing and timeseries functions, totals and subtotals, column filters, and Natural Language Queries (NLQ).
- Faceting applies only to the primary dataset when you blend data from multiple datasets.

Change the Chart Scale

With some data, a chart displays values that range widely on an axis and make the visualization more difficult to understand. You can constrict the range's scale by applying a logarithmic scale. Scaling is available for certain charts only.

1. To open the chart's properties, click ⚙.

2. Open the panel for the vertical axis, and click **Log Scale**.

   Depending on the type of chart being used, the vertical axis, or y axis, could also be called the left or right axis.
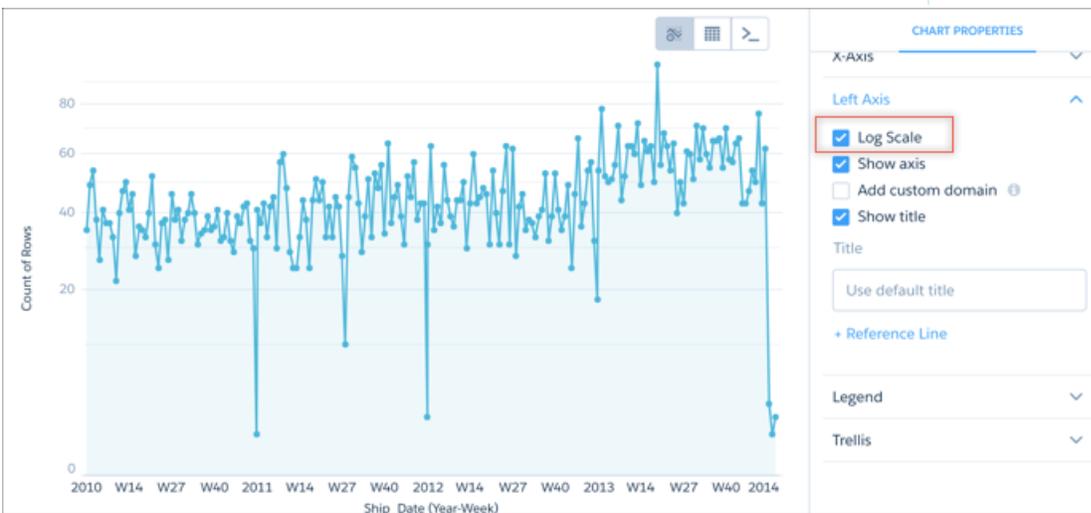
Return to a Previous View by Using History

Exploring your data can lead you down many paths. Not all of them are fruitful, and that's OK. Explorer keeps a full history of your activity in a lens. If you've changed your lens in undesired ways, or just want to see those nifty animations again, use history to backtrack quickly to a previous state in your visualization.

1.

Click the **View History** button ⏰ in the header.
A sidebar lists a history of all the actions that you've taken in the lens.
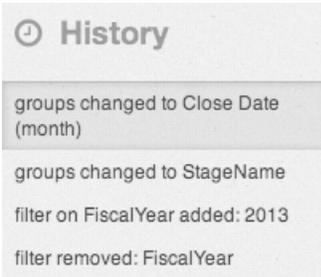
2. Select any item in the list to view the exploration at that point. Or use the undo and redo buttons (1) step backward and forward through the list.

Limit the Size of Your Query Results

Interested in reviewing just the top 10 accounts in your product pipeline—without excessive scrolling? To fast-track to meaningful insights and for improved performance on your dashboards, you can limit the size of query results in a lens.

1. To set the limit for your query, click the gear icon, and then select **Set Limit**.

2. Enter the size of the query result and click **Apply**.

If you edit your query in SAQL, you also need to set the query limit in SAQL.

💡 Tip: For a smoother scrolling experience, we recommend using the query result limit along with a chart's auto-fit feature.

View the Query Behind Your Lens

Tableau CRM uses Salesforce Analytics Query Language (SAQL) or Salesforce Object Query Language (SOQL) behind the scenes in lenses and dashboards to gather data for visualizations. You can view and edit the underlying code that's written as you explore, or copy it for use elsewhere.

1. Explore a dataset to get the visualization or table of data you want.

2. Click Query Mode to view the query's SAQL or SOQL code.

3. To change the query, edit the code and then click **Run Query**.

Query results appear in a table below the query editor.

> 💡 Tip: To work with multiple datasets in explorer simultaneously, reference them in the JSON editor for the currently running lens. Press Ctrl + E or Cmd + E to open the editor, and use the `load` statement to load each dataset you want to explore.

If you run a query while exploring, the filters become read-only while in chart and table modes. To make the filters editable, click and return to a point in the history before you ran the query.

> 📝 Note: If you save the lens with filters in read-only mode, they remain in read-only mode. Your exploration history is not saved with the lens and cannot be used to undo the read-only state.

For more information about SAQL, see the *Analytics Cloud SAQL Reference*.

Clip a Lens to a Dashboard

To include the lens in a dashboard, clip it. When you clip the lens, Tableau CRM adds a query to the most recently used dashboard. If a dashboard isn't open, Tableau CRM adds the query to a new dashboard in dashboard designer. After you add the query to the designer, you can apply it to a widget in the dashboard. The widget overrides the visualization settings of the lens—the widget determines how to display the results of the query.

1. Open the lens.

2. Click ✂ .

   Tableau CRM creates a query, and adds it to the query panel of the dashboard designer.

3. To save the query, save the dashboard.

   📝 Note: Changes to the query properties don't affect the lens.

Explore Data with Tables

You can use tables to get a view of the data that is close to the underlying dataset, and you can use tables to manipulate and extend the data to expose fresh insights. With values, compare, and pivot tables, Tableau CRM explorer gives you options for both of those goals.

Manage the View of Data in a Values Table

The values table displays all the columns in your dataset. You can customize the view of columns in a values table, and you can sort the row values using any column except for the index column.

Navigate Compare Table Columns

Create columns and navigate between existing ones using clicks or keystrokes.

Add Custom Formulas to Columns

Use the column editor to define custom formulas in compare tables or charts. View measures side by side, and perform math across the table's columns and rows. Use string values to create labels, concatenate dimension values, provide simple buckets, or add image URLs.

Window Functions Available in the Compare Table

The compare table's function menu provides UI-based access to useful window functions, saving you the time of having to write them in SAQL. Once you choose a function, you can edit the generated SAQL in the formula editor.

Use Raw Field Names in Explorer Formulas

When writing compare-table formulas, you can reference measure and group (dimension) columns by name. Referencing measure and group columns by name lets you write more intuitive and easily understandable formulas. Even better, the formula editor includes nifty function and formula pickers that save you time as you enter your SAQL formulas.

Table Properties

Use table properties to control a table's appearance.

Column Properties

The Column Properties panel lets you jazz up the columns you want in compare, values, and pivot tables. To highlight data that meets specific criteria, set conditional formatting rules on a column. Or, highlight data based on conditions set on a reference column. To make a column stand out without applying any conditions, you can use text alignment and text styles. In addition to displaying data in the text format, you can set a measure column's data type as bar.

Manage the View of Data in a Values Table

The values table displays all the columns in your dataset. You can customize the view of columns in a values table, and you can sort the row values using any column except for the index column.

On the left-hand side of the values table are measure and dimension (grouping) tiles representing all the visible columns in your dataset. To move a measure or grouping, click and hold its tile, then drag the tile to a new position. Release the tile when the position indicator is in the correct position (1).

To remove a measure or grouping from the table, click the x button in its tile (1). To access the Edit Columns dialog, where you can hide or show any column, click the pencil button (2). You can also add, edit, or remove filters (3). To sort the rows by a column, click its heading cell until an arrow appears in the direction you want to sort (4).

📝 Note: When sorting a values table by a column, that column's null records are excluded from the table. To prevent the exclusion of null records, assign a default value to those records.

Navigate Compare Table Columns

Create columns and navigate between existing ones using clicks or keystrokes.

You can navigate columns in the column editor using arrow buttons or your keyboard's left and right arrow keys.

When you reach the right-most column, the right arrow button becomes an add-column button. You can click the add-column button or use your keyboard's Enter key to add a formula column.

📝 **Note:** You can add up to 26 columns in the Compare Table.

As you navigate through a compare table, you can freeze up to five columns while you scroll through the remaining data. From a column header's dropdown menu, select **Freeze Columns Up to Here**.

Alternatively, go to the Layout section in the table properties panel (1). Select **Customize Frozen Columns** in the Layout section (2) and set the number of table columns to freeze (3). A dark gray line indicates the end of the freeze pane.

You can also delete columns from the compare table by selecting **Delete** in the context menu of any measure or dimension.

Use the **Close** button to exit the column editor without applying changes to the column.

Add Custom Formulas to Columns

Use the column editor to define custom formulas in compare tables or charts. View measures side by side, and perform math across the table's columns and rows. Use string values to create labels, concatenate dimension values, provide simple buckets, or add image URLs.

1. In the Explorer, click the Table Mode icon and select **Compare Table**.

2. On the Data tab, click ➕ under Measures.

3. Add dimensions by clicking ➕ in the **Group by** panel.

   📝 Note: Null values that occur in the data are displayed as minus signs ("-").

4. Change the first measure by clicking the initial selection ("Count of Rows"). Add more measures by clicking ➕ under Measures in the Data tab.

	Product_Category	Product_Sub_Category	Sum of Profit	Avg of Discount
Measures	Furniture	Chairs & Chairmats	165,348.8828	0.0501
Sum of Profit		Office Furnishings	92,209.2259	0.0485
Sum of Profit > 5,000.0	Office Supplies	Appliances	121,651.3914	0.0508
Avg of Discount		Binders and Binder Accessories	226,572.523	0.0503
		Envelopes	46,133.2238	0.0492
Group by		Labels	17,775.3205	0.0492
Product_Category		Paper	35,361.6217	0.0492
Product_Sub_Cat...		Storage & Organization	8,078.8047	0.0511
	Technology	Computer Peripherals	87,917.8425	0.0486
Filter by		Copiers and Fax	129,156.684	0.0517
Access filters from each measure's menu.		Office Machines	168,072.8333	0.0507
		Telephones and Communication	297,950.5249	0.0478

5. To enter a custom formula, click ➕ under Measures and click **Add Formula**.

In the column editor, you can enter text string formulas, SAQL formulas, including math functions, or choose a windowing function to calculate the values in the column. You can also name the column and choose a number format. If any of the formulas contain division by zero, the result is displayed as a minus sign ("-").

If you want to preserve an existing column, select **Clone Column** from the original column's action menu. You can edit the cloned column instead of the original.

6. In the **Formula** field, create formulas using operators from this table.

Symbol	Examples	Calculation
+	A+B, A+100	Addition, applied per row.
-	A-B, A-100	Subtraction, applied per row.
/	A/B, A/100	Division, applied per row.
*	A*B, A*100	Multiplication, applied per row.
( )	(B-A)/(A*100)	Parentheses for grouping operations.

> **Note:** Dimension columns are on the left and measure columns are on the right. The measure columns are identified in formulas as A, B, C, and so on, with A being the far-left measure column.

You can also create string formulas. For example, this formula creates simple buckets.

> **Tip:** Columns based on string formulas can't be used as dimensions for grouping in charts, but they can be used to apply conditional formatting in charts.

7. To use SAQL, enter it directly into the formula editor, then click **Apply** to see the result.

See the *Analytics SAQL Reference* for more information on writing SAQL.

> **Note:** Some SAQL functions cannot use letter references, such as A, B, C, etc., for columns. `percentile_desc` and `percentile_cont` functions, for example, require raw field names as arguments. See Use Raw Field Names in Explorer Formulas for more information.

8. To use a function, choose the function editor and then choose one of the windowing functions from the menu. After you configure the function, click **Apply** to see the result.

Switch to the formula editor to see the SAQL generated by the chosen function. You can edit the SAQL in the formula editor after generating it with a function as if you'd written it.

9. To open a column's action menu, click the down arrow.

10. To change a column's name or formula, click **Edit this Column** arrow.

📝 Note: Numeric formats use the period as a decimal separator and the comma as a grouping symbol. The currency formats are restricted to dollars and cents.

Window Functions Available in the Compare Table

The compare table's function menu provides UI-based access to useful window functions, saving you the time of having to write them in SAQL. Once you choose a function, you can edit the generated SAQL in the formula editor.

Sliding Window

Applies an aggregate function to the current row with respect to a configurable range of rows. The function can be reset on a grouping defined in the table. Use the sliding window for calculating values such as running totals and moving average.

Percentage of Group

Calculates the percentage each row is of its group total, or of the grand total. The function can be reset on a grouping defined in the table.

Rank Within Group

Provides rank, dense rank, cumulative distribution, and row-number functions. Ranking can be reset on a grouping defined in the table.

The Order menu determines the direction of ranking based on the values being ranked. Ascending ranks the lowest value as number 1, while descending ranks the highest value as number 1.

Period Over Period

Compares periods of time to calculate changes in values. For example, year-over-year sales, or quarter-over-quarter closed opportunities, can be calculated by choosing the column and setting the granularity of the time period.

Always group data by year first, then (optionally) grouped by a finer granulation of time if available. Compound dates, if available, can be used directly with the period-over-period function. Examples of compound dates include "2016-01" (month of January 2016) and "2016-1" (first quarter of 2016). Comparisons can be shown as percentages or units.

Change from Previous

Compares the value of the current row with that of the previous row and calculates the difference. For example, to calculate the growth in number of accounts between the first and fourth quarters, choose the column showing number of accounts from **Column**. To restrict that calculation to each country, choose the country column from **Reset Group**.

Running Total

Calculates the total value of the current row summed with all previous rows. For example, to calculate the monthly total of closed opportunities as a running total, choose the column showing closed opportunities from **Column**. To calculate that running total within each region, choose the region column from **Reset Group**.

Use Raw Field Names in Explorer Formulas

When writing compare-table formulas, you can reference measure and group (dimension) columns by name. Referencing measure and group columns by name lets you write more intuitive and easily understandable formulas. Even better, the formula editor includes nifty function and formula pickers that save you time as you enter your SAQL formulas.

You can use certain SAQL aggregate functions that require raw field names for arguments directly in the formula editor. These functions include `percentile_desc` and `percentile_cont`. You can also use the date function `daysBetween` in the formula editor.

> 📝 **Note:** In a compare-table formula, when you reference columns by default column names or user-defined column aliases, you can't use raw field names in that formula.

> 📝 **Note:** You can use the date functions `year_last_day`, `quarter_last_day`, `week_last_day`, `quarter_days`, `day_in_week`, and `day_in_quarter` in the SAQL editor. Click `>_` to access the SAQL editor.

See these examples for how to use SAQL functions with raw-field references in the formula editor. Also see the *Tableau CRM SAQL Reference* for more information on SAQL functions.

```
sum(Amount)/100
percentile_cont(0.95) within group (order by 'LoadTime' asc)
regr_slope(Amount,Closed_On_epoch)
daysBetween(toDate(last(ModifiedDate_epoch), "yyyy-MM-dd"), now())
coalesce(sum(Amount), 0)
len(first(Description))
```

Table Properties

Use table properties to control a table's appearance.

Display Totals and Subtotals

Add totals and subtotals to tables in the table properties panel or from the $\boxed{\Sigma\text{-}}$ button.

Measure functions are calculated according to all data in the relevant subgrouping. For example, the **Total** of *Max Annual Revenue* returns the maximum annual revenue of the entire column. If the formula for the column is **A/B**, the subtotal is the *subtotal of A/subtotal of B*.

> **Note:** Groups can't be sorted when subtotals are displayed. Measures can only be sorted within groups (inner sorts) when subtotals are displayed.

> **Note:** Totals and subtotals work by using the `rollup` statement to calculate totals of grouped data. For more information on the functionality and limitations of totals and subtotals, refer to the `rollup` section of the `group` topic in the *Analytics SAQL Reference*.

If a time series, a window function, or an unsupported aggregate function is used on the data:

- Totals are determined using a window function grand total or weighted average calculation instead of the `rollup`.
- Subtotals aren't calculated or displayed, the setting is ignored.

Because subtotals and grand totals use the `rollup` statement, which is computed before any aggregate filter is applied, subtotals and totals amounts don't reflect the filter. For example, if you filter out the rows where the sum or amount is greater than $1000, you do not see any rows greater than $1000, but the totals still include those rows in their value. An aggregate filter can also filter out the total rows themselves. As a best practice, it's not recommended to use aggregate filters with totals or subtotals.

Choose a Table Theme

Click a theme to apply it to a table.

Set the Spacing for Columns and Rows

To increase or decrease the whitespace in all cells, click a Cell Padding tile.

To set column width, select an option from the **Column Width** drop-down menu, then enter a value or values in **Width in Pixels**. If you choose **Fit to Widget**, all columns are brought into view.

Set Header Row Style

For the header row, you can set the cell background color, and the color and font size for text.

Set Body Row Text Attributes

For body rows, you can set the font size and color for text, and the number of text lines each row displays.

Note: The value you choose in the **Number of Lines per Row** drop-down menu does not affect text wrapping. It sets the number of lines each row displays regardless of the length of the text in each cell.

Set Border Styles

To style the table's external border, choose a thickness from **Border Thickness** and a color from **Border Color**. For the remaining cell borders, choose colors from **Major Gridlines Color** and **Minor Gridlines Color**.

Choose Colors for Body Rows

To set a background color for body rows, choose a color from **Table Background Color**. To alternate row colors, click **Alternate Row Colors**, then choose colors for the odd and even rows.

Note: Rows in dimension columns retain the table background color even if you use alternate row colors.

Add an Index Column to Values Tables

To add or remove an index column from a values table, click **Show Row Index Column**.

Dress Up Tables with Images

To add or remove an index column from a values table, click **Show Row Index Column**.

Include reps' Chatter profile pictures next to their KPIs in leaderboards. Or, add flags to your compare table showing worldwide sales data.

In the column you want to display images, include the Salesforce URLs that point to the images. In Column Properties, set the data type to **Image**. If you set the cell height or width for this column, we ensure that the images maintain their aspect ratio.

Column Properties

The Column Properties panel lets you jazz up the columns you want in compare, values, and pivot tables. To highlight data that meets specific criteria, set conditional formatting rules on a column. Or, highlight data based on conditions set on a reference column. To make a column stand out without applying any conditions, you can use text alignment and text styles. In addition to displaying data in the text format, you can set a measure column's data type as bar.

To set a column's properties, select the column in the table or from the **Choose Column** list in Column Properties.

You can also get to the Column Properties panel from a column's context menu.

Visualize Data With Charts

If your goal is to understand vast amounts of business data, and to communicate that understanding with coworkers, partners, and customers, being able to visualize your data is critical. Tableau CRM provides a chart for every need, each a means for illustrating key aspects of your business in just the right way.

Tableau CRM charts are integrated with the Tableau CRM tools you already use to explore data. Start with a dataset, explore with a lens, clip it to dashboard designer, and create a stunning dashboard.

Customize Charts with Chart Properties

To create the best visualizations for presenting your data, use chart properties to configure Tableau CRM charts.

Bar Charts

Tableau CRM provides the following types of bar charts: Bar, Column, Stacked Bar, and Stacked Column. Use the Bar or Column chart to provide a quick visual comparison of related values. Use the Stacked Bar or Stacked Column chart to show groups within each bar.

Calendar Heat Map Charts

Calendar heat maps are useful for visualizing recurring discrete activities, such as closing accounts, over long periods of time. In this chart, you can easily change the granularity of the time-based grouping, such as from month to week or day.

Column Charts

Use a column chart (also called vertical bar chart) to show relative counts of things, such as leads or dollars. Use a stacked column chart to show groups within each bar.

Combo Charts

Use a combo chart to display at least two related series of data, such as discrete grouped values as vertical bars overlaid with a line chart representing an average value.

Donut Charts

Use a donut chart when you have a grouping and want to show not only the proportion of a single value for each group member against the total, but also the total amount itself. If you remove the donut's center by selecting **0%** from the **Center Size** menu, this visualization is commonly referred to as a pie chart.

Dot Plot Charts

Horizontal and vertical dot plots use the size and coloring of bubbles to display multiple dimensions and measures. Use dot plots to visualize related data to compare performance or locate unusual values.

Funnel Charts

Use a funnel chart to visualize sequential data that can be broken up into stages, such as a sales cycle.

Gauge Charts

Use gauge charts to track progress along a single measure, such as how much revenue has been realized versus the target.

Bullet Charts

Bullet charts are perfect for comparing metrics against quantitative benchmarks and references, like current revenue with target revenue.

Heat Maps

Use a heatmap to visually enhance high- and low-value data when there's a single measure and multiple dimensions.

Line Charts

Use a line chart when you have one important grouping representing an ordered set of data and one or more values to show.

Map Charts

Use a map if you have data with a geographical component. Maps can shade areas in proportion to mapped values, allowing for visual pop-out of high-value areas. You can find the map chart in the dashboard designer.

Bubble Map Charts

Use a bubble map chart to visually indicate measure values in specified geographical areas.

Geo Map Charts

Geo maps allow you to visualize data that contains geographical coordinates (latitude and longitude). Using coordinates allows you to place visual data elements more precisely on map charts.

Custom Map Charts

Use a custom map if you have data with a geographical component that doesn't match any of the standard maps provided by Tableau CRM charts, such as custom regions. You can also use custom maps for areas that can be represented with polygons, such as stadium seat sections, city blocks, or floor plans.

Matrix Charts

Use matrix charts to create a table that represents measures visually, allowing you to quickly spot extreme values.

Origami Charts

Use an origami chart to create a striking horizontal visualization for easily identifying high- and low-value data when there's a single measure and a single grouping.

Parallel Coordinates Charts

Use a parallel coordinates chart when you have multiple measures and a single grouping. Parallel coordinate charts are useful for displaying how data elements in a grouping stack up against each other.

Pyramid Charts

Use a pyramid chart to visually highlight relative sizes of stacked dimension values.

Radar Charts

Use radar charts to display a small dataset with one dimension and at least three measure columns.

Rating Charts

Use a rating chart to get a sense of how well a measured quality, such as customer satisfaction, is meeting expectations.

Sankey Charts

Use a sankey chart to visualize the distribution of a measure from one grouping to another grouping.

Scatter Charts

Use a scatter chart to visualize correlation between two groups of data.

Timeline Charts

A timeline chart is a line chart with one axis dedicated to a time dimension. Use a timeline chart to show how a value changes over time.

Time Bar Charts

A time bar chart is a column chart with the horizontal axis showing a time dimension. Use a time bar chart to visualize changes over time with vertical bars, which is a great way to illustrate relative changes over time and highlight missing data.

Time Combo Charts

A time combo chart shows two or more measures over time. You can display each measure as a line or bar. You can display the charts on the same or separate axes.

Treemap Charts

Use treemaps to visualize hierarchical quantitative data, where containing rectangles designate relationship using color, and "leaf" rectangles represent quantity using area.

Waterfall Charts

Use a waterfall chart to show the cumulative effect of sequentially introduced positive or negative values with breakdowns of value totals. Also known as "flying bricks" or "Mario" charts. To include breakdowns of value totals, use a stacked waterfall chart.

Customize Charts with Chart Properties

To create the best visualizations for presenting your data, use chart properties to configure Tableau CRM charts.

To display the chart properties in the explorer, click ![icon] . The chart properties appear in the right panel and are grouped into sections. The properties differ based on the chart type. Properties that are unique to a chart type are grouped in their own section. For example, the map chart has the "Map" section.
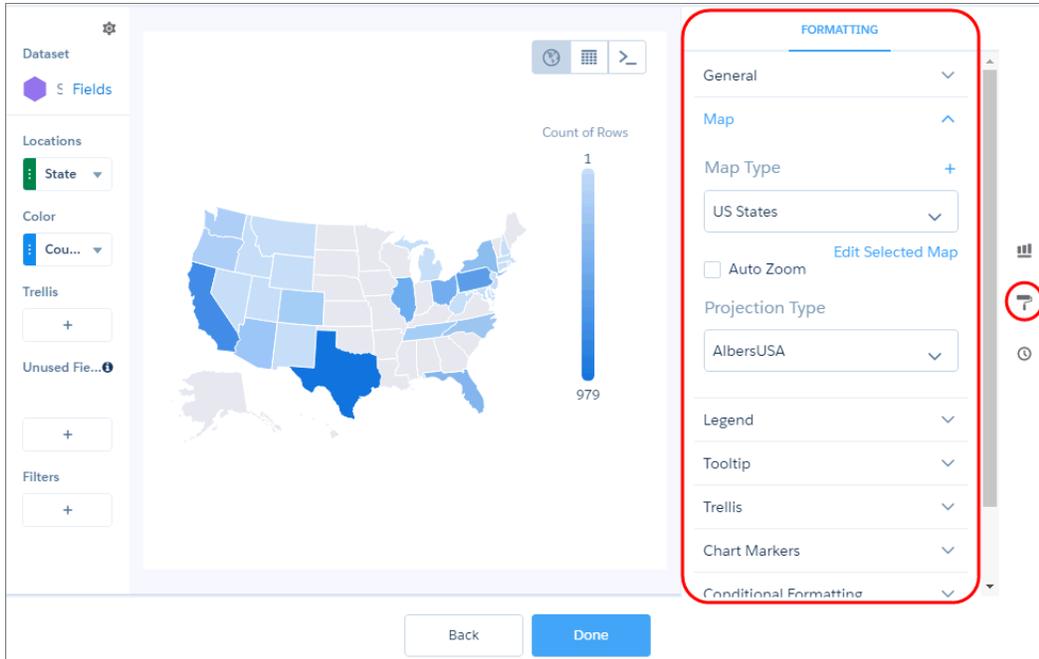
If you add a chart to a dashboard, edit the dashboard and select the chart widget in the dashboard designer to change the chart properties for the widget.
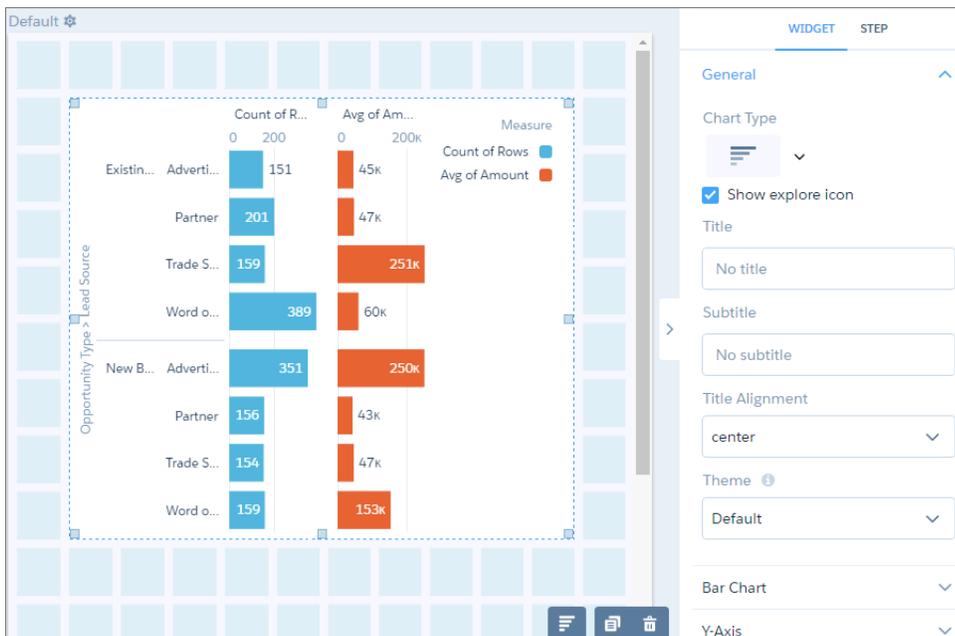
Or, to change the chart properties at the query level, double-click the widget in the dashboard designer (which opens the query in the explorer) and click 🖌. If the query powers multiple charts in the dashboard, changes to the chart properties at the query level only affect chart that you double-clicked.

SEE ALSO:

Use a Suggested Chart to Display Results

Using a sophisticated algorithm that analyzes your lens's composition, explorer suggests chart types that can better communicate the essential message your data is communicating.

To view suggested charts, click the gear menu in chart mode, then click **Suggested Charts**.
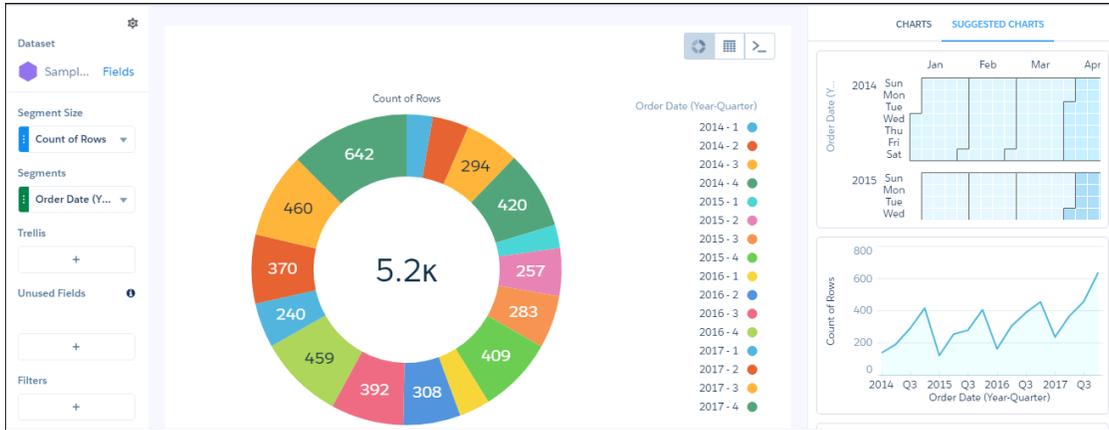
Show Each Measure on a Separate Set of Axes

Charts that display multiple measures are sometimes hard to read. To fix that, display each measure separately.
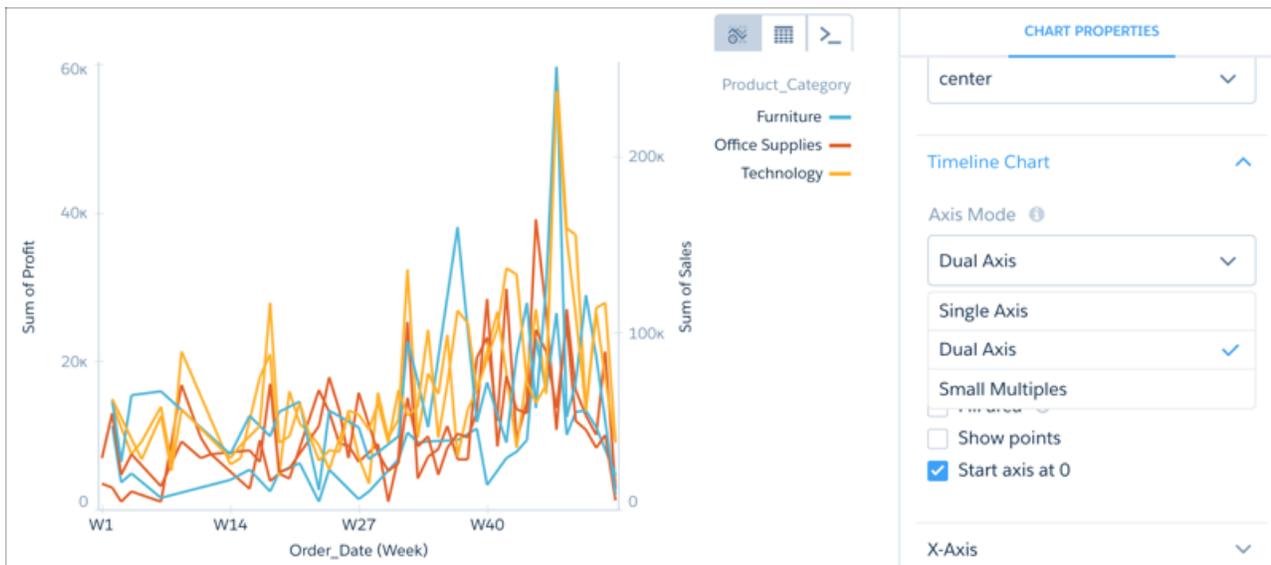
Select **Small Multiples** from Axis Mode to separate the measures and make the chart easier to understand.

Tip: While viewing the dashboard, click the title of a measure axis to sort the results. Each click cycles through the sort order options: ascending, descending, and alphabetical. You can use this method to sort all charts with measure axes, except scatterplot or charts that also have a time axis.

Automatically Highlight Data with Conditional Formatting

Don't delay important decisions because you miss critical changes in your data. With clicks, not code, add rules to automatically format results to quickly catch changes and take immediate action. For example, highlight accounts in red in a bar chart when their CSAT score drops below 80% so that you can get on the phone with those customers.

Use rules to automatically highlight charts and tables in an explorer lens or dashboard as well as number widgets in your dashboard. Stay on top of each KPI by highlighting the highs and lows based on ranges and colors that you specify. Or, distinguish each measure or dimension value (group) by color-coding them.

For example, you can apply conditional formatting to accomplish the following goals.

- Determine how well you're doing on a KPI by coloring the number widget based on its value.
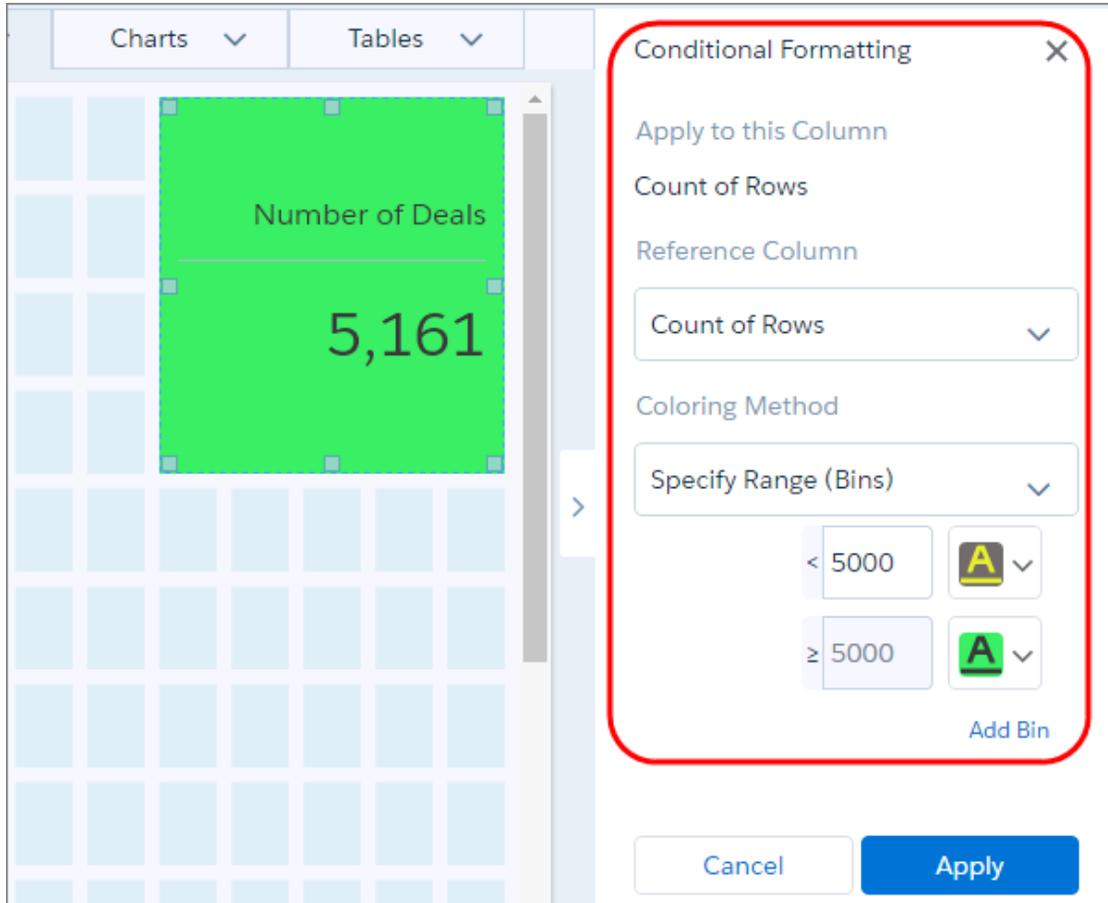
- Distinguish each measure in a chart with a unique color.
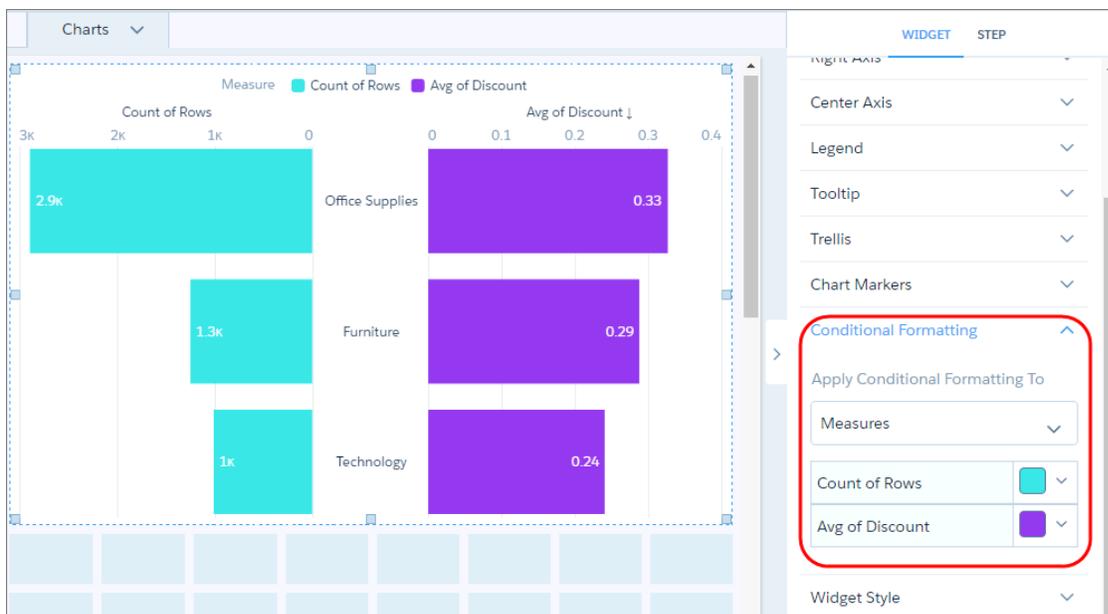
- Similarly, distinguish each dimension value.
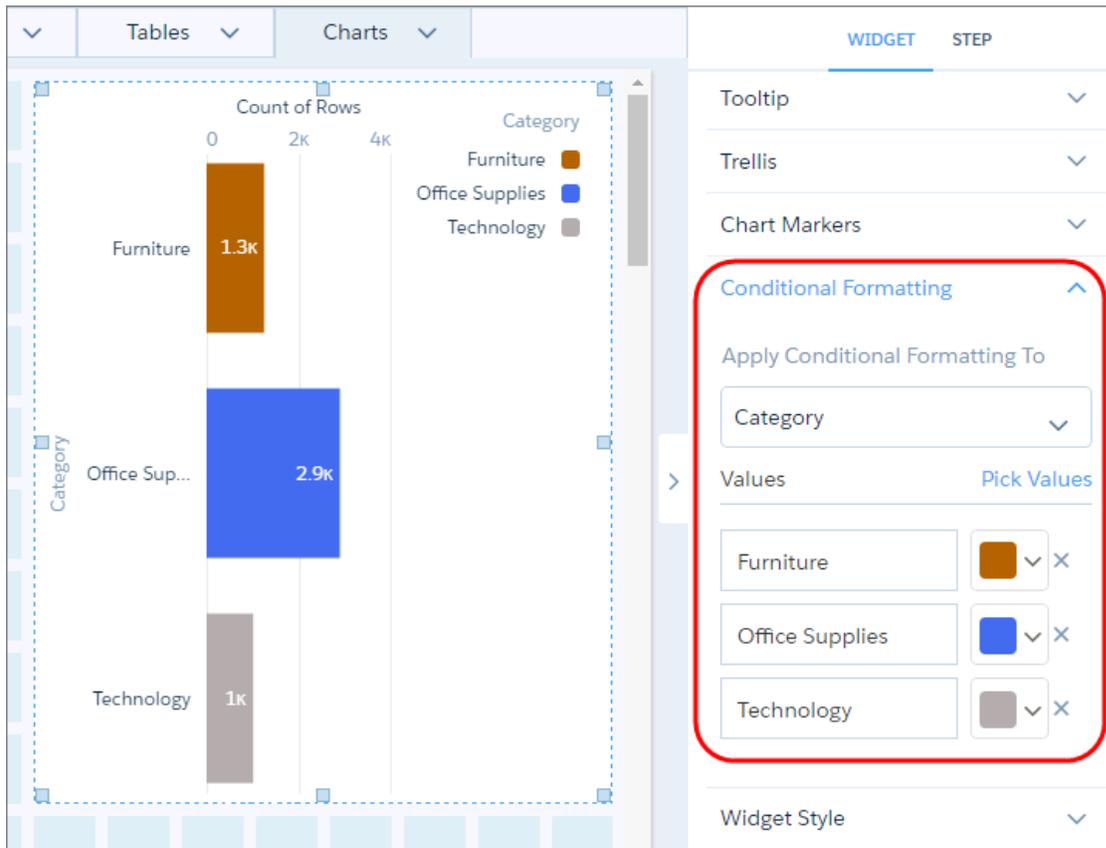
- Determine your top and bottom performers by categorizing numeric values into colored bins. You can add up to 10 bins.
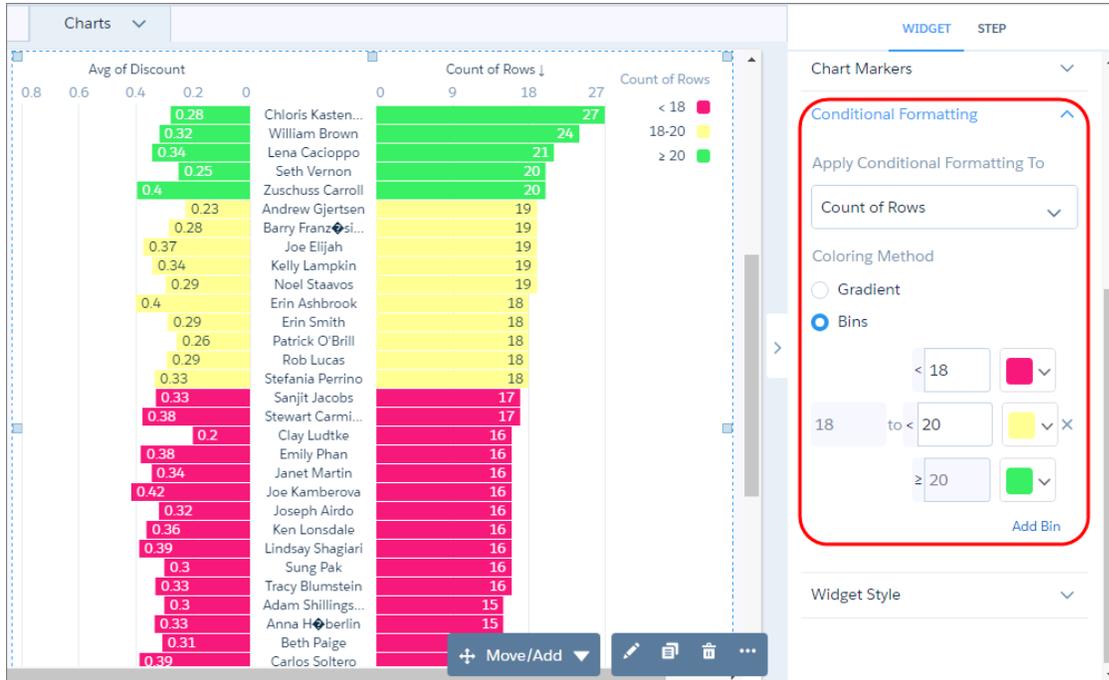
- Focus on key table records by color-coding the text or background of a column's cells. In the following example, the Customer Name column is colored based on the value of the Sales field.
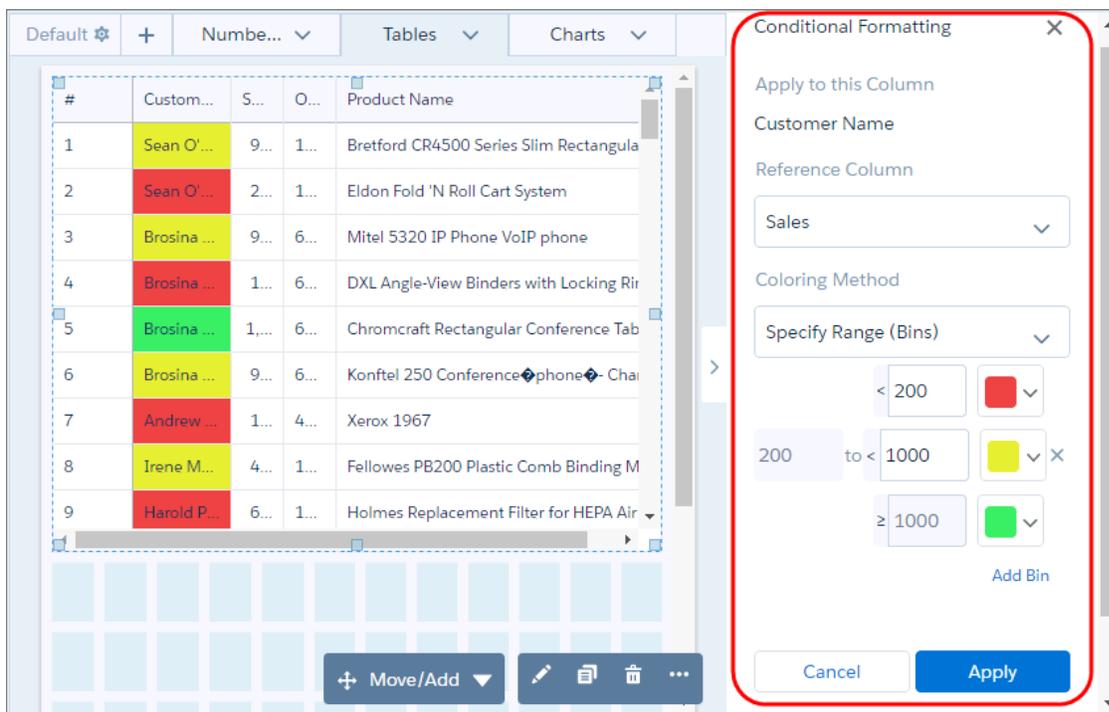
Set up conditional formatting in the widget properties panel. Although you set it up at the widget level, the formatting applies to the underlying query. Conditional formatting affects all widgets that use the query. If you change the widget or chart type, the formatting

is retained. A message appears if the formatting can't be retained, like when changing between a table and chart widget. Conditional formatting overrides colors set on dimension values in the dataset XMD.

For charts and number widgets, set up conditional formatting in the widget properties. Select **Measures** to apply a format to each measure. Or select one measure or dimension to apply a format to each of its values. Each chart or number widget can have one conditional formatting rule.
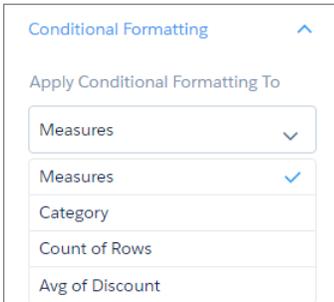
For tables, in the Column tab of the widget properties, select the column that you want to format and then set the rule used to format it. Each table can have multiple conditional formatting rules. You can apply only one rule for each style, like background color or text color in a table.
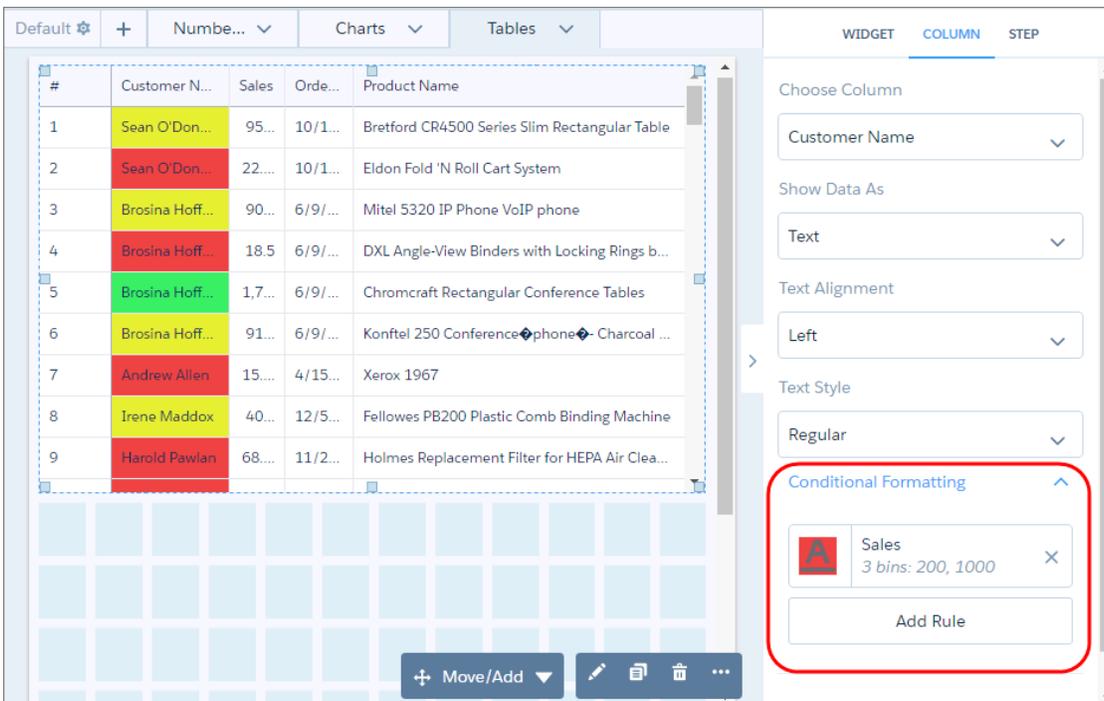
For calculated dimensions, select the dimension and click **Pick Values**. For Select Values, you can choose previously defined values or enter custom values.
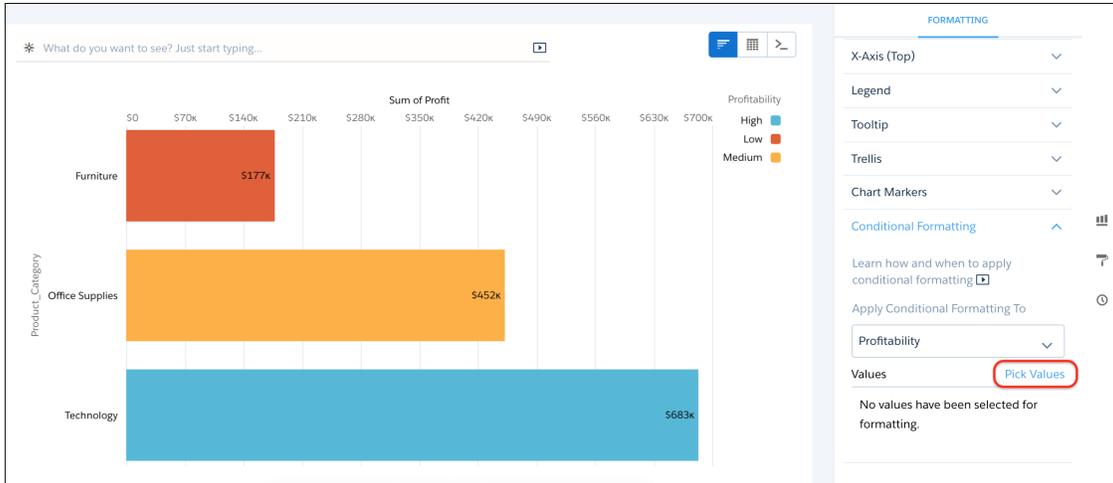
If you enter a custom value, then click to add it to the Select Values list.

Conditional Formatting Considerations

Keep these considerations in mind when setting up conditional formatting.

You can set up conditions based on a single reference column in the widget properties. However, sometimes, you have to create more complex conditions. To set up rules based on multiple reference columns for number and table widgets, use SAQL.

SEE ALSO:

Conditional Formatting Considerations

Keep these considerations in mind when setting up conditional formatting.

- Conditional formatting can cause deployment failures. Before deploying a dashboard, we recommend removing all conditional formatting in the widget properties.

- You can't apply conditional formatting to a query that contains a period (.) in the query ID. To apply it, modify the query ID.

- In a pivot table, you can apply conditional formatting to all columns other than the pivoted dimension.

- Conditional formatting is removed if you manually change the query ID or dev column name in the dashboard JSON or SAQL, and save the dashboard or clip the lens to a dashboard.

- Conditional formatting set in the widget properties doesn't appear in the dashboard JSON. Any conditional formatting previously set up in the dashboard JSON is retained.

Set Up Complex Conditional Formatting

You can set up conditions based on a single reference column in the widget properties. However, sometimes, you have to create more complex conditions. To set up rules based on multiple reference columns for number and table widgets, use SAQL.

1. Set up the conditions in the SAQL query. Define the rules and output the colors to a new column. Specify the colors in hex codes (like #CCCC00) or RGB codes (like rgb(255,255,255) or rgba(255,255,255,0)).
   For example, the following query specifies the rule and outputs the hex color code for each condition in the "FormatColor" column.

```
"query": "q = load \"Sample_Superstore_with_Goals\";\n
    result = group q by 'Segment';\n
    result = foreach result generate
        q.'Segment' as 'Segment',
        sum(q.'Discount') as 'Discount',
        sum(q.'Profit') as 'Profit',
        sum(q.'Quantity') as 'Quantity',
        sum(q.'Sales') as 'Sales',
        (case
            when Segment == \"Consumer\" and sum(q.'Discount') >= 1000 then \"#008000\"

             when Segment == \"Consumer\" and sum(q.'Discount') < 1000 then \"#CCCC00\"

            when Segment == \"Corporate\" and sum(q.'Discount') >= 500 then \"#008000\"

             when Segment == \"Corporate\" and sum(q.'Discount') < 500 then \"#CCCC00\"

           when Segment == \"Home Office\" and sum(q.'Discount') >= 200 then \"#008000\"

            when Segment == \"Home Office\" and sum(q.'Discount') < 200 then \"#CCCC00\"

             else \"#000000\" end) as 'FormatColor';\n
```

```
result = order result by ('Sales' desc);\n
result = limit result 2000;",
```

2. To apply the text color to the Sales table column, select the new color column ("FormatColor") in the Reference Column widget property and then select **Use Color in Reference Column** in the Coloring Method option.

The following example conditionally colors the text in the Sales table column based on the logic used to determine the color in the "FormatColor" reference column.

Set a Custom Domain to Focus the Results

If the domain on the chart is too large, chart elements can be difficult to view, like the bars in this chart.
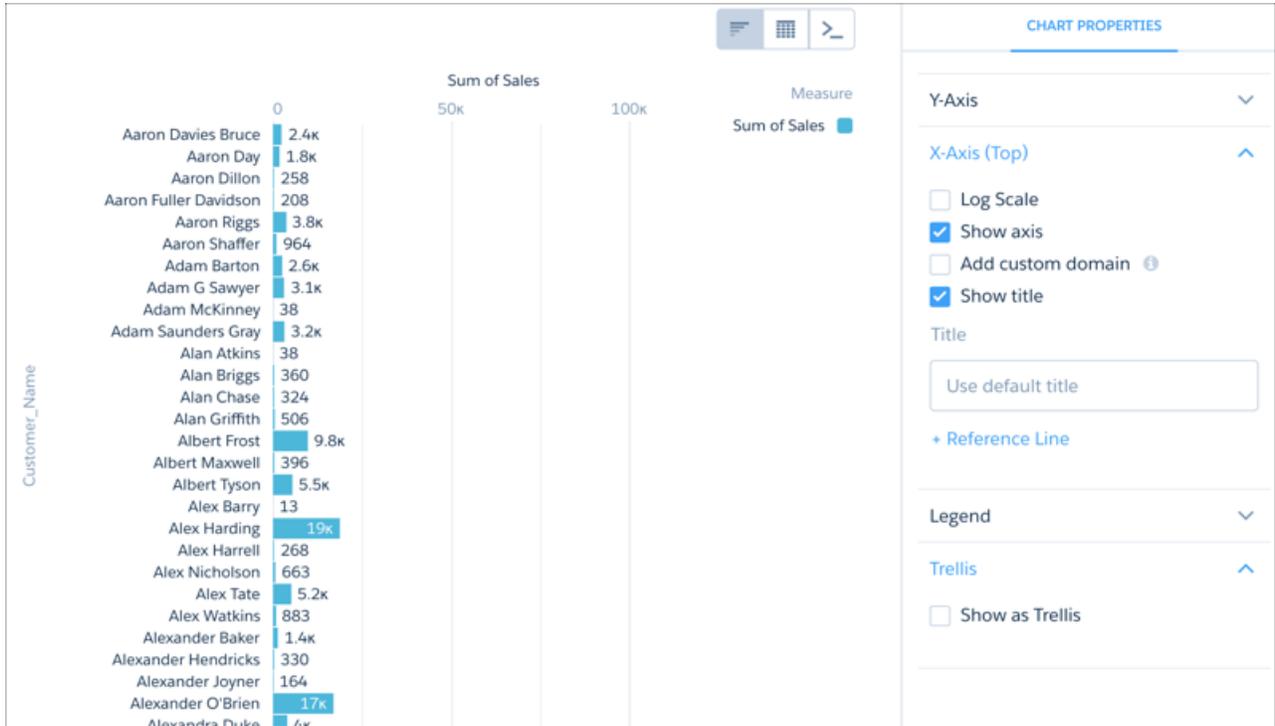
Set a custom domain to restrict the view to values that bring smaller values into view. Click **Add custom domain**, then set the domain's bounds be entering minimum and maximum values.
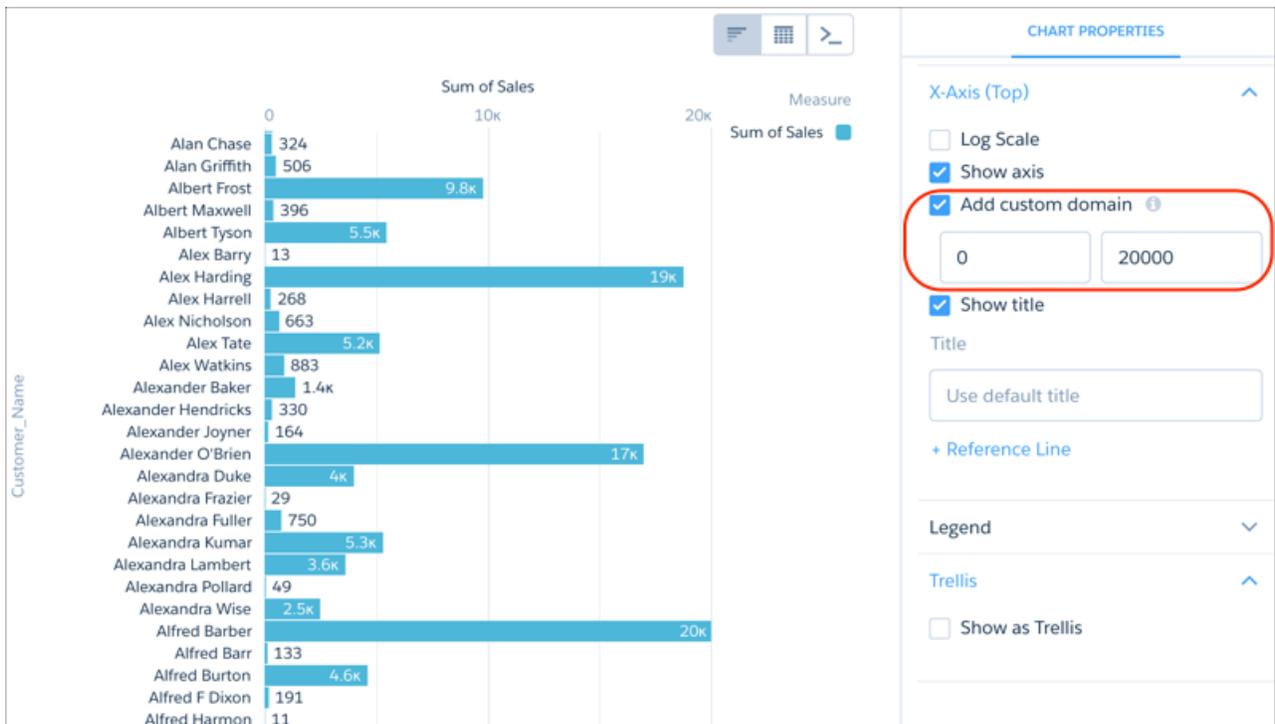
Fill the Area Between Lines

You can shade the area between lines in line and timeline charts to accentuate the spread between two measures.

To show the spread between minimum and maximum values of closed opportunities in a timeline chart, for example, click **Shade Area**.
Then choose the measures from the **Measure 1** and **Measure 2** menus, and pick a color for the shading from **Color for Shading**.
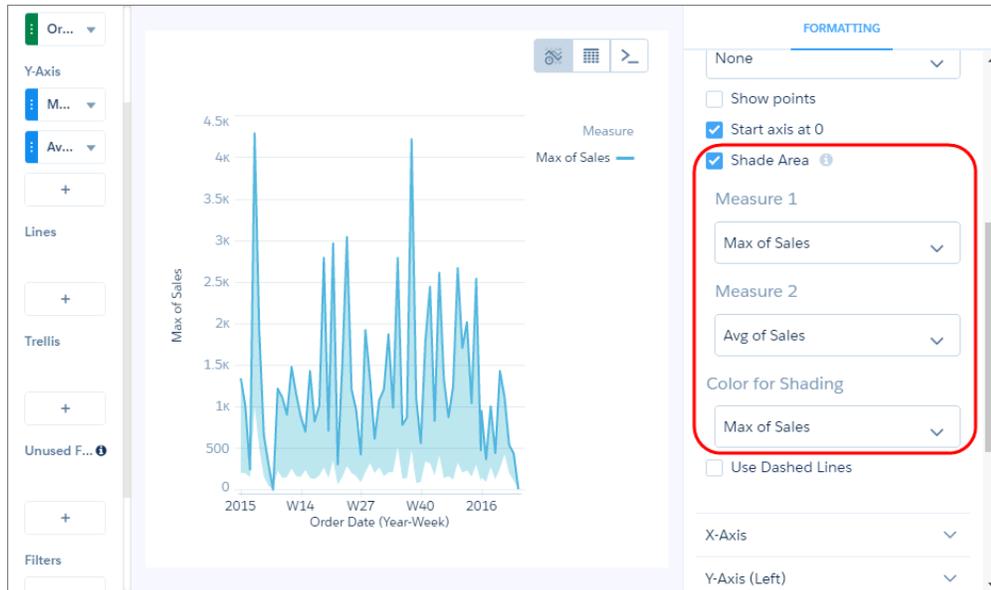
To focus on the shading, and not the lines, the chart hides the lines and legend for measures specified in **Measure 1** and **Measure 2**,
unless you select them in the **Color of Shading** field.

Note: The colors of the measures are based on the theme in the legend. The measure colors can change when you select the
shade color or if lines are removed from the chart.

Use Dashed Lines for Effect

To differentiate a measure in a line or timeline chart, use a dashed line.

To contrast the line representing the average between the minimum and maximum values of closed opportunities in a timeline chart,
for example, click **Use Dashed Lines**, then choose the measure for the average from the multiselect menu below it.
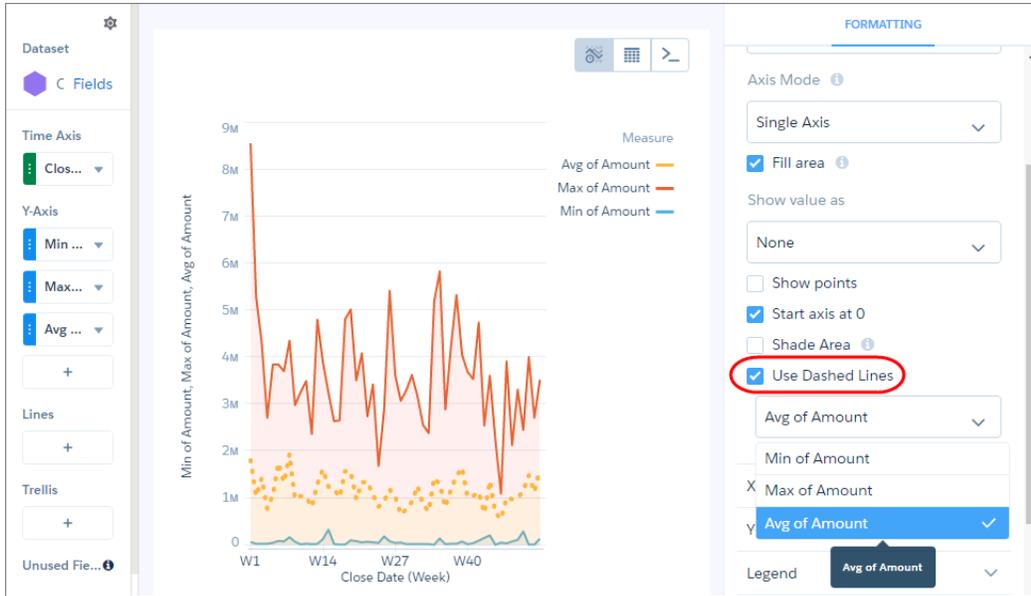
To make other measure lines dashed instead of solid, select them from the same menu.

Handle Missing Values in Line Charts

Charts can display gaps to highlight missing data. Or charts can ignore the missing data and show a continuous line that connects the data points. For periods of missing data in timeline charts, you can make cumulative lines horizontal to indicate that no change occurs.

To configure how a chart handles missing data, set one of the following options in the Missing Value chart property.

Show as gaps
The graph connects contiguous data points and shows gaps whenever data is missing. In the following example, there's a separate graph for customer and partner account types.
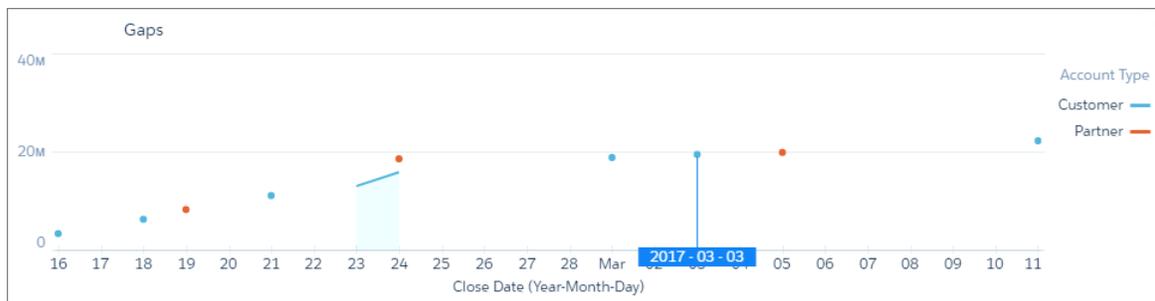
Ignore, show continuous line
The graph connects all data points, glossing over missing data. The graph starts at the first data point and ends at the last for each grouping.
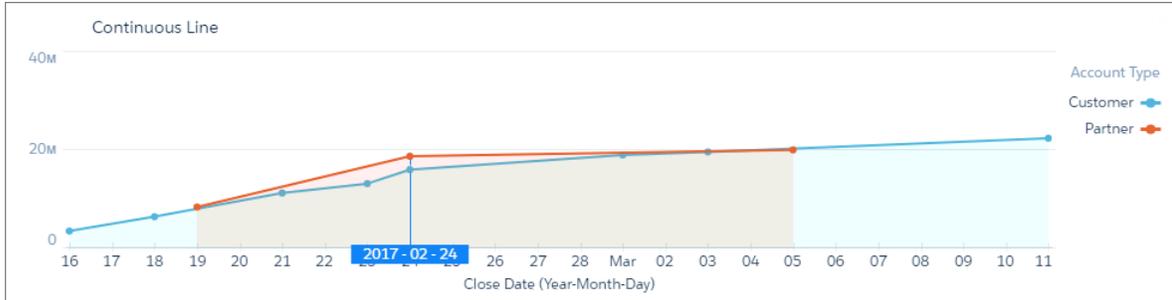
Use prior values

Uses the value prior to the missing data point to indicate no change. Use this option for charts that show cumulative values, such as timeline.

Let's look at an example. Because no deals closed between February 20 and 23, the cumulative value doesn't change for the partner line. To show no change, the chart uses the same value from the previous data point on February 19. This chart is more accurate than the previous graph, which is misleading because it shows an upward trend during this same time period.
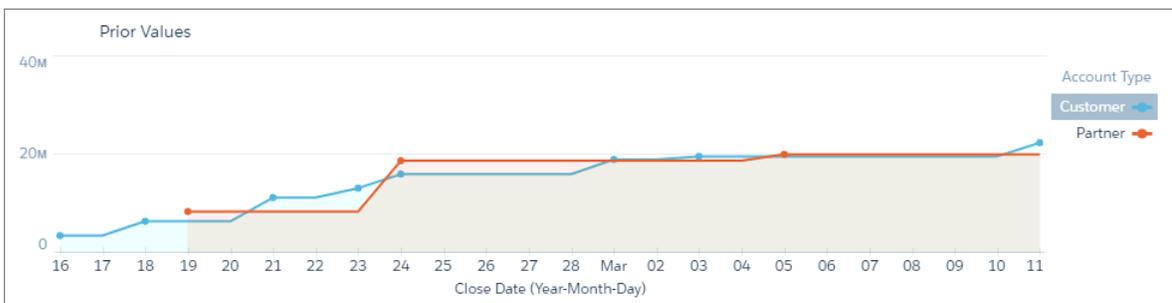
Use prior values (trimmed)

The graph ends at the last data point. For instance, looking at the following example, this graph is the same as the Use Prior Values graph, except that it's truncated on the right because there's no data after March 5.
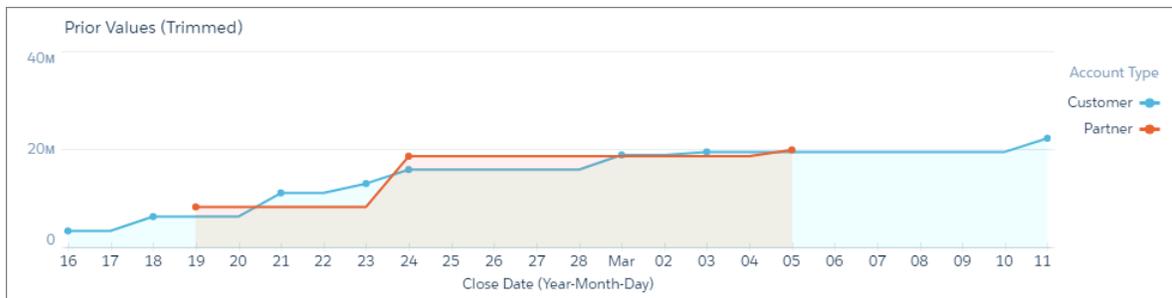
Add Multiple Reference Lines to Charts

You can use any number of static or dynamic reference lines to mark up a chart, such as when you indicate points of reference on a timeline before and after a predictive line. Dynamic reference lines are powered by existing queries in the dashboard.
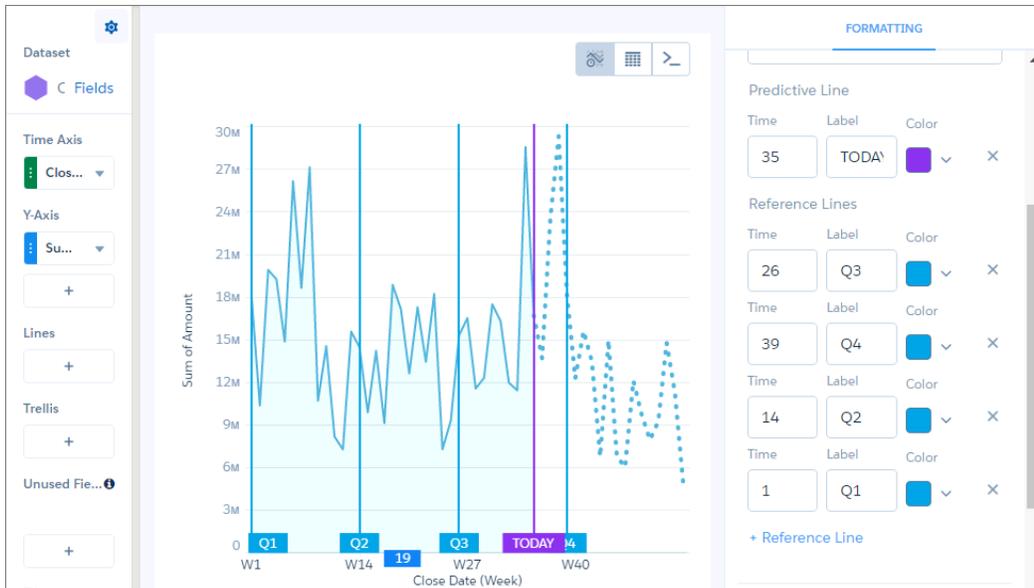
1. Tap **+Reference Line** in the X-Axis or Y-Axis section of the chart widget panel.

2. Add either a static or dynamic value for the reference line.

   - To add a static value, type a numeric value in the **Value** field.

   - To add a dynamic value, tap ⟲ .

a. Select a source query.

b. Select a reference line value based on the query.

c. Select an interaction type.

   To learn more about interaction types, see Result Binding and Selection Binding.

d. Tap **Apply** to add the value to your reference line.

3. Add either a static or dynamic value for the reference line.

   • To add a static label, type custom text in the **Label** field.

   • To add a dynamic label, tap ⇄ .

a. Select a source query.

b. Use the query fields and custom text to create your label.

c. Select an interaction type.

   To learn more about interaction types, see Result Binding and Selection Binding.

d. Tap **Apply** to add the value to your reference line.

4. Select a default or custom color.

Annotate Data Points with Tooltips and Markers

Create your own tooltips and markers in charts to show details about data points. Create a tooltip to specify which details appear when you hover over any data point. To annotate a specific data point, add a marker and, to draw more attention, make it blink.

The tooltips and markers appear when you hover over the data point.
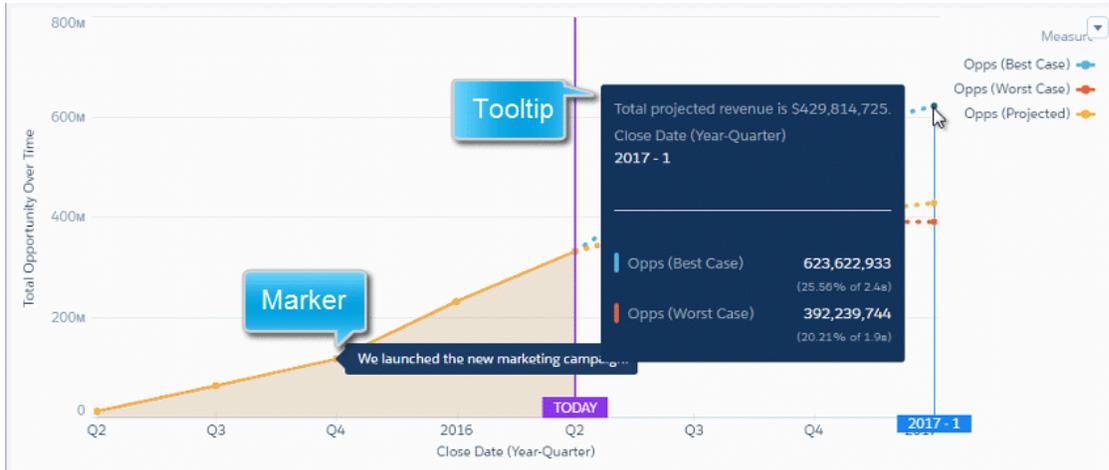
1. Edit the dashboard and select the chart widget.

2. To make it easier for users to locate data points on the line, select **Show points**.

3. To add a tooltip, which appears for all data points, expand the Tooltip section of the widget properties.
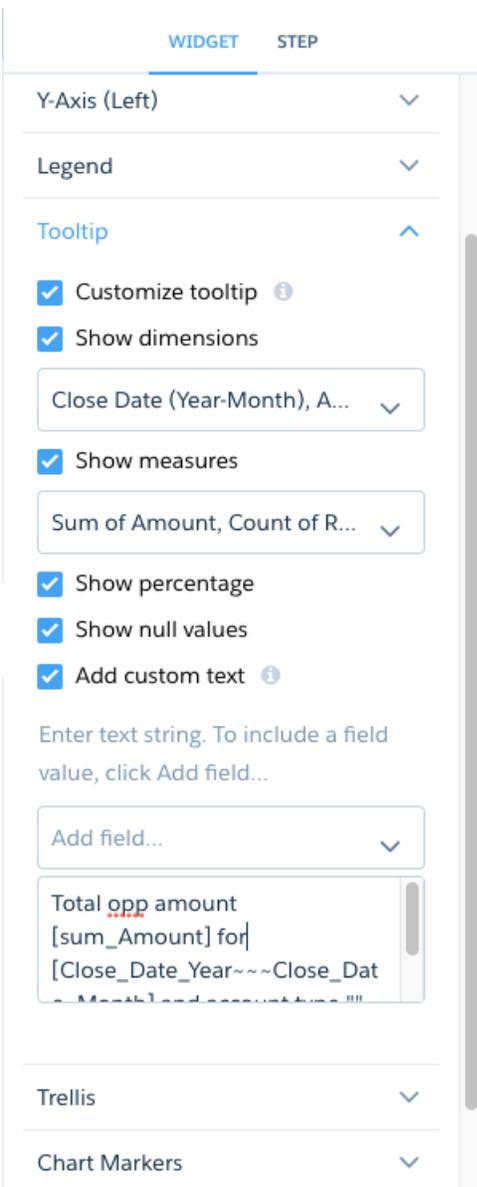
4. Enter the following tooltip details.

Property	Description
Customize tooltip	Shows a tooltip when you hover over any data point in the chart. The tooltip can show your own text and the values of measures and dimensions.
Show dimensions	Shows the values of all selected dimensions.
Show measures	Shows the values of all selected measures.
Show percentage	If you show measures, enable this option to show each measure's percentage of the total.

Property	Description
Show null values	Show null values in tooltips.
Add custom text	Specify your own text. To insert the value of a field in the text, click **Add field**. If manually entering a field, reference the field by its alias (not label) and enclose the alias in square brackets.

5. To add a marker for a data point, expand the Chart Markers section in the widget properties, and then click **Edit Chart Markers**.

6. Select a data point in the chart where you want to add a marker.
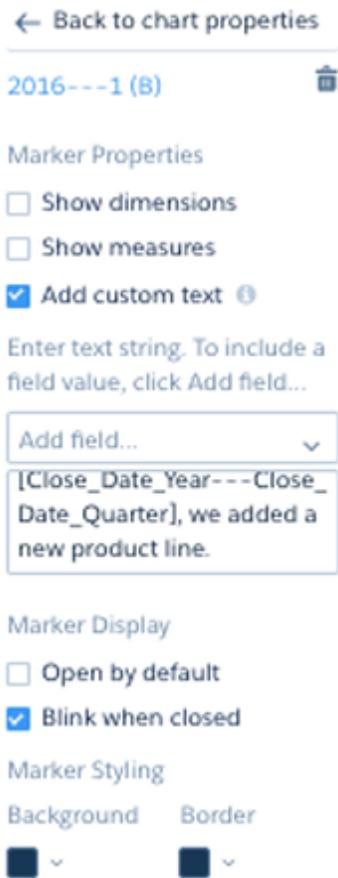
7. Enter the following marker details.

Property	Description
Show dimensions	Shows the values of all selected dimensions.
Show measures	Shows the values of all selected measures.
Add custom text	Specify your own text. To insert the value of a field in the text, click **Add field**. If manually entering a field, reference the field by its alias (not label) and enclose the alias in square brackets.

Property	Description
Open by default	Opens the marker and shows its text, by default. Otherwise, the dashboard viewer must click the marker to view the text.
Blink when closed	Makes the marker blink when it's closed.
Background	Background color of the marker.
Border	Border color of the marker.

8. Preview the changes.

9. Save the dashboard.

Compare Results for Different Groups with Trellis

Enable trellis on a chart to display a separate chart for each group. Because the charts in a trellis have the same scale, you can quickly compare results across all groups.

Regular map charts, for example, can display one grouping. This map of the United States shows a sum grouped by state.
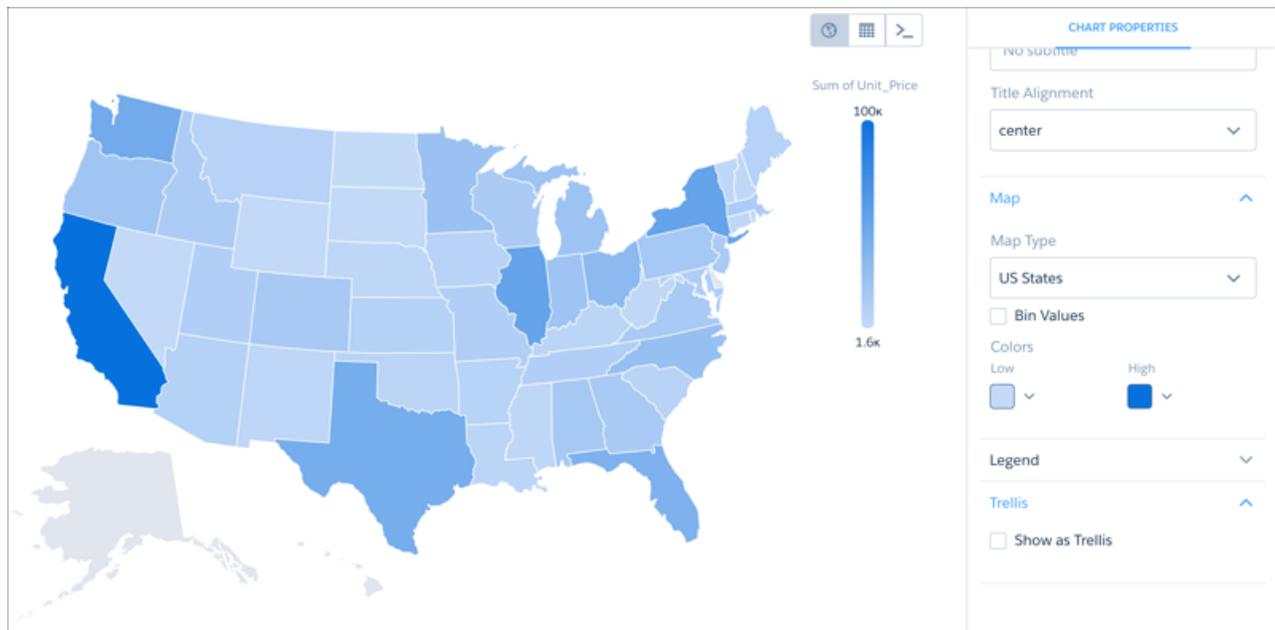
In trellis view, more groupings can be added. This trellis of maps of the United States shows a sum grouped by billing state *and* by product.
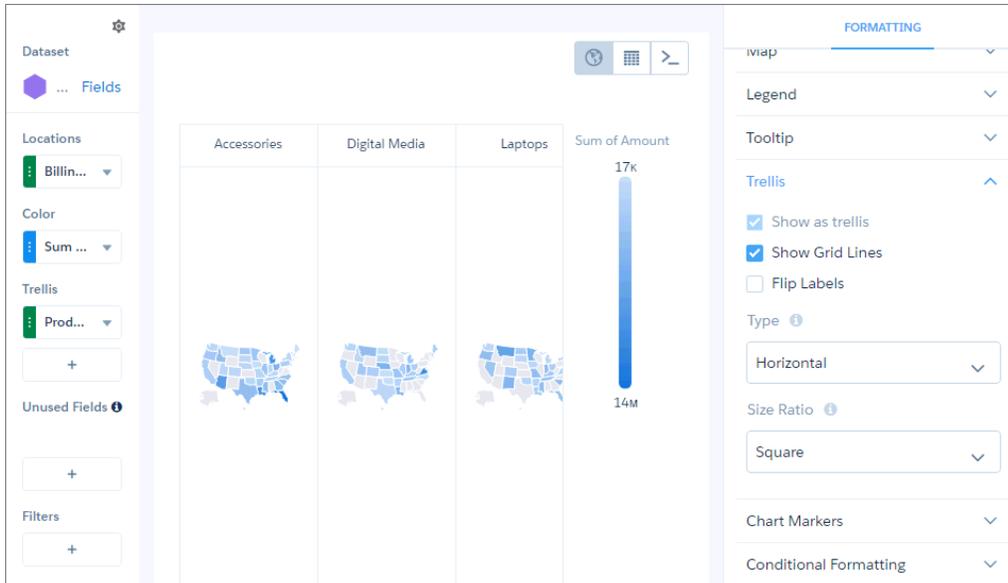
Trellis view can also be used to visualize values that may otherwise be difficult to see in a chart. For example, some of the values shown by this stacked bar chart are difficult to pick out.

By using trellis view, you can separate the values so that smaller values can be picked out more easily.
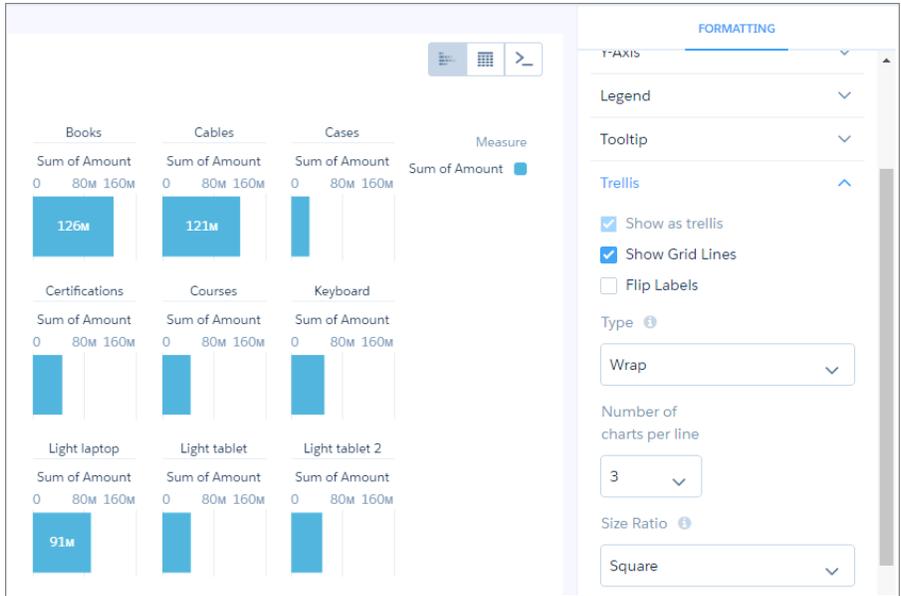
In any trellis view except for Wrap, you can remove the grid lines.
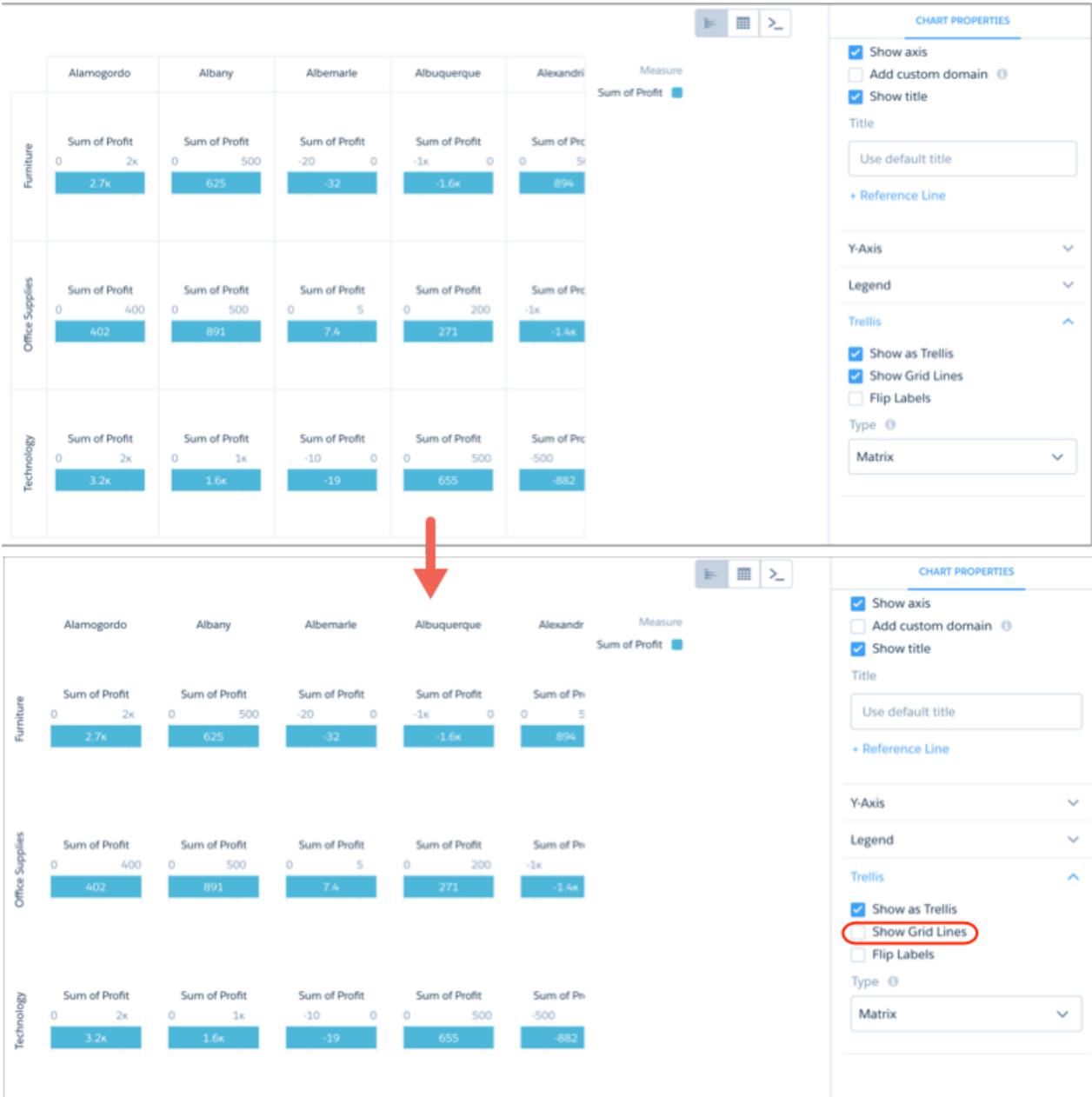
You can also move the labels to the other side of the chart.
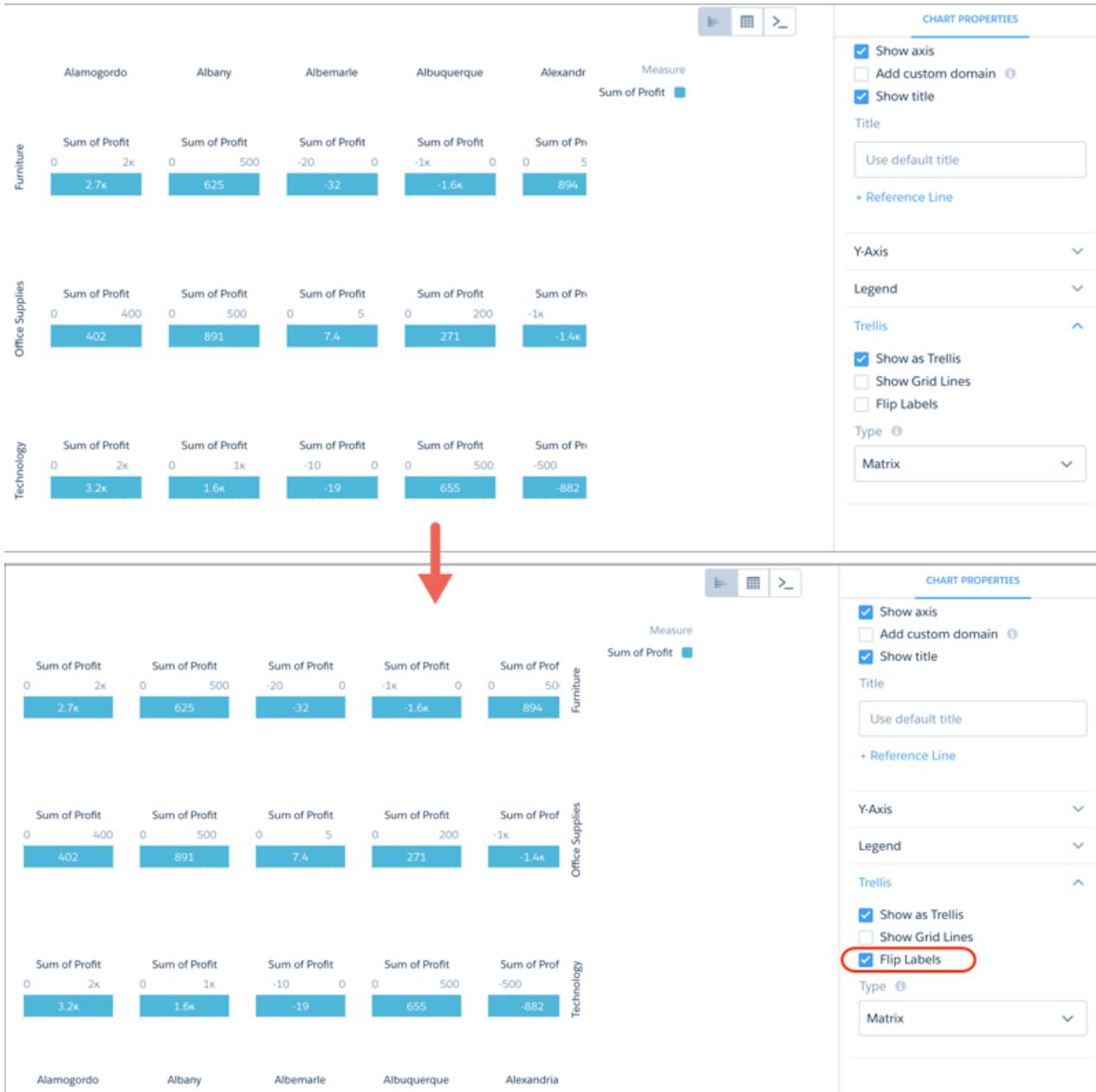
Visualizing a large number of charts in a trellis can create a view that is difficult to read.
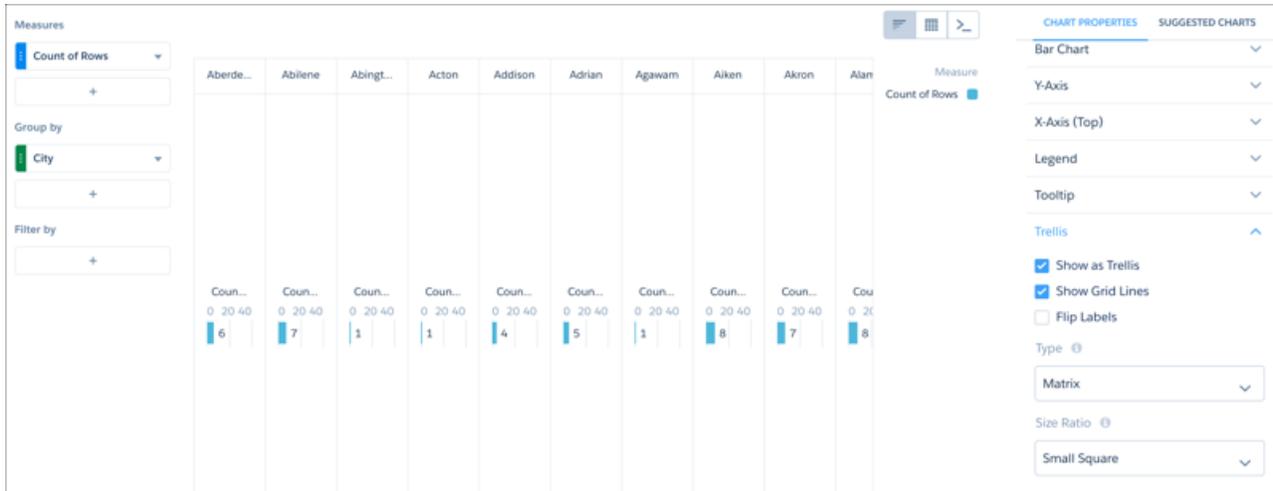
Adjust the matrix type and the size ratio to create a trellised chart that is easier to view and understand.

Add Icons to Tableau CRM Charts

Bar charts (but not stacked bars), dot-plot charts, and line charts can have icons along with dimension values as labels. Polar gauge charts can include icons in their centers.

To add icons to a chart, use a dataset with an ungrouped URL dimension whose values are URLs to the icon images. For example, each row in the following dataset has a URL that points to a flag image stored in your org:

```
Country,Country_code,Region,flag_img_url,Accounts,Value
Australia,aus,South
Pacific,https://MyDomainName--c.documentforce.com/file-asset/flagaustraliapng,1898,22930651

China,chn,East
```

```
Asia,https://MyDomainName--c.documentforce.com/file-asset/flagchinapng,2051,29754009

Europe,eur,Europe,https://MyDomainName--c.documentforce.com/file-asset/flageuropepng,4668,61238042

France,fra,Europe,https://MyDomainName--c.documentforce.com/file-asset/flagfrancepng,2303,28746829

India,ind,Asia,https://MyDomainName--c.documentforce.com/file-asset/flagindiapng,2721,32037499

Italy,ita,Europe,https://MyDomainName--c.documentforce.com/file-asset/flagitalypng,1722,26340986

    Japan,jpn,East
Asia,https://MyDomainName--c.documentforce.com/file-asset/flagjapanpng,2872,36916012
    Russian Federation,rus,East
Europe,https://MyDomainName--c.documentforce.com/file-asset/flagrussiapng,1847,27387456
    South
Africa,zaf,Africa,https://MyDomainName--c.documentforce.com/file-asset/flagsouthafricapng,879,9794857

    South Korea,kor,East
Asia,https://MyDomainName--c.documentforce.com/file-asset/flagsouthkoreapng,1643,21934785

    United States of America,usa,North
America,https://MyDomainName--c.documentforce.com/file-asset/flagusapng,6527,87044191
```

After you upload the dataset to Tableau CRM, explore the dataset using *only one* grouping. Then edit the SAQL query to add the image URLs to the result set.

1. Explore the dataset to visualize the data you want to view in chart mode.
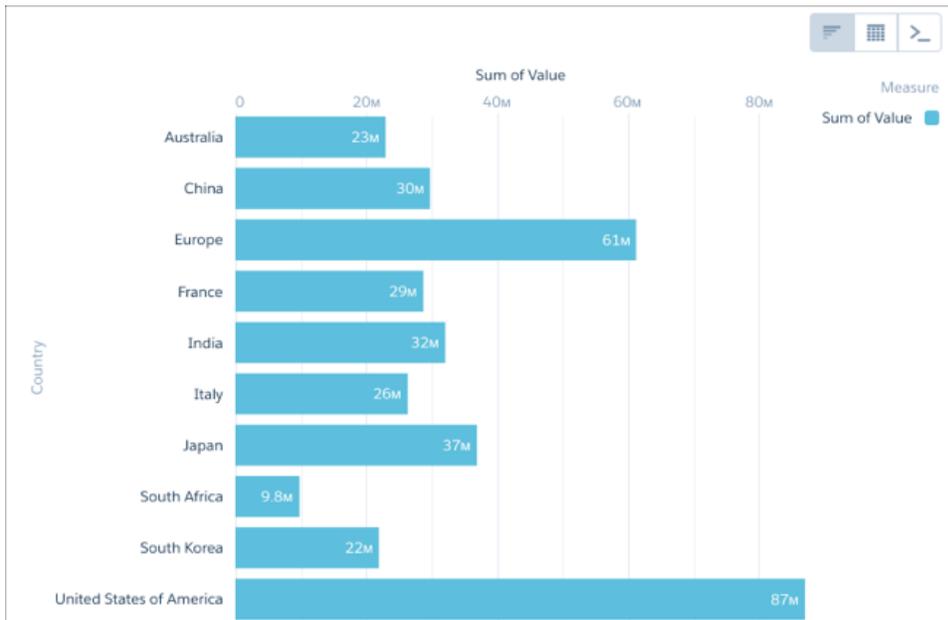
2. Switch to SAQL mode and edit the query using a non-grouping SAQL function such as `first()`.

3. Return to chart mode, then click the gear menu to open the chart properties.

4. Click **Use Icons**, then select the name of the dimension containing the icon URLs from the **Icon Source** menu.

    Use the **Fit Mode** and **Shape** menus to adjust the icons' appearance.

    📝 Note: The **Use Icons** menu contains the name of every non-grouped dimension used in your query, even if these dimensions do not contain image URLs.

To use Chatter profile images, use your browser's development tools to find the image's underlying URL.
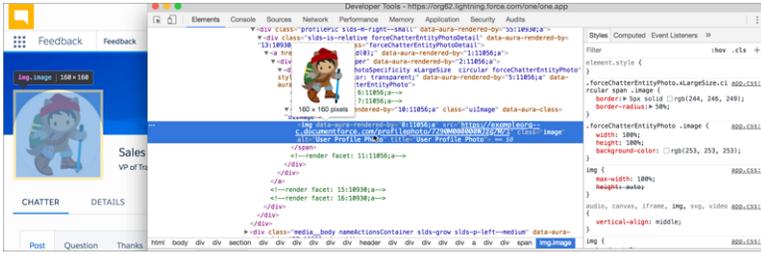
Once you've collected the URLs, add them to a column in your dataset before uploading the dataset to Tableau CRM.

```
name,url_src,num_accts,value_accts,territory

astro,https://MyDomainName--c.documentforce.com/profilephoto/7290M000000WJzg/M/1,829,1309742,North
 America
 ...
```

Bar Charts

Tableau CRM provides the following types of bar charts: Bar, Column, Stacked Bar, and Stacked Column. Use the Bar or Column chart to provide a quick visual comparison of related values. Use the Stacked Bar or Stacked Column chart to show groups within each bar.

Create a Horizontal Bar Chart

Use a horizontal bar chart to compare values across one or more categories. Using the auto-fit feature, you can fit a bar chart into a crowded dashboard without the necessity of long scroll bars or loss of information.

For example, create a bar chart that analyzes the average opportunity amount by opportunity owner.

1. In the explorer, click ▥ and then select the **Bar** chart type.

2. In the Bar Length field, add one or more measures.

3. In the Bars field, add one or more dimensions to analyze the measures by.

4. To rank the records and see the highest or lowest values, click the down arrow next to the measure and sort the results.

5. To change the chart display, click 🖌 and set the chart properties in the Formatting panel.

   For example, to show all bars without scrolling, select **Fit** in the Auto Fit field.

> 💡 **Tip:** If you resize the bar chart while arranging a dashboard and lose the labels, choose **Preserve labels** in the Auto Fit field to restore them.

Create a Stacked Column Chart

Use a stacked column or bar chart to compare parts of a whole and show values across one or more categories.

For example, create a stacked column chart that analyzes the total opportunity amount by opportunity owner and breaks down total opportunity amount by opportunity type within each bar.

1. In the explorer, click ▥ and then select the **Stacked Column** chart type.

2. In the Bar Length field, add the Sum of Amount measure.

3. In the Bars field, add one or more dimensions to analyze the measures by.

4. To rank the records and see the highest or lowest values, click the down arrow next to the measure and sort the results.
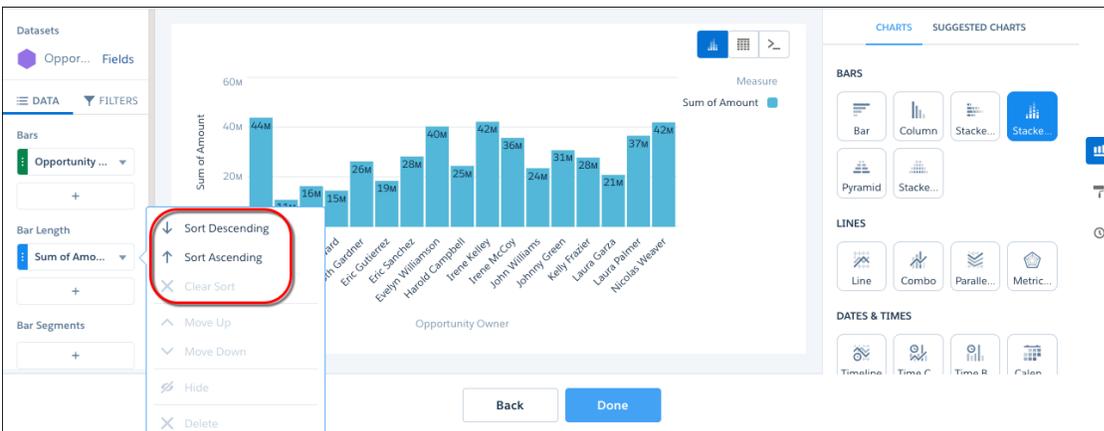
5. In the Bar Segments field, select **Stage**.
   Each chart bar breaks up the total amount to show the value contributed by each opportunity stage.

6. To change the chart display, click ⬤ and set the chart properties in the Formatting panel.

   For example, to show the total for each bar, select a value in the Show Total field. To subtract negative values from the total, select **Net**. To ignore the negative signs and add the absolute values, select **Absolute**.
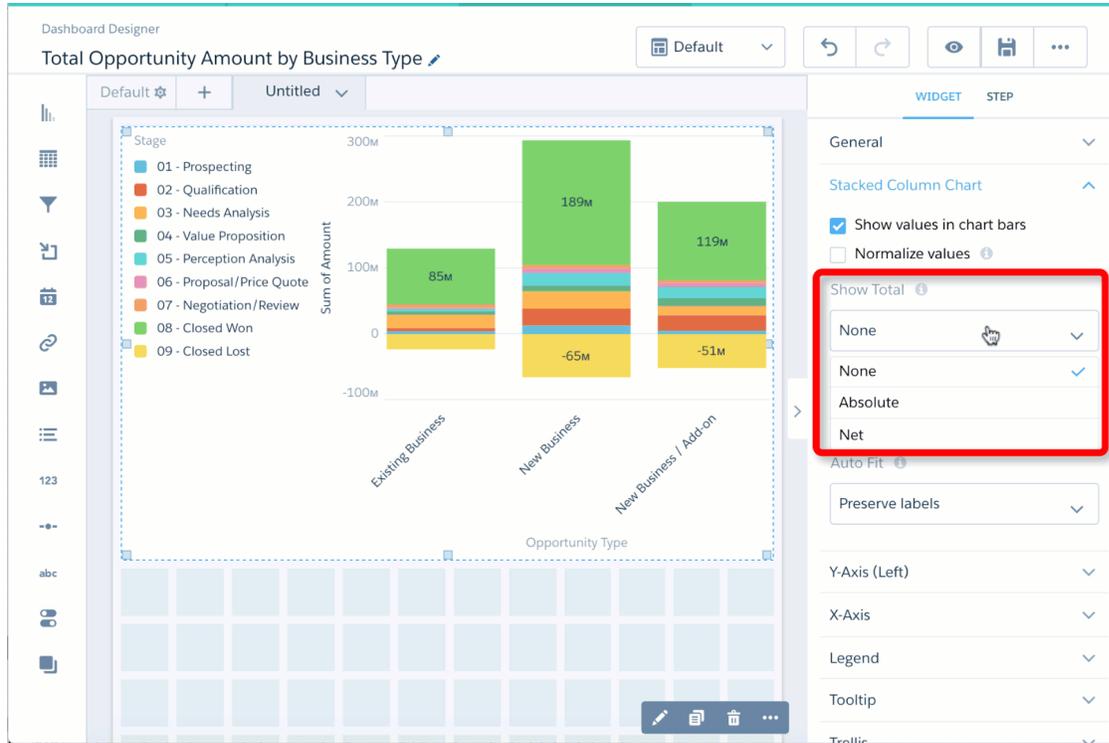
Calendar Heat Map Charts

Calendar heat maps are useful for visualizing recurring discrete activities, such as closing accounts, over long periods of time. In this chart, you can easily change the granularity of the time-based grouping, such as from month to week or day.

To use a calendar heat map, set a measure in the Color field and a date dimension in the Time Periods field. The underlying query groups the measure by the specified time period. The chart shows a color for each time period based on the measure.
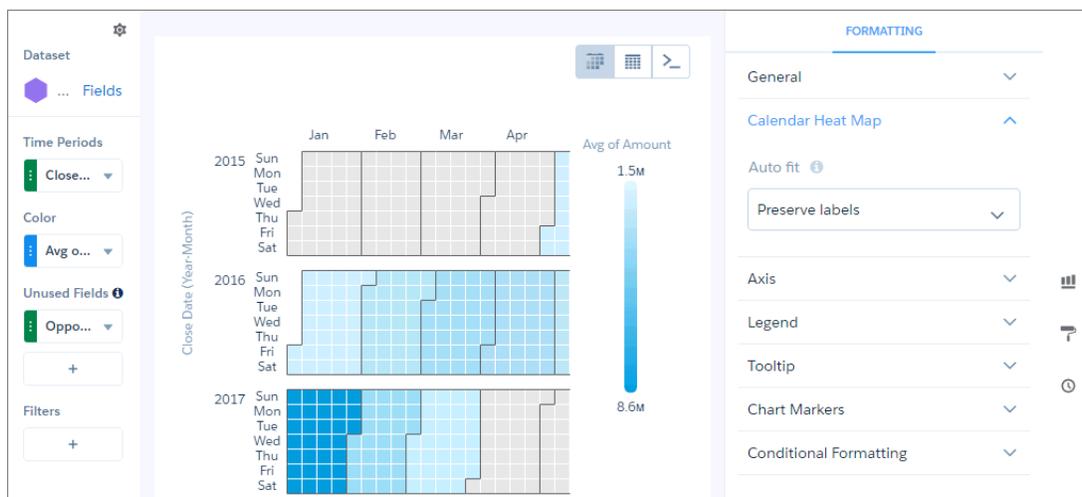
Column Charts

Use a column chart (also called vertical bar chart) to show relative counts of things, such as leads or dollars. Use a stacked column chart to show groups within each bar.
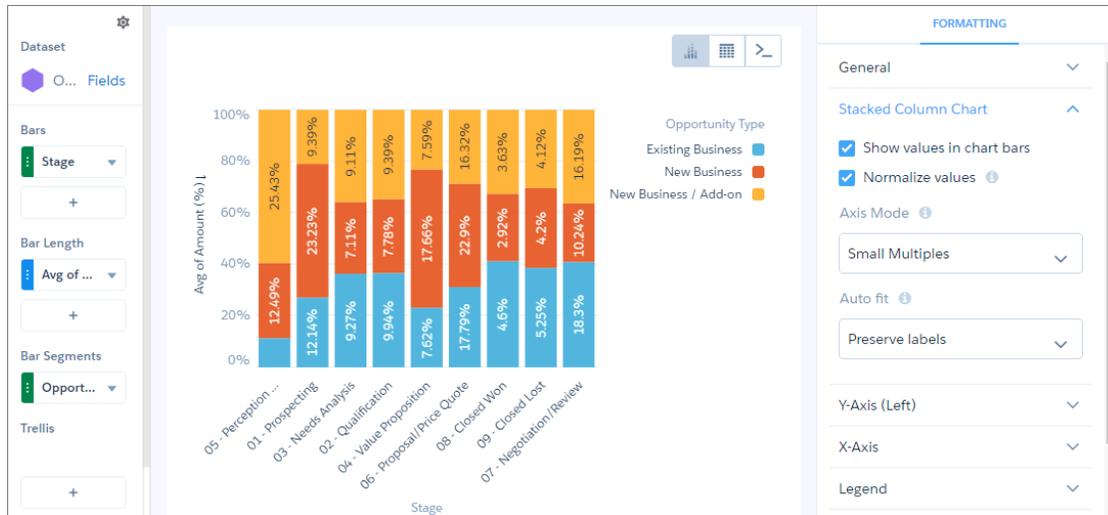
Combo Charts

Use a combo chart to display at least two related series of data, such as discrete grouped values as vertical bars overlaid with a line chart representing an average value.

Create a Combo Chart

Use a combo chart to compare two measures by the same dimension. For example, when comparing deal sizes across industries, use a combo chart to analyze the total amount and the number of deals for each industry.

1. In the explorer, click ![icon] and then select the **Combo** chart type.

2. In the X-Axis field, add the dimension to analyze the measures by. For example, select **Industry**.

3. In the Y-Axis field, add at least two measures.

4. To change the chart display, click ![icon] and set the chart properties in the Formatting panel.

   For example, set the axis mode to specify whether both measures appear on the same axis.
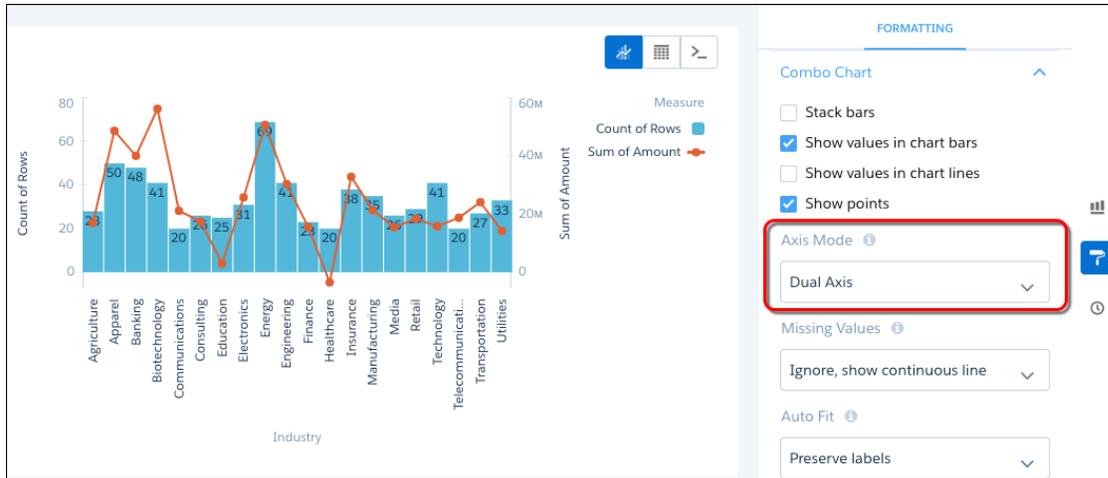
Create a Combo Chart with Stacked Bars

In a combo chart, you can change the bar to a stacked bar to show and compare parts of a whole. For example, add a combo chart that analyzes opportunities by industry. Use the line to show total number of opportunities and the bar to show total opportunity amount. You can then divide each bar by forecast type to show the contribution of each type to the total amount. Let's learn how to build this example.

1. In the dashboard, create a combo chart that shows total opportunity amount as a bar and count of rows as a line, grouped by industry.
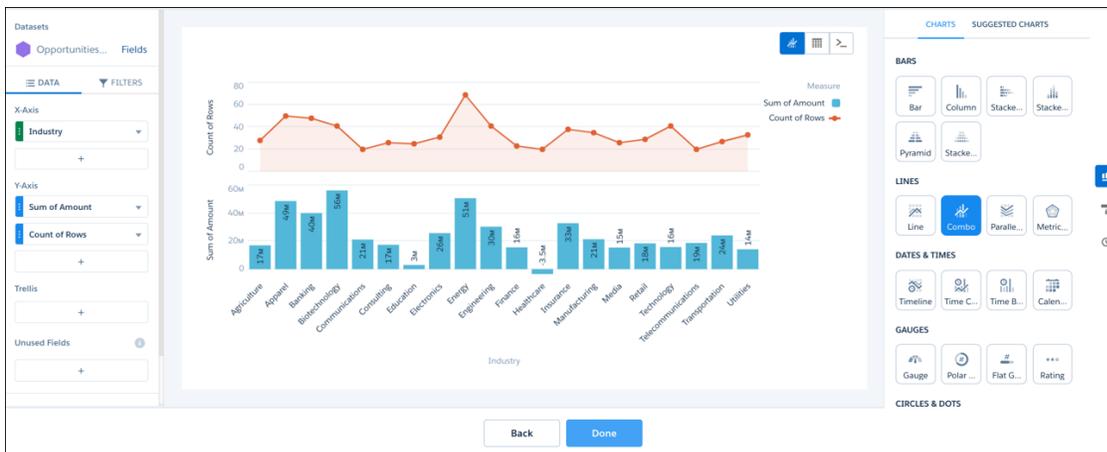
2. Click ▦ , and then click ▦ .

   We switched to the compare table so that we can add multiple Sum of Amount measures, each with their own filter. Let's create a Sum of Amount measure for each forecast category: Best Case, Closed, Commit, and Pipeline. Let's start by filtering the existing Sum of Amount measure by the Best Case forecast category.

3. To filter the sum by Best Case category, click the dropdown next to Sum of Amount, and select **Add a Filter**.
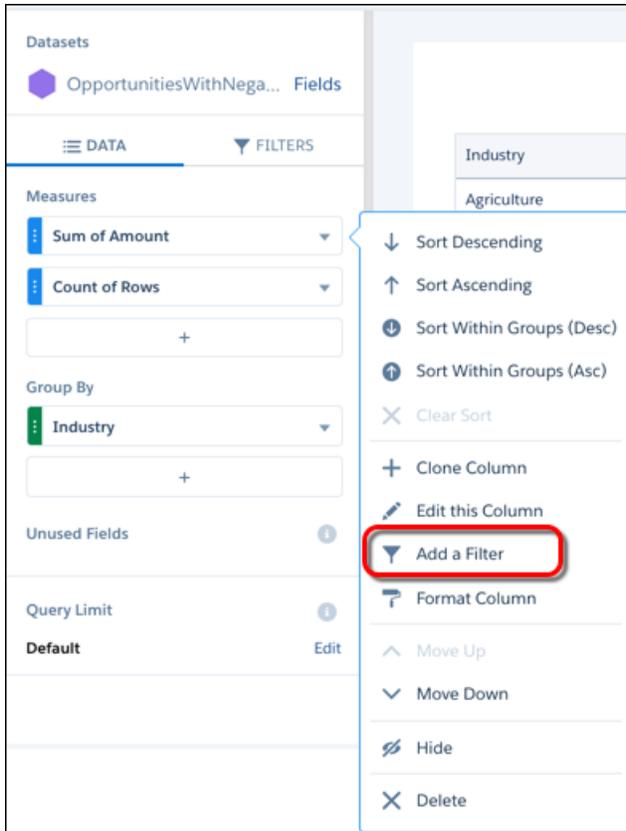
4. Select **Forecast Category** > **Best Case**, and then click **Add**.

This filter applies only to this measure.
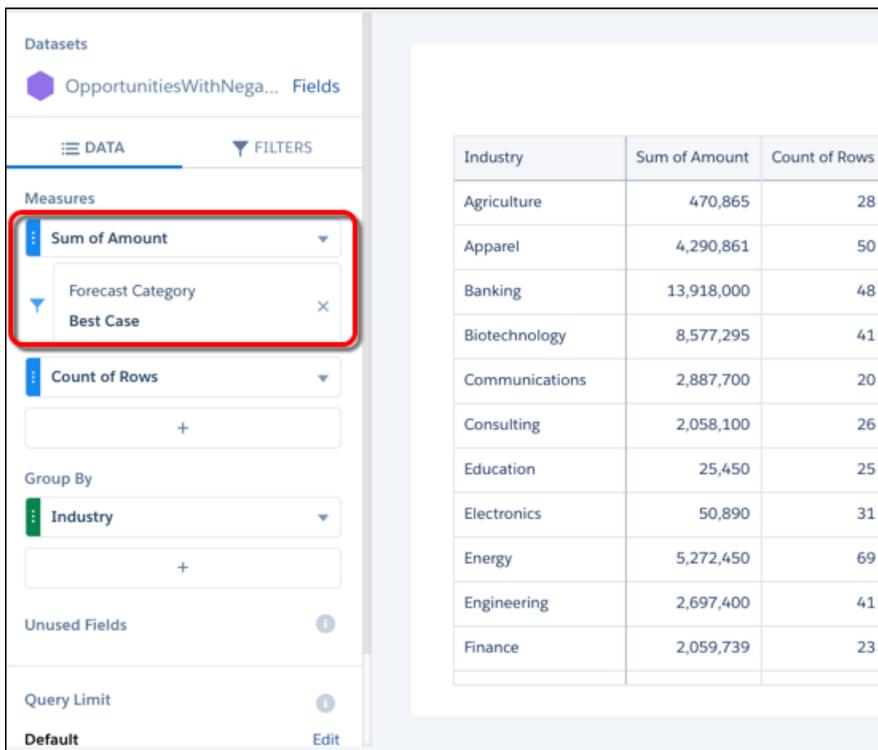
5. To create three more instances of the Sum of Amount measure, click the dropdown arrow next to Sum of Amount, and select **Clone Column**. Repeat this step two more times so that you have four Sum of Amount measures.
   You now have four of the same Sum of Amount measures with the same filter.

6. For second, third, and fourth Sum of Amount measures, change the Forecast Category filter to **Closed**, **Commit**, and **Pipeline**, respectively.
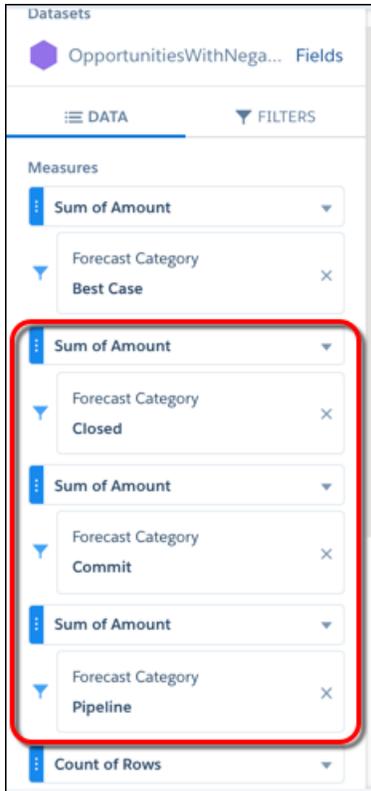
7. To make the measure labels more descriptive, click the dropdown next to the top Sum of Amount measure, and select **Edit this Column**. Change the column header to "Best Case Total." Use a similar label for each of the remaining Sum of Amount measures.

8. Show the widget properties by clicking 🖌.

9. In the widget properties, expand the Series section, and then click **Customize series**.

10. To show all Total measures as bars, select each Total measure in the Measure field and then **Bar** in the Show As field. Leave the Count of Rows measure showing as a line.

> 📝 **Note:** You can also choose **Dot** in the Show As field to create a bullet chart.

11. Under the Combo Chart section, change the Axis Mode to **Dual Axis**, and select **Stack bars** to stack all the bars for each industry.

> **Note:** The Stack bars option is available when Axis Mode is set to **Single Axis** or **Dual Axis**, not **Small Multiples**. Also, when you set Axis Mode to **Dual Axis**, the first measure appears on the left axis. If the first measure shows as a bar and you stack bars, all bar measures appear on the left axis. Similarly, if the first measure shows as a line and you stack bars, the line measure appears on the left axis and the bar measures appear on the right.
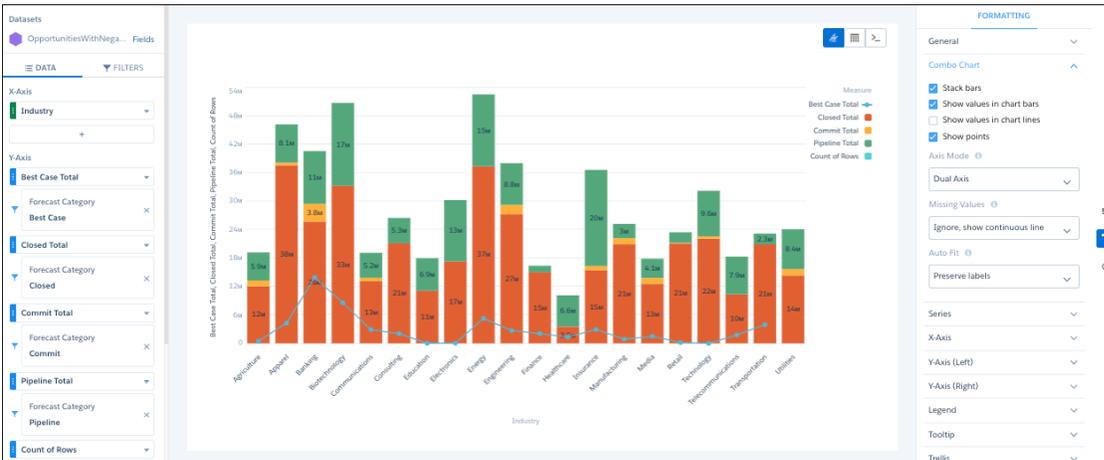
Donut Charts

Use a donut chart when you have a grouping and want to show not only the proportion of a single value for each group member against the total, but also the total amount itself. If you remove the donut's center by selecting **0%** from the **Center Size** menu, this visualization is commonly referred to as a pie chart.

Create a Donut Chart

Donut charts are useful visualizations when comparing a measure across a few categories, which makes the chart readable.

For example, showing opportunities that cover a limited number of product lines can be visualized effectively using a donut chart.

1. In the explorer, click 📊 and then select the **Donut** chart type.

2. In the Segment Size field, add the measure that determines the size of each segment. For example, select the measure that shows the total opportunity amount.

3. In the Segment By field, add the dimension to group the data by. For example, select **Product Family**.
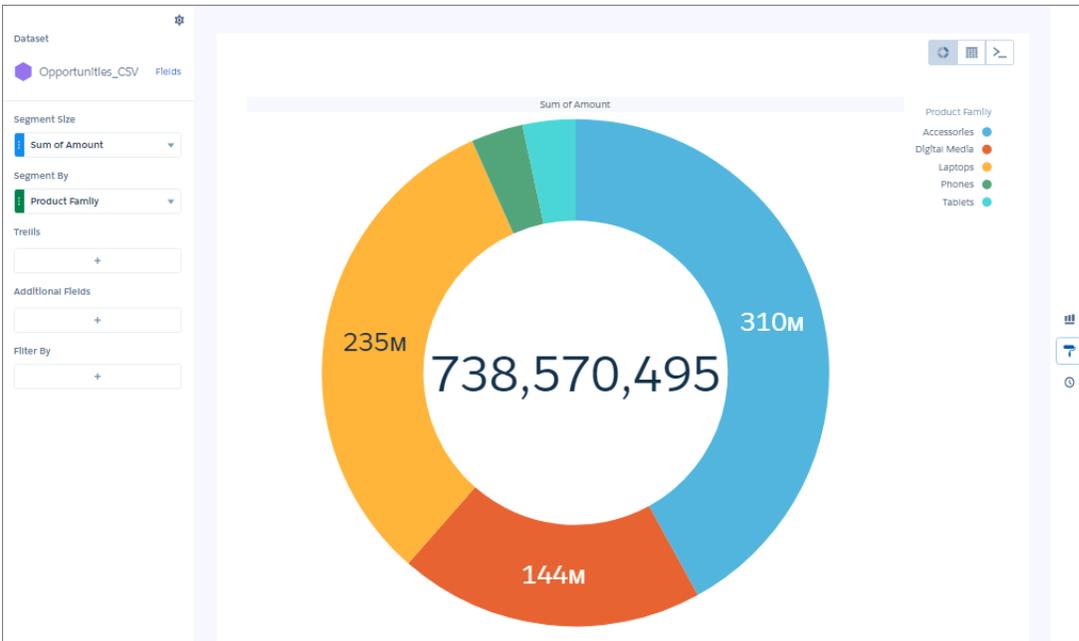
4. To highlight slices, click individual them.

   The total in the center changes to reflect the total of the selected slices.
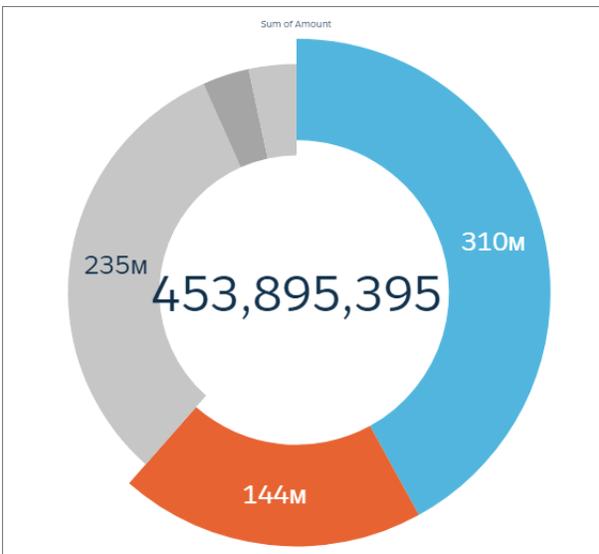
5. To change the chart display, click 🖌 and set the chart properties in the Formatting panel.

For example, select **Custom Text** from the Center Value menu, then enter custom text in the Custom Text field.
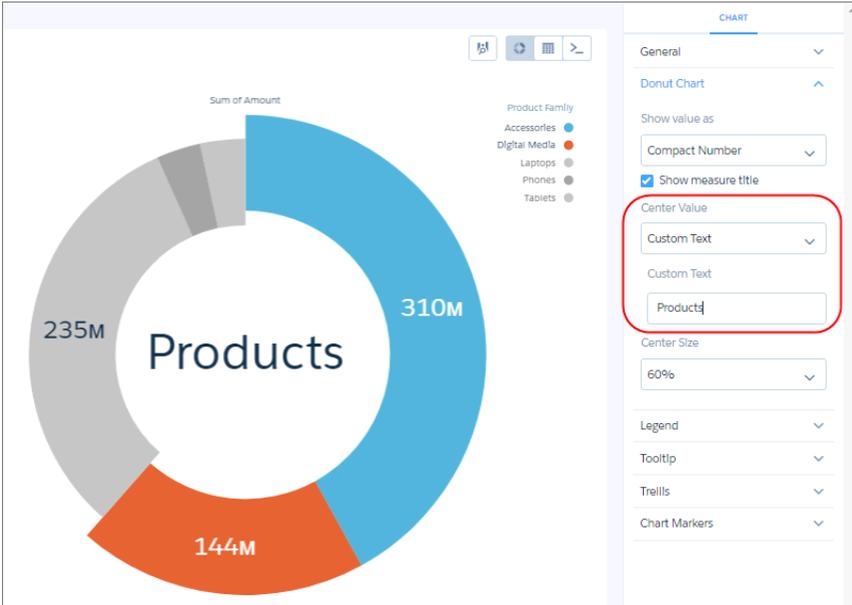
Dot Plot Charts

Horizontal and vertical dot plots use the size and coloring of bubbles to display multiple dimensions and measures. Use dot plots to visualize related data to compare performance or locate unusual values.

Create a Horizontal Dot Plot

Create a horizontal dot plot chart to analyze one or two measures across one or more dimensions.

For example, you can create a horizontal dot plot that shows the number of opportunities for each account owner and record type.

1. In the explorer, click ▥ and then select the **Horizontal Dot Plot** chart type.

2. In the X-Axis field, add the measure that you want to analyze. For example, select count of rows.

3. In the Y-Axis field, add at least one dimension that you want to analyze the measure by. For example, select **Account Owner**.

4. In the Bubble Size field, add the measure that determines the bubble size, such as sum of amount.

5. In the Bubbles field, add dimension that determines the different types of bubbles to show. For example, select **Account Type** to show bubbles for each record type.
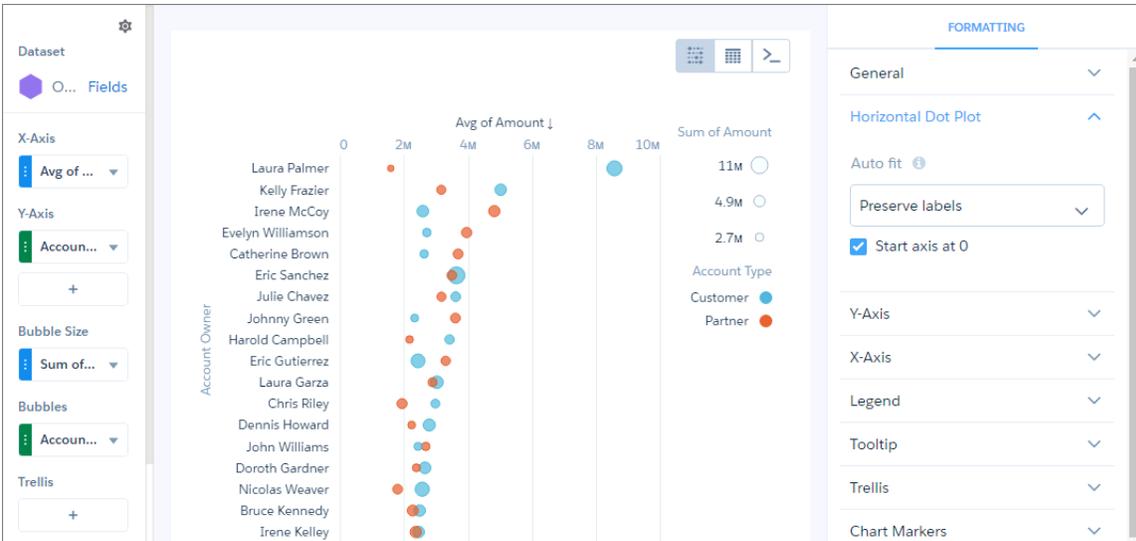
6. To change the chart display, click 🖌 and set the chart properties in the Formatting panel.

Funnel Charts

Use a funnel chart to visualize sequential data that can be broken up into stages, such as a sales cycle.

Create a Funnel Chart

Use a funnel chart to compare a measure throughout a process.

For example, you can use this chart to compare the opportunity amounts through different pipeline stages.

1. In the explorer, click 📊 and then select the **Funnel** chart type.

2. In the Segment Size field, add the measure that determines the size of each segment. For example, select the measure that shows the total opportunity amount.

3. In the Segments field, add the dimension to group the data by. For example, select **Stage**.
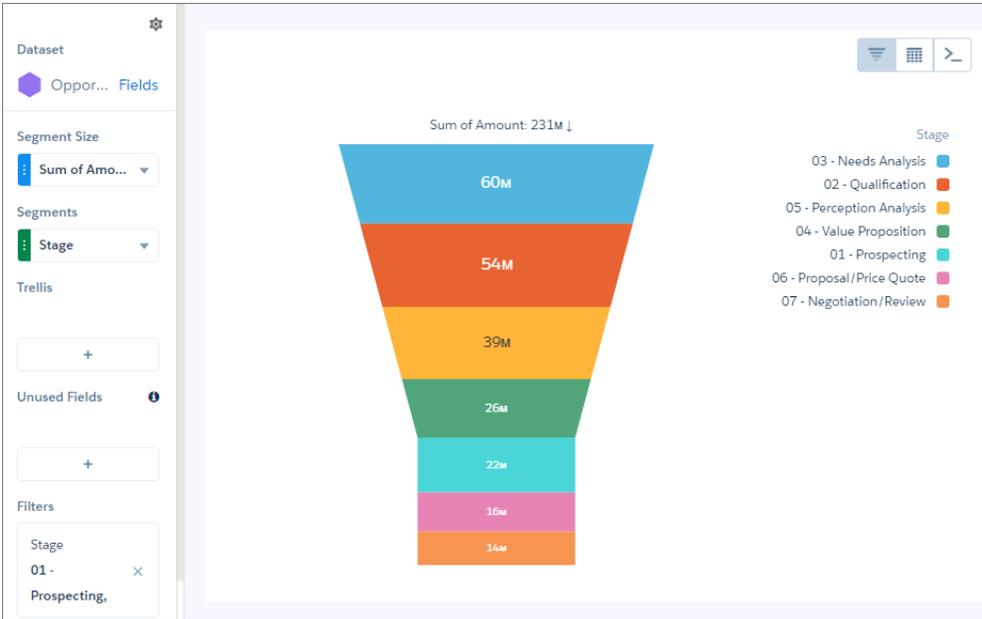
> 📝 **Note:** This funnel chart also has a filter that includes only open pipeline.

4. To change the chart display, click 🖌 and set the chart properties in the Formatting panel.
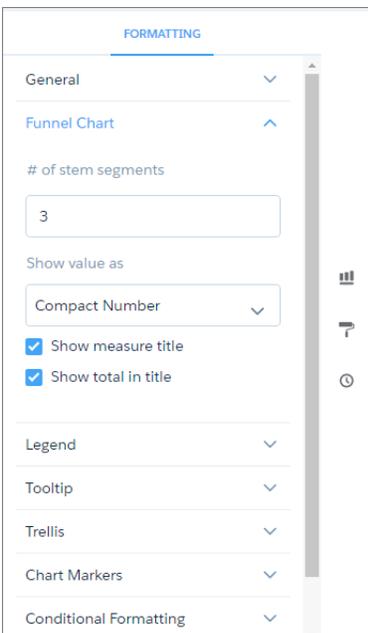
Gauge Charts

Use gauge charts to track progress along a single measure, such as how much revenue has been realized versus the target.

Create a Gauge Chart

If you've set goals for, as an example, average value for each account manager, gauge charts are a good choice for showing how close you are to that goal.

You've got a dataset with account information. Now you want to present it in a chart that shows how close each account is in reaching the associated goal. Each gauge chart can display one value per account manager, and highlight value ranges.

1. In the explorer, click and then select the **Polar Gauge** (round) chart type.

    A gauge (angular) and flat gauge (linear) are also available.

2. In the Value field, add the measure that you want to analyze, like average opportunity amount.

3. To compare the measure across every member of a category, like every account owner, add the dimension in the Trellis field.
    A separate gauge appears for each value of the selected dimension.

4. To rank the performance based on the measure, sort the measure in ascending or descending order. Default sort order is alphabetical order.
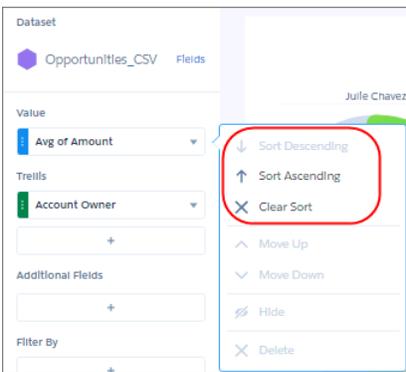
5. To change the chart display, click and set the chart properties in the Formatting panel.

    For example, set the Trellis Type field to **Wrap** so the chart can be viewed all at once, without having to scroll. Also, set the bins in Conditional Formatting to create band colors. The band makes it easy to see how close managers are to hitting their goal. The band color makes it easy to spot which managers are performing well and which can use some help.

The breakpoints set the range values where the color bands transition. In this example, the maximum (goal) specified in the Range Values section of the widget properties is $2 million. If account managers hit at least $1.4 million, they're in the green . If they hit between $1.2 and $1.4 million, they're in the yellow. Otherwise, they're in the red.

Bullet Charts

Bullet charts are perfect for comparing metrics against quantitative benchmarks and references, like current revenue with target revenue.

Create a Bullet Chart

Measure current revenue against target revenue and compare it with last quarter's revenue with a bullet chart.

You've got a dataset with opportunity information. From that dataset, you can measure current revenue by finding the sum of the amount field. With a bullet chart, you can then measure current revenue against qualitative values, like target revenue.

The background bar represents target revenue, divided into three sections that represent qualitative performance: red for bad, yellow for good, and green for great. The black bar (also known as a bullet) shows the current revenue. The blue reference line marks last quarter's revenue.

1. In the explorer, click 📊 and then select the **Bullet** chart type.

2. In the Value field, add the measure that you want to analyze, like sum of amount.

3. To customize the bullet chart, click 🎨 and set the chart properties in the Formatting panel.

   In this current revenue by target revenue example, set a target revenue, band colors, and a reference line.

   First, set target revenue. Expand **Range Values** and set **Max** to your target revenue. For example, $300,000,000.00.

   Let's make the bar and bullet a bit bigger. Expand **Bullet** and set **Range weight** to 200 and **Bar weight** to 40.

   To add last quarter's revenue, click **+ Reference Line**, set value to last quarter's revenue (for example, $190,000,000.00), and set **Label** to *Last Quarter*.

   Set bins in Conditional Formatting to create band colors on the background bar to create bands that represent qualitative performance: red for bad, yellow for good, and green for great. The black bar (also known as a bullet) shows the current revenue.

The bullet chart now makes it easy to see current revenue against target revenue.
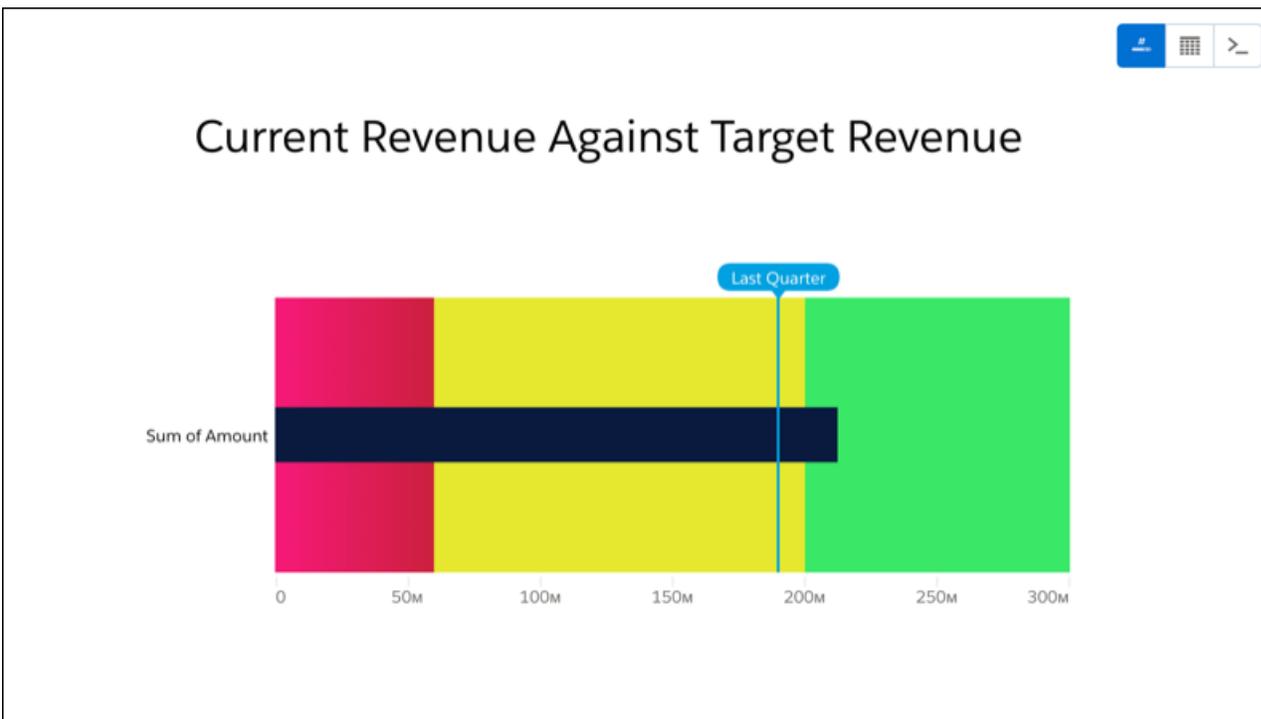
Heat Maps

Use a heatmap to visually enhance high- and low-value data when there's a single measure and multiple dimensions.

Create a Heat Map Chart

Use a heat map chart to analyze the distribution of data to find if there are concentrations in particular segments.

For example, use a heat map to determine which functional areas and quarters have the most support cases.

1. In the explorer, click 📊 and then select the **Heat Map** chart type.

2. In the X-Axis field, add a dimension to analyze the measure by. For example, add the Created Date dimension and choose **Year-Quarter**.

3. In the Y-Axis field, add another dimension. For example, select **Billing State/Province**.

4. In the Color field, add the measure that you want to analyze across both dimensions. For example, select **Avg** > **Amount** to look at the average opportunity amount.
   By default, the heat map shows a color gradient based on two colors.
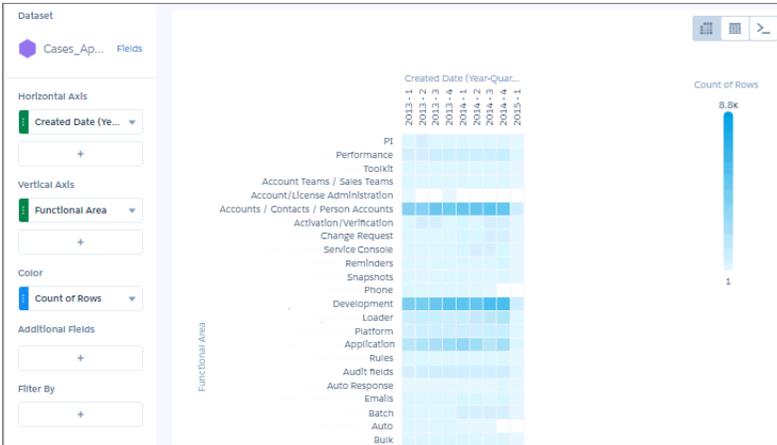
💡 **Tip:** When it's hard to tell the difference between gradients, use bins.

5. To change the chart display, click 🖌 and set the chart properties in the Formatting panel.

For example, to categorize values in high, medium, and low bins, select **Bins** under the Conditional Formatting section, and set them up based on thresholds.
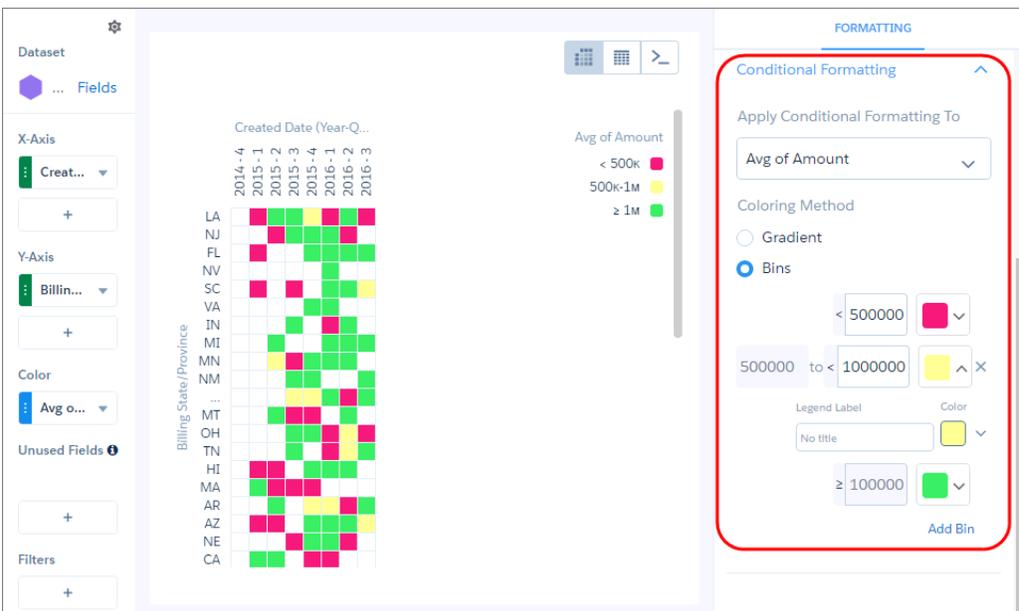
Line Charts

Use a line chart when you have one important grouping representing an ordered set of data and one or more values to show.

Line charts are useful for comparing the value of measures, such as sales and profit totals, over a series of categories (grouped dimensions) such as industries and sales territories.
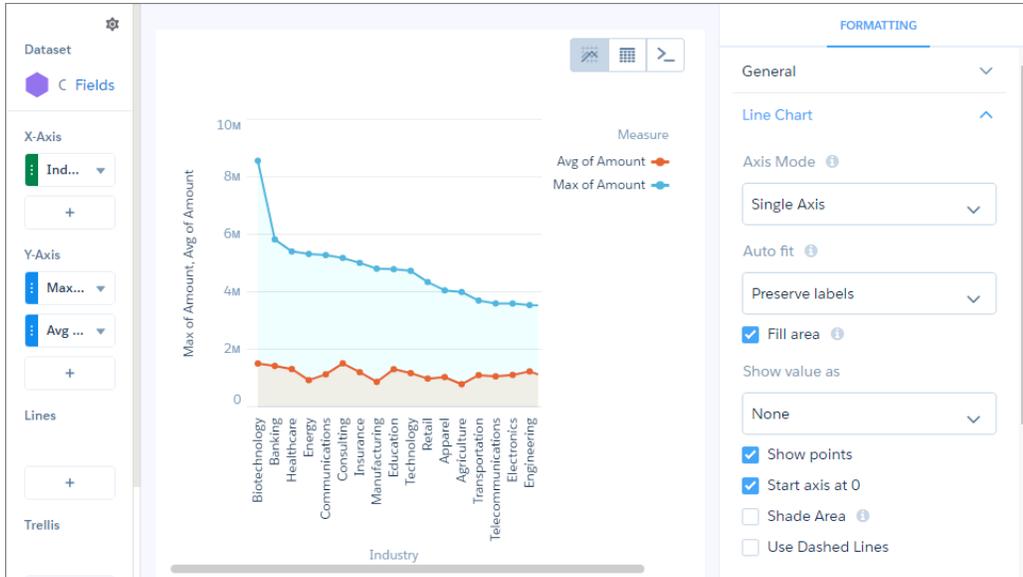
Map Charts

Use a map if you have data with a geographical component. Maps can shade areas in proportion to mapped values, allowing for visual pop-out of high-value areas. You can find the map chart in the dashboard designer.

Data for map charts must contain geographical names corresponding to the geographical entities represented on the map. Tableau CRM map charts are not case-sensitive, but specific codes or names for geographical entities such as US states and world countries are expected.

For maps of the USA, each row of data must contain a single field identifying the state by its two-letter postal code (recommended) or its full common name. A row with data linked to the state of California, for example, must have an identifying field with "CA" or "California", but not "Calif".

For global maps, the identifying country names must follow ISO 3166-1 (alpha 3) three-letter code (recommended) or be spelled out using the country names set by Tableau CRM. A row with data linked to the United Kingdom, for example, must have an identifying field with "GBR" or "United Kingdom", but not "Great Britain".

Create a Map Chart

Use a map chart to analyze data across geographical regions.

For example, you can use color saturation in a map chart to indicate the number of opportunities in the pipeline for each country in which you do business.

1. In the explorer, click ▥ and then select the **Map** chart type.

2. Click 🖌 and choose the type of map in the Map Type field. For example, to analyze measures by countries, choose **World Countries**.
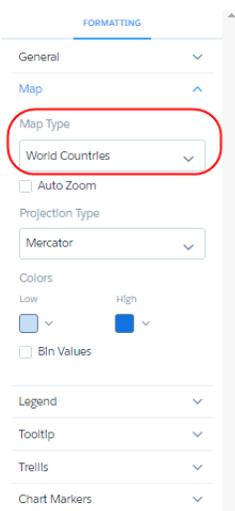
3. In the Locations field, add the dataset field that contains the locations. For example, select **Billing Country**, which corresponds to the World Countries map type.

4. In the Color field, add the measure that determines the color of each region. For example, select **Count of Rows** to show colors based on the number of opportunities for each country.
    By default, the map shows a color gradient based on two colors.
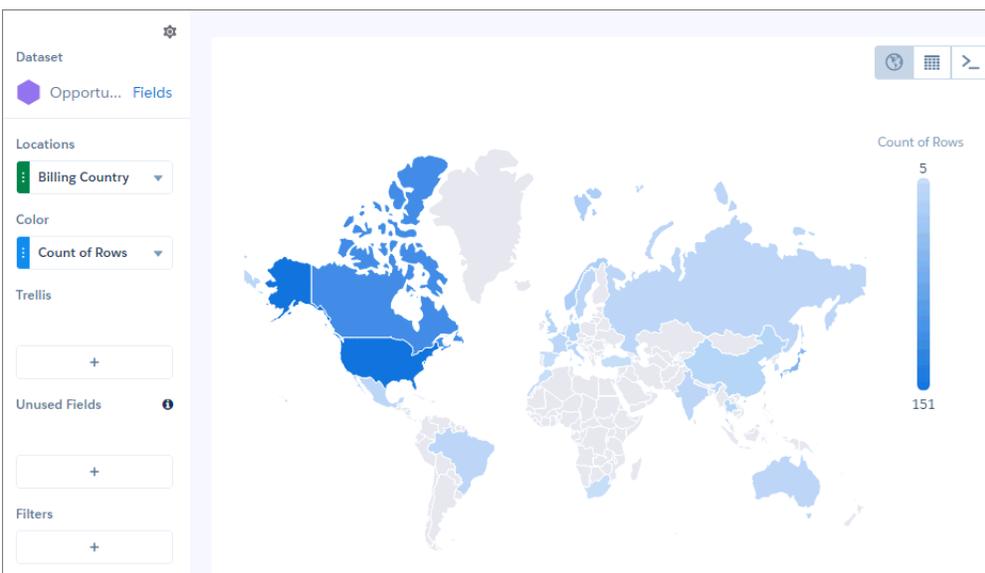
💡 Tip: When it's hard to tell the difference between gradients, use bins.

5. To bin values in high, medium, and low categories, select **Bins** under Conditional Formatting, and choose the low and high value thresholds and the legend labels and colors.
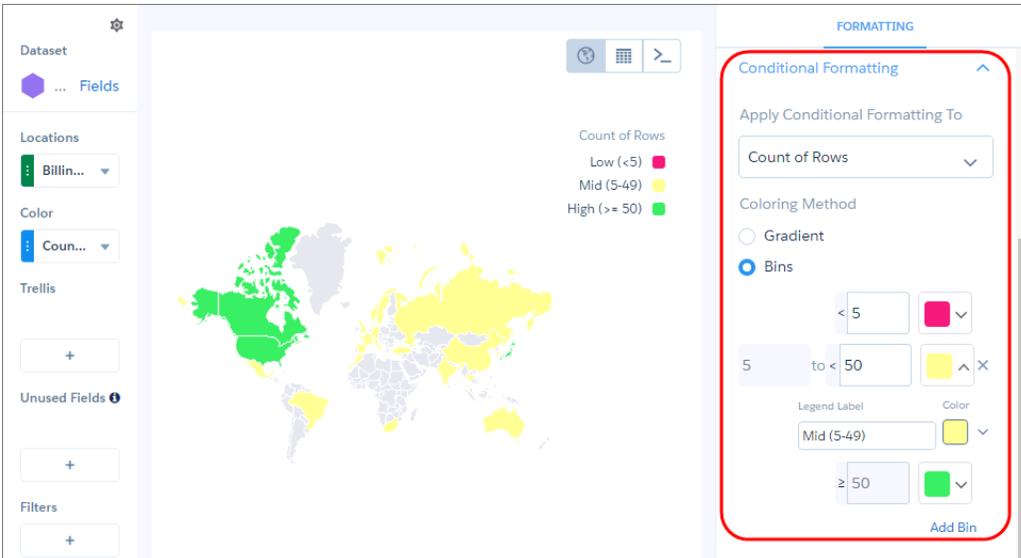
6. To make more changes to the chart display, set the chart properties in the Formatting panel.

Country Codes and Names

For data to show up correctly in a map of countries, the data must identify countries using either the ISO 3166-1 (alpha 3) three-letter country codes (recommended) or the spelled-out country names set by Tableau CRM.

The following table lists the supported country name for each ISO 3166-1 (alpha 3) country code.

Country Code	Country Name
AFG	Afghanistan
AGO	Angola
ALB	Albania
ARE	United Arab Emirates
ARG	Argentina
ARM	Armenia
ATF	French Southern and Antarctic Lands
AUS	Australia
AUT	Austria
AZE	Azerbaijan
BDI	Burundi
BEL	Belgium
BEN	Benin
BFA	Burkina Faso

Country Code	Country Name
BGD	Bangladesh
BGR	Bulgaria
BHR	Bahrain
BHS	The Bahamas
BIH	Bosnia and Herzegovina
BLR	Belarus
BLZ	Belize
BMU	Bermuda
BOL	Bolivia
BRA	Brazil
BRN	Brunei
BTN	Bhutan
BWA	Botswana
CAF	Central African Republic
CAN	Canada
CHE	Switzerland
CHL	Chile
CHN	China
CIV	Ivory Coast
CMR	Cameroon
COD	Democratic Republic of the Congo
COG	Republic of the Congo
COL	Colombia
CRI	Costa Rica
CS-	Kosovo
CUB	Cuba
CYP	Northern Cyprus
CYP	Cyprus
CZE	Czech Republic
DEU	Germany

Country Code	Country Name
DJI	Djibouti
DNK	Denmark
DOM	Dominican Republic
DZA	Algeria
ECU	Ecuador
EGY	Egypt
ERI	Eritrea
ESH	Western Sahara
ESP	Spain
EST	Estonia
ETH	Ethiopia
FIN	Finland
FJI	Fiji
FLK	Falkland Islands
FRA	France
FRO	Faeroe Is.
GAB	Gabon
GBR	United Kingdom
GEO	Georgia
GHA	Ghana
GIN	Guinea
GLP	Guadeloupe
GMB	Gambia
GNB	Guinea Bissau
GNQ	Equatorial Guinea
GRC	Greece
GRL	Greenland
GTM	Guatemala
GUF	French Guiana
GUY	Guyana

Country Code	Country Name
HKG	Hong Kong
HND	Honduras
HRV	Croatia
HTI	Haiti
HUN	Hungary
IDN	Indonesia
IND	India
IRL	Ireland
IRN	Iran
IRQ	Iraq
ISL	Iceland
ISR	Israel
ITA	Italy
JAM	Jamaica
JEY	Jersey
JOR	Jordan
JPN	Japan
KAZ	Kazakhstan
KEN	Kenya
KGZ	Kyrgyzstan
KHM	Cambodia
KOR	South Korea
KWT	Kuwait
LAO	Laos
LBN	Lebanon
LBR	Liberia
LBY	Libya
LKA	Sri Lanka
LSO	Lesotho
LTU	Lithuania

Country Code	Country Name
LUX	Luxembourg
LVA	Latvia
MAR	Morocco
MDA	Moldova
MDG	Madagascar
MEX	Mexico
MKD	Macedonia
MLI	Mali
MLT	Malta
MMR	Myanmar
MNE	Montenegro
MNG	Mongolia
MOZ	Mozambique
MRT	Mauritania
MUS	Mauritius
MWI	Malawi
MYS	Malaysia
NAM	Namibia
NCL	New Caledonia
NER	Niger
NGA	Nigeria
NIC	Nicaragua
NLD	Netherlands
NOR	Norway
NPL	Nepal
NZL	New Zealand
OMN	Oman
PAK	Pakistan
PAN	Panama
PER	Peru

Country Code	Country Name
PHL	Philippines
PNG	Papua New Guinea
POL	Poland
PRI	Puerto Rico
PRK	North Korea
PRT	Portugal
PRY	Paraguay
PSE	West Bank
QAT	Qatar
REU	Reunion
ROU	Romania
RUS	Russia
RWA	Rwanda
SAU	Saudi Arabia
SDN	Sudan
SEN	Senegal
SGP	Singapore
SLB	Solomon Islands
SLE	Sierra Leone
SLV	El Salvador
SOM	Somaliland
SOM	Somalia
SRB	Republic of Serbia
SSD	South Sudan
SUR	Suriname
SVK	Slovakia
SVN	Slovenia
SWE	Sweden
SWZ	Swaziland
SYR	Syria

Country Code	Country Name
TCD	Chad
TGO	Togo
THA	Thailand
TJK	Tajikistan
TKM	Turkmenistan
TLS	East Timor
TTO	Trinidad and Tobago
TUN	Tunisia
TUR	Turkey
TWN	Taiwan
TZA	United Republic of Tanzania
UGA	Uganda
UKR	Ukraine
URY	Uruguay
USA	United States of America
UZB	Uzbekistan
VEN	Venezuela
VNM	Vietnam
VUT	Vanuatu
YEM	Yemen
ZAF	South Africa
ZMB	Zambia
ZWE	Zimbabwe

Properties of Maps

The map chart is configured with the properties in the dashboard designer. These properties have equivalent JSON properties.

Configure Maps to Auto-Zoom on Selected Areas

When you select datapoints in a particular geographical area of your map chart, the data in that area is highlighted.
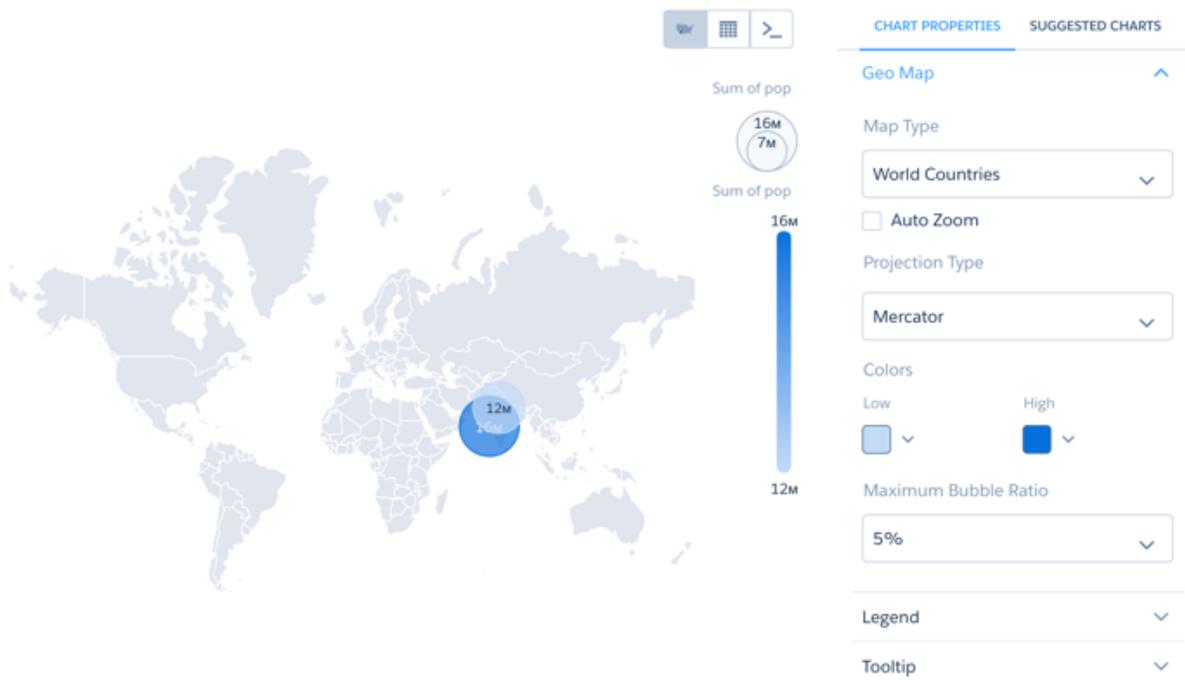
With auto-zoom, the map chart both highlights the data and zooms in the geographical area.
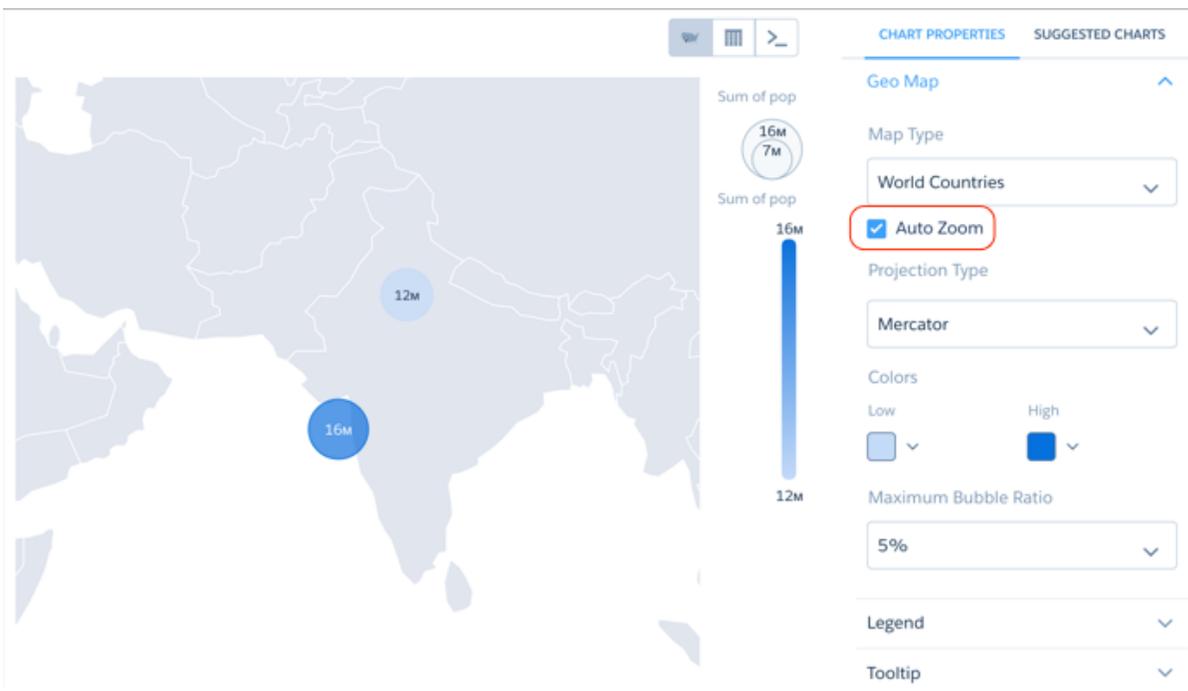
Any filtering that occurs on the visualized data can trigger auto-zoom, including faceting in dashboards, allowing you to create dynamic maps.

Set the Map Projection Type for Map Charts

A map projection is a method for representing shapes from the curved surface of the Earth (or any other curved surface) using shapes on a flat surface. Some map projection types preserve the accuracy of shape outlines while sacrificing the correct representations of area sizes, while others preserve relative size but distort the outlines. You can apply the projection of your choice using the Projection Type menu.
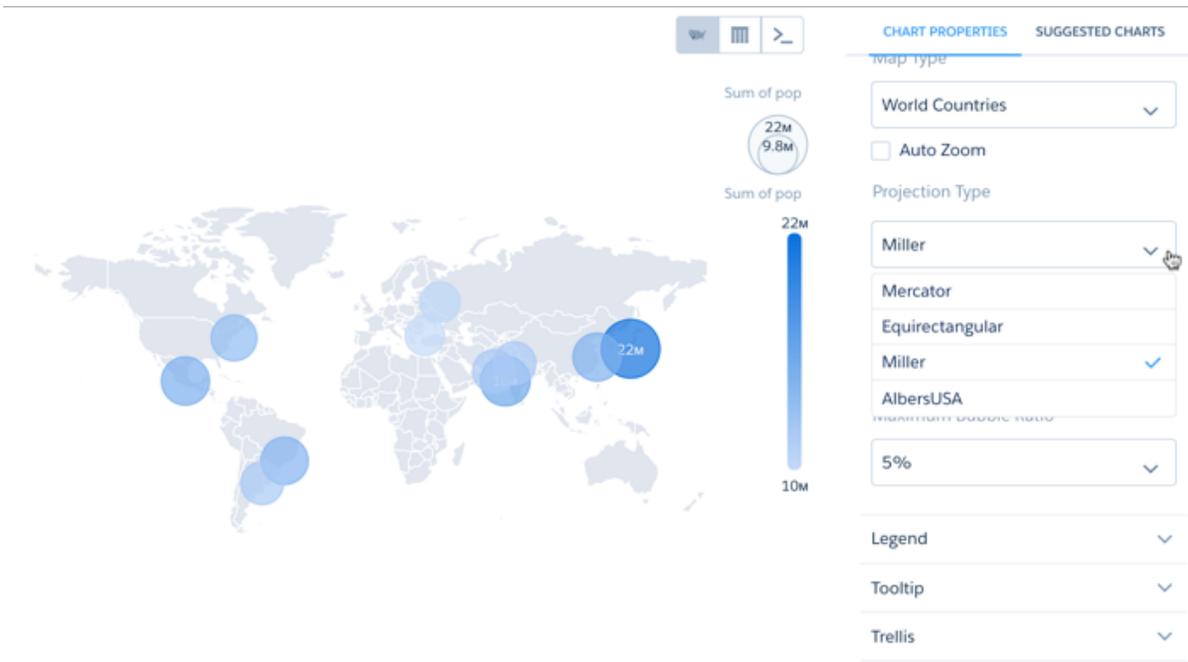
JSON Representation Map Chart

This is an example map JSON with a United States map.

```
{
  "visualizationType": "choropleth",
  "step": "step_newmap",
  "map": "US States",
  "legend": {
    "show": false,
    "inside": false,
    "showHeader": true,
    "position": "right-top"
  },
  "title": {
    "label": "",
    "subtitleLabel": "",
    "align": "center"
  },
  "theme": "wave",
  "lowColor": "#C5DBF7",
  "highColor": "#1674D9",
  "trellis": {
    "enable": false,
    "type": "x",
    "chartsPerLine": 4
  }
}
```

Other available values for the `map` property are:

"World Countries", "Americas", "North America", "South America", "Central America", "Caribbean", "Europe", "Eastern Europe", "Northern Europe", "Southern Europe", "Western Europe", "Central Europe", "Scandanavia", "Asia", "South Asia", "South East Asia", "East Asia", "Australasia", "Africa", "East Africa", "Middle Africa", "North Africa", "Southern Africa", "Western Africa", "EMEA", "APAC".

Bubble Map Charts

Use a bubble map chart to visually indicate measure values in specified geographical areas.

Create a Bubble Map Chart

Tableau CRM bubble-map charts use the same maps available to map charts, giving you another choice representing measure values on the map.

For example, you can use bubbles instead of color saturation alone to indicate the number of opportunities in the pipeline for each state in the United States.

1. In the explorer, click 📊 and then select the **Bubble Map** chart type.

2. Click 🖌 and choose the type of map in the Map Type field. For example, to analyze measures by state, choose **US States**.

3. In the Locations field, add the dataset field that contains the locations. For example, select **Billing State/Province** if the map type is US States.

4. In the Bubble Size field, add the measure that determines the bubble size.

5. In the Bubble Color field, add the measure that determines the bubble color.

6. Set the chart properties in the Formatting panel to make other adjustments, such as entering the chart title.
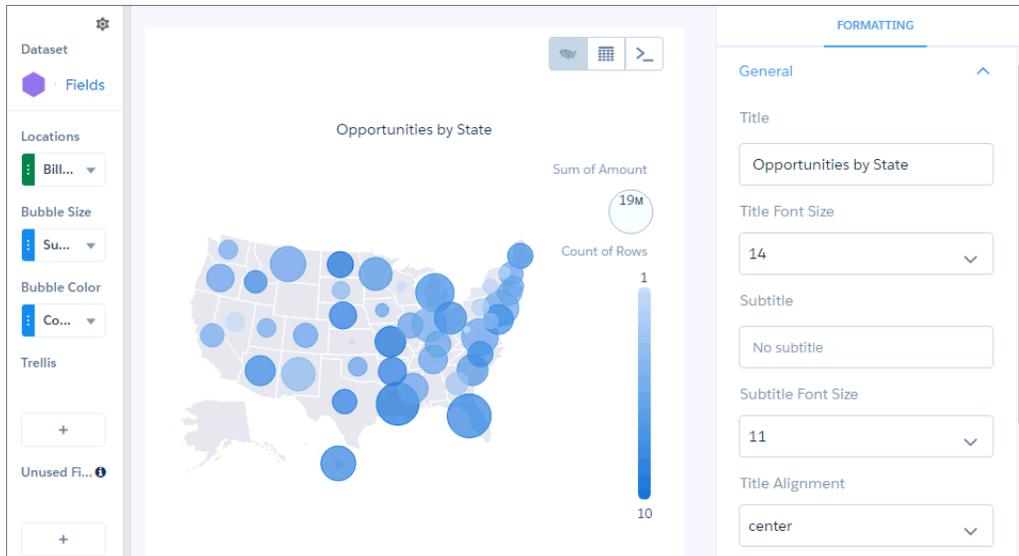
Geo Map Charts

Geo maps allow you to visualize data that contains geographical coordinates (latitude and longitude). Using coordinates allows you to place visual data elements more precisely on map charts.

Create a Geo Map Chart

A dataset with geographical location data, such as U.S. states, can be overlaid on a U.S. map. The bubbles representing a measure appear centered on the state areas. If the locations are tied to particular areas in those states, coordinates can be used to locate the bubbles more accurately.

For example, with airline data tied to particular airports, use airport latitude and longitude coordinates to visualize the data at each airport's actual location. The dataset could look like the following:

```
Airport,Year,Quarter,lat,lng,flights,passengers
JFK,2014,1,40.639751,-73.778925,120000,67276
SFO,2014,1,37.618972,-122.374889,30000,7383
ORD,2014,1,41.978603,-87.904842,50000,90908
HNL,2014,1,21.318681,-157.922428,10000,5262
ATL,2014,1,33.640411,-84.419853,217089,24044724
DFW,2014,1,32.89748,-97.040443,169955,15888600
DEN,2014,1,39.849312,-104.673828,141381,13368128
SEA,2014,1,47.443546,-122.301659,84283,10585134
SLC,2014,1,40.758701,-111.876183,81238,5285402
PHX,2014,1,33.457439,-111.727386,107615,10533665
JFK,2014,2,40.639751,-73.778925,170000,77276
SFO,2014,2,37.618972,-122.374889,40000,8383
ORD,2014,2,41.978603,-87.904842,50000,100908
HNL,2014,2,21.318681,-157.922428,11000,6262
ATL,2014,2,33.640411,-84.419853,227089,34044724
DFW,2014,2,32.89748,-97.040443,170955,25888600
DEN,2014,2,39.849312,-104.673828,151381,23368128
SEA,2014,2,47.443546,-122.301659,85283,20585134
```

```
SLC,2014,2,40.758701,-111.876183,82238,6285402
PHX,2014,2,33.457439,-111.727386,117615,11533665
JFK,2014,3,40.639751,-73.778925,210000,87276
SFO,2014,3,37.618972,-122.374889,50000,9383
ORD,2014,3,41.978603,-87.904842,50000,90908
HNL,2014,3,21.318681,-157.922428,12000,7262
ATL,2014,3,33.640411,-84.419853,237089,35044724
DFW,2014,3,32.89748,-97.040443,180955,26888600
DEN,2014,3,39.849312,-104.673828,161381,24368128
SEA,2014,3,47.443546,-122.301659,95283,22585134
SLC,2014,3,40.758701,-111.876183,92238,6585402
PHX,2014,3,33.457439,-111.727386,127615,13533665
JFK,2014,4,40.639751,-73.778925,220000,117276
SFO,2014,4,37.618972,-122.374889,60000,7000
ORD,2014,4,41.978603,-87.904842,50000,90908
HNL,2014,4,21.318681,-157.922428,13000,8262
ATL,2014,4,33.640411,-84.419853,247089,36044724
DFW,2014,4,32.89748,-97.040443,190955,27888600
DEN,2014,4,39.849312,-104.673828,171381,25368128
. . .
```

1. In the explorer, click 📊 and then select the **Geo Map** chart type.

2. To select the map type in the chart properties, click 🖌️ .
   The map type depends on the region that you want to show in the map. For example, if your flights are only in the Unitied States,
   select **US States**.
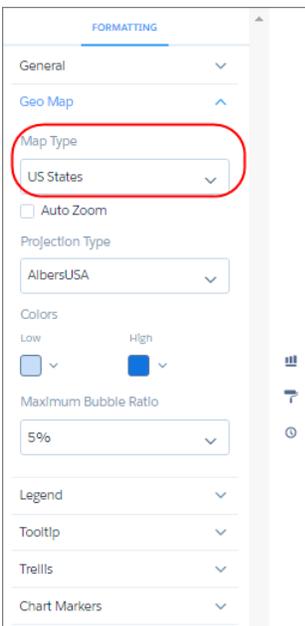
3. In the Locations field, select the location field that determines where to add bubbles. For example, select the Airport field.

4. In the Latitude and Longitude field, add two measures: the first contains average latitude coordinates and the second contains
   average longitude coordinates.

Tip: If you're adding measures in a SAQL query, add the latitude and longitude measures first, and then any other measures.

5. In the Bubble Size field, add a measure that determines the bubble size, such as sum of flights.

6. In the Bubble Color field, add a measure that determines the color, such as sum of passengers.
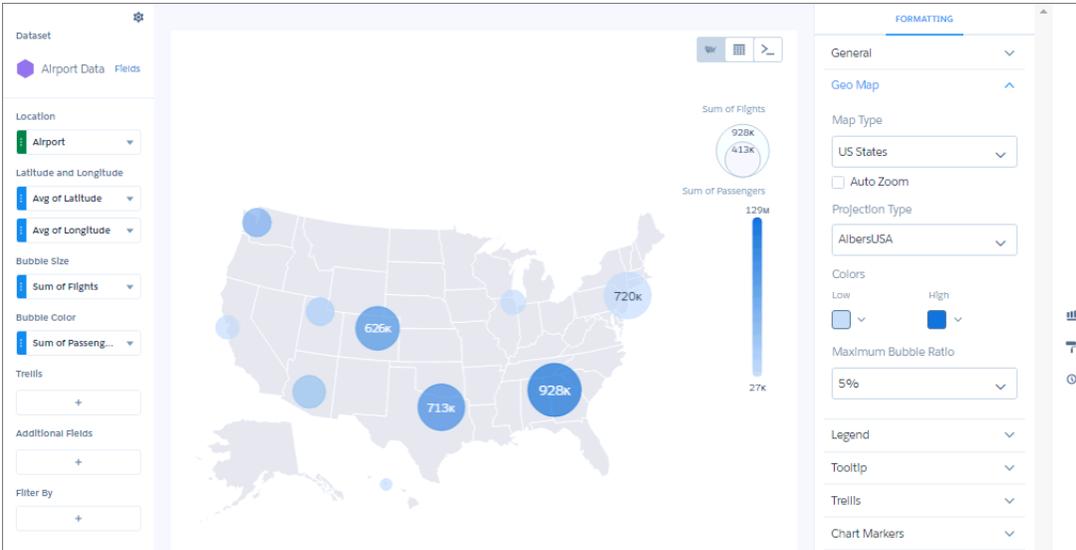
7. To make additional changes, such as setting the colors of the geo map chart bubbles, set the chart properties in the Formatting panel.

Custom Map Charts

Use a custom map if you have data with a geographical component that doesn't match any of the standard maps provided by Tableau CRM charts, such as custom regions. You can also use custom maps for areas that can be represented with polygons, such as stadium seat sections, city blocks, or floor plans.

Custom map charts can be used in lenses and dashboards. You can create and manage custom maps in explorer and dashboard designer.

Custom map charts in Tableau CRM can be created from GeoJSON with a geometry type of Polygon or MultiPolygon. See the reference section for more information. Data is overlaid on a custom map using either an id or name property as a key.

Create a Custom Map

Tableau CRM comes with prepackaged maps that you can use in map, bubble map, and geo map charts. If you need a map that isn't available, upload the geoJSON file to the user interface to create it. Set the boundaries to focus on a specific region.

Adding, editing, and deleting custom maps, and uploading geoJSONs requires the Manage Analytics Custom Maps user permission. Users with this permission can edit and delete any custom map, including maps created by other users. All users in the org, even those without this permission, can use all maps in their lenses and dashboards.

1. Select the map widget, and click ➕ next to the Map Type widget property.
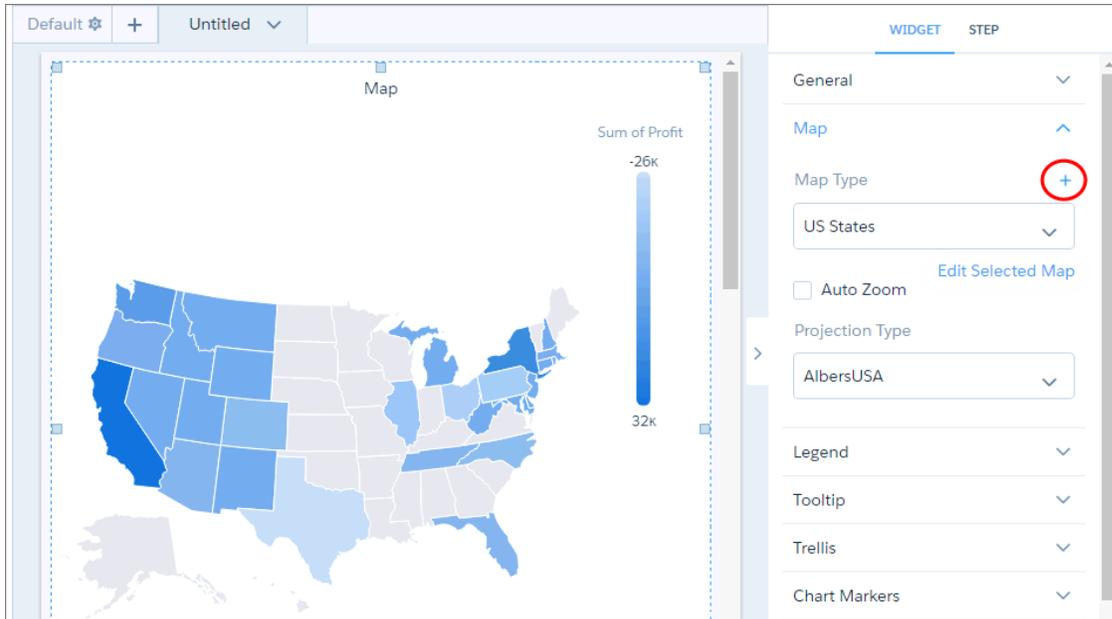
2. In the left pane, choose an existing geoJSON or upload a new one (max 10 MB). Each map is based on an underlying geoJSON.

Tip: You can create multiple maps from the same geoJSON by applying unique boundaries that focus on different regions. To edit or delete an existing geoJSON, hover over the geoJSON and click the appropriate button. You can delete only unused geoJSONs, those that aren't used to define maps. To delete a used geoJSON, first remove it from the maps in which it's used.
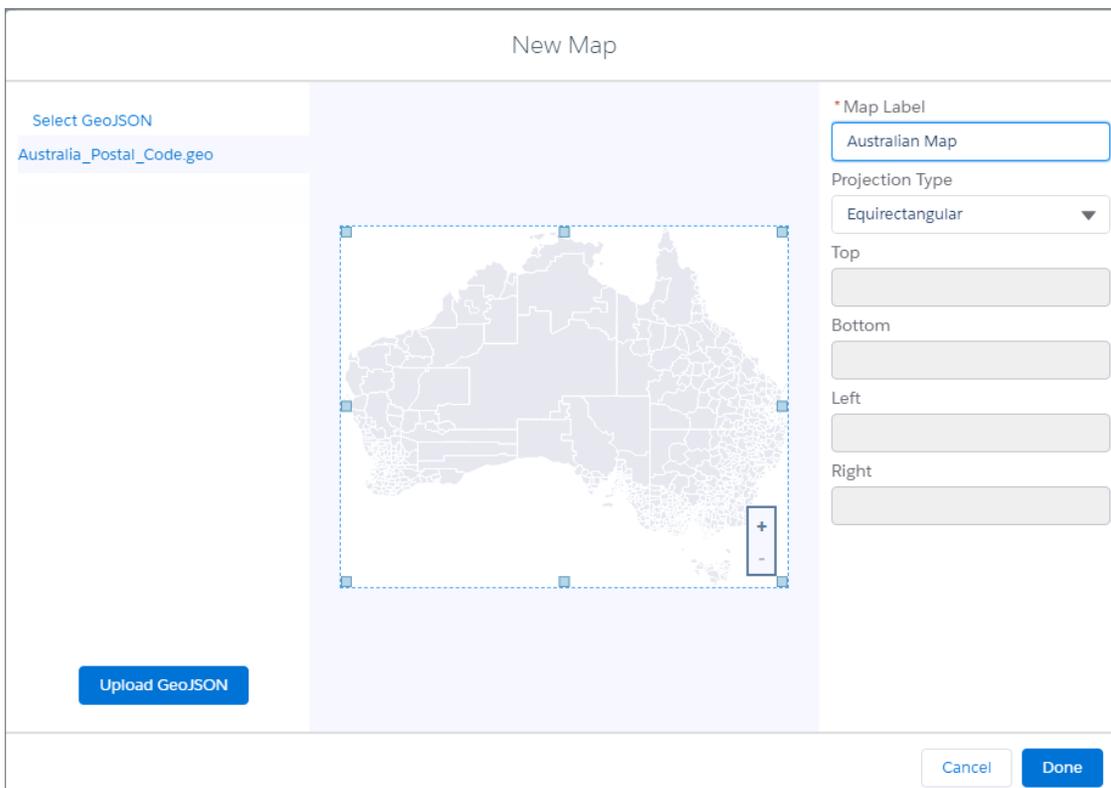
3. Specify the label and projection type of the new map.

   Mercator is most suitable for traditional geographical maps. Use AlbersUSA for a map of the United States of America that includes Hawaii and Alaska near the rest of the United States. Use Equirectangular for simple geometric shapes such as floor plans, city blocks, or zip-code areas.

   📝 Note: When you use the map in a chart, you can override this projection type in the widget properties.

4. In the center pane, drag the handles of the map to change the boundaries, and zoom in on a particular region. The boundaries are listed in the right pane.
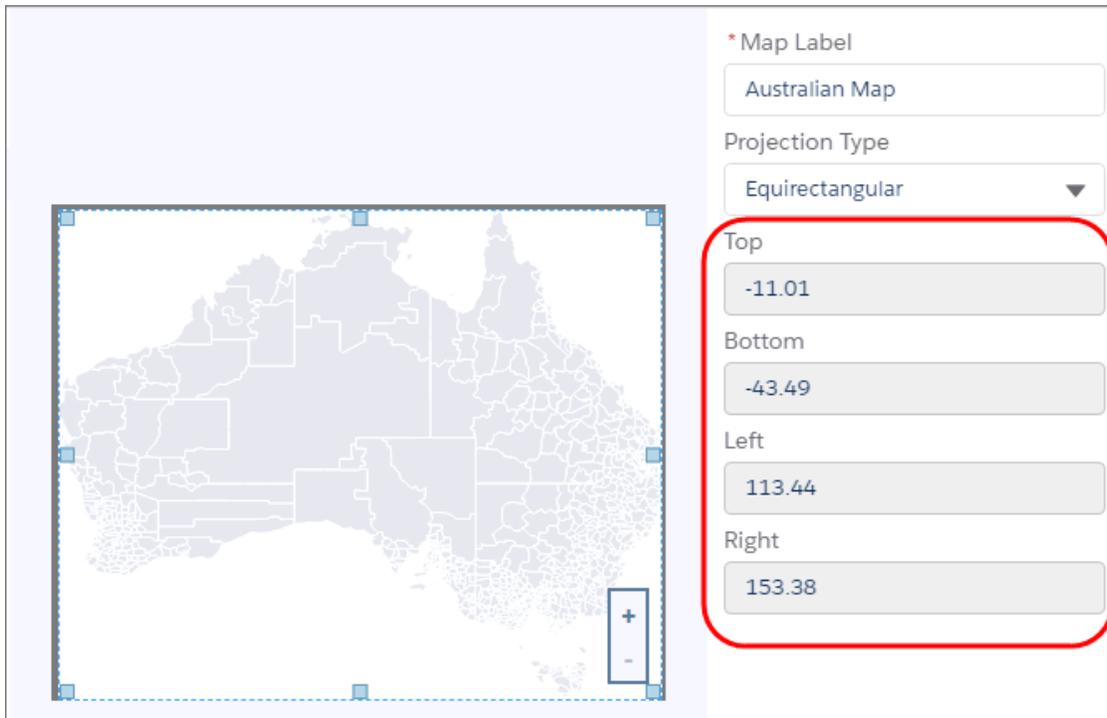
Edit and Delete a Custom Map

Edit a map to change its boundaries.

1. In the Map Type property, select the map type and then click **Edit Selected Map**.
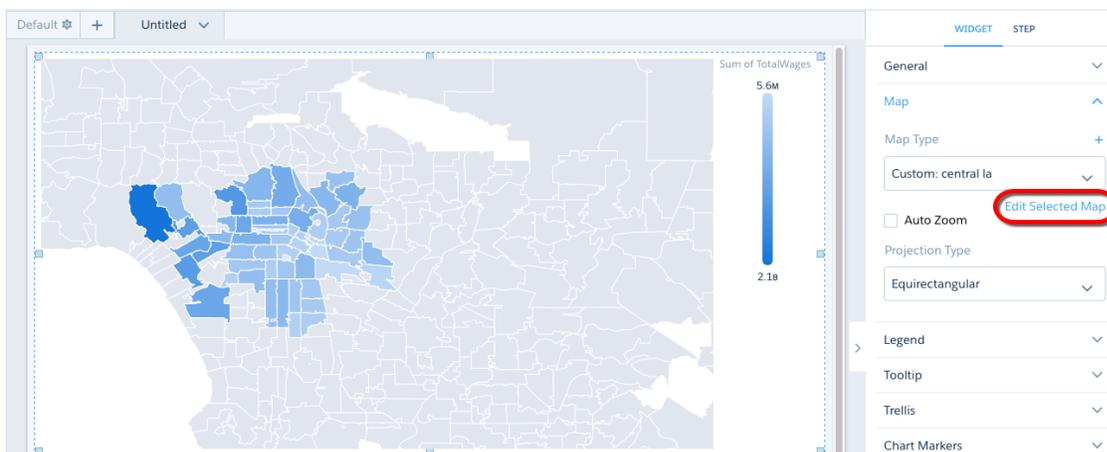
2. Drag the handles of the map to change the bound box.

The new coordinates appear in the right pane.

Edit Map
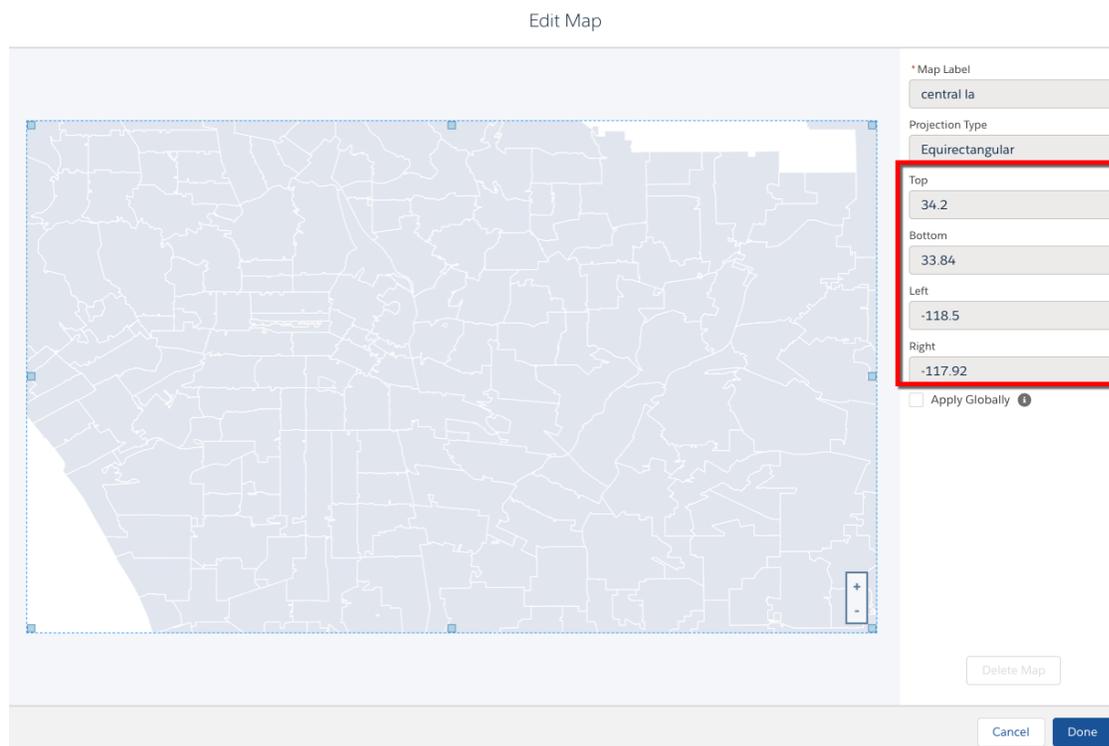
3. To apply the changes to all lenses and dashboards in your org that use this map, select **Apply Globally**. If left unchecked, the changes only apply to this instance of the map.

4. To delete the map, click **Delete Map**.

   You can't delete maps that are prepackaged with Tableau CRM.

5. Click **Done**.

Properties of Custom Maps

Proper formatting of the underlying GeoJSON is critical for successfully overlaying data onto a custom map. For improved presentation, it may be necessary to adjust the map's bounding-box coordinates to zoom in on the overlaid areas.

GeoJSON Structure

Standard GeoJSON typically has a structure that contains feature blocks similar to the following:

```
{
  "type": "Feature",
  "geometry": {
    "type": "MultiPolygon",
    "coordinates": [
      [125.6, 10.1],
      [126.1, 11.0], ...
    ]
```

```
        },
        "id": "SI",
        "properties": {
          "name": "Sandwich Islands",
          "property1": "value1"
        }
      }
```

Each feature block defines a shape, or map area. To correctly overlay data onto the map, the data must have a column with values uniquely matching one of the properties in the GeoJSON. For example, if the data has values in a column called Country_Code, then each feature block in the GeoJSON should have a property called "id" with values that exactly match values in Country_Code. The feature block with `"id": "SI"` would match a row in the data that has "SI" in the Country_Code column, and that data can then be overlaid on the area defined by that feature block. To overlay the data on the matching areas, you'll have to group on Country_Code when exploring the dataset in Tableau CRM.

The "id" property is required, and it has to be at the same level as "properties" in the GeoJSON. Often, however, it's a child of "properties":

```
{
   "type": "Feature",
   "geometry": {
      "type": "MultiPolygon",
      "coordinates": [
         [125.6, 10.1],
         [126.1, 11.0], ...
      ]
   },
   "properties": {
      "name": "Sandwich Islands",
      "id": "SI",
      "property1": "value1"
   }
}
```

Sometimes "id" doesn't exist at all, and must be created. In either case, the original GeoJSON must be edited to contain the "id" property at the level of the "properties" property. In a single feature block, doing that is trivial, but a large GeoJSON file could contain hundreds of GeoJSON feature blocks. Using a script is recommended.

📝 Note: For GeoJSON that lacks an "id" property, Tableau CRM falls back to using the values in properties.name. If your GeoJSON has a property called "name" under "properties", and "name" has values that match a column in your data, it can be used instead of "id". Do not change the position of name. It must remain a child of "properties".

Bounding Boxes

You can "zoom" into a map area by redefining the area of the map to display. This is done by giving the boundaries of the new area in terms of coordinates. These coordinates are called a bounding box. A bounding box limits the map to the shapes within the coordinates, or borders, of the bounding box.

When you create a map from a GeoJSON without a bounding box, every defined shape is displayed as an area on the map. For example, the following map of zip codes in the Los Angeles area does not use a bounding box:
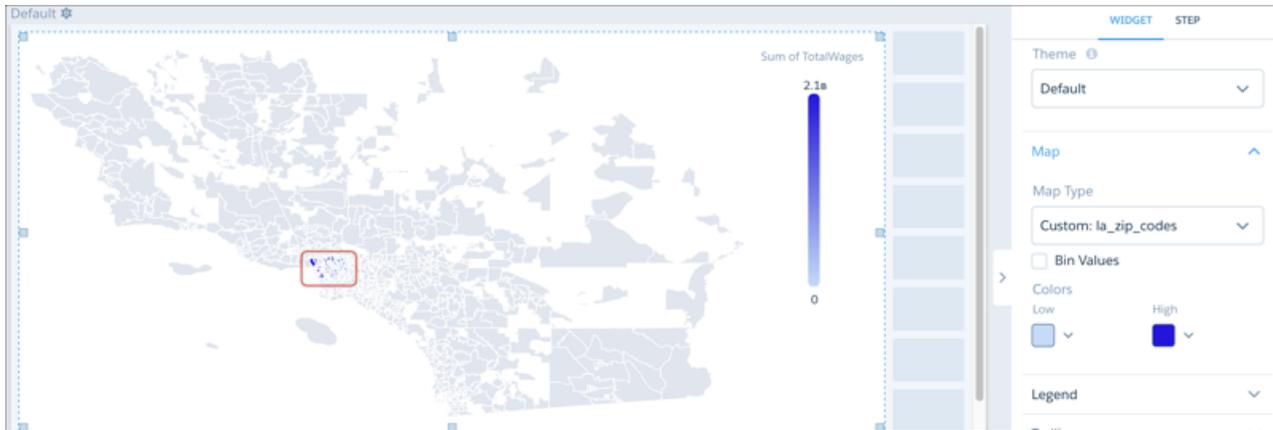
The areas of interest, where data has been overlaid on the map, are indicated by the red rectangle. To make the map more readable, the view should zoom in on the zip-code areas that have data. The coordinates (in this case, latitude and longitude) bordering the area of interest can be found using a GeoJSON tool that gives the coordinates of the areas in the map. A second map is then created from the original GeoJSON, this time with bounding-box coordinates.

When viewed in Tableau CRM, this new map is zoomed in on the area of interest.
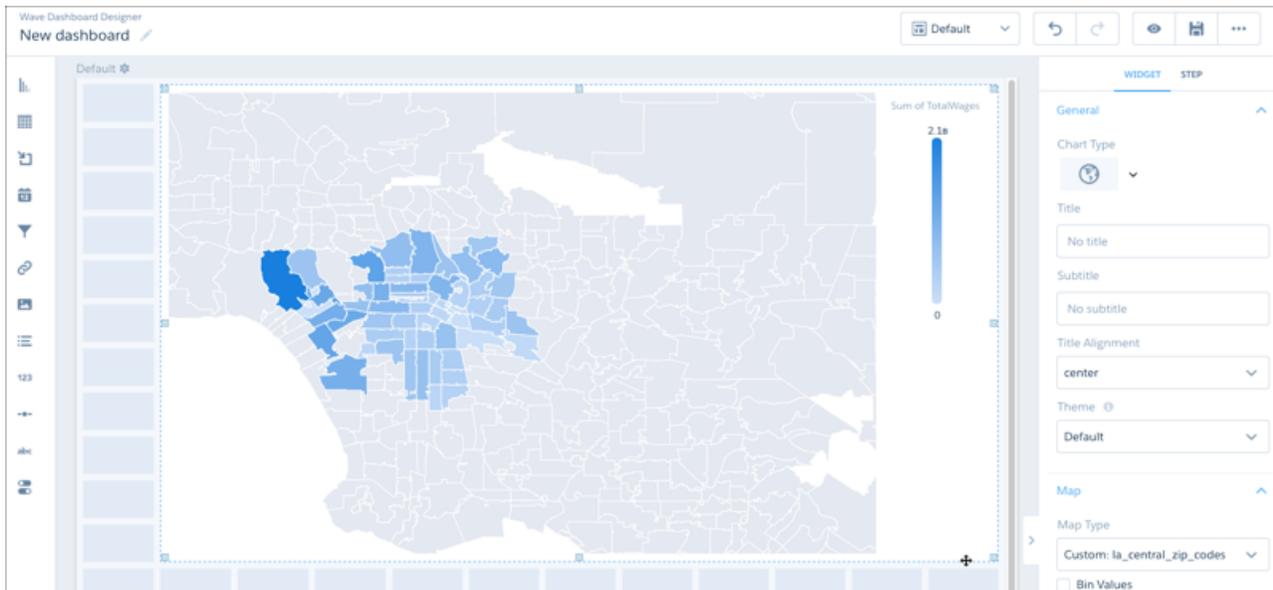
You can set the bounding box while creating or editing a map.

Matrix Charts

Use matrix charts to create a table that represents measures visually, allowing you to quickly spot extreme values.

For example, you can show which product families are generating the most opportunities.
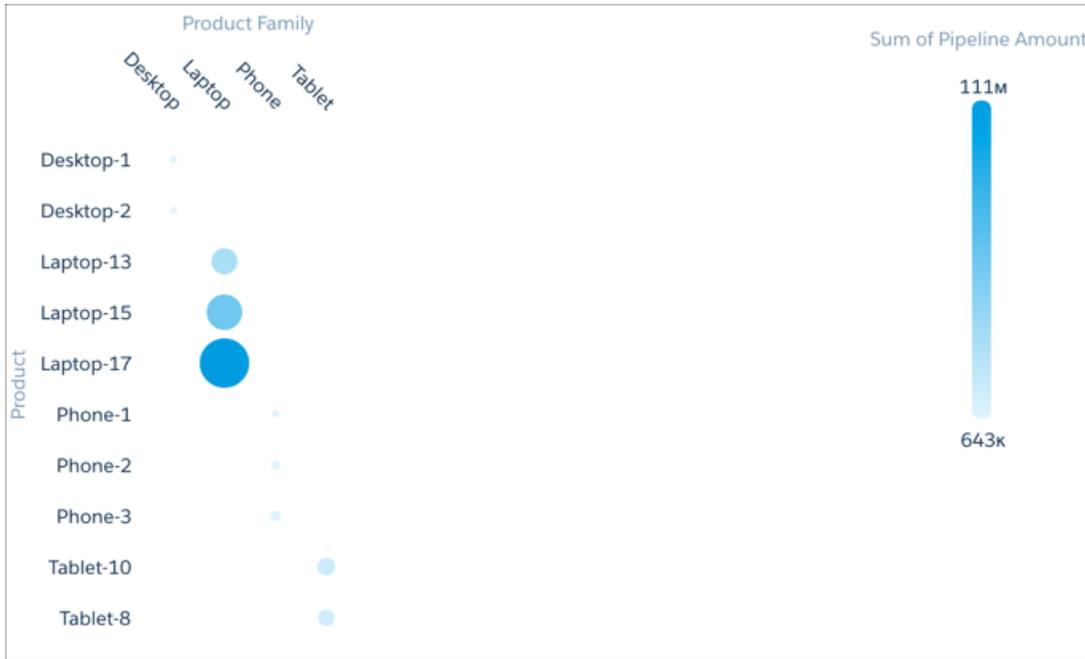
Create a Matrix Chart

Use a matrix chart to compare one or two measures across two dimensions.

For example, use a matrix chart to compare the number of opportunities by sales rep and pipeline stage.

1. In the explorer, click ![icon] and then select the **Matrix** chart type.

2. In the X-Axis field, add a dimension to analyze the measure by. For example, select **Stage**.

3. In the Y-Axis field, add another dimension. For example, select **Opportunity Owner**.

4. In the Bubble Size field, add the measure that you want to analyze across both dimensions. For example, select average opportunity amount.

5. In the Bubble Color field, add another measure that you want to analyze across both dimensions. For example, select **Count of Rows** to look at the number of opportunities.
   By default, the matrix shows a color gradient based on two colors.

6. To change the chart display, click ![icon] and set the chart properties in the Formatting panel. For example, change the two colors under the Conditional Formatting section.

Origami Charts

Use an origami chart to create a striking horizontal visualization for easily identifying high- and low-value data when there's a single measure and a single grouping.
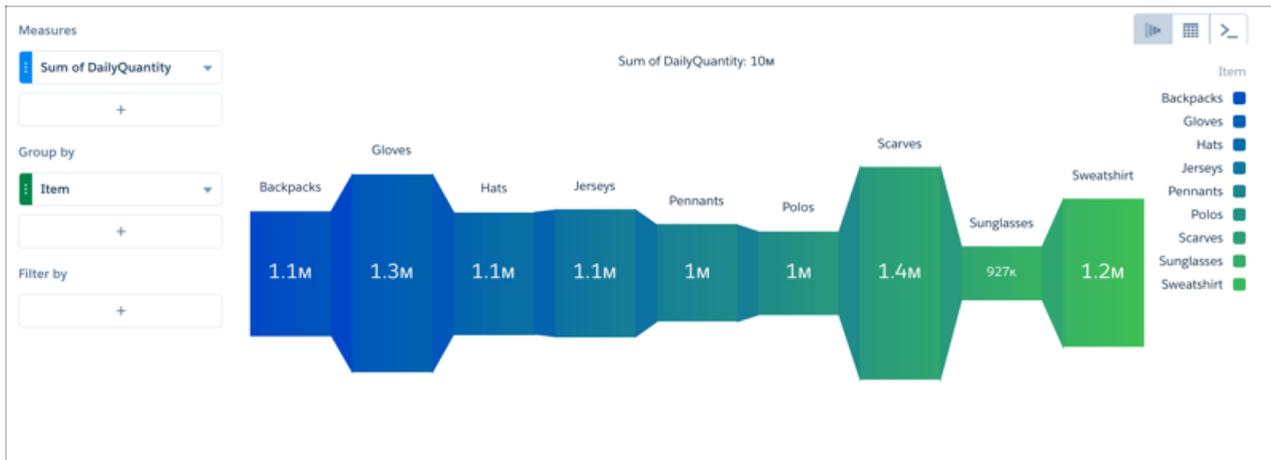
Parallel Coordinates Charts

Use a parallel coordinates chart when you have multiple measures and a single grouping. Parallel coordinate charts are useful for displaying how data elements in a grouping stack up against each other.

Create a Parallel Coordinates Chart

If you're trying to spot outliers or patterns based on related metric factors, parallel coordinates charts can help tease them out.

For example, you have product pricing data but you aren't sure why some higher-volume products are more profitable than others.

1. In the explorer, click ▥ and then select the **Parallel Coordinates** chart type.

2. In the Node Measures field, add the measures you're interested in.

   To compare product pricing, add average unit cost, discount, shipping cost, and profit.

3. In the Lines field, add the dimension that you'd like to group the data by. Each group gets its own line. For example, select **Product Name**.

   The parallel coordinates chart shows that higher discounts aren't adding up to higher profits.
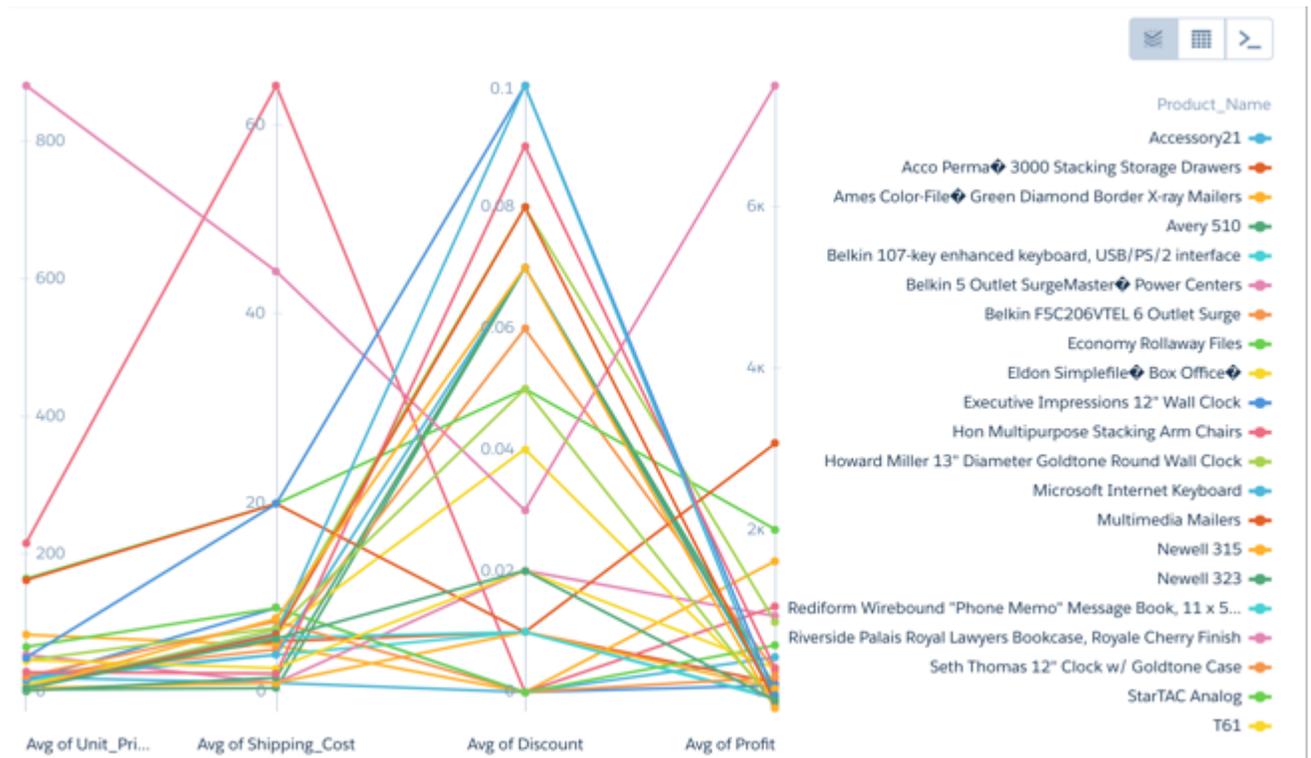
4. To change the chart display, click 🖌 and set the chart properties in the Formatting panel. For example, indicate whether to show every measure title in the chart.

Pyramid Charts

Use a pyramid chart to visually highlight relative sizes of stacked dimension values.

Create a Pyramid Chart

A pyramid chart is a good choice for presenting related measures for one particular dimension, such as number of accounts and account values in a particular category.

For example, you're interested in looking at the overall value and number of accounts in countries where your company does business. A bar chart can show that data in order, but it's not necessarily a compelling or easy-to-read visualization. Changing it to a pyramid chart improves both overall appearance and readability.
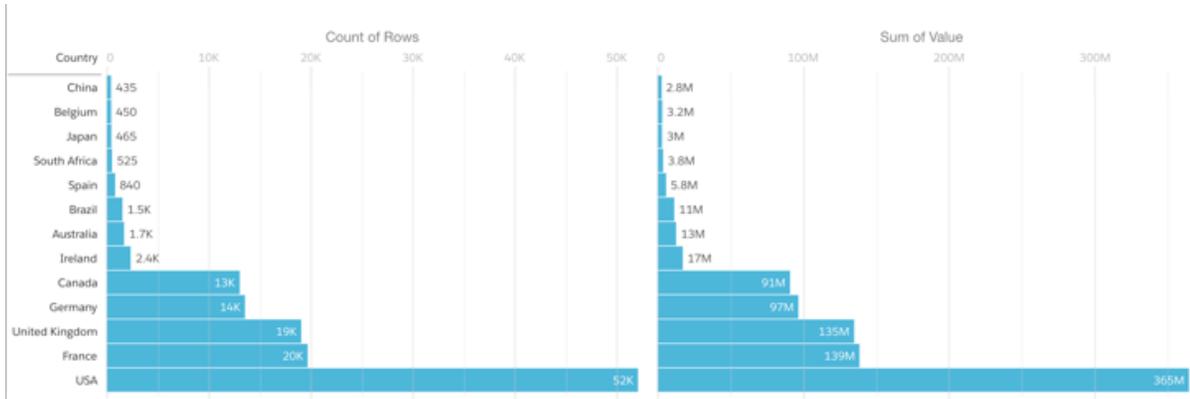
1. In the explorer, click ▦ and then select the **Pyramid** chart type.

2. In the Left Axis field, add the measure to include on the left side of the pyramid. For example, select **Count of Rows**.

3. In the Right Axis field, add the second measure. For example, select **Sum of Amount**.

4. In the Bars field, add a dimension to analyze the measures by, like Billing Country.

5. To sort the results by a measure, select the down arrow next to the measure and select the sort order.
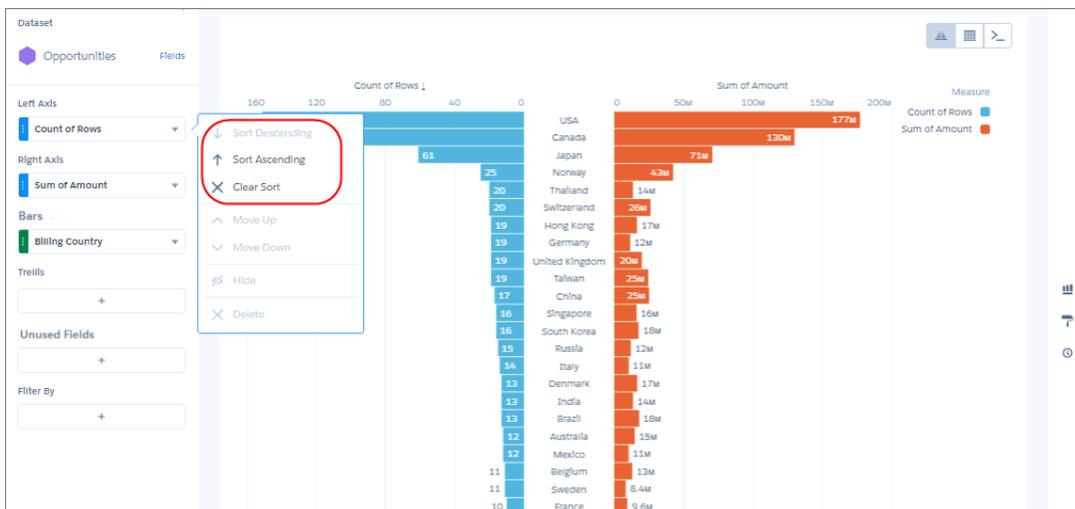
6. To change the chart display, click 🖌 and set the chart properties in the Formatting panel.

   For example, specify whether to show measure axes and titles.
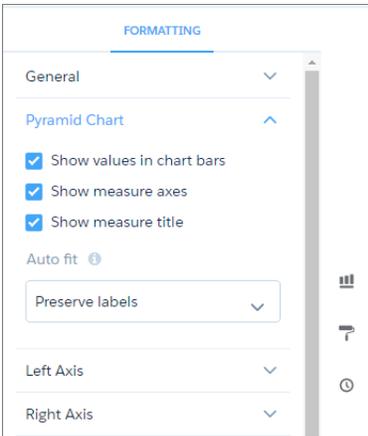
Radar Charts

Use radar charts to display a small dataset with one dimension and at least three measure columns.

Create a Radar Chart

A radar chart is useful when you're trying to visually represent a dimension along three or more measure axes that have different scales.

For example, you want to visualize the key metrics for all sales reps (opportunity owners) to determine who has superior performance.

1. In the explorer, click ![icon] and then select the **Metrics Radar** chart type.

2. In the Axes field, add at least three measures. For example, add average opportunity amount, total opportunity amount, and count of rows.

3. In the Polygons field, add the dimension to analyze the measures by. For example, select **Account Type**.

4. To change the chart display, click ![icon] and set the chart properties in the Formatting panel.

   For example, show all measures in the tooltip when you hover over a data point.
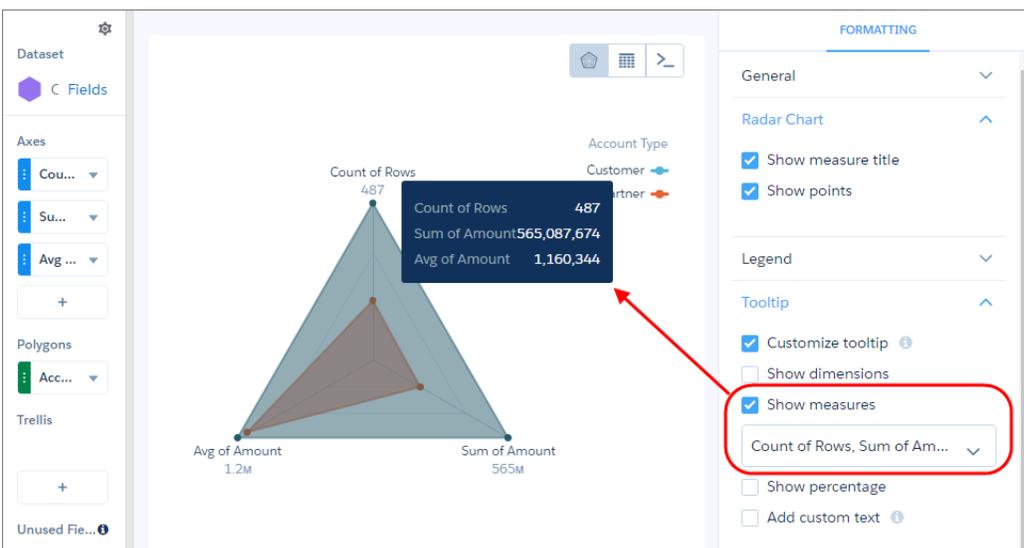
5. To view a specific grouping, click it in the legend.
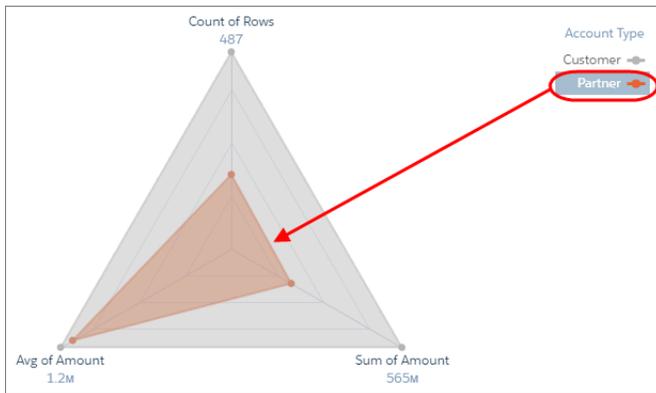
💡 **Tip:** Use a metrics radar chart instead of a stacked bar chart when the measures are similar across groupings. Although stacked bar charts can analyze multiple measures, they don't provide enough contrast when comparing values with subtle differences. For example, it's difficult to determine which customer segment has the highest average discount in this stacked bar chart.
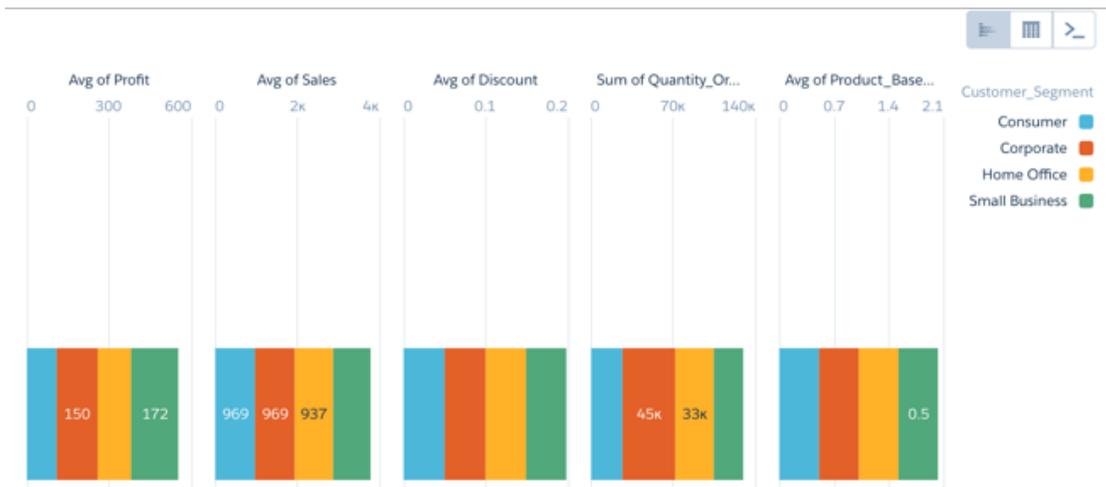
Rating Charts

Use a rating chart to get a sense of how well a measured quality, such as customer satisfaction, is meeting expectations.

Create a Rating Chart

If you've set a goal for, as an example, customer satisfaction in each of the countries where your company operates, rating charts are a good choice for showing how close you are to that goal.

You've collected, averaged, and loaded the data from the surveys. Now you want to present it in a way that makes sense for customer satisfaction scores. Each rating chart can display 1 value per country, and highlight value ranges.

1. In the explorer, click 📊 and then select the **Rating** chart type.

2. In the Value field, add the measure that you want to analyze, like average opportunity value.

3. To compare the measure across each member of a category, like opportunity owner, add the dimension in the Trellis field.
A separate rating chart appears for each value of the selected dimension.

4. To change the order of multiple rating charts, sort the measure in ascending or descending order. Default sort order is alphabetical order.
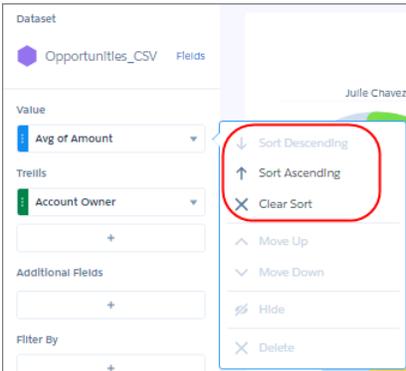
5. To change the chart display, click 🖌 and set the chart properties in the Formatting panel.

For example, show the rating as a series of stars.
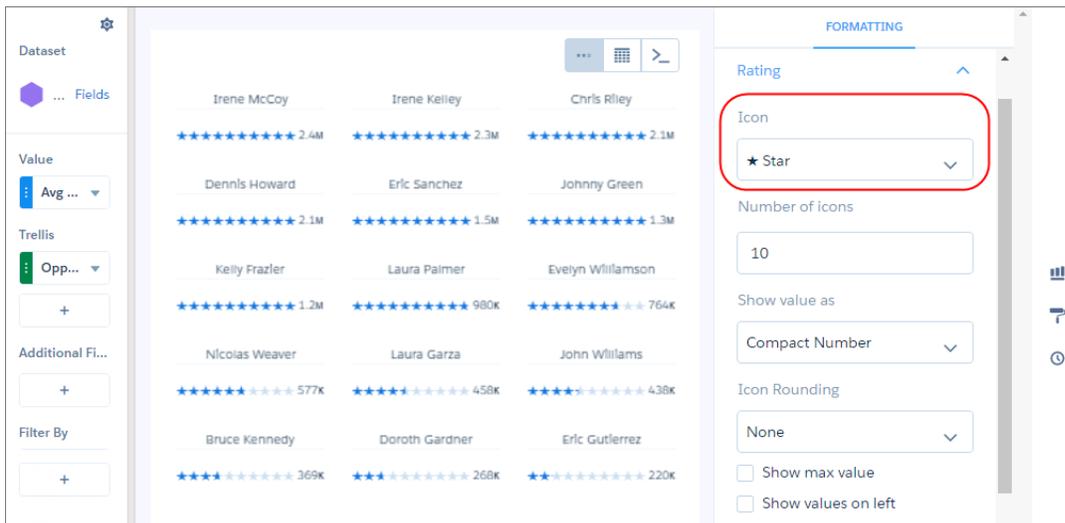
Sankey Charts

Use a sankey chart to visualize the distribution of a measure from one grouping to another grouping.

The sankey chart performs calculations based on the absolute value of each measure. For example, you have the following data:

#	Type	Name	Amount
1	New Business	SalesRep1	200
2	New Business	SalesRep2	200
3	New Business	SalesRep3	-100
4	Add-On	SalesRep2	-50
5	Add-On	SalesRep3	50
6	Add-On	SalesRep4	75

Based on the data in the table, total new business is 300 (200 + 200 + -100) and total add-on business is 75 (-50 + 50 + 75). When you visualize this data in a sankey chart, notice the different results: new business is 500 and add-on business is 175. These values (3) also impact the total (1) and difference between the two groupings (2).
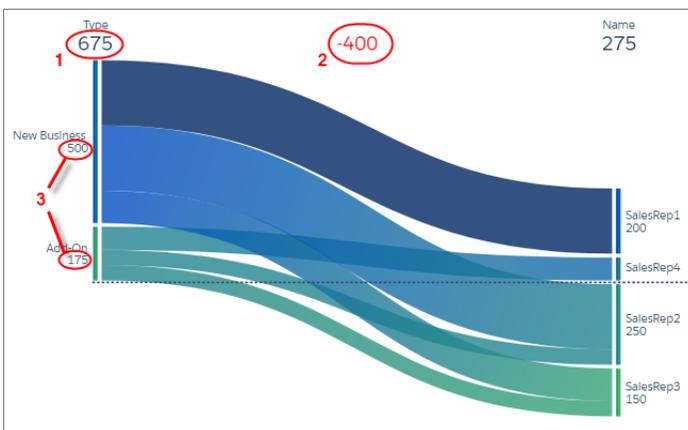
> **Note:** Sankey charts don't support scroll and don't render when there's not enough space. For high cardinality situations, consider alternative visualizations.

Scatter Charts

Use a scatter chart to visualize correlation between two groups of data.

Create a Scatter Plot Chart

Use a scatter chart to see the correlation between two measures across a single dimension.

For example, use this chart to analyze the correlation between total opportunity amount and number of opportunities for product families.

1. In the explorer, click ▥ and then select the **Scatter Plot** chart type.

2. In the X-Axis field, add the first measure. For example, select the count of rows.

3. In the Y-Axis field, add the second measure, such as sum of amount.

4. In the Bubble Size field, add a third measure that determines the bubble size. For instance, add average amount.

5. In the Bubble field, add the dimension that determines the bubbles that appear in the chart. For example, select **Product Family** to create a bubble for each family.

6. To change the chart display, click 🖌 and set the chart properties in the Formatting panel.
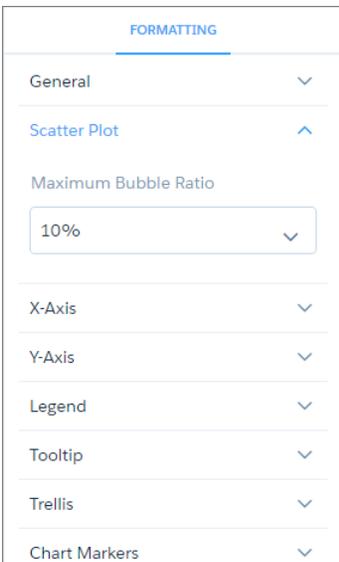
Timeline Charts

A timeline chart is a line chart with one axis dedicated to a time dimension. Use a timeline chart to show how a value changes over time.

Create a Timeline Chart

Use a timeline chart to analyze changes to a measure over time using a line.

For example, a timeline chart can show how profit changes over a year. Exploring a year's worth of data, you want to visualize not just the total profit, but how profits changes from month to month.

1. In the explorer, click ![icon] and then select the **Timeline** chart type.

2. In Time Axis, select the date field and the granularity that you want to analyze the results by. For example, select **Closed Date** > **Year-Quarter**.

3. In Y-Axis, add at least one measure to analyze over time, such as the total opportunity amount.

4. If needed, add a filter to restrict the results to a particular time period, like a single year.
The timeline chart analyzes the measures with the time dimension shown along the x-axis.

5. To change the chart display, click and set the chart properties in the Formatting panel.

   For example, if some of the dataset includes predictions of future amounts, click **Predictive Line** under the X-Axis section in the charts properties panel. Or add a reference line to provide a visual cue about a particular point in time.

6. To add a reference line, set the position of the reference line by specifying the date in **Time**.

   The numerical date you specify should match the granularity selected in the Time Axis field. For example, if you selected Year-Month and want the reference line to start just after August 2016, enter *2016/8*. You can also specify a label in **Label** for the line, and a color.

Time Bar Charts

A time bar chart is a column chart with the horizontal axis showing a time dimension. Use a time bar chart to visualize changes over time with vertical bars, which is a great way to illustrate relative changes over time and highlight missing data.

Create a Time Bar Chart

Use a time bar to analyze a measure over time using vertical bars.

For example, a time bar chart can show how total opportunity closes over time, where the lack of bars highlights no sales.

1. In the explorer, click 📊 and then select the **Time Bar** chart type.

2. In Time Axis, select the date field and the granularity that you want to analyze the results by.

3. In Y-Axis, choose the measure to analyze over time, such as the total opportunity amount.
The time bar chart appears as a column chart, with the time dimension shown along the X-axis.

4. To change the chart display, click 🖌 and set the chart properties in the Formatting panel.

For example, if some of the dataset includes predictions of future amounts, click **Predictive Line** under the X-Axis section in the charts properties panel. Or add a reference line to provide a visual cue about a particular point in time.

5. To add a reference line, set the position of the reference line by specifying the date in **Time**.

    The numerical date you specify should match the granularity selected in the Time Axis field. For example, if you selected Year-Month and want the reference line to start just after August 2016, enter *2016/8*. You can also specify a label in **Label** for the line, and a color.

Time Combo Charts

A time combo chart shows two or more measures over time. You can display each measure as a line or bar. You can display the charts on the same or separate axes.

If the time combo chart shows multiple bars, you can stack them to show and compare parts of a whole. For example, add a chart that analyzes opportunities over time. Then stack the bars to compare how much you lost and won over each time interval.

Create a Time Combo Chart

Use a time combo chart analyze two or more measures over time.

For example, this chart can show how the total opportunity amount and the number of opportunities change over time.

1. In the explorer, click 📊 and then select the **Time Combo** chart type.

2. In Time Axis, select the date field and the granularity that you want to analyze the results by.

3. In Y-Axis, choose at least two measures to show, such as the total opportunity amount and count of rows.
    The first measure displays on the time bar graph and each subsequent measure displays on a timeline graph.

4. To change the chart display, click and set the chart properties in the Formatting panel.

For example, if some of the dataset includes predictions of future amounts, click **Predictive Line** under the X-Axis section in the charts properties panel. Or add a reference line to provide a visual cue about a particular point in time.

5. To add a reference line, set the position of the reference line by specifying the date in **Time**.

The numerical date you specify should match the granularity selected in the Time Axis field. For example, if you selected Year-Month and want the reference line to start just after August 2016, enter *2016/8*. You can also specify a label in **Label** for the line, and a color.

Treemap Charts

Use treemaps to visualize hierarchical quantitative data, where containing rectangles designate relationship using color, and "leaf" rectangles represent quantity using area.

Create a Treemap Chart

Treemap charts are especially useful for visualizing groupings that are difficult to organize well using other types of charts.

For example, you can show the number of opportunities by state or province in each country in a way that reveals top markets.

1. In the explorer, click ![icon] and then select the **Treemap** chart type.

2. In the Segment Size field, add a measure, such as total opportunity amount.

3. In the Segments field, add one or more dimensions to analyze the measure by. For example, select **Billing Country** and **Billing State/Province**.

   In the treemap chart, each color represents a country, as indicated in the legend. Each rectangle represents a state or province in a country. The high-value states and provinces are easy to spot. Using tooltips, you can view details for any rectangle representing a state.
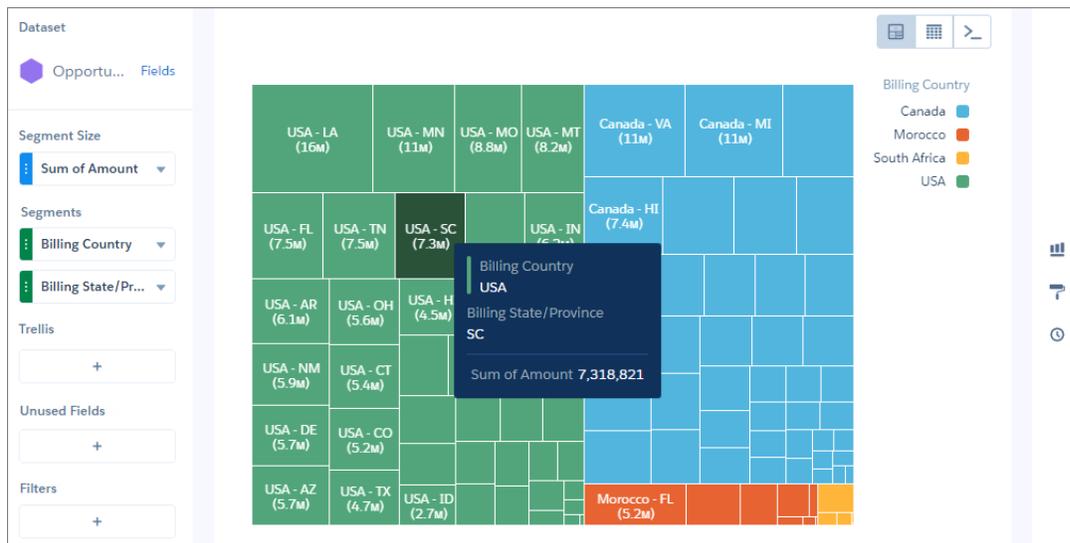
4. To change the chart display, click ![icon] and set the chart properties in the Formatting panel.
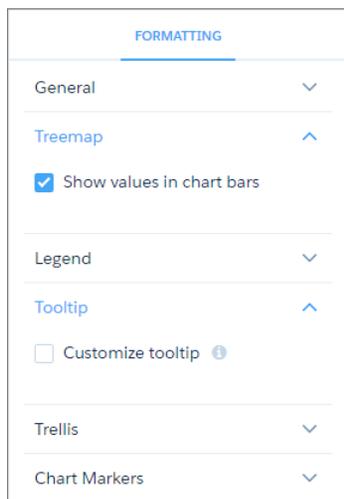
Waterfall Charts

Use a waterfall chart to show the cumulative effect of sequentially introduced positive or negative values with breakdowns of value totals. Also known as "flying bricks" or "Mario" charts. To include breakdowns of value totals, use a stacked waterfall chart.

Create a Waterfall Chart

Use a waterfall chart to analyze the smaller segments that comprise a total value.

For example, use this chart to determine the relative contribution each stage adds to the total open opportunity amount,

1. In the explorer, click 📊 and then select the **Waterfall** chart type.

2. In the X-Axis field, add the dimension to analyze the measures by. For example, select **Stage**.

3. In the Y-Axis field, add the measure, such as the sum of amount.
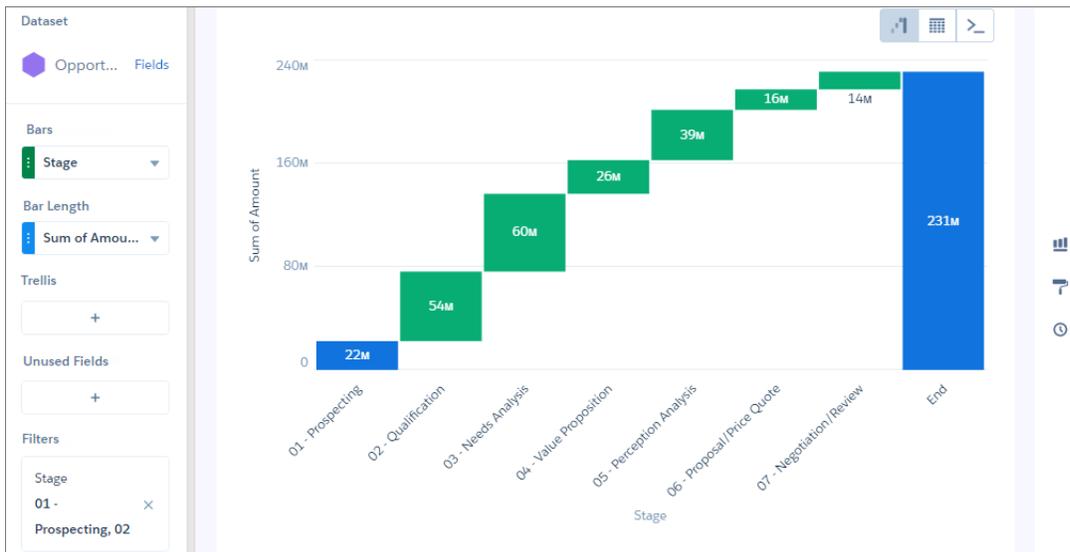
4. To change the chart display, click 🖌 and set the chart properties in the Formatting panel.

   For example, set the label for the total column.
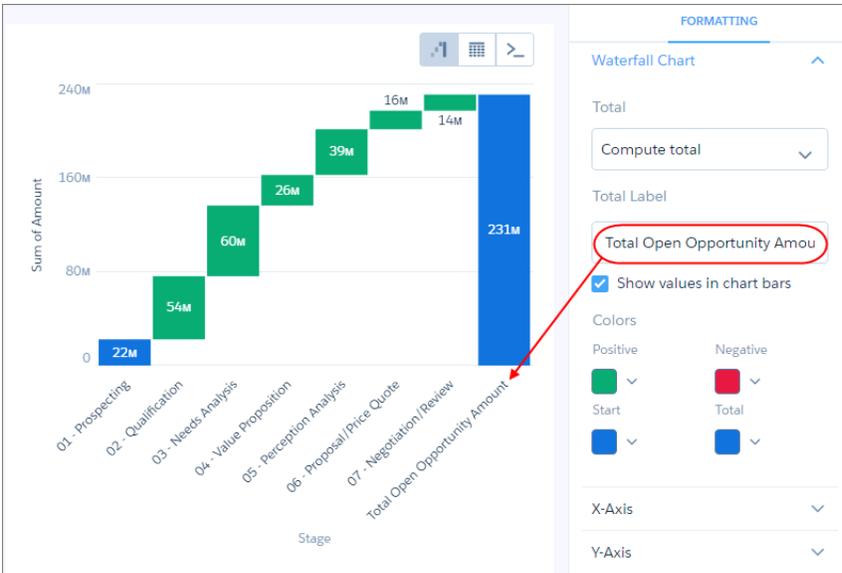
Clone a Lens

Build upon a visualization by cloning its containing lens to a new tab where you can continue exploring. You can save the original in its own tab.

1. Click an existing Tableau CRM lens to open it.

   If you're using a new, unsaved lens, you must save it before it can be cloned.

2. Select **Clone in a New Tab** from the More menu.
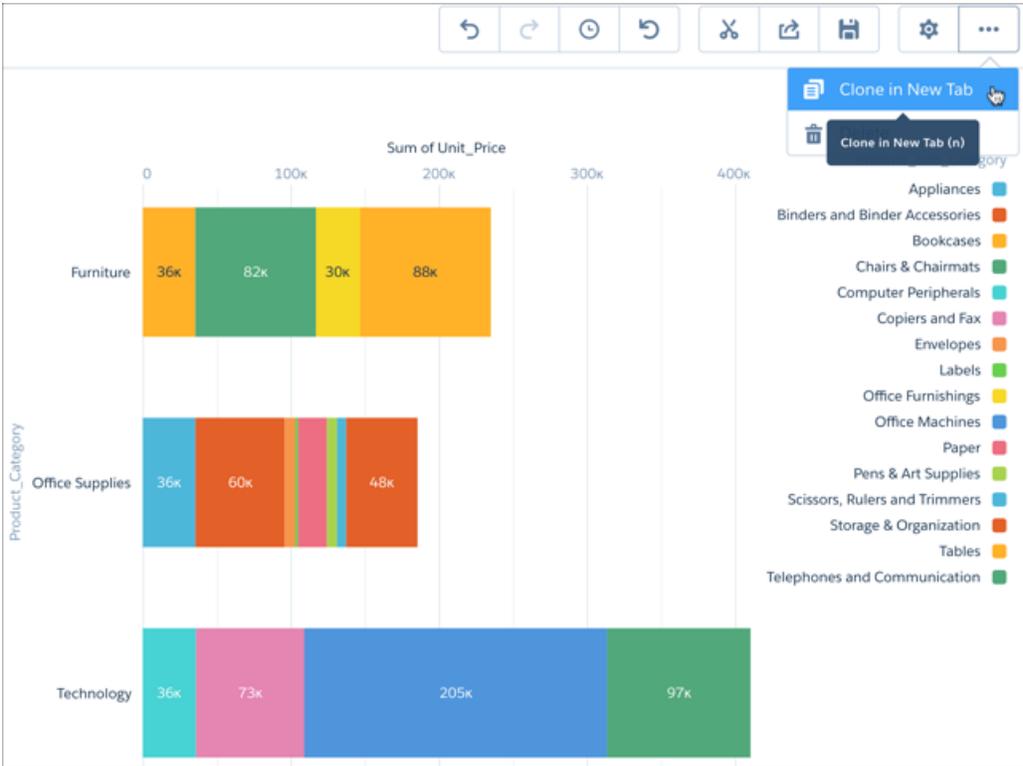
A new, unsaved copy of the cloned lens opens in a new tab.

Save a Visualization

Save your visualization as a lens.

1. Click the **Save** icon.

2. Enter a title and a description for your lens, and select the app to save it in.

3. Click **Save**.

Share a Visualization

Share a visualization with your colleagues by posting to Chatter, getting its unique URL, downloading a screenshot of it, or downloading its filtered data. A Chatter post provides an image and a link to the asset—lens, dashboard, or app—in Tableau CRM. Colleagues with the link and access to the asset can drill down and explore the information that's presented. To share without giving access to the asset, use the download options.

1. Click **Share**.

2. Click the tab for the sharing method that you want:

   a. **Post to Feed** posts to Chatter an image and the link to the lens, dashboard, or app. Select **User** or **Group** feed, enter the name, and then type your comment. You can also remove the image from your post by hovering over it and clicking the **x**. Note that posted images are public.

   b. **Get URL** provides a unique URL to the asset. Copy the link and then paste it wherever you want to share it.

   c. **Download** gives you options to share a visualization without giving access to it. You can download it as an image (`.png`) or as filtered data in Microsoft® Excel® (`.xls`) or comma-separated values (`.csv`) format.

3. If applicable, click **Give Access** to set the level of sharing access.

4. Click **Done**.

   📝 Note: To share a visualization, you must give access to the app that contains it. Lenses, datasets, and dashboards within the default Shared App are accessible to all Tableau CRM users, unless administrators have restricted access. All other apps are private unless someone with Manager access to the app has shared it with a specific user, group, or role.

Build Tableau CRM Dashboards

Build a Tableau CRM dashboard to continuously monitor key metrics of your business, analyzing the results by key dimensions, like region, products, and time period. Add interactive charts that synthesize information into an easy-to-read format. To complement the charts, add tables that show record-level details. Add filters to allow dashboard viewers to change the focus of the results. Create customized layouts to optimize the display of a dashboard on different types of devices, like mobile phones, tablets, and desktops.

📝 Note: Prior to the Winter '20 release, queries were called steps.

Watch a Demo: ▶ Build Interactive Tableau CRM Dashboards (English Only)

Get Hands-On Training with Trailhead: Tableau CRM Dashboard Building Basics

📝 Note: For you to create and edit dashboards with the dashboard designer, the admin must enable it in Tableau CRM setup—it's enabled by default—and must grant you the Create and Edit Tableau CRM Dashboards user permission.

This section discusses how to build dashboards with the dashboard designer. The dashboard designer is the newer designer that offers additional features to help you build dashboards more quickly and easily. Use a template to create a dashboard with a predefined design. Use the wizards to quickly build widgets. Use drag-and-drop, snap-to-grid, and browser reflow to effortlessly add, position, and align widgets in a dashboard.

1. Create a Tableau CRM dashboard with a Template

   Dashboard templates speed your analytics development by automatically creating dashboards. Some provide blank layouts that you populate with data, while other "smart" templates create dashboards that require little to no additional configuration.

2. Create and Manage Dashboards with Reusable Components

   Dashboard components are a type of dashboard widget that can contain other widgets and pages. Use dashboard components to manage and reuse groups of charts, tables, filters, text, and more in multiple dashboards.

3. Create Dashboard Pages

   Make the information on a dashboard easier to digest by chunking the content into multiple pages. And with fewer queries per page, dashboard performance increases. With pages, you can tell a story by creating a dynamic pathway through your dashboard. Depending on how you lay out your pages, you can also create some cool effects as you transition from one page to the next.

4. Add Widgets to the Dashboard

   Widgets are the basic building blocks of a dashboard. In the dashboard designer, you can add different widgets to perform functions. For example, widgets can calculate key performance indicators, filter dashboard results, visualize your data using interactive charts, and show record-level details in tables.

5. Manage Queries for Widgets

   Queries return results that are displayed in widgets. For example, a number widget displays the result of a calculation that is defined in a query. Queries can be built on a data source, like a dataset or a Salesforce object. They can also be "custom queries" created with user-defined values.

6. Make the Dashboard Widgets Interactive

   Tableau CRM dashboards have unique features that allow you to make the widgets interactive. For example, widgets in the dashboard can be filtered to show only results for the region that's selected in a list widget. Or, when the value of a number widget can change to red when it falls below a threshold.

7. Set Initial Selections and Global Filters in the Dashboard

   Set the initial selections and global filters that appear when the dashboard first opens. To analyze the results from a different angle, the dashboard viewer can change the initial selections and, if configured, global filters while viewing the dashboard.

8. Generate Unique Tableau CRM Dashboard Layouts for Different Devices

   After you add widgets to the dashboard, optimize the layout for each device on which the dashboards can be viewed. For example, you can remove widgets from a mobile phone layout to reduce the dashboard size for the smaller screen. You can also move widgets around in one layout and it doesn't affect the other layouts.

9. Optimize Dashboard Performance

   Before you finalize the dashboard, run a performance check on the dashboard and its queries to ensure that everything is running optimally. The dashboard inspector identifies different types of bottlenecks, like query issues and redundant queries, and provides recommendations to improve performance. Because dashboard layouts can contain different widgets (and queries), run the inspector on each layout. If a dashboard contains multiple pages, run the inspector on each page. The inspector provides results only for the current page.

10. Keyboard Shortcuts for Building Tableau CRM Dashboards and Lenses

    You can do some basic actions from your keyboard.

11. Restore a Previous Version of a Dashboard

    Tableau CRM uses version history to back up dashboard versions when you edit them so you can restore a previous version.

12. Collaborate on a New Dashboard Version Behind the Scenes (Pilot)

Tableau CRM users can add themselves to the list of dashboard publishers so they can edit and test new versions of the dashboard. Other Tableau CRM users continue to see a live version until a publisher makes the draft the live version.

SEE ALSO:

What Is a Dashboard?

Best Practices for Building Your Own Tableau CRM Dashboard

Learn Tableau CRM with In-App Examples

Create a Tableau CRM dashboard with a Template

Dashboard templates speed your analytics development by automatically creating dashboards. Some provide blank layouts that you populate with data, while other "smart" templates create dashboards that require little to no additional configuration.

Watch a Demo: ▶ Quickly Create Wave Dashboards with Templates (English Only)

Each dashboard template provides a prebuilt layout with empty widgets, sections, and a color scheme. Use the widget wizards to populate the widgets with your data. Choose the blank template if you prefer to create your own layout.

Choose smart templates to create complete dashboards already populated with your data. They also include a configuration wizard to guide you through the creation process. You don't have to create queries or do any manual configuration—just answer a few questions about the data you'd like to include.

1. From the home page, select **Create** > **Dashboard**.

Create Analytics Dashboard
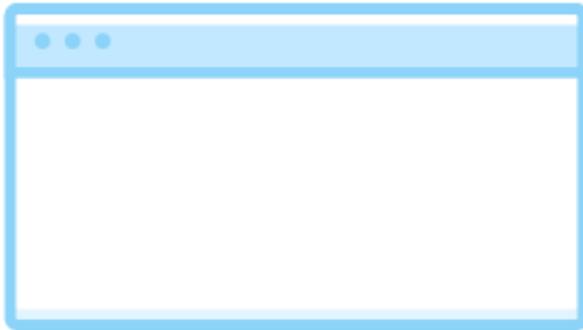
Blank Dashboard

Create your dashboard from scratch.

Create

Dashboard from

Choose a template to get you

Create

2. Choose **Blank Dashboard** or **Dashboard from Template**, and click **Create**.

3. Choose the template to use from the template picker. The **SMART** icon in the upper-right corner of a template thumbnail designates a smart template.
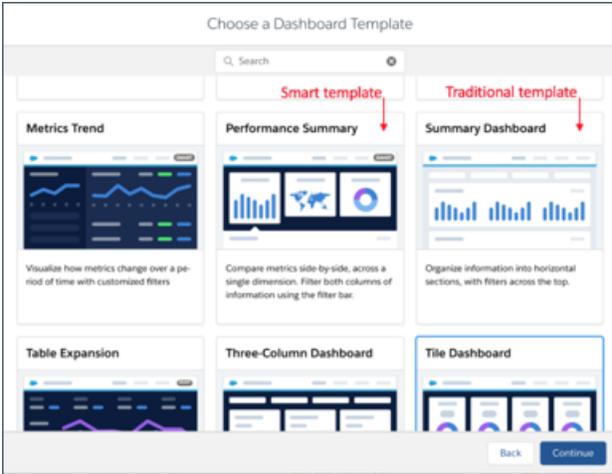
4. If you use a smart template, follow the instructions in the wizard. They prompt you to choose datasets and the specific data to display in the dashboard.

> ⛔ **Important:** Creating dashboards from smart templates requires the 'Use Any API Client' permission. Without that permission, you may not see all available datasets in the wizard.. To assign that permission, request API Access Control from Salesforce Customer Support. See Restrict Access to APIs with Whitelisted Connected Apps.

5. If you use a traditional template, add a query to each widget by clicking the icon inside the empty widget. A wizard guides you through adding the query. When you complete the wizard, Tableau CRM populates the widget based on the results of the query. Tableau CRM also adds the query to the query panel. If you use a smart template, you're asked a set of questions about your data before Tableau CRM creates the dashboard. Dashboards created from smart templates usually don't require any more configuration work, but you can customize them as you would any other dashboard.

6. To set the dashboard properties, click ⋯ and then click **Dashboard Properties**.

    The dashboard properties include the default widget properties, which you can use to ensure that the background and borders of all widgets are consistent. When you create a widget, you can specify whether the widget uses these default properties.

7. To save the dashboard, click ⊟ , enter a title and description. Also enter a brief description in the **Version History** field to help you remember what's unique about it in case you ever want to revert to it. For example, since it's the first version, enter `Original`. Select the app in which you want to save the dashboard, then click **Save**.

The permissions on the app determine each user's access to the dashboard. To ensure that the right users have the right access to the dashboard, review the app permissions.

Tableau CRM Dashboards Enabled with Time Zone (Pilot)

Follow best practices to design and build useful, effective Tableau CRM dashboards, while minimizing rework and addressing potential gaps.

Dashboard Properties

These properties apply to a dashboard designer dashboard. Some of them are the default widget properties that you can apply to each widget to ensure a consistent appearance.

Smart Dashboards

Use a smart dashboard template to create a ready-made dashboard that's prepopulated with your Salesforce data.

Follow best practices to design and build useful, effective Tableau CRM dashboards, while minimizing rework and addressing potential gaps.

Dashboard Properties

These properties apply to a dashboard designer dashboard. Some of them are the default widget properties that you can apply to each widget to ensure a consistent appearance.

General

Select **Available on mobile** to show this dashboard in the Tableau CRM mobile app. When not selected, mobile users can't access this dashboard. Use this option to hide dashboards that don't perform well on the mobile app.

Widget Default Properties

Property	Description
Background Color	Default background color for widgets
Border	Default border around widgets
Border Color	Default color for widget borders
Border Width	Default width of widget borders (in pixels)
Border Radius	Default roundness of the corners of widget borders

Smart Dashboards

Use a smart dashboard template to create a ready-made dashboard that's prepopulated with your Salesforce data.

The Tableau CRM dashboard template picker includes the following smart dashboards. Each one lets you select data from an individual Tableau CRM dataset to include in the dashboard. After selecting the template to use, follow a series of wizard questions to indicate specific fields, dimensions, measures, and so on to display in visualizations.

- **Performance Summary**. Compares metrics side by side, across a single dimension. The wizard lets you select metrics to display, such as sales revenue, product mix, or geography. Also select dates to trend the data over a certain period, and indicate how to group the data.

- **Metrics Trend**. Visualize how metrics change over time with customized filters. Select the metrics to display, such as revenue, profit, or quantity, and the time period to trend the data.

- **Table Expansion**. Visualize how metrics change over time with customized filters. The dashboard details table expands into a full page. Select measures dimensions to include in the dashboard and the time period to trend the data.

- **Time Series**. Look at future metrics trends based on historical data. The template gives you the option of using the SAQL timeseries function. Select measures dimensions to include in the dashboard and the time period to trend the data.

Best Practices for Building Your Own Tableau CRM Dashboard

Follow best practices to design and build useful, effective Tableau CRM dashboards, while minimizing rework and addressing potential gaps.

- Before you build the dashboard, take into account the following design best practices:

  - Sketch your dashboard on paper or a whiteboard before you start building.

  - Prioritize elements, top left to bottom right. With languages that are read left to right, people start by looking at the top left corner and working their way down. Consider the audience's language and design for it. If your audience has limited time or attention, place important elements where they will be noticed.

  - Place high-level, easy-to-read, actionable widgets near the top left, and place widgets with supporting information lower. For example, place numbers that display a single measure, such as revenue for the current quarter, high and to the left.

    Highlight facts that the audience wants to know at a glance.

  - Group filters together at the top or left so that they are quickly noticeable. You can use a container widget to section them off in the dashboard.

  - Keep in mind that a chart in Analytics Cloud is primarily a way to ask questions, not a way to illustrate a conclusion. A good dashboard invites the audience to drill down and seek ever more focused and useful information.

  - Choose chart types based on the characteristics of the data, not for look or variety. For example, if most of your charts display value changes over time, it's OK if they're all line graphs.

  - If a chart seems to need a lengthy caption or title, reconsider whether the chart is doing its job. Well-chosen data often speaks for itself.

- Use container widgets to frame and organize related elements in the dashboard.
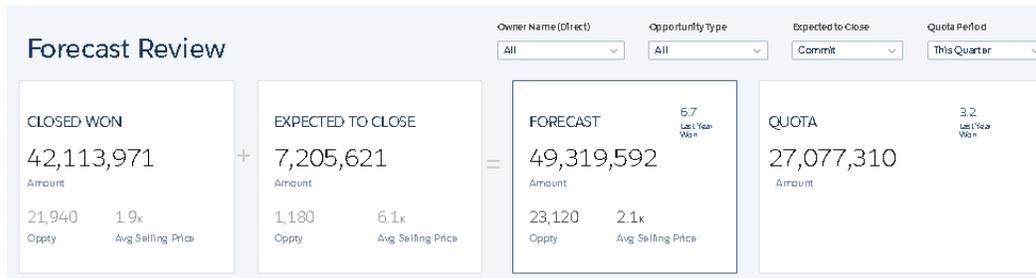
- While you build the dashboard:
  - Apply labels to sections and charts to annotate the dashboard.

  - Use colors to define sections.
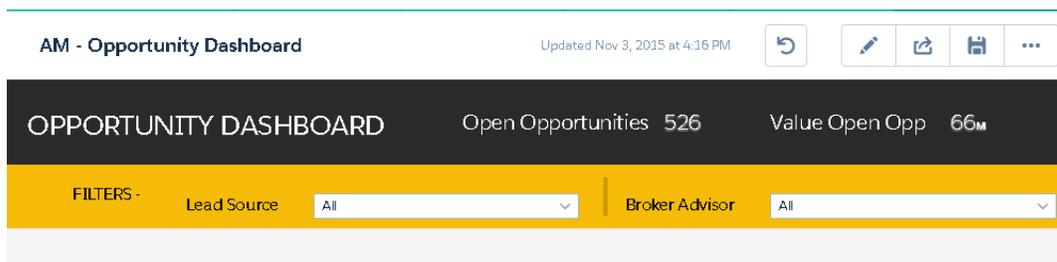
  - Don't clutter the dashboard—leave some empty space. If needed, break a dashboard into a series of dashboards, using link widgets to help the user navigate them.

- After you build the dashboard:
  - Have users review the dashboard before making it final. You can post a dashboard to Chatter to get feedback. Users can annotate the widgets in the dashboard to have conversations about the content.

SEE ALSO:

Build Tableau CRM Dashboards

Create and Manage Dashboards with Reusable Components

Dashboard components are a type of dashboard widget that can contain other widgets and pages. Use dashboard components to manage and reuse groups of charts, tables, filters, text, and more in multiple dashboards.

Watch a Demo: ⏵ Save Time Creating and Managing Dashboards with Reusable Components (English Only)

Create and Reuse Dashboard Components

Use dashboard components to manage and reuse groups of charts, tables, filters, text, and more in multiple dashboards.

Edit Dashboard Components

When you edit a component that's used in other dashboards, you can edit the original component or make a copy. If you edit the original component, keep in mind that changes are applied wherever the component is used.

Preview and Save Dashboard Components

Preview and save components the same way you preview and save a dashboard.

Delete Dashboard Components

If you delete a component that's used in other dashboards, keep in mind that you're deleting the contents of the component only. The component will be empty wherever it's used in other dashboards. To completely delete a component in other dashboards, open and delete the component in dashboards that use it.

Manage Global Filters in Dashboard Components

Global filters let you apply the same filter to multiple queries in a dashboard, or components in a dashboard.

Manage Datasets in Dashboard Components

You can control whether components inherit data source linking from the dashboard in which the component is created.

Create and Reuse Dashboard Components

Use dashboard components to manage and reuse groups of charts, tables, filters, text, and more in multiple dashboards.

Watch a Demo: ⏵ Save Time Creating and Managing Dashboards with Reusable Components (English Only)

1. To create a dashboard component, drop a component widget (▦) onto a dashboard.

2. Click the component. You can create a new component, or use an existing one.

   When you save a component, you can use it in other dashboards as much as you want.

   💡 **Tip:** If your component contains multiple pages, you can select which page to show as the default initial page of the component.

   📝 **Note:** Layouts aren't supported in dashboard components.

SEE ALSO:

Manage Global Filters in Dashboard Components

Manage Datasets in Dashboard Components

Edit Dashboard Components

When you edit a component that's used in other dashboards, you can edit the original component or make a copy. If you edit the original component, keep in mind that changes are applied wherever the component is used.

1. To edit a component, open a dashboard that contains the component you want to edit.

2. Click **Edit**.

3. Select the component you want to edit and click **Edit Component** in the widget properties.

   The component opens in a new tab.

Preview and Save Dashboard Components

Preview and save components the same way you preview and save a dashboard.

👁 Example: With the dashboard component open in Tableau CRM, click **Preview** or **Save**.

💡 Tip: You can save components in a private app or in shared apps. Keep in mind that components saved in a private app can be used in other apps, but the contents of the component are visible only by you.

Delete Dashboard Components

If you delete a component that's used in other dashboards, keep in mind that you're deleting the contents of the component only. The component will be empty wherever it's used in other dashboards. To completely delete a component in other dashboards, open and delete the component in dashboards that use it.

1. To delete a component, open a dashboard that contains the component you want to delete.

2. Click **Edit** (✏).

3. Select the component you want to delete.

4. Click **More** (⋯), and then select **Delete**.

Manage Global Filters in Dashboard Components

Global filters let you apply the same filter to multiple queries in a dashboard, or components in a dashboard.

To manage global filters in components, click the global filters icon (🔽), then select **COMPONENT** in the sidebar.

When components and dashboards contain global filters that use the same field, dashboard global filters override global filters in components.

📝 Note: If two components in the same dashboard have global filters on the same field, the filter applied is based on the alphanumeric order of the components' API names. For example, a global filter in a component with a name "corporate_info_1" takes precedence over a global filter in a component named "corporate_info_2". To change which global filter is used in components, clone and rename the component with the global filter you want applied. Make sure the component's API name precedes other component API names in alphanumeric order.

SEE ALSO:

[Create and Reuse Dashboard Components](#)

Manage Datasets in Dashboard Components

You can control whether components inherit data source linking from the dashboard in which the component is created.

To manage data filtering and faceting in a component, select a component and then select or deselect the **Allow Filtering and Faceting** check box.

When selected, filtering and faceting is available in both the dashboard and components in the dashboard. When deselected, filtering and faceting is available in the dashboard only.

Keep these considerations in mind when managing components and datasets.

- Components follow all dashboard-level data viewing, editing, and security rules.
- You can use data source linking to facet data in components the same way you facet data in dashboards. Use the dashboard designer to link datasets in components.
- Component access is based on app access. If you have access to an app, you also have access to components and datasets in that app.
- Components saved in a private app aren't available to other users or public apps. This means the component can be used in other apps, but its contents are visible only by you.

SEE ALSO:

Create and Reuse Dashboard Components

Configure Cross-Dataset Faceting with Connected Data Sources

Salesforce Data Access in Tableau CRM

Create Dashboard Pages

Make the information on a dashboard easier to digest by chunking the content into multiple pages. And with fewer queries per page, dashboard performance increases. With pages, you can tell a story by creating a dynamic pathway through your dashboard. Depending on how you lay out your pages, you can also create some cool effects as you transition from one page to the next.

Share widgets, queries, global filters, and selections across pages to keep the same focus as you go from one page to the next. For example, a selection in one page also filters the results of the other pages.

Pages apply to the layout in which they're created. When you create a layout, all pages in the current layout are copied to the new layout.

1. To add another page, click ➕ .

   Each page appears as a separate tab. You can add up to 20 pages per dashboard. When you preview a page, it looks just like a dashboard.

2. Click ⌄ next to a page to configure it, like rename it or hide it from the navigation widget when a user views the dashboard.
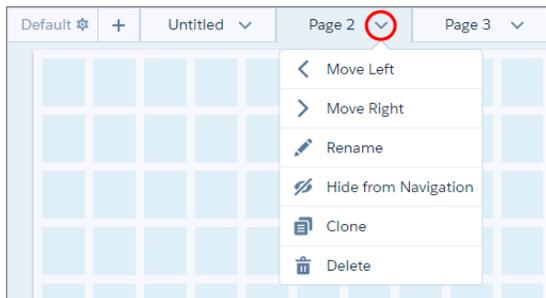
3. To enable pages to link to each other, add a navigation widget that contains a tab for each visible page.

   When a navigation widget is added, mobile users can no longer swipe to access other pages—they must use the navigation widget.
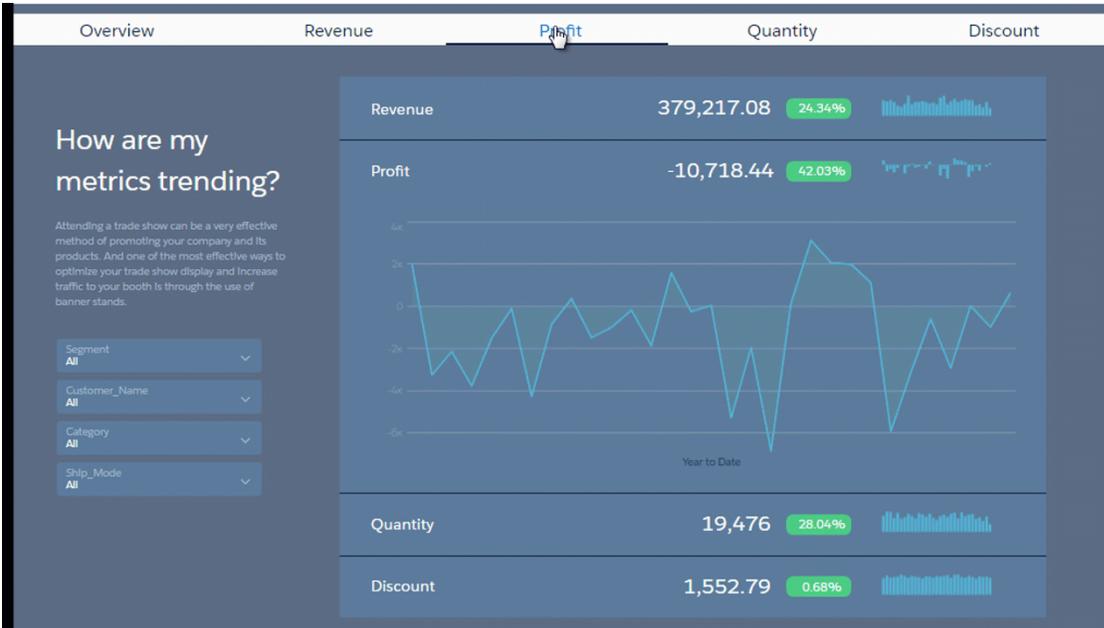
> **Note:** If you don't add a navigation widget, mobile app users swipe to get to the next page. Use the **Move Left** and **Move Right** actions to reorder the pages.

4. To create a path for viewers to navigate to a specific page, add a link widget. In the link widget properties, you can link to a dashboard, page, query, lens, or web page.

5. To share widgets on other pages, select the widgets, click **Add to Page**, and then specify the pages.

   For example, use it to add the navigation widget to all pages. When you use the Add to Page option to add a widget to another page, you add an instance of the same widget to the page. Basically the widget is shared across both pages, making it easier to maintain. The styling properties you set for the widget reflects across all pages.
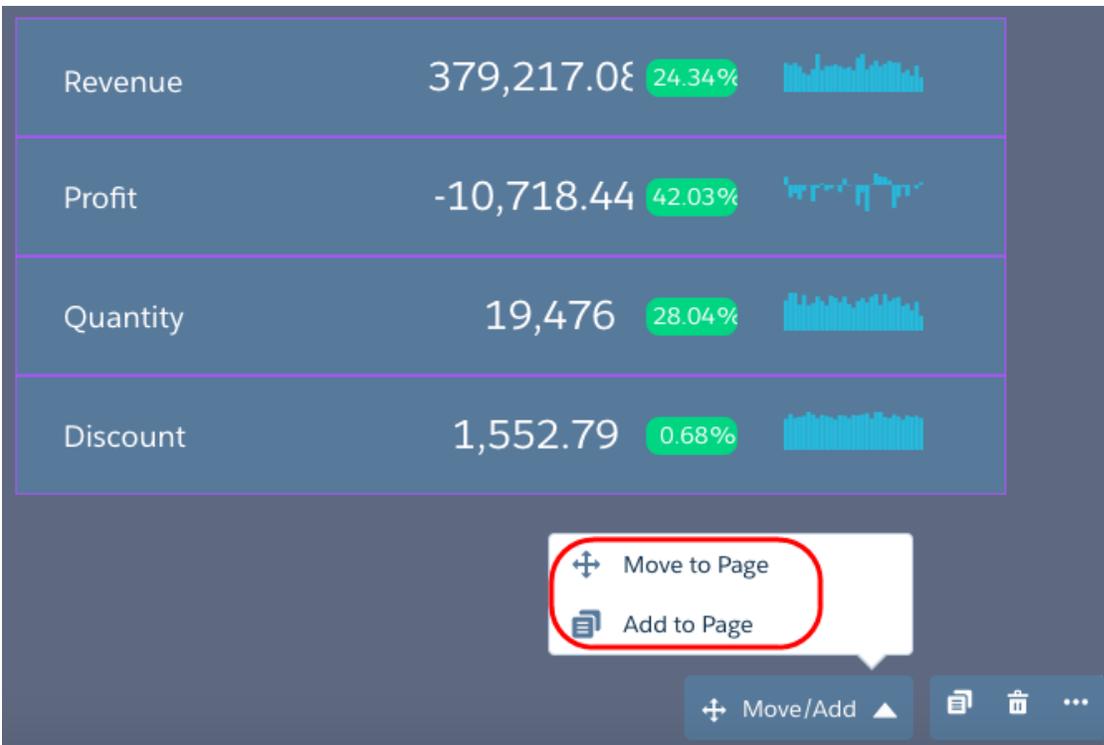
6. If needed, repeat these steps to set up pages for other layouts.

7. Save the dashboard to save changes to pages. When you save the new version, enter a brief description of the changes in the **Version History** field. The description can help you remember what's unique about this version of the dashboard in case you want to restore this version after you make changes and save a new version. For example, enter `Added Pages`.

Manage Widgets on a Page

You can move widgets to another page. To share them and their queries across pages, add them to other pages. If you no longer need a widget, delete it from the page.

SEE ALSO:

Add a Link Widget to Link to Other Assets

Manage Widgets on a Page

You can move widgets to another page. To share them and their queries across pages, add them to other pages. If you no longer need a widget, delete it from the page.

1. To move a widget to another page, select the widget, click **Move/Add** in the widget actions menu, and then click **Move to Page**.
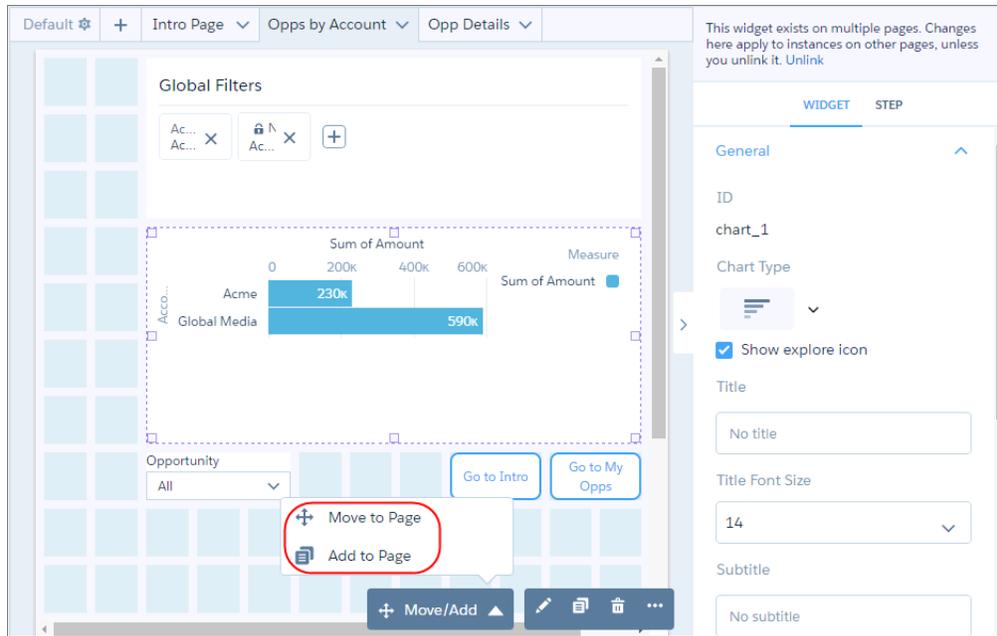
2. To add a widget to another page, select **Add to Page**.

When you add a widget from one page to another, Tableau CRM adds an instance of the same widget. If you change a widget property on one page, it affects all instances on other pages. If a widget is used in multiple pages, dashboard designer highlights the widget in purple and adds a banner in the right pane when you select it.
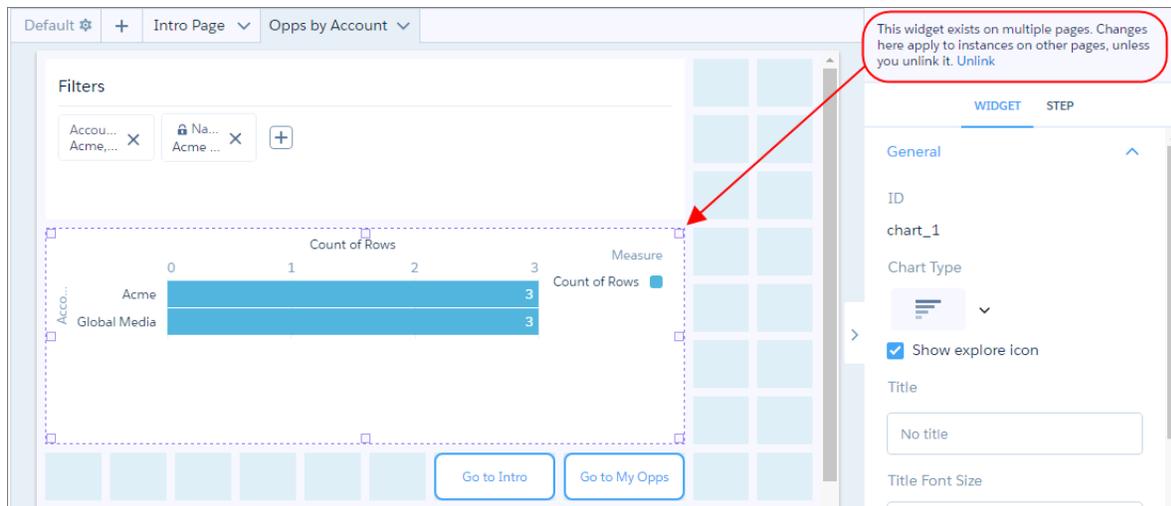
3. To make a widget work independently from other instances, select the widget, and then click **Unlink** in the banner.

4. To delete a widget from the page, select the widget, and then press the Delete key.

If you delete a widget from a page:

- and the widget is used in other pages, Tableau CRM removes the widget only from the current page.
- and the widget isn't used in other pages but is used in other layouts, Tableau CRM removes it from the page and current layout.
- and the widget isn't used in other pages or layouts, Tableau CRM deletes it from the dashboard.

5. To save your changes, save the dashboard.

Add Widgets to the Dashboard

Widgets are the basic building blocks of a dashboard. In the dashboard designer, you can add different widgets to perform functions. For example, widgets can calculate key performance indicators, filter dashboard results, visualize your data using interactive charts, and show record-level details in tables.

Watch a Demo: ▶ Get to Know Wave Dashboard Widgets (English Only)

Widgets that display data require a query. A widget determines how to use or display the results returned by a query. For example, a query calculates the average opportunity amount, grouped by region. A bar chart widget displays the results in a bar chart format with each bar showing the average opportunity amount for a region. If the query is faceted, when you select a bar in the chart, other widgets are filtered, showing results for only the selected region.

Most widgets have a wizard that you can use to build the widget, including the underlying query. You can also build your own custom query and apply it to a widget. Multiple widgets can use the same query, unless it's a date or range widget.

Clone widgets to create similar looking components in a Tableau CRM dashboard. For example, you can clone a chart widget and change the chart type to get a different view of the same results. When you clone a widget, you also clone its query. You can also quickly create three number widgets by cloning one of them and modifying their queries' filters to show total open, total won, and total lost opportunity amounts. To clone widgets, remove queries, clone queries, and view query properties, select the widget while editing the dashboard, and then click the appropriate button.
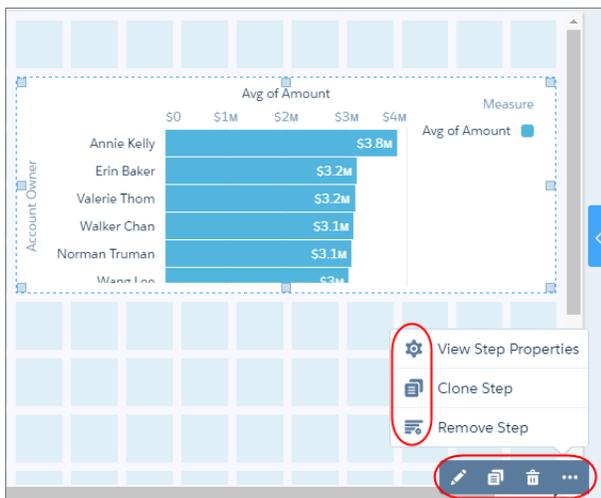
To configure how a widget appears, set the properties in the Widget tab. For example, you can show data values and the axis title for a chart widget. You can also apply default properties to ensure that widgets look consistent, like having the same border and background color.

To configure how the widget behaves, set query properties in the **Query** tab. For example, you can choose which measure to display in the widget when its query contains multiple measures. Changes to query properties impact all widgets that are built on the query.

When you save the new version of your dashboard, it's a good idea to enter a brief description of your changes to the **Version History** field. The description can help you remember what's unique about this version of the dashboard in case you want to restore this version after you make changes and save a new version. For example, enter `Added Number Widget`.

Add Number Widgets to Include Key Metrics

Key metrics help you track critical measurements of your business. For example, you can show the total number of open cases or total opportunity amount for the last quarter or the northern region.

Add Chart Widgets to Visualize Data

Charts in a Tableau CRM dashboard allow you to visually summarize and trend your data. For example, a donut chart can show the distribution of opportunities across industry type.

Add a Global Filter Panel Widget to Filter Data from the Dashboard

The panel can contain single or multiple global filters. Use a global filter to apply the same filter to multiple queries in the dashboard. For example, a dashboard has two charts comparing two sales reps' performance. To compare apples to apples, both charts must have the same filters. When possible, use a global filter instead of a selection-based filter to improve dashboard performance. If configured, users can change a global filter while viewing the dashboard.

Add Selection-Based Filter Widgets to Enable Users to Filter the Results

Users viewing the dashboard can make selections in these types of widgets, which filter the results of other faceted widgets. Selection-based filter widgets give users the ability to slice and dice the data to view results from different angles. For example, the dashboard viewer can select a region in a list widget to focus the dashboard results on a specific sales location. Unlike with a filter defined by a query, the viewer can filter the results differently (by making another selection) or remove the filter altogether.

Add a Table Widget to Show Record Details and Create Calculated Columns

You can add different types of tables to a Tableau CRM dashboard. To view record-level details, add a values table. A values table can show details like how long a case has been open and who owns it. To create custom columns in a table based on calculations from existing fields, add a compare table. If the underlying query contains at least one grouping, you can create a pivot table.

Add a Container Widget to Create Sections in the Dashboard

Use a container in a Tableau CRM dashboard to group related widgets. For example, you can use a container widget to create a filter panel on the left side of the dashboard to store all selection-based filter widgets. To distinguish the filter panel from the rest of the dashboard, you can apply a background color and border to the container. When you move this widget, the contained widgets also move while maintaining their spacing and alignment.

Add an Image Widget to Display Graphics in the Dashboard

You can include company logos to brand the Tableau CRM dashboard, icons to categorize results, or graphics to animate the dashboard.

Add Text Widgets to Label Parts of the Dashboard

Label your Tableau CRM dashboard so that users understand what they are looking at. You can also add text that describes the dashboard or provides usage information for widgets.

Add a Navigation Widget to Access Dashboard Pages

Give dashboard viewers an easy way to navigate dashboard pages with a navigation widget. Reuse your navigation widget for a consistent look across dashboard pages.

Add a Link Widget to Link to Other Assets

Take dashboard viewers straight to where they need to go with links to specific assets, like dashboards, pages, queries, lenses, and even web pages. You can also use links to create analysis paths for users to follow, helping them navigate your analytics app, or to perform mass quick actions on record lists.

Widget Properties for Tableau CRM Dashboards

Widget properties define how widgets appear in the Tableau CRM dashboard. For example, you can apply labels, backgrounds, and borders. To make widgets look consistent, you can set up default widget properties in the layout and then assign them to each widget. Widget properties vary based on the widget type and the designer used to create them. The properties listed here apply only to widgets created in the dashboard designer.

Add Number Widgets to Include Key Metrics

Key metrics help you track critical measurements of your business. For example, you can show the total number of open cases or total opportunity amount for the last quarter or the northern region.

The number widget requires a query that has at least one measure. If the query has multiple measures, you can choose which measure to display in the widget properties—by default, the first measure shows. If you include groupings, the widget displays the measure for the first group.

1. Drag the number widget to the dashboard canvas.

2. To open the wizard, click the button inside the widget.
   One of the following screens appears.

   - The Choose Data screen appears, showing the most recently used datasets first. Select the dataset in the Create Query tab or select an existing query in the Use Existing Query tab.

   - The explorer appears, showing a bar chart with the Count of Rows measure. Tableau CRM selects the dataset that you used to create the previous query. To use a different dataset or an existing query, click **Back**.

3. In the explorer, click **Untitled Query** and enter the query label.
   Tableau CRM creates the query ID from the label. After you create the query, you can't change the query ID. Tableau CRM refers to queries by their ID so dashboards don't break if you change the labels.

4. To add a measure, click **+** under the Bar Length field.
   The Count of Rows measure shows, by default. To change a measure, click the measure and then choose a new one.

5. To sort, reorder, or delete a measure, click the down arrow to the right of the measure and select the option.

Alternatively, to quick sort on a measure, click the measure's field header. Quick sort isn't available for charts with these underlying queries:

- When the compact-form query has interactions present on the sort order
- When the SAQL query contains union statements or interactions
- When the query is in PIGQL

1216

6. To add a filter, select **Filters**, click **+** under Filters, and then select the field to filter by.

You can add multiple filters.
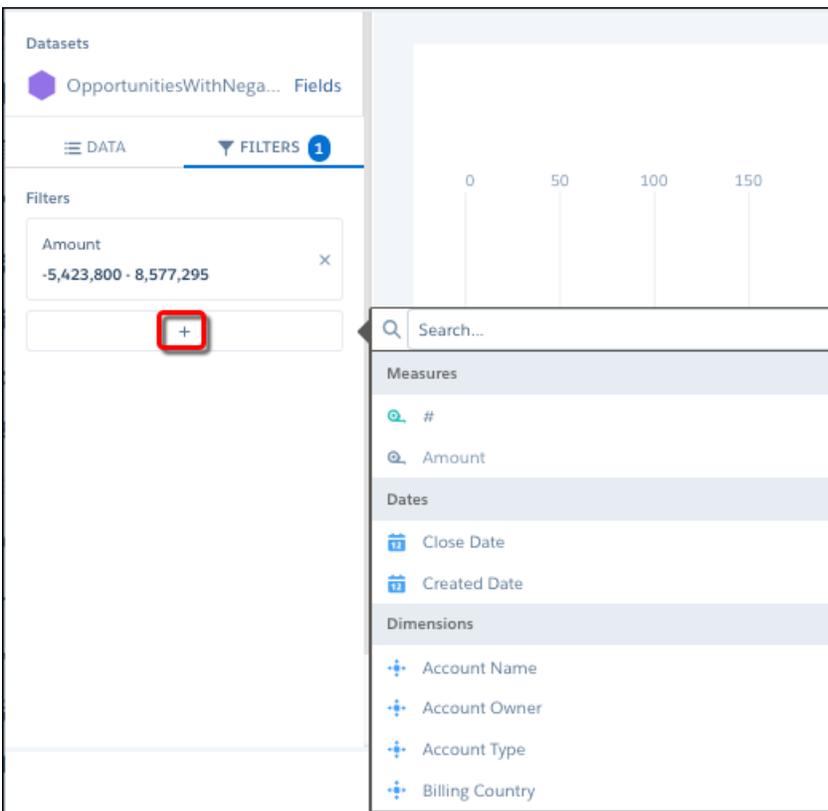
7. To delete the filter, click ╳ .

8. Click **Done**.

Behind the scenes, Tableau CRM created a query based on what you configured in the explorer. Tableau CRM adds the query to the query panel, making it available for use by other widgets in the dashboard. In addition, the widget shows the calculated measure.

9. To change the widget and query properties, select the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.
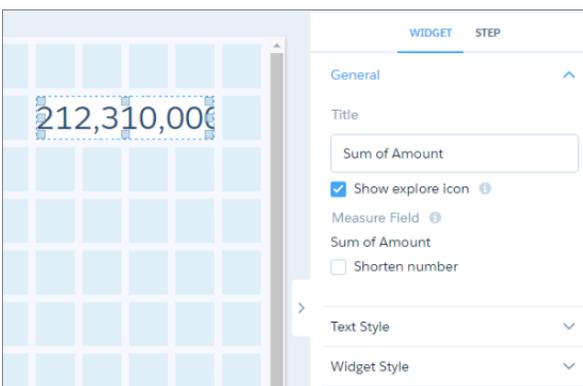
For example, to shorten the number that appears in the widget, select **Shorten number** in the widget properties. Check out the unit symbols for shortened numbers.

Symbol	Unit	Example
K	Thousand (10^3)	4K = 4,000
M	Million (10^6)	4M = 4,000,000
B	Billion (10^9)	4B = 4,000,000,000
T	Trillion (10^12)	4T = 4,000,000,000,000
P	Quadrillion (10^15)	4P = 4,000,000,000,000,000
X	Quintillion (10^18)	4X= 4,000,000,000,000,000,000
Z	Sextillion (10^21)	4Z = 4,000,000,000,000,000,000,000
Y	Septillion (10^24)	4Y = 4,000,000,000,000,000,000,000,000
m	Thousandth (10^-3)	4m = 0.004
μ	Millionth (10^-6)	4μ = 0.000004
n	Billionth (10^-9)	4n = 0.000000004
p	Trillionth (10^-12)	4p = 0.000000000004
f	Quadrillionth (10^-15)	4f = 0.000000000000004
a	Quintillionth (10^-18)	4a = 0.000000000000000004
z	Sextillionth (10^-21)	4z = 0.000000000000000000004
y	Septillionth (10^-24)	4y = 0.000000000000000000000004

10. To customize the appearance of the widget, set the widget properties.

When you set a property, the widget immediately shows the effects of your change.

11. To show and set the query properties, click **Query** in the right panel.

12. To preview your changes to the dashboard, click 👁 .

13. Save the dashboard.

SEE ALSO:

Widget Properties for Tableau CRM Dashboards

Query Properties for Tableau CRM Dashboards

Manage Queries for Widgets

Add Chart Widgets to Visualize Data

Charts in a Tableau CRM dashboard allow you to visually summarize and trend your data. For example, a donut chart can show the distribution of opportunities across industry type.

Tableau CRM supports multiple chart types. Each type has unique widget properties and query requirements. For details about each chart type, see Visualizing Data With Charts.

When previewing a dashboard, some charts show values in segments. Values appear as shortened numbers, rounded to the nearest whole number. For example, 1,111,556 appears as 1 M, 999.95 appears as 1 K, and 111,045 appears as 111 K. You can hover over a segment to view the actual value.

Check out the unit symbols for shortened numbers.

Symbol	Unit	Example
K	Thousand (10^3)	4K = 4,000
M	Million (10^6)	4M = 4,000,000
B	Billion (10^9)	4B = 4,000,000,000
T	Trillion (10^12)	4T = 4,000,000,000,000
P	Quadrillion (10^15)	4P = 4,000,000,000,000,000
X	Quintillion (10^18)	4X= 4,000,000,000,000,000,000
Z	Sextillion (10^21)	4Z = 4,000,000,000,000,000,000,000
Y	Septillion (10^24)	4Y = 4,000,000,000,000,000,000,000,000
m	Thousandth (10^-3)	4m = 0.004
μ	Millionth (10^-6)	4μ = 0.000004
n	Billionth (10^-9)	4n = 0.000000004
p	Trillionth (10^-12)	4p = 0.000000000004
f	Quadrillionth (10^-15)	4f = 0.000000000000004

Symbol	Unit	Example
a	Quintillionth (10^-18)	4a = 0.000000000000000004
z	Sextillionth (10^-21)	4z = 0.000000000000000000004
y	Septillionth (10^-24)	4y = 0.000000000000000000000004

1. Drag the chart widget to the dashboard canvas.

2. To open the wizard, click the button inside the widget.
   One of the following screens appears.

   - The Choose Data screen appears, showing the most recently used datasets first. Select the dataset in the Create Query tab or select an existing query in the Use Existing Query tab.

   - The explorer appears, showing a bar chart with the Count of Rows measure. Tableau CRM selects the dataset that you used to create the previous query. To use a different dataset or an existing query, click **Back**.

3. In the explorer, click **Untitled Query** and enter the query label.

   Tableau CRM creates the query ID from the label. After you create the query, you can't change the query ID. Tableau CRM refers to queries by their ID so dashboards don't break if you change the labels.

4. To add a measure, click **+** under the Bar Length field.

   The Count of Rows measure shows, by default. To change a measure, click the measure and then choose a new one.

5. To sort, reorder, or delete a measure, click the down arrow to the right of the measure and select the option.

6. To group the measures by a date or dimension, click **+** under Bars, and then select the field.

   You can add multiple groupings. To change a grouping, click the grouping and select a new field. To reorder or delete a grouping, click the down arrow to the right of the grouping and then select the option.

7. To add a filter, select **Filters**, click **+** under Filters, and then select the field to filter by.

   You can add multiple filters.
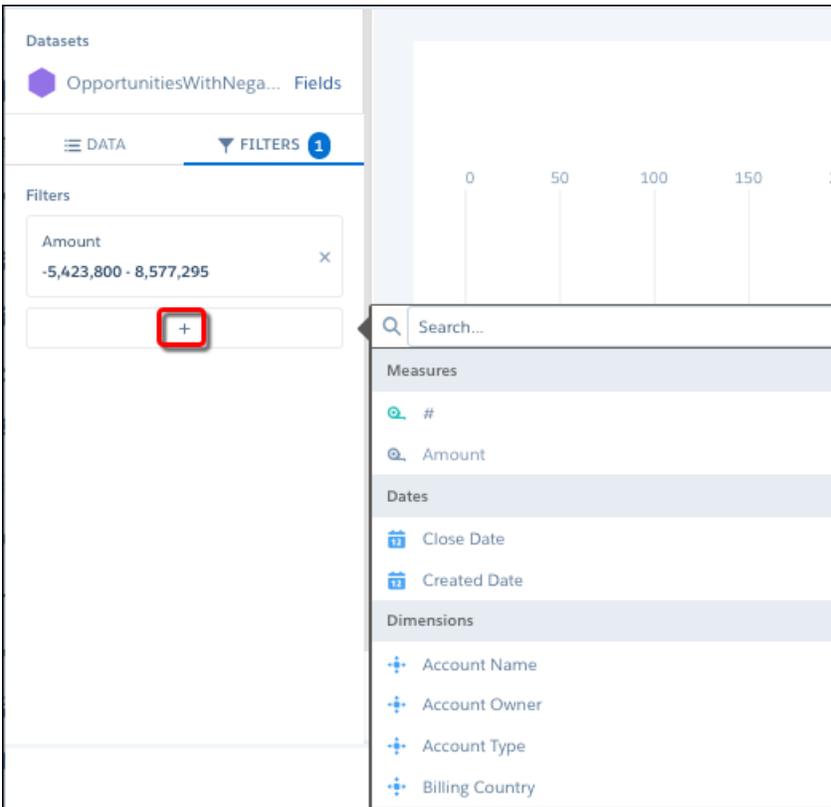
8. To change the chart type, click ▥ and the select the new one.

Although Tableau CRM saves the chart type with the query, you can override the chart type at the widget level by specifying a different chart in the widget properties. Setting the chart type at the widget level is useful when you use the same query for multiple widgets, but want to display the results differently in each widget.

9. Click **Done**.

Tableau CRM adds the query to the query panel, making it available for use by other widgets in the dashboard. In addition, the widget shows the results in a bar chart.

10. To change the widget and query properties, select the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.
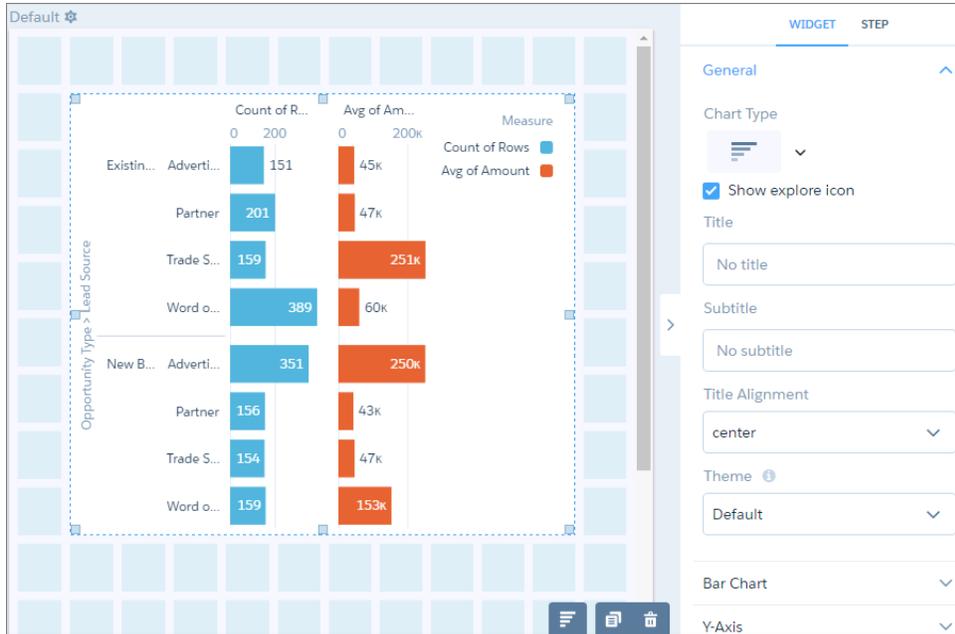
11. To customize the appearance of the widget, set the widget properties.

For example, add a title in the Title field.

12. To show and set the query properties, click **Query** in the right panel.

For example, to enable this widget to filter other faceted widgets, select **Broadcast selections as facets**.

While viewing the dashboard, a user can select a segment of a chart to filter the results of other faceted widgets. To allow the viewer to make single or multiple selections, configure this widget's query properties. You can specify whether selections are optional or required. You can also set an initial selection that applies when the dashboard opens. While viewing the dashboard, viewers can undo the selection or change the selection to filter the results differently. For information about setting an initial selection, see Set Initial Selections in the Dashboard .

13. To preview your changes to the dashboard, click 👁 .

14. Save the dashboard.

Project the Same Query Results Differently in Charts and Tables
If charts and tables use the same query, each widget can show a different set of fields from the query results. You can also specify a different field order for each widget. For example, to change the measure that appears on each axis in a scatter plot, reorder the measures for that widget.

SEE ALSO:

Widget Properties for Tableau CRM Dashboards

Query Properties for Tableau CRM Dashboards

Manage Queries for Widgets

Project the Same Query Results Differently in Charts and Tables

If charts and tables use the same query, each widget can show a different set of fields from the query results. You can also specify a different field order for each widget. For example, to change the measure that appears on each axis in a scatter plot, reorder the measures for that widget.

If charts and tables use the same query, each widget can show a different set of fields from the query results. You can also specify a different field order for each widget. For example, to change the measure that appears on each axis in a scatter plot, reorder the measures for that widget.

The following compare table query shows multiple measures grouped by a dimension. The Expected Amount measure is calculated based on the other two measures: Amount * Probability.

The following compare table query shows multiple measures grouped by a dimension. The Expected Amount measure is calculated based on the other two measures: Amount * Probability.

To show only the Expected Amount and Name fields for this widget, edit the widget and then select **Hide** for the other measures. (To show a hidden field, select **Show**. To reorder a field, select **Move Up** or **Move Down**.)

The following screenshot shows two charts based on the same query. The second chart shows the effect of hiding the other two measures.

The **Show** and **Hide** options are only available for `aggregateflex` (for compare tables) and `saql` query types. To show or hide fields for tables or charts based on other query types, configure the widget properties in the dashboard JSON. Add a "columns" attribute and specify the API names of the fields that you want to appear in the widget.

```
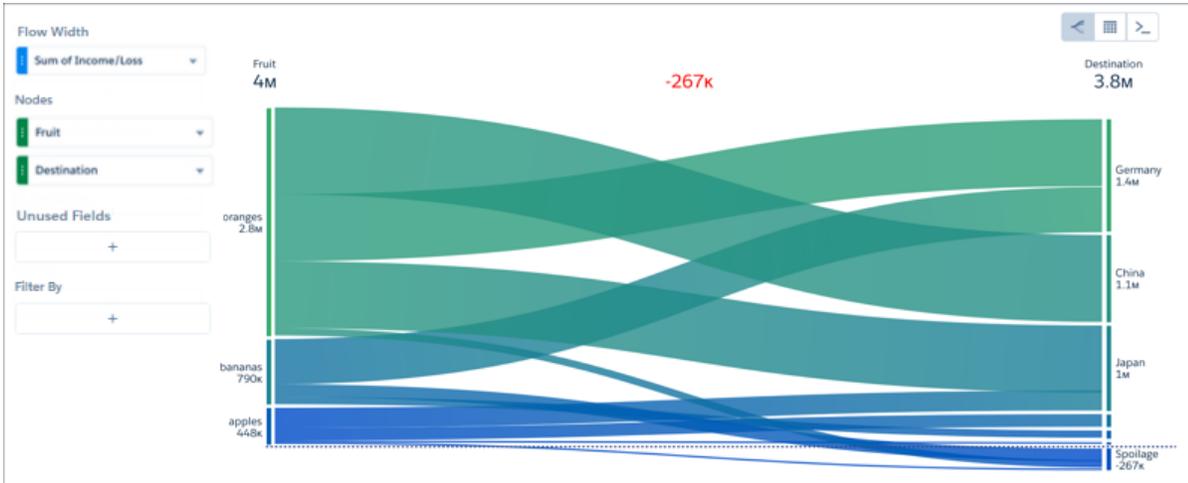"chart_2": {
    "parameters": {
        "columns": [ "Name", "C" ],
        "visualizationType": "hbar",
        "step": "Amount_Prob_1",
        ...
    },
    "type": "chart",
    ...
}
```

Consider the following guidelines when configuring the "columns" JSON property.

- If fields are listed in "columns" and you add a measure or grouping to the query, the new field doesn't appear in the widget until you add it to "columns."

- If "columns" is not specified in the widget properties or is set to an empty array, the widget shows all fields available in the query results.

- Tableau CRM ignores fields specified in "columns" that don't exist in the query. If none of the fields in "columns" exist in the query, the widget doesn't show any fields.

- The order in which you specify fields in "columns" is the order in which they appear in the widget.

  📝 Note: There are two ways to show, hide, or reorder columns. Do it by editing the query or editing the widget. If you perform these actions at the query level, your changes impact all widgets that use the query and can break widgets that use the query. For example, a widget breaks if you hide a required measure.

SEE ALSO:

Add a Global Filter Panel Widget to Filter Data from the Dashboard

The panel can contain single or multiple global filters. Use a global filter to apply the same filter to multiple queries in the dashboard. For example, a dashboard has two charts comparing two sales reps' performance. To compare apples to apples, both charts must have the same filters. When possible, use a global filter instead of a selection-based filter to improve dashboard performance. If configured, users can change a global filter while viewing the dashboard.

Each global filter must be based on a single measure, dimension, or date field. You can't create a composite filter based on multiple fields, like Region="West" and Industry= "CPG."

If multiple datasets are linked using a data source connection, only one of the common fields used in the connection can be included in a global filter. If you include multiple common fields, they can generate redundant, conflicting filters and not work properly. For example, you connect the Opportunity dataset using the Account.Name field to the Account dataset using the corresponding Name field. To filter based on account name, add one global filter based on either Account.Name from the Opportunity dataset or Name from the Account dataset. Because the datasets are connected, the global filter applies to both datasets.

Global filters only apply to widgets that accept them. To apply global filters to a widget, select **Apply global filters** in the query properties of the widget. By default, Tableau CRM applies global filters to all widgets that have queries based on the same dataset or connected data sources.

1. Drag the global filter panel widget to the dashboard canvas.

   An empty panel appears in the designer.
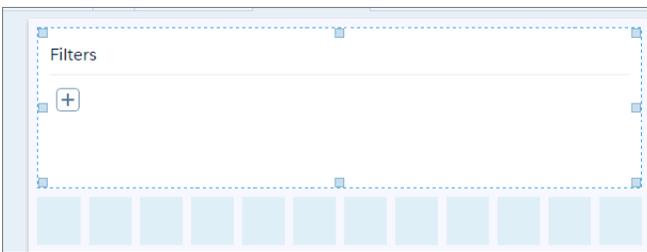
   > **Note:** If the dashboard contains a global filter created before Winter '18, a pop-up message asks if you want Tableau CRM to convert it to the new version. If you agree, Tableau CRM converts all global filters to the new design, showing all global filters in a panel. During the conversion, Tableau CRM doesn't add unsupported global filters to the new global filter panel widget. Unsupported global filters are composite filters and invalid ones that reference datasets or fields that don't exist. If you don't convert to the new design, you can continue to use the legacy global filter, but can't add global filters to the dashboard.

2. To add a global filter to the panel:

   a. In the global filter panel widget, click ➕.

   b. Select a dataset field and click **Create**.

      To use a different dataset, click **Change Dataset**.

   c. Select the filter operator and field values.
      These filters apply when a user opens the dashboard.

1227
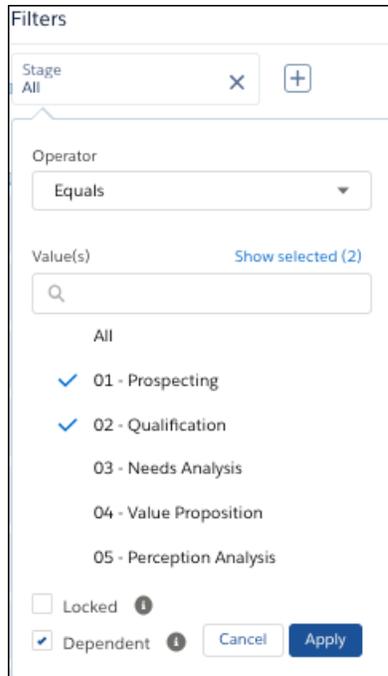
d. To prevent users viewing the dashboard from changing the global filter, select **Locked**.

The Locked option also ensures that incoming filters passed from a linked dashboard don't override the global filter. If the global filter is locked and the incoming filter is defined on the same field, Tableau CRM ignores the incoming filter. If it's unlocked, the incoming filter overrides the global filter defined in the dashboard.

e. To limit the global filter's picklist values based on selections in other global filters, select **Dependent**.

For example, you select an opportunity in the Opportunity Name global filter. Only products sold in the selected opportunity appear in the Product Name global filter picklist. The Dependent option applies to global filters that use the same dataset or a connected data source.

f. Click **Apply**.

3. To change the widget properties, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.

4. To customize the appearance of the widget, set the widget properties.

When you set a property, the widget immediately shows the effects of your change.

5. To edit a global filter in the panel, access the global filter panel by clicking the ▼ icon at the top of the dashboard.

For example, you might edit a global filter to change the selections.

6. To delete a global filter from the panel, click ✕ .

7. To delete all global filters from the dashboard, click **Delete All Global Filters** button in the Global Filters panel.

> Note: If you delete the global filter panel widget, Tableau CRM removes the panel from the dashboard, but the global filters still apply. To delete the global filters later, use the global filter property panel.

8. To preview your changes to the dashboard, click 👁 .

9. Save the dashboard.

Add Selection-Based Filter Widgets to Enable Users to Filter the Results

Users viewing the dashboard can make selections in these types of widgets, which filter the results of other faceted widgets. Selection-based filter widgets give users the ability to slice and dice the data to view results from different angles. For example, the dashboard viewer can select a region in a list widget to focus the dashboard results on a specific sales location. Unlike with a filter defined by a query, the viewer can filter the results differently (by making another selection) or remove the filter altogether.

A selection-based filter is different from a query. When a selection is made, all values in the original query still appear in the widget, just with particular values highlighted. However, a query modifies the widget, reducing the number of results that appear in the widget. A global filter is a query that can be applied to multiple queries in a dashboard.

The following illustration shows the difference between a selection and a query on the same data.
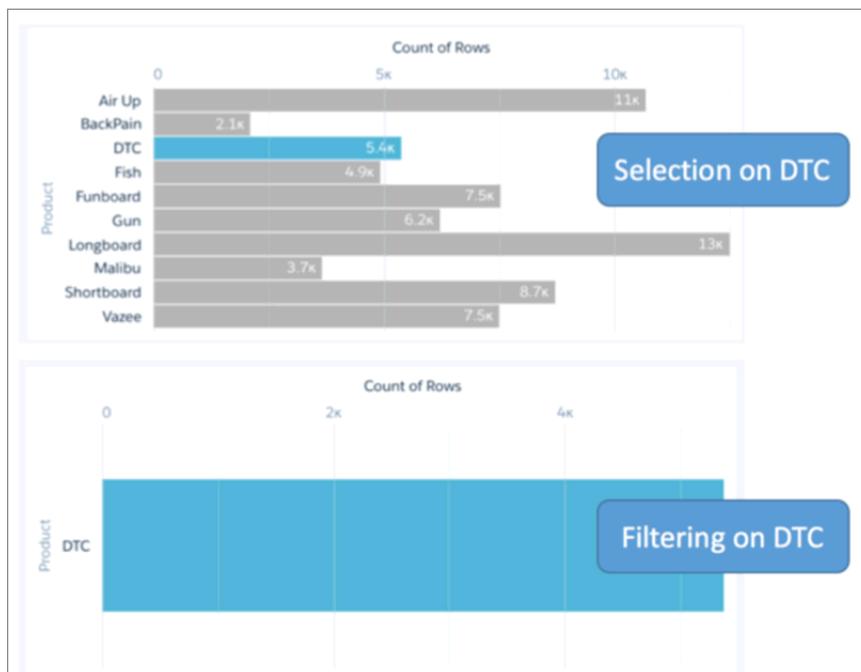
You can create different types of selection-based filter widgets based on measure, date, and dimension fields. You can also set an initial selection for each of them. For example, when the viewer opens the dashboard, the date widget can select last year, by default, to show dashboard results for last year only.

Selection-based filter widgets can be styled using pill or combobox styling to easily mix-and-match selection-based filters and global filters on the same dashboard.

SEE ALSO:

Set Initial Selections in the Dashboard

Add a Date Widget to Filter Dashboard Results Based on a Date

A date filter widget allows dashboard viewers to filter the Tableau CRM dashboard results based on a selected date range. The range can use relative dates based on UTC (like 2 years ago), absolute dates (like 1/15/2016), or presets (like Current Fiscal Year To Date). For example, you can add a date widget to filter support cases, showing only cases that closed this month or last month.

The date widget requires a query that has a single grouping based on a date field.

1. Drag the date widget to the dashboard canvas.

2. To open the wizard, click the button inside the widget.
   The Select a Field screen appears, showing a list of all date fields in the specified dataset.
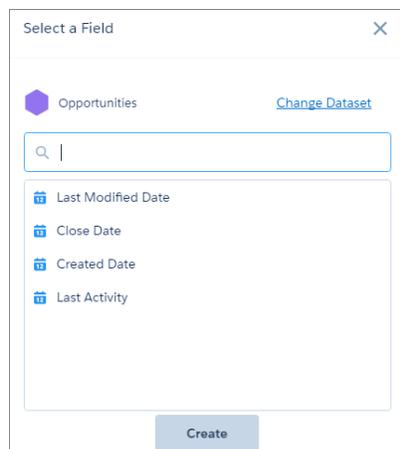
3. To use a different dataset, click **Change Dataset**.

4. Select the date field that you want to filter on.

5. Click **Create**.

Behind the scenes, Tableau CRM creates a query grouped by the field that you selected. Tableau CRM adds the query to the query panel, but you can't reuse it in other widgets. Tableau CRM doesn't apply any selections to the widget, by default.

6. To change the widget and query properties, select the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.

7. In the widget properties, specify whether to show absolute, relative, or preset dates, select the default calendar type, and specify whether users can switch calendar types while viewing the dashboard.
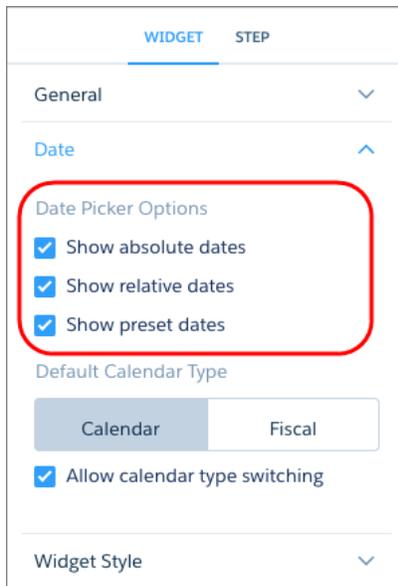
> **Note:** If you enable absolute and relative dates, absolute dates show, by default. To show relative dates, by default, set an initial selection based on a relative date.

8. To show and set the query properties, click **Query** in the right panel.

For example, to enable this widget to filter other faceted widgets, select **Broadcast selections as facets**.

9. To preview your changes to the dashboard, click ⊙ .

10. Save the dashboard.

You can set an initial selection that applies when the dashboard opens. The selected range filters other faceted widgets. While viewing the dashboard, viewers can change the selection to filter the results differently. For information about setting an initial selection, see Set Initial Selections in the Dashboard .

SEE ALSO:

Widget Properties for Tableau CRM Dashboards

Query Properties for Tableau CRM Dashboards

Manage Queries for Widgets

Add a List Widget to Filter Dashboard Results Based on a Dimension

A list filter widget allows dashboard viewers to filter the Tableau CRM dashboard results based on a dimension. For example, a list widget can filter results based on a particular case status or region.

The list widget requires a query that has a single grouping based on a dimension. The values of the grouping appear as the values in the list.
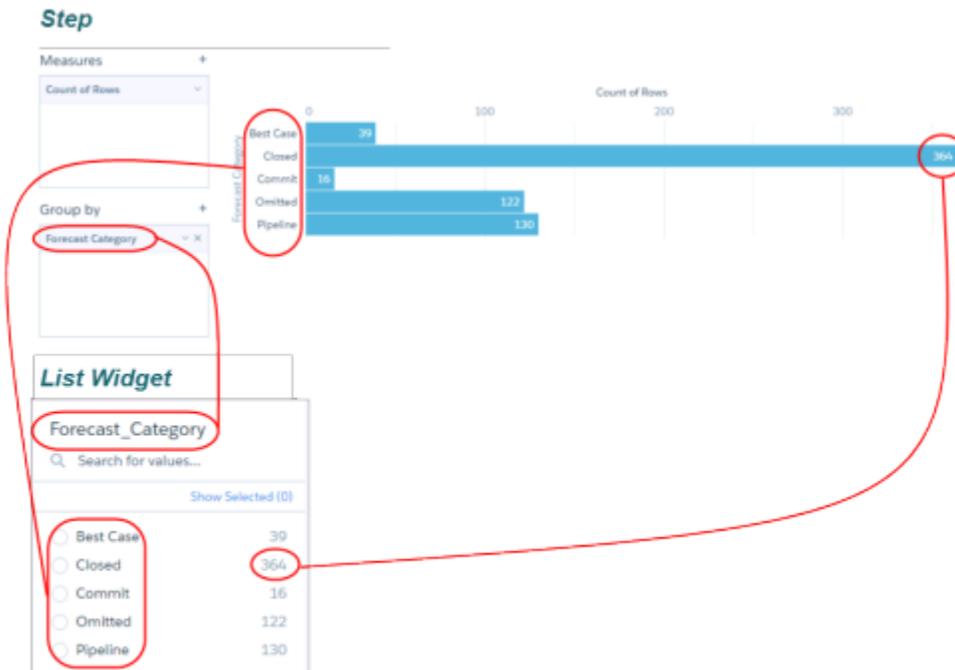
> 🔷 **Note:** You can also create a custom query to specify a user-defined list of values for the list widget. For more information, see Create a Custom Query with User-Defined Values.

1. Drag the list widget to the dashboard canvas.

2. To open the wizard, click the button inside the widget.
   The Select a Field screen appears, showing a list of all dimension fields in the specified dataset.
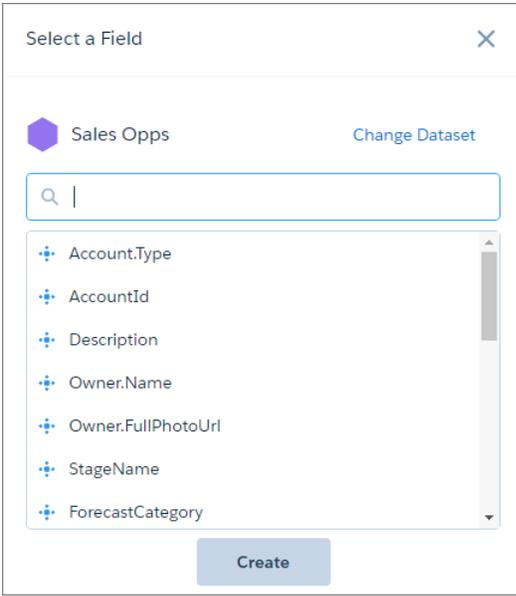
3. To use a different dataset, click **Change Dataset**.

4. Select the date field that you want to filter on.

5. Click **Create**.

💡 Tip: To create a user-defined list of values instead of using values from a dataset, click **Customize**.

Behind the scenes, Tableau CRM creates a query grouped by the field that you selected. Tableau CRM adds the query to the query panel, making it available for use by other widgets in the dashboard. In addition, by default, the list widget is collapsed and Tableau CRM doesn't apply any selections to the widget.
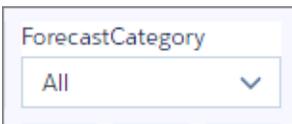
6. To expand the list and show more values, increase the size of the widget.
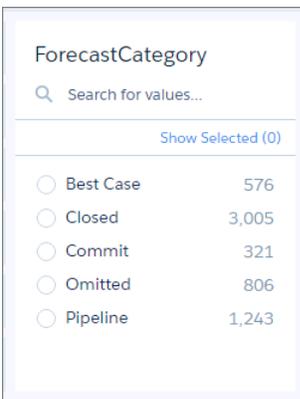
> **Note:** If you don't expand the list, when a user views the dashboard, they can select the drop-down arrow to see the values in the list.

7. To change the widget and query properties, select the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.

8. To customize the appearance of the widget, set the widget properties.

When you set a property, the widget immediately shows the effects of your change.

9. To show and set the query properties, click **Query** in the right panel.

For example, to enable this widget to filter other faceted widgets, select **Broadcast selections as facets**.
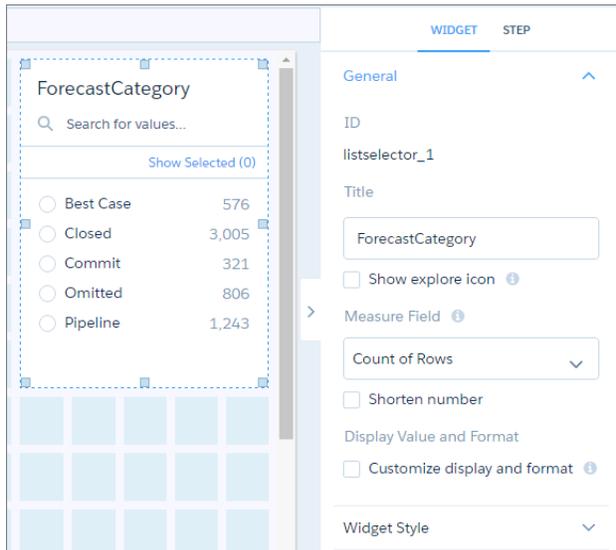
While viewing the dashboard, a user can select a dimension in this widget to filter other faceted widgets. To allow the viewer to make single or multiple selections, configure this widget's query properties. You can specify whether selections are optional or required. You can also set an initial selection that applies when the dashboard opens. While viewing the dashboard, viewers can undo the selection or change the selection to filter the results differently. For information about setting an initial selection, see Set Initial Selections in the Dashboard .

10. To preview your changes to the dashboard, click 👁 .

11. Save the dashboard.

SEE ALSO:

Widget Properties for Tableau CRM Dashboards

Query Properties for Tableau CRM Dashboards

Manage Queries for Widgets

Add a Range Widget to Filter Dashboard Results Based on a Measure

A range widget allows dashboard viewers to filter the Tableau CRM dashboard results based on a range of values for a measure. For example, a range widget can show only opportunities with amounts between $100,000 and $900,000.

The range widget requires a query that has a single measure and no groupings.

1. Drag the range widget to the dashboard canvas.

2. To open the wizard, click the button inside the widget.
   The Select a Field screen appears, showing a list of all measure fields in the specified dataset.
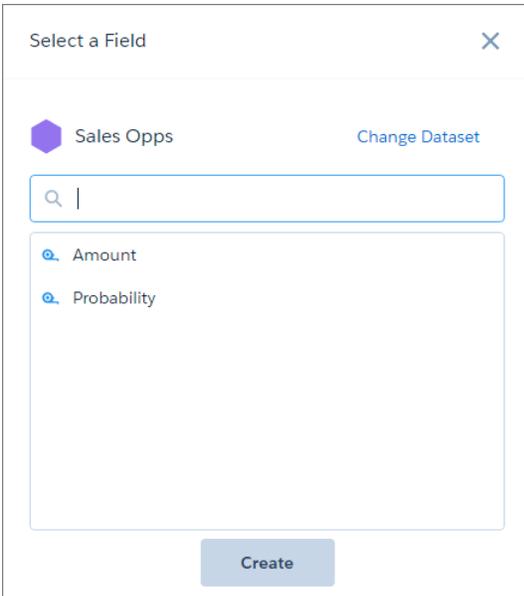
3. To use a different dataset, click **Change Dataset**.

4. Select the date field that you want to filter on.

5. Click **Create**.

   Behind the scenes, Tableau CRM creates a query with the measure that you selected. Tableau CRM adds the query to the query panel, but you can't reuse it in other widgets. Tableau CRM doesn't apply any selections to the widget.

6. To change the widget and query properties, select the widget.

   The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.
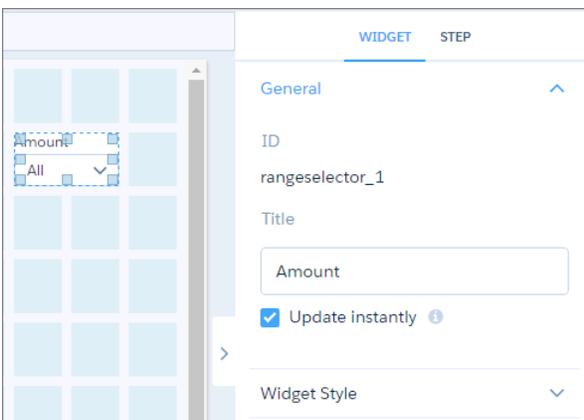
7. To customize the appearance of the widget, set the widget properties.

   When you set a property, the widget immediately shows the effects of your change.

8. To show and set the query properties, click **Query** in the right panel.

   For example, to enable this widget to filter other faceted widgets, select **Broadcast selections as facets**.

9. To preview your changes to the dashboard, click 👁 .

The default range shown is the minimum to the maximum values of the measure.

10. Save the dashboard.

You can set an initial selection that applies when the dashboard opens. The selected range filters other faceted widgets. While viewing the dashboard, viewers can change the selection to filter the results differently. For information about setting an initial selection, see Set Initial Selections in the Dashboard .

SEE ALSO:

Widget Properties for Tableau CRM Dashboards

Query Properties for Tableau CRM Dashboards

Manage Queries for Widgets

Add a Toggle Widget to Filter Dashboard Results Based on a Dimension

A toggle widget allows dashboard viewers to filter the Tableau CRM dashboard results based on a date or dimension. For example, a toggle widget can show results based on a specific record type.

The toggle widget requires a query that has a single grouping based on a dimension. The values of the grouping appear as the toggle options.
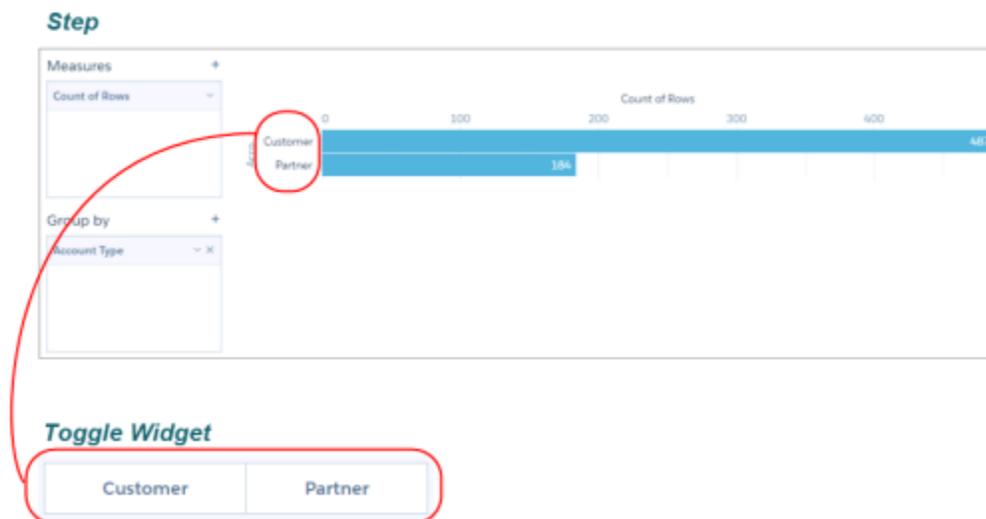
> 📝 **Note:** You can also create a custom query to specify a user-defined list of values for the toggle widget. For more information, see Create a Custom Query with User-Defined Values.

1. Drag the toggle widget to the dashboard canvas.

2. To open the wizard, click the button inside the widget.

The Select a Field screen appears, showing a list of all dimension fields in the specified dataset.
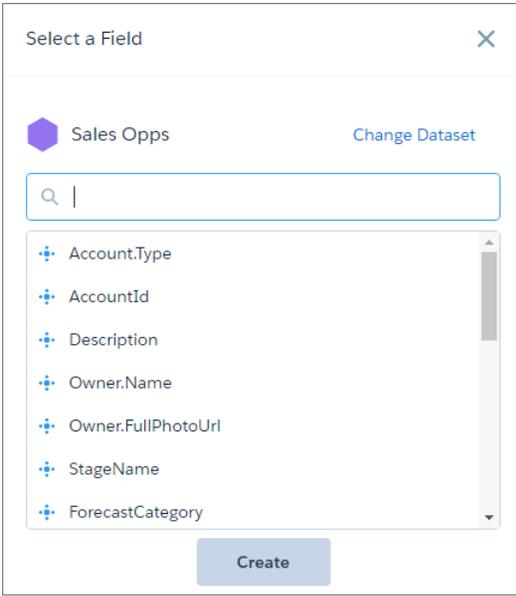
3. To use a different dataset, click **Change Dataset**.

4. Select the dimension field to show each of its values as a toggle option.

5. Click **Create**.

   Behind the scenes, Tableau CRM creates a query grouped by the field that you selected. Tableau CRM adds the query to the query panel, making it available for use by other widgets in the dashboard. Tableau CRM doesn't apply any selections to the widget.

6. If needed, increase the size of the widget to show the display names of all toggle options.
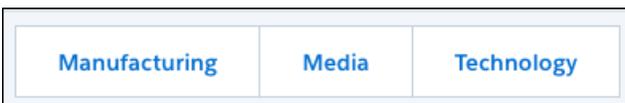
7. To change the widget and query properties, select the widget.

   The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.

8. To customize the appearance of the widget, set the widget properties.

   For example, you can set the styling properties for all toggle options in the "General" section, except the selected option, which you set under the "Selected Tab Style" section. You might choose different styling for the selected toggle option to indicate which option is selected while viewing the dashboard.
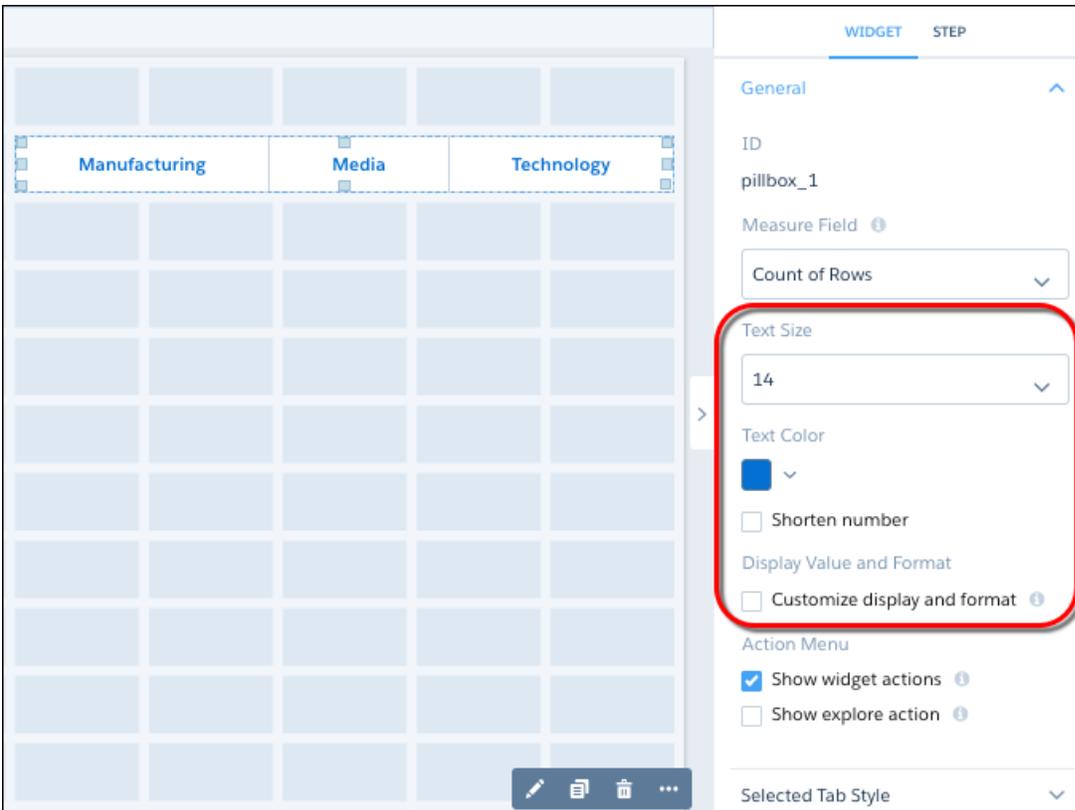
9. To show and set the query properties, click **Query** in the right panel.

For example, to enable this widget to filter other faceted widgets, select **Broadcast selections as facets**.

While viewing the dashboard, a user can select a toggle option in this widget to filter other faceted widgets. To allow the viewer to make single or multiple selections, configure this widget's query properties. You can specify whether selections are optional or required. You can also set an initial selection that applies when the dashboard opens. While viewing the dashboard, viewers can undo the selection or change the selection to filter the results differently. For information about setting an initial selection, see Set Initial Selections in the Dashboard .

10. To preview your changes to the dashboard, click 👁 .

11. Save the dashboard.

SEE ALSO:

Widget Properties for Tableau CRM Dashboards

Query Properties for Tableau CRM Dashboards

Manage Queries for Widgets

Add a Table Widget to Show Record Details and Create Calculated Columns

You can add different types of tables to a Tableau CRM dashboard. To view record-level details, add a values table. A values table can show details like how long a case has been open and who owns it. To create custom columns in a table based on calculations from existing fields, add a compare table. If the underlying query contains at least one grouping, you can create a pivot table.

The steps vary depending on the type of table you create. For a values table, choose which dataset fields to show as columns. For a compare table, select which dataset fields to show as columns and define the calculations for custom columns. For a pivot table, define at least one grouping. The values of the last grouping appear as columns.

1. Drag the table widget to the dashboard canvas.

2. To open the wizard, click the button inside the widget.
   One of the following screens appears.

   - The Choose Data screen appears, showing the most recently used datasets first. Select the dataset in the Create Query tab or select an existing query in the Use Existing Query tab.
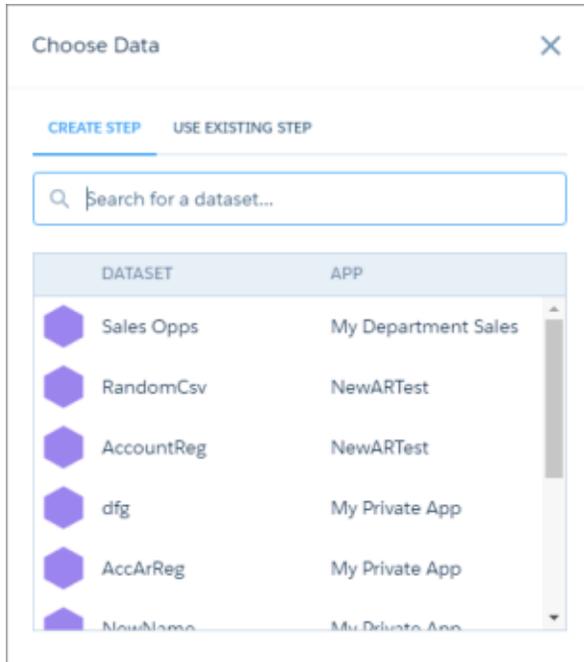
   - The explorer appears, showing a bar chart with the Count of Rows measure. Tableau CRM selects the dataset that you used to create the previous query. To use a different dataset or an existing query, click **Back**.
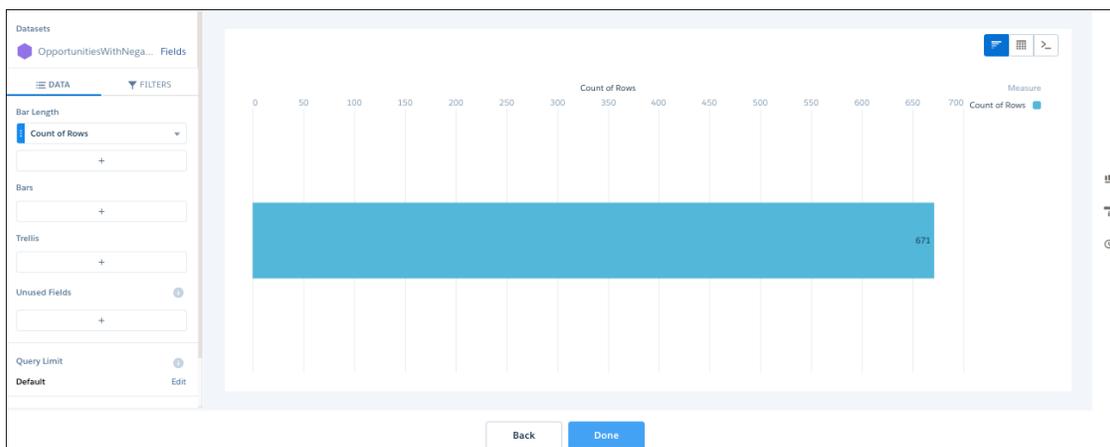
3. Select [grid icon] and then choose the table type.

Values table
Use to show dataset fields as columns. You can add, remove, or reorder columns. You can quick sort on a column by clicking its field header. You can also add filters.

Compare table
Use to add calculated columns to the table. By default, the table shows the previously selected measure as the first column. You can add measures and filters, and group by dimensions. To quick sort on a column, click its field header. To add a calculated column, click the down arrow next to a measure, and select **Clone Column** and then edit the new column.

For more information, see Creating Calculated Columns Using a Compare Table.

Pivot table

Use to show groupings on the x- and y-axes. The last grouping determines the columns, and the other groupings determine the rows. The pivot table shows the first specified measure in the table cells. To quick sort on a column, click its field header.

4. To show totals in a values or compare table, click [icon] and then select **Show Summary**.

5. To increase the number of records that appear in a values table, click [icon] and change the `limit` parameter in the SAQL query.

> **Note:** By default, the query for a values table widget returns up to 100 results. For more information about the `limit` parameter, see Analytics Cloud SAQL Reference.

6. Click **Done**.

The widget shows the table and results. Tableau CRM adds the query to the query panel.

7. To change the widget and query properties, select the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.
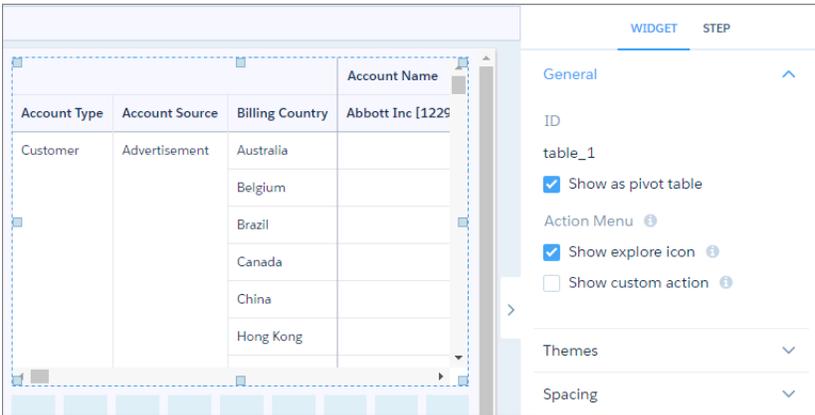
8. To customize the appearance of the widget, set the widget properties.

The widget properties vary based on the table type.

9. To show and set the query properties, click **Query** in the right panel.

10. To preview your changes to the dashboard, click 👁 .

11. Save the dashboard.

While viewing a dashboard, a table's groupings appear in the left columns (1) and the measures appear in the right ones (2). Although the grouping columns are fixed, if needed, you can scroll across the measures. If your table isn't wide enough to display any measures, you won't see the scrollbar.
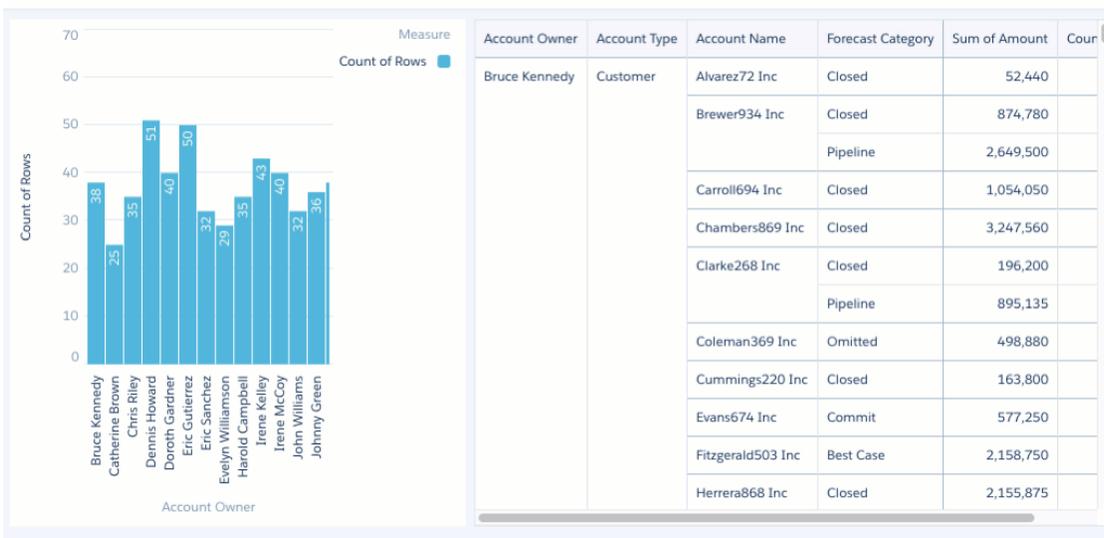
Quick sort isn't available for tables with these underlying queries:

• When the compact-form query has interactions present on the sort order

- When the SAQL query contains union statements or interactions
- When the query is in PIGQL

SEE ALSO:

Add a Container Widget to Create Sections in the Dashboard

Use a container in a Tableau CRM dashboard to group related widgets. For example, you can use a container widget to create a filter panel on the left side of the dashboard to store all selection-based filter widgets. To distinguish the filter panel from the rest of the dashboard, you can apply a background color and border to the container. When you move this widget, the contained widgets also move while maintaining their spacing and alignment.

You can place any widget inside a container widget, except another container widget.

1. Drag the container widget to the dashboard canvas.
   An empty container appears.

2. To change the widget properties, select the widget.

   The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.
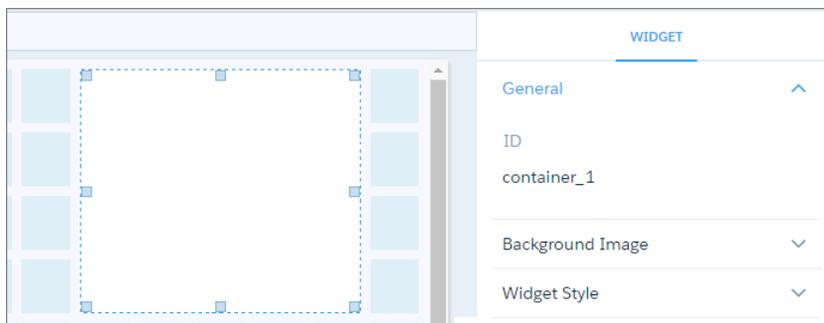
3. To customize the appearance of the widget, set the widget properties.

   For example, apply a background color or image.

4. Drag other widgets inside the container.

   You can increase the size of the container to fit more widgets.

5. To preview your changes to the dashboard, click 👁 .

6. Save the dashboard.

SEE ALSO:

Add an Image Widget to Display Graphics in the Dashboard

You can include company logos to brand the Tableau CRM dashboard, icons to categorize results, or graphics to animate the dashboard.

To upload image files to Tableau CRM, Chatter must be enabled in the org.

You can also add images as backgrounds for container widgets and layouts. Tableau CRM in a web browser supports .svg, .png. jpeg, .jpg, .gif, .bmp. The Tableau CRM mobile app supports all these file types, except .svg.

> 💡 **Tip:** To overlay one image on top of another, place an image widget inside a container widget. To display a background image for the entire dashboard, add a background image in the layout properties. Click ⚙ in the upper left corner to show the layout properties.

1. Drag the image widget to the dashboard canvas.

2. Click **Image** inside the image widget.
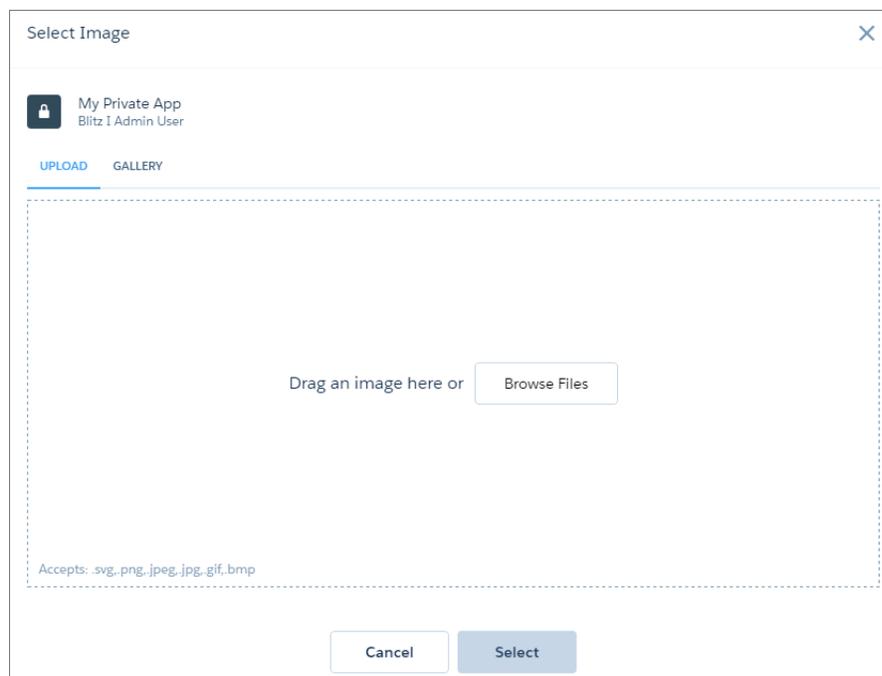
3. To upload a new image, click **Browse Files**. Or, select an uploaded image from the Gallery tab.
   Tableau CRM uploads images to the gallery for the app displayed at the top left corner of the screen. Each image file must be no more than 2 GB.

4. To change the widget properties, select the widget.

   The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.
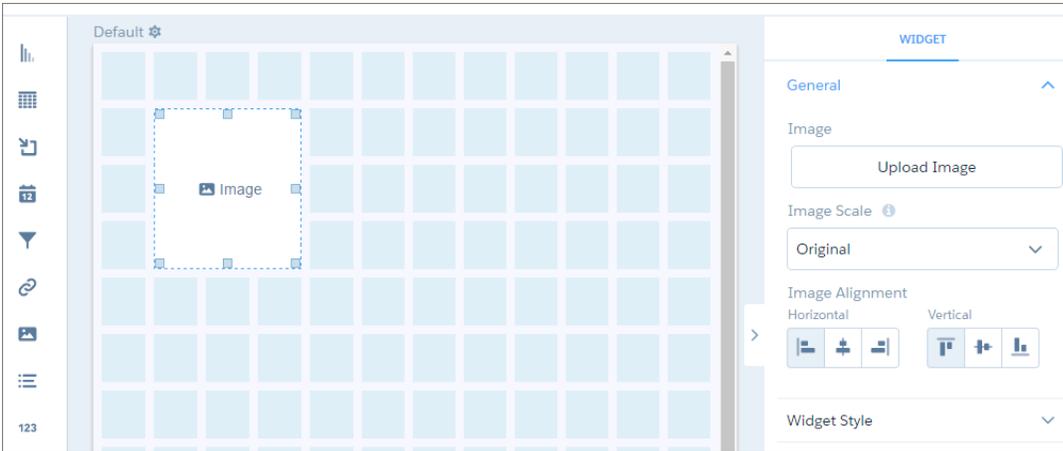
5. To preview your changes to the dashboard, click 👁 .

6. Save the dashboard.

Add Text Widgets to Label Parts of the Dashboard

Label your Tableau CRM dashboard so that users understand what they are looking at. You can also add text that describes the dashboard or provides usage information for widgets.

1. Drag the text widget to the dashboard canvas.

2. To change the widget properties, select the widget.

   The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.
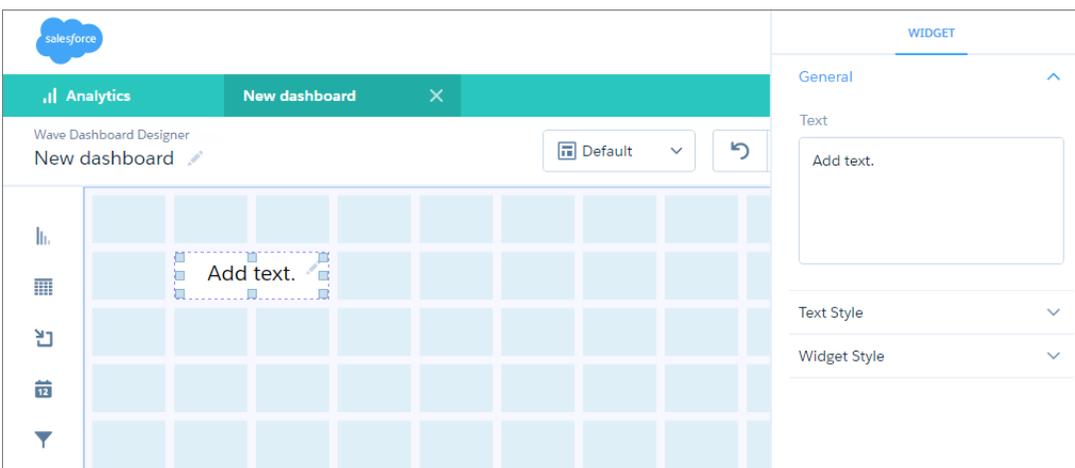
3. To customize the appearance of the widget, set the widget properties.

For example, enter the text that you want to appear in this widget.

4. To preview your changes to the dashboard, click 👁 .

5. Save the dashboard.

SEE ALSO:

[Widget Properties for Tableau CRM Dashboards](#)

Add a Navigation Widget to Access Dashboard Pages

Give dashboard viewers an easy way to navigate dashboard pages with a navigation widget. Reuse your navigation widget for a consistent look across dashboard pages.

For example, here's a navigation widget that provides access to each of the 5 defined pages.
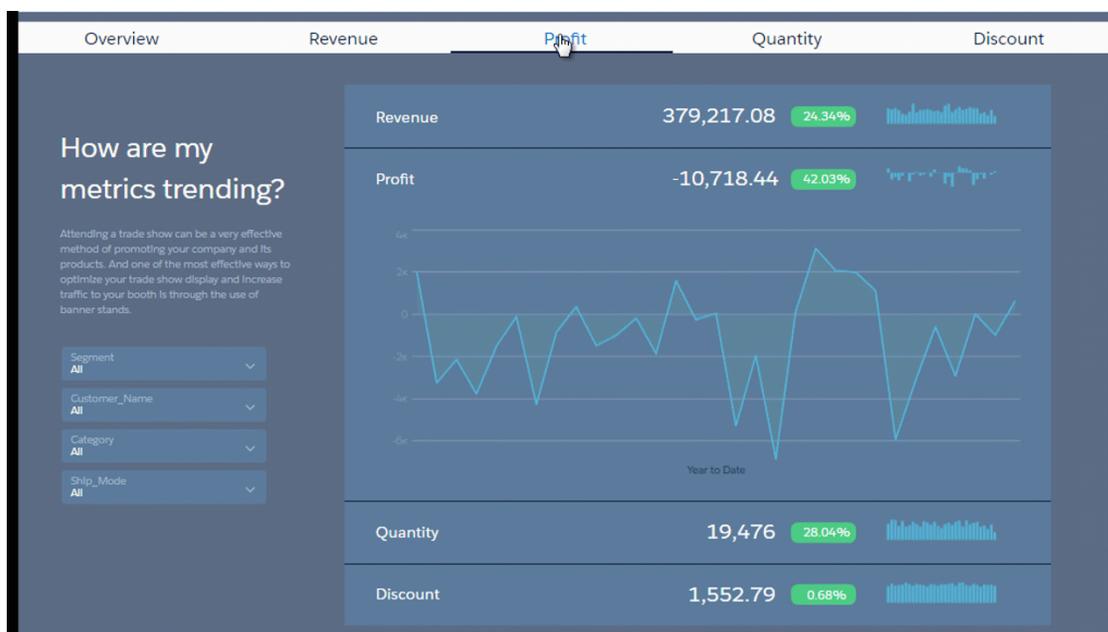
1. Drag the navigation widget to the dashboard canvas.
The widget shows all pages in the dashboard. The underlined page in the widget is the one that you're currently viewing.
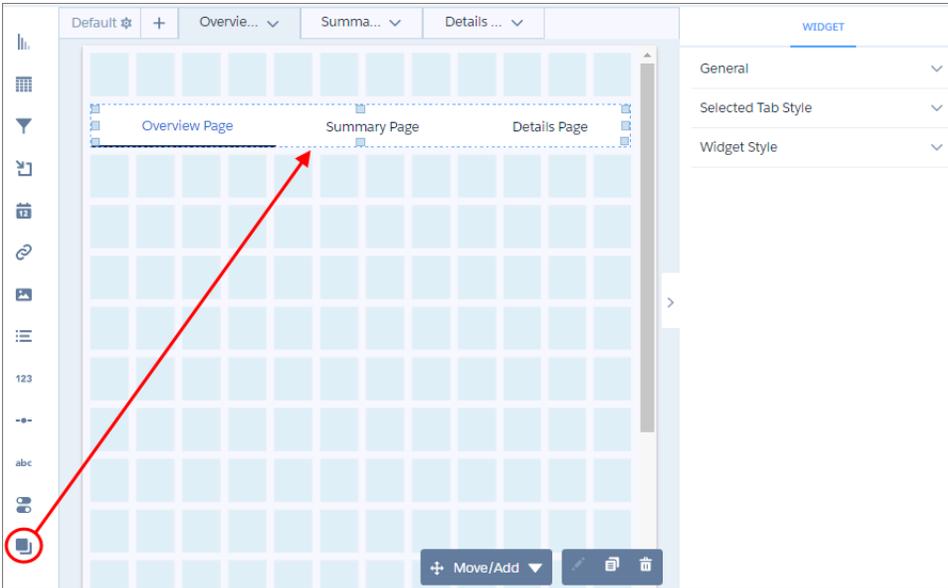
2. To hide a page while viewing the dashboard, click ⌄ and then **Hide from Navigation**.

   Dashboard viewers see only visible pages. The link to the hidden page remains hidden from the navigation widget until you show it again.
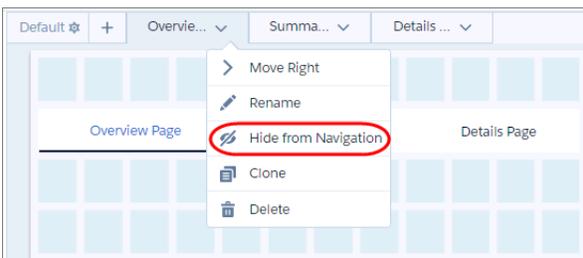
3. From the same page menu, you can also move, rename, clone, and delete pages.

   The order of the page tabs determines the order of the page links in the navigation widget.

   💡 Tip: To avoid truncation, keep page names short.

4. To set the widget properties, select the navigation widget.

   For example, set the styling properties for all page links in the "General" section, except the selected page's link. Set that under the "Selected Tab Style" section. You can choose different styling for the selected page to indicate which page is selected while viewing the dashboard.
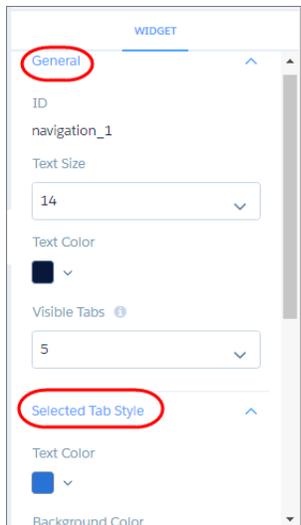
5. To enable navigation from all pages, select the navigation widget that you created on this page and use the **Add to Page** action to add it to other pages.

Using **Add to Page** ensures that all of the styling properties are consistent across pages.
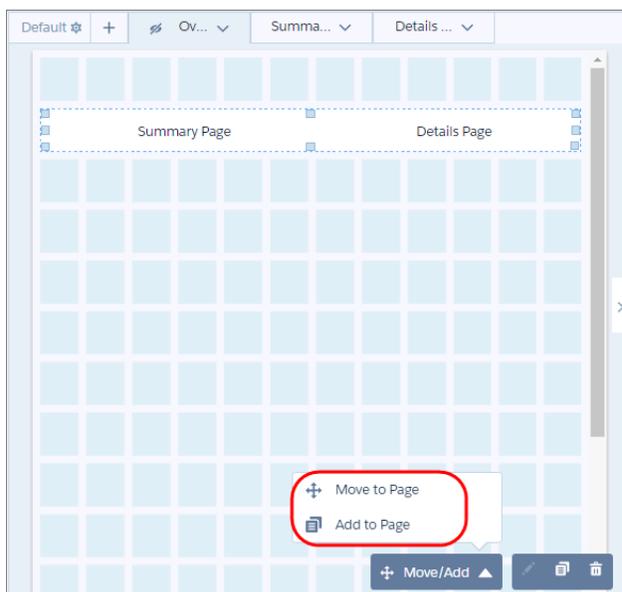
6. To preview your changes to the dashboard, click 👁 .

7. Save the dashboard.

Add a Link Widget to Link to Other Assets

Take dashboard viewers straight to where they need to go with links to specific assets, like dashboards, pages, queries, lenses, and even web pages. You can also use links to create analysis paths for users to follow, helping them navigate your analytics app, or to perform mass quick actions on record lists.

In the Tableau CRM Studio, clicking a link widget in a dashboard opens the link in a new tab. In presentation mode and in the Analytics tab (in Lightning Experience or Salesforce Classic), the link opens in the same window. For embedded dashboards, you can set a link to open in the same or new window.

1. Drag the link widget to the dashboard canvas.

2. To change the widget properties, select the widget.

   The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.
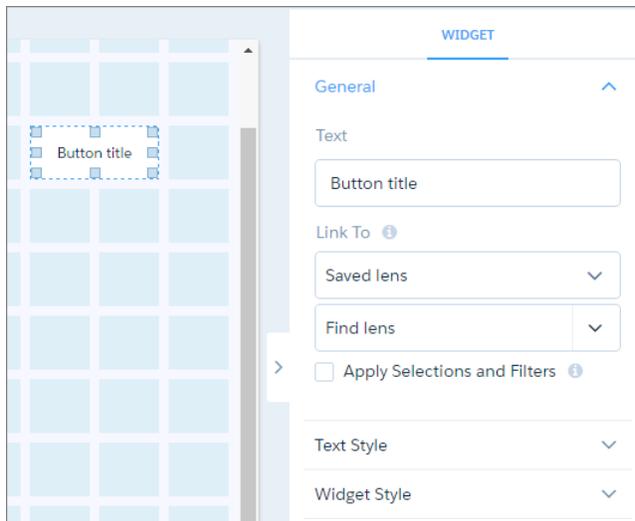
3. In the Text field, enter the label that appears on the link widget.

4. In the Link To field, select whether you want to link to a query, lens, dashboard, page, URL, or mass action, and then select the asset.

   If you link to a dashboard that has multiple pages, you can specify which page to view when the dashboard first opens.

   a. To pass global filters and selection-based filters (also called facets) to a linked Tableau CRM asset, select **Apply Selections and Filters**.

      Tableau CRM passes selections in chart, list, toggle, range, and date widgets as selections in the linked asset if they apply. For example, you select North America in a list widget based on the Region dataset field. Tableau CRM applies that same selection to faceted queries in the linked dashboard that have a grouping based on the Region field in the same dataset.

      📝 Note: A selection is applied to a query in the linked dashboard only if the selected value exists in the initial result set for that query. If your selection isn't applied, try increasing the limit for the query in the linked dashboard so that the value can be found.

      Tableau CRM passes global filters as filters to a linked asset as long as the filters apply. Tableau CRM ignores a global filter if the linked asset doesn't use the dataset that the global filter is defined on. If the global filter is locked and the incoming filter is defined on the same field, Tableau CRM ignores the incoming filter. If it's unlocked, the incoming filter overrides the global filter defined in the dashboard.

      When you open a linked dashboard, the passed global filters appear as external filters. To view the external filters, click 🔽 as highlighted in the screenshot. The External Filters panel shows filters passed from linked dashboards (and filters preconfigured in embedded dashboards). You can remove external filters while viewing the dashboard.

Note: To pass a filter selection to a linked asset, the asset must contain a query based on the same dataset and grouping used to define the filter.
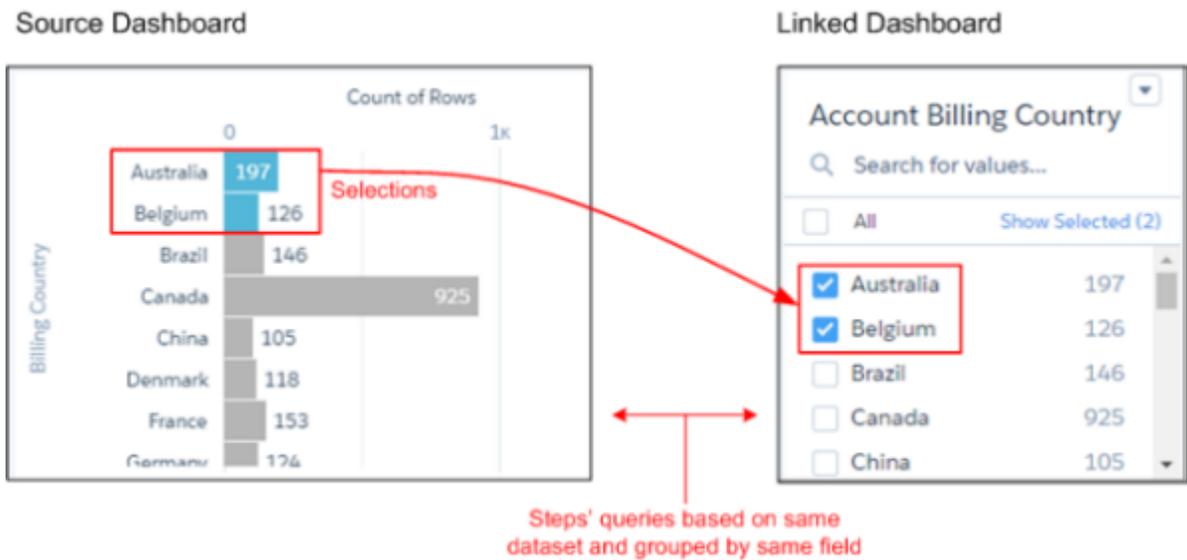
To pass a filter selection based on a custom query to a linked asset, the custom query must be linked to the dataset using a data source connection. Also, the linked asset must contain a query based on the same dataset and grouping used to define the data source connection.

b. If the link is to a URL, enter the following properties.

Destination

Enter the full URL. The URL field supports http or https. It doesn't support passing Salesforce parameters in the URL, like Salesforce object or Tableau CRM dataset fields.

Link Tooltip

Enter a description of the link or the URL so users know where the link takes them.

c. If the link is for a mass action, see Configure Mass Quick Actions on Multiple Salesforce Records from Tableau CRM Dashboards on page 1371 for configuration steps.

5. To customize the appearance of the widget, set the widget properties.

6. To preview your changes to the dashboard, click 👁 .

7. Save the dashboard.

Widget Properties for Tableau CRM Dashboards

Widget properties define how widgets appear in the Tableau CRM dashboard. For example, you can apply labels, backgrounds, and borders. To make widgets look consistent, you can set up default widget properties in the layout and then assign them to each widget. Widget properties vary based on the widget type and the designer used to create them. The properties listed here apply only to widgets created in the dashboard designer.

Tableau CRM groups widget properties into sections. After the first section ("General"), the remaining sections are listed in alphabetical order.

General

Property	Description
Chart Type	Select the type of chart to visualize the data.
Show as pivot table	Indicates whether the table is pivoted. A pivot table requires the underlying query to have at least one grouping. Tableau CRM pivots the table on the last defined grouping. Disabled, by default.
Show widget actions	Hide a widget's action menu, which includes options to download, share, and explore the widget's results. Hide the menu to hide these options or when the action dropdown arrow is too distracting in smaller widgets.
Show explore action	Allow dashboard viewers to click a link in the widget to open the underlying query in explorer. This option appears when you select **Show widget actions**. 📝 Note: This option only applies to widgets based on queries in compact form. You can't explore widgets that are built on SAQL form queries or multiple datasets.
Title	Title of the widget.
Subtitle	Subtitle of the chart. Appears below the title in a smaller font.
Title Alignment	Horizontal alignment of the title.
Title Color	Color of title.
Title Size	Font size of title.
Title Separator Color	Color of the line that separates the widget title from the global filters.
Filters Per Row	Number of global filters to show per row in the global filter panel.
Theme	Set a color scheme for the chart.
Update instantly	Instantly update the dashboard results based on a filter selection. Otherwise, you have to click **Update** to apply the filter.

Property	Description
Measure Field	The measure to show in the widget. You can use this field to specify the measure to show when multiple measures are defined in the underlying query. You can also use this option to not show the measure in the widget.
Shorten number	Display shortened numbers, rounded to the nearest tenths place. For example, when shortened, 1,111,556 appears as 1.1 M, 999.95 appears as 1.0 K, and 111,045 appears as 111.0 K.
Text	Widget label.
Link To	Link to a dashboard, lens, query, page, or URL. If the dashboard contains no pages, the option to link to a page doesn't appear.
Page	Specify the page that you want the link widget to link to.
Destination	The URL of the page that the link widget links to.
Link Tooltip	Enter a tooltip that appears when the user hovers over the link. For example, if the link is to a website, enter the URL as the tooltip so the user knows where the link takes them.
Customize display and format	Customize the display values for a list or toggle widget. The widget can display values from multiple fields. For example, a list widget can show sales rep and region, with values like SalesRepA-West, SalesRepB-West, and SalesRepC-North—allowing you to filter on both fields. You can also display measures as part of the value, and format the display. Click **Add field...** dropdown, select a grouping or measure defined in the query. You can insert more characters directly into the editor box to format the display. For instance, you can add parentheses to enclose the measure value as shown here.
Apply Selections and Filters	Tableau CRM passes selections in chart, list, toggle, range, and date widgets as selections in the linked asset if they apply. For example, you select North America in a list widget based on the Region dataset field. Tableau CRM applies that same selection to faceted queries in the linked dashboard that have a grouping based on the Region field in the same dataset. Tableau CRM passes global filters as filters to a linked asset as long as the filters apply. Tableau CRM ignores a global filter if the linked asset doesn't use the dataset that the global filter is defined on. If the global filter is locked and the incoming filter is defined on the same field, Tableau CRM ignores the incoming filter. If it's unlocked, the incoming filter overrides the global filter defined in the dashboard.
Image	The file name of the image. The image must be available in the gallery of the Tableau CRM app that contains the dashboard. This property applies to image widgets created during or after Spring '17.
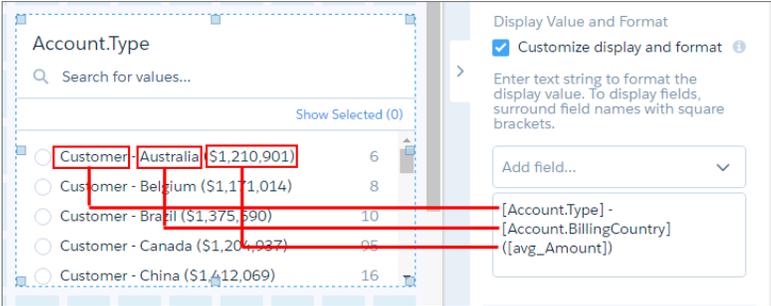
Property	Description
Image ID	15-character ID of the image. The image must be uploaded to the Documents tab of Salesforce and the Externally Available Image option must be enabled. This property applies to image widgets created before Spring '17.
Image Scale	Indicate how to fit the image.
Image Alignment	Horizontal and vertical alignment of the image.
Show custom action	Show the widget properties that you set to show custom bulk actions on a table widget's action menu.
Custom Action Label	Display label for the button in the table widget's action menu. The dashboard viewer clicks the button to execute the action.
Visualforce Page Name	Name of the Visualforce page that defines the bulk action.
Visualforce Namespace Prefix	Optional. Namespace prefix of the Visualforce page that defines the bulk action.

Background Image

Property	Description
Image	The file name of the image. The image must be available in the gallery of the Tableau CRM app that contains the dashboard. This property applies to container widgets created during or after Spring '17.
Image ID	15-character ID of the image. The image must be uploaded to the Documents tab of Salesforce with the Externally Available Image option enabled. This property applies to container widgets created before Spring '17.
Image Scale	Indicate how to fit the image.
Image Alignment	Horizontal and vertical alignment of the image.

Band Colors

Property	Description
Low Band	The color range for the low band.
Medium Band	The color range for the medium band.
High Band	The color range for the high band.

BreakPoints

Property	Description
Min	Minimum value of the low color band.
Medium	The value separating the low and medium colors.

Property	Description
High	The value separating the medium and high colors.
Max	Maximum value of the high color band.

Center Axis, Left Axis, X-Axis, Y-Axis

Property	Description
Start axis at zero	Start the axis at 0.
Show axis	Show the axis.
Show title	Show the title of the axis.
Title	Enter title of the axis. The default title is the name of the dimension for x-axis or names of the measures for y-axis. Multiple measures appear as a comma-separated list.
Reference Line	Show a reference line, which you can use to show a threshold, boundary, or reference point in the chart. For x-axis, it's a vertical line. For y-axis, it's a horizontal line.

Chart Details

Property	Description
Compute total	Computes the total based on all segments of the waterfall or stacked waterfall chart.
Show values in chart bars	Show the measure value of each bar in the chart if there is enough space to do so. Available space will is determined by fit mode of axis, axis type (such as *date-time*), available space for the widget, and other factors.
Normalize values	Show stacked groups as a percentage of the total.

Axis Mode	Specify how multiple measures appear in the chart.
	
	You can select one of the following values:
	
	Single Axis
	    Show all measures on the same axis.
	
	Dual Axis
	    Show the first measure on its own axis and the remaining measures on a separate axis.
	
	Small Multiples
	    Show each measure in a separate graph.

Auto fit	Fit entire chart inside the widget. Else, if needed, scroll to see the remainder of the chart.
	
	You can select one of the following values:
	
	None
	    No scaling is applied. Scroll bars appear for charts larger than the widget.
	
	Fit
	    Scale the chart to exactly fit the widget.

Property	Description
	Preserve labels Scale the chart to fit the widget while preventing labels from overlapping. Scroll bars appear for charts larger than the widget.
Handling Missing Values	Specify whether to connect data points regardless of missing data. You can select one of the following values: **connect** Show a continuous graph, ignoring missing values. **disconnect** Show gaps in the graph where values are missing.
Fill area	Fill area under the timeline chart.
Show points	Show data points on the line.
# of stem segments	The number of values that appear in the funnel's stem.
Icon	Icon used in the ratings chart.
Number of icons	Number of icons that appear for each rating.
Show value as	Specify how to show values. You can select one of the following choices: **None** Do not show values. **Number** Show as numbers. **Compact Number** Show compact form of numbers, summarized using letter suffixes. **Percentage** Show as percentages of the total.
Show max value	Show the maximum measure value in each rating.
Show values on left	Show the measure value on the left of each rating. Default is on the right.
Show measure title	Show the title of the measure in the chart.
Show measure axes	Show the axes for measures in the chart.
Show total in title	Show the total in the measure title.
Show total in center	Show the total of the measure in the center of the donut chart.
Center Size	The percentage of the donut chart or polar gauge used to create the center circle.
Map Type	Select the type of map.

Property	Description
Colors	Depending on the chart type, specifies the colors of the low and high values, or colors for the start, total, positive, and negative values.

Date

Property	Description
Default Date Mode	Specify whether the date widget displays absolute or relative dates, by default.
Allow date mode switching	Indicates whether the dashboard viewer can switch between absolute and relative dates in the date widget.
Default Calendar Type	Specifies whether the date widget displays dates using the calendar or fiscal year, by default.
Allow calendar type switching	Indicates whether the dashboard viewer can switch between the fiscal and calendar year in the date widget.

Filter Style

Property	Description
Field Color	Color of the field name.
Value Color	Color of the field value.
Background Color	Color of the filter background.
Border Color	Color of the filter border.
Border Radius	Radius of filter border.

Legend

Property	Description
Show legend	Show the chart legend.
Show legend header	Show legend header.
Show legend inside chart area	Show legend inside the chart area to reduce space required for the legend. The legend might overlap with the chart.
Position	Location of legend inside the chart widget.

Onboarding

Property	Description
Resource Title	Specify a title for the frame that presents your resource.

Property	Description
Resource URL	Specify the URL for your video or webpage on a site that doesn't prevent embedding. For best results with a YouTube video, specify the Embed URL. To get the Embed URL in YouTube, click **Share** and then **Embed**. The URL must have this format: `https://www.youtube.com/embed/xxxxxxxxxx` 📝 Note: The Onboarding property loads content from any URL into a Tableau CRM inline frame. If the URL allows execution within an inline frame, then all content at the URL becomes accessible to dashboard viewers. Salesforce is not responsible for third-party URLs.

Table Options

Property	Description
Column Size	You can set the table column size to: **Fit to Data** Dynamically set the column widths based on the length of the data currently visible in the table. You can also set a minimum and maximum width to ensure that the column isn't too wide or narrow. **Fixed Width** Set each column to the same width. **Fit to Widget** Fit the columns to the size of the widget. Use this setting to ensure that no empty space appears at the right side of the table, as shown here.
Column Width	The width of each column in the table. Configure the widths to minimize whitespace and ensure widget alignment in a dashboard. If you selected **Fit to Data** in Column Size, specify the minimum and maximum column widths to ensure that the columns aren't too wide or narrow.
Show Totals	Shows the totals for all measures across the top of a compare or values table. 📝 Note: The Show Totals option appears in the widget properties only if the Show Totals option is set in the query.
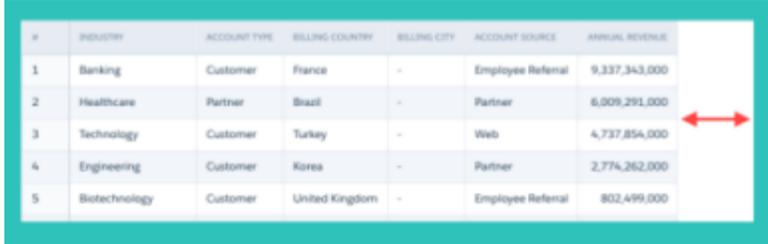
Text Style

Property	Description
Text Color | Color	Color of the text.
Text Size | Size	Size of the text.
Number Color	Color of the number.
Number Size	Size of the number.
Title Color	Color of the title.
Title Size	Size of the title.
Alignment	Horizontal alignment of the text.

Trellis

Property	Description
Show as trellis	Create separate graphs based on groupings.
Type	Select the type of trellis, which affects how the graphs appear. You can select one of the following values: **Horizontal** Separate horizontally, with a graph for each value in the first grouping. **Vertical** Separate vertically, with a graph for each value in the first grouping. **Matrix** Display graphs in a table, with the first grouping in columns and the second grouping in rows. **Wrap** Display graphs line-by-line for each value in the first grouping, wrapping at the value set in the per-line setting.
Number of charts per line	Number of charts per line when the trellis wraps to multiple lines.

Widget Style

Property	Description
Use defaults	Use the default widget properties defined in the dashboard properties.
Background Color	Background color of the widget.
Border	Border around the widget.
Border color	Color of the widget border.
Border Width	Width of the widget border (in pixels).

Property	Description
Border Radius	Roundness of the corners of the widget border.

Manage Queries for Widgets

Queries return results that are displayed in widgets. For example, a number widget displays the result of a calculation that is defined in a query. Queries can be built on a data source, like a dataset or a Salesforce object. They can also be "custom queries" created with user-defined values.

Watch a Demo: ⊙ Use Steps to Build Wave Dashboard Widgets (English Only)

To create a query, you can clip a lens from an exploration, use the widget wizard, or create a query manually in dashboard designer. The widget wizard creates a basic query that meets the minimum requirements for the widget. If you need a more robust query, like one that has customized SAQL, manually create your own. The method you choose depends on how much flexibility you need.

Behind the scenes, Tableau CRM assigns a unique ID to each query in the dashboard. Because query IDs, like "Amount_1," are not always descriptive, add a query label when you create the query. For example, to clarify the "Amount_1" query, you can add the label "Won Opportunity Amount." Tableau CRM generates the query ID based on the label. After creating the query, you can't change the ID, unless you edit the JSON.

To improve dashboard performance, minimize the number of queries in a dashboard. When possible, reuse a query for multiple widgets. You can reuse any query, except a range or date widget's query.

Queries and Custom Queries

Widgets that display data require queries. A widget displays the results returned by a query. For example, a query calculates the average opportunity amount, grouped by region. The widget displays the results in a bar chart format with each bar showing the average amount for a region.

Create a Query with the Explorer

Manually create a query from the dashboard designer when you're concerned about the results it returns and not the widget that displays those results. For example, you might build a query that calculates three measures. You can display all measures in a combo chart and display each measure in a separate number widget as well. You can reuse queries in multiple widgets.

Create a Query with the Query Editor

Use the query editor to create or customize SAQL or SOQL queries. The query editor gives you additional flexibility over queries created by widget wizards or in the chart and table modes of the explorer.

Create a Custom Query with User-Defined Values

Use custom queries to apply user-defined values to dimensions, measures, and limits of other queries. For example, you can create a toggle widget that filters case records based on the following user-defined values: High Priority Cases, Medium Priority Cases, and Low Priority Cases. To enable these custom values to filter your data, you map them to your data using a binding.

Find a Query

You can find all queries defined in the dashboard in the query panel. The panel organizes queries into categories and provides a search bar to help you locate your queries faster.

Edit a Query

When you edit a query, you can change the details, like the query label. These changes affect all widgets that use the query. You can also change the visualization properties, like chart type. Depending on the method that you use to edit the query, Tableau CRM handles visualization changes differently.

SEE ALSO:

Queries and Custom Queries

Widgets that display data require queries. A widget displays the results returned by a query. For example, a query calculates the average opportunity amount, grouped by region. The widget displays the results in a bar chart format with each bar showing the average amount for a region.

In the dashboard designer, a "lens" and a "query" are similar, but not the same. Unlike a query, a lens can be saved as a standalone asset in an app. For example, you can save a lens that displays opportunity trends as a timeline chart. We use "query" to refer to the underlying calculation used to populate a widget in a dashboard.

You can build a widget using one of the following types of queries.

Query

Most queries are this type. The type of query is based on the data source. For instance, Tableau CRM can execute a SAQL query against one or more datasets or a SOQL query against a Salesforce object. Tableau CRM provides multiple ways to create queries.

You can use the Chart mode or Table mode of the explorer on a single dataset. You can calculate measures, group results by dates and dimensions, and filter records based on dataset fields. Here's an example of an explorer-built query that populates a chart widget.
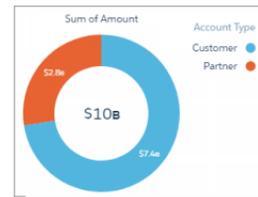
To create customized SAQL or SOQL queries, use the Query mode of the explorer. For example, in Query mode, you can create a query based on multiple datasets. The following chart widget shows the results from the SAQL query. For more information about SAQL queries, see the Analytics Cloud SAQL Reference.
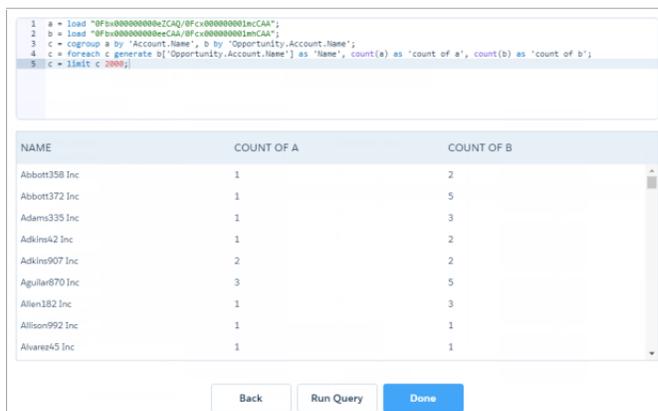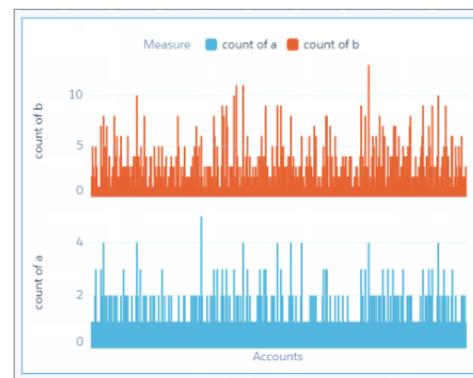
Tableau CRM creates different query types based on the widget type. Some query types have a limit on the number of results that they can return. For more information about all query types, including those not available in the explorer, see the Analytics Cloud Dashboard JSON Reference.

Custom Query

A custom query is more rare. It doesn't run calculations on your data. Instead, the query provides a static list of user-defined values. For example, a toggle widget can show static values "$," "#" and "AVG." To make the toggle widget interact with other widgets in the dashboard, use data source linking to connect the custom query with other faceted queries. Or bind the static values to the queries of other widgets in your dashboard. For instance, you can bind "$" and "#" to show different measures: "$" to show the value of opportunities, and "#" to show the number of opportunities. In this example, "$" is selected in the toggle widget, and the dashboard shows the value of opportunities.
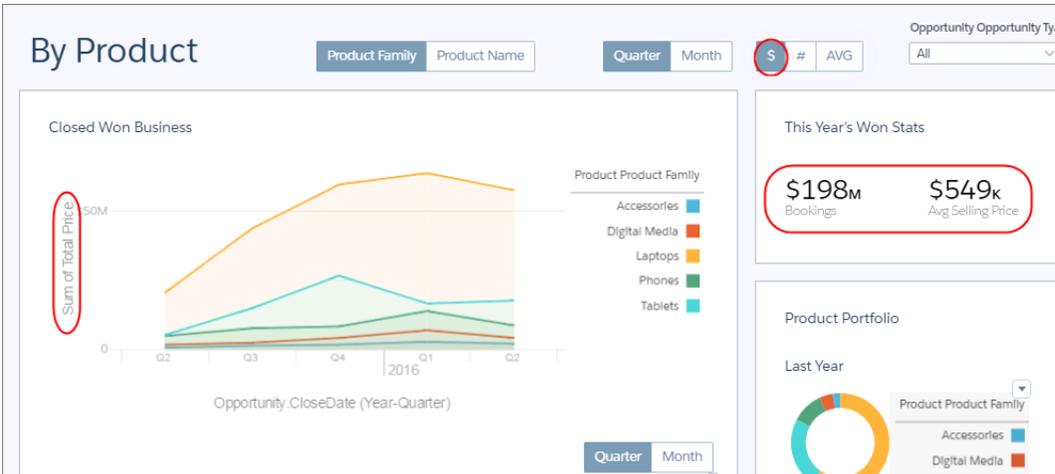
For more information, see Create a Custom Query with User-Defined Values.

SEE ALSO:

Manage Queries for Widgets

Create a Query with the Explorer

Create a Query with the Query Editor

Create a Custom Query with User-Defined Values

Create a Query with the Explorer

Manually create a query from the dashboard designer when you're concerned about the results it returns and not the widget that displays those results. For example, you might build a query that calculates three measures. You can display all measures in a combo chart and display each measure in a separate number widget as well. You can reuse queries in multiple widgets.

For information about clipping a lens to create a query, see Clip a Lens to a Dashboard.

1. To open the queries panel, click an empty space in the dashboard canvas in the dashboard designer.
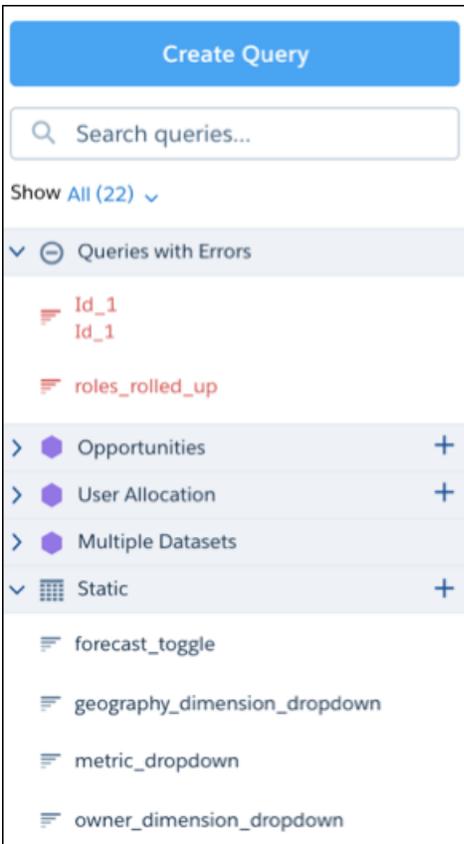
2. If the queries panel has existing queries organized in sections, click ＋ next to the a section to create a query of that type.

   For example, click ＋ next to the Static section to create a custom query. This feature is available for all section types, except `Apex`, `Soql`, and multiple-dataset queries.

3. Click **Create Query**.

4. Click **Dataset** for a Tableau CRM dataset or **Salesforce Direct** for a Salesforce object.

5. Select the dataset or Salesforce object that contains the data that you want to include in the query. Enter the name in the search box if your data source isn't easy to find.
   The explorer appears showing the default measure, Count of Rows.

6. In the explorer, click **Untitled Query** and enter the query label.
   Tableau CRM creates the query ID from the label. After you create the query, you can't change the query ID. Tableau CRM refers to queries by their ID so dashboards don't break if you change the labels.

7. To add a measure, click **+** under the Bar Length field.
   The Count of Rows measure shows, by default. To change a measure, click the measure and then choose a new one.
   You can add as many measures as needed.

8. To group the measures by a date or dimension, click **+** under Bars, and then select the field.

You can add multiple groupings. To change a grouping, click the grouping and select a new field. To reorder or delete a grouping, click the down arrow to the right of the grouping and then select the option.

9. To add a filter, select **Filters**, click **+** under Filters, and then select the field to filter by.

You can add multiple filters.

10. To change the chart or table type for the query, click ☰ or ▦ , respectively.

11. To set the widget properties at the query-level, click 🛢 .
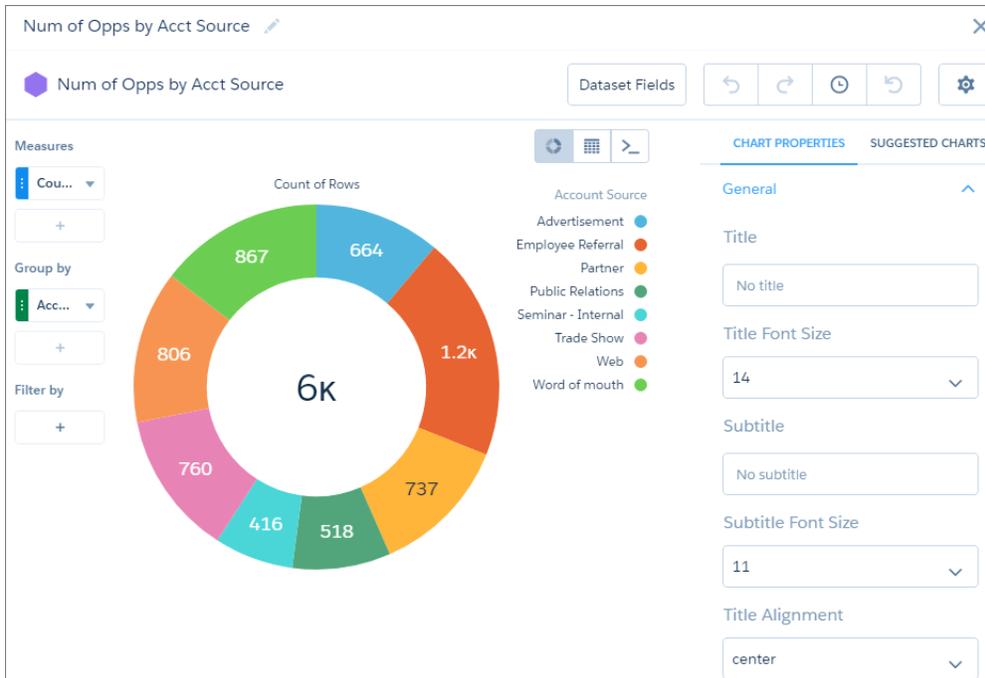
12. To write a custom SAQL or SOQL query, click >_ .
For more information, see Create a Query with the Query Editor .

13. Click **Done**.
Tableau CRM adds the query to the query panel, making it available for use by other widgets in the dashboard.

> 💡 **Tip:** If you drag the query to the dashboard designer canvas, Tableau CRM displays the query results as the chart specified while creating the query. To display the widget differently in the dashboard than what was specified at the query level, select the widget in the dashboard and then set the widget properties. The widget properties override the query settings.

14. Save the dashboard.

Tips for Salesforce Direct Data Queries

Direct Data is an option for querying live Salesforce data without having to set up replication or create datasets. With Direct Data, you can query Salesforce objects in the dashboard designer's explorer and edit Salesforce Object Query Language (SOQL) queries in the query editor.

SEE ALSO:

Query Properties for Tableau CRM Dashboards

Tips for Salesforce Direct Data Queries

Direct Data is an option for querying live Salesforce data without having to set up replication or create datasets. With Direct Data, you can query Salesforce objects in the dashboard designer's explorer and edit Salesforce Object Query Language (SOQL) queries in the query editor.

Direct Data is available only in the dashboard designer. Click **Create Query**, and then select the Salesforce Direct tab. Select an object and the dashboard designer's explorer opens with the object's data ready to explore.
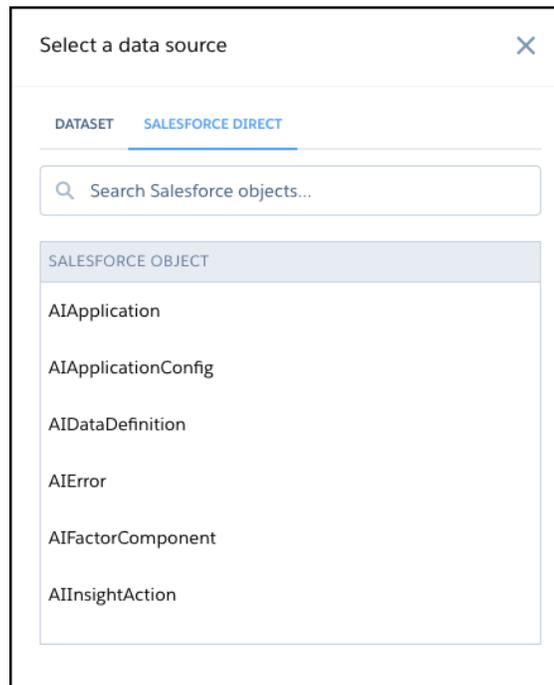
(!) Important: Direct Data widgets aren't automatically refreshed but are updated when the dashboard is refreshed.

Take the following into consideration.

- For best performance, use Direct Data only where data changes often and scheduled dataflows aren't frequent enough. Queries on Salesforce objects with a large number of records could increase query time. Use filters to limit the data that's queried. For example, rather than querying all cases opened, query only cases opened within a defined period of time.

- Direct Data is accessible only in the dashboard designer. After a Direct Data widget is added to a dashboard, you can select **Explore** from the widget's dropdown and open a Direct Data exploration in a new tab. However, saving custom queries on Direct Data lenses isn't supported.

- In the explorer, these features aren't available: compare tables; totals and subtotals in tables; conversational querying; suggested charts; record actions; drilling into data by hour, minute, or second; and applying filters based on aggregated measures or Boolean logic.

- In the dashboard designer, only chart, number, and table widgets are supported, and these options aren't available: creating a query from an sObject in the queries panel (+ button); broadcasting or faceting between SOQL queries; data source linking; record actions; and global filters.

- These sharing options aren't available from a Direct Data widget: Download as CSV, Download as Excel, and Export to Quip.

- These actions aren't available from a Direct Data widget: Annotate, Set Notification, and Subscribe.

- Direct Data queries are run in user context, and the current user's permissions, field-level security, and sharing rules apply.

Maximum limits on Tableau CRM API calls and on queries apply to Direct Data.

- Maximum concurrent Tableau CRM API calls per org is 100.
- Maximum Tableau CRM API calls per user per hour is 10,000.
- Maximum concurrent queries per organization is 50 per platform.
- Maximum concurrent queries per user is 10.

SEE ALSO:

> Get Started with Data Integration

Create a Query with the Query Editor

Use the query editor to create or customize SAQL or SOQL queries. The query editor gives you additional flexibility over queries created by widget wizards or in the chart and table modes of the explorer.

📝 Note: If you modify a query using the query editor, you can no longer add dimensions, measures, and filters if you switch back to the table or chart view. However, you can still change the visualization parameters.

1. To open the queries panel, click an empty space in the dashboard canvas in the dashboard designer.
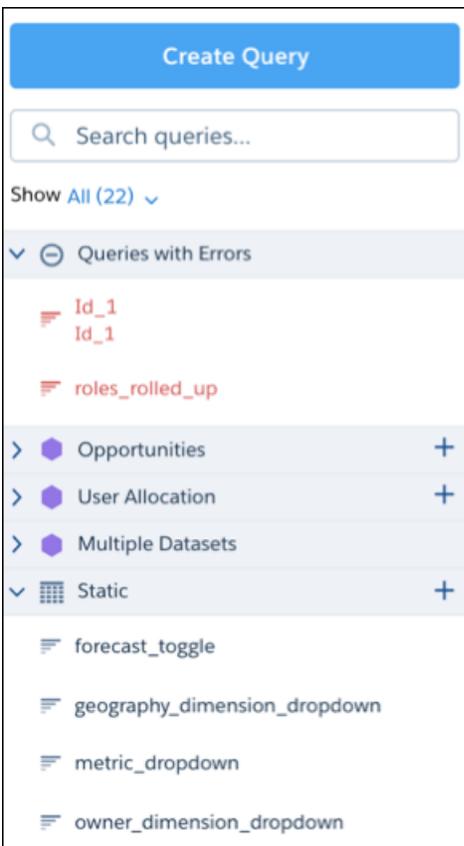
2. Click **Create Query**.
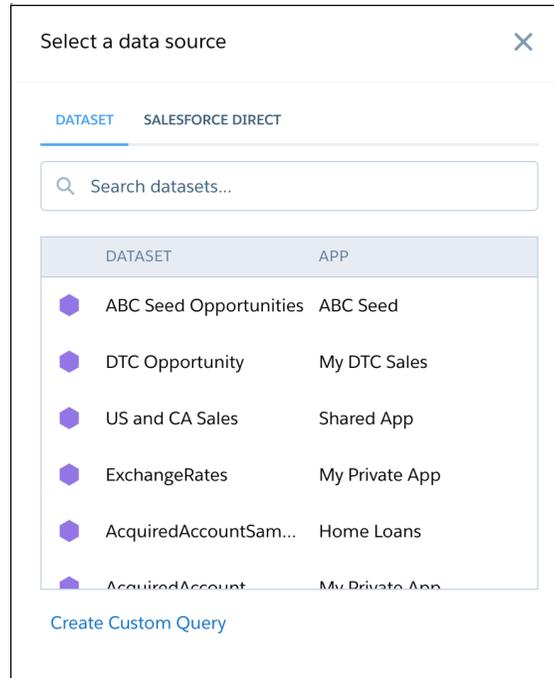
3. Click **Dataset** for a Tableau CRM dataset or **Salesforce Direct** for a Salesforce object.

4. Select the dataset or Salesforce object that contains the data that you want to include in the query. Enter the name in the search box if your data source isn't easy to find.
   The explorer appears showing the default measure, Count of Rows.
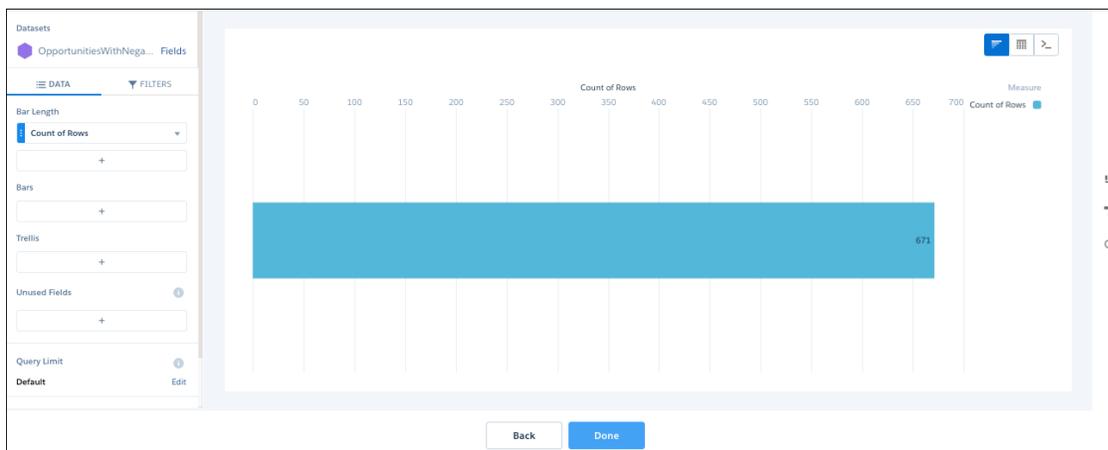
5. In the explorer, click **Untitled Query** and enter the query label.
   Tableau CRM creates the query ID from the label. After you create the query, you can't change the query ID. Tableau CRM refers to queries by their ID so dashboards don't break if you change the labels.

6. To write a custom SAQL or SOQL query, click [icon].
   The query editor appears, showing the code for the current query.

7. Edit the query as needed.

For SAQL, the editor provides contextually smart, auto-completion suggestions. For more information about SAQL queries, see the *Analytics Cloud SAQL Reference.*

8. Click **Run Query** to validate the query.

The **Done** button becomes active if the query is valid.

9. Click **Done**.
Tableau CRM adds the query to the query panel, making it available for use by other widgets in the dashboard.

10. Save the dashboard.

Tips for Working with SAQL Queries in the Query Editor

Queries created or edited in the query editor have unique characteristics compared to other queries, like those queries created in the explorer. Review the following tips for working with queries created in the query editor.

SEE ALSO:

Query Properties for Tableau CRM Dashboards

Tips for Working with SAQL Queries in the Query Editor

Queries created or edited in the query editor have unique characteristics compared to other queries, like those queries created in the explorer. Review the following tips for working with queries created in the query editor.

Sorting in a Stacked Bar or Stacked Column Chart

Sorting works differently in stacked bar and stacked column charts depending on whether the underlying query is created in the explorer or query editor. To see the difference, let's start by creating a stacked bar chart in the explorer. The chart shows the number of cases, grouped by industry and account type. The chart is sorted in descending order. Notice how the total bar size shortens. (For advanced users who modify the dashboard JSON, charts created in the explorer have query type `aggregateflex`.)

If you customize the query in the query editor, the results sort differently—even if you don't change the sort order. The same bars appear, but in a different order. (For advanced users, the query type changes from `aggregateflex` to `saql` in the dashboard JSON when you edit the query in the query editor.)

Although it looks like the results aren't sorted, they are. The results are sorted in descending order based on the first account-type segment for each industry. Notice how the blue segments get shorter as you scan down the chart.

If you're wondering how to sort the results based on total bar size, like we saw when we first created the chart in the explorer, perform the following tasks.

1. In the explorer, create the chart by adding the measure and groupings (one for Bars and another for Bar Segments).
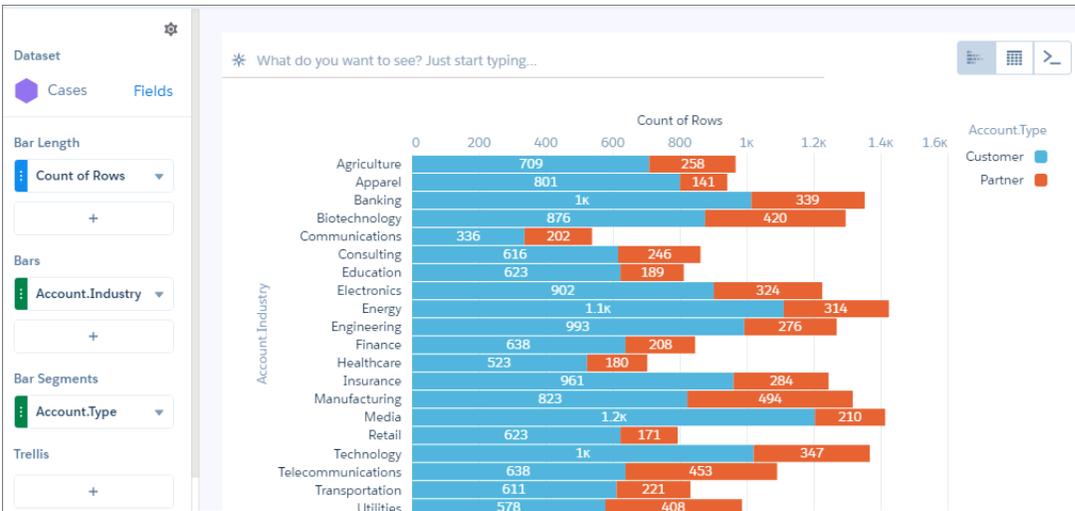
2.
Click and select to switch to a compare table.

3. Add another Count of Rows measure by clicking **+** under Measures, and then clicking **Count** > **Rows**. You're going to total the values for all account types for each industry, and then sort based on that total using the windowing function.

4. For the second Count of Rows measure, click the down arrow and select **Edit this Column**.
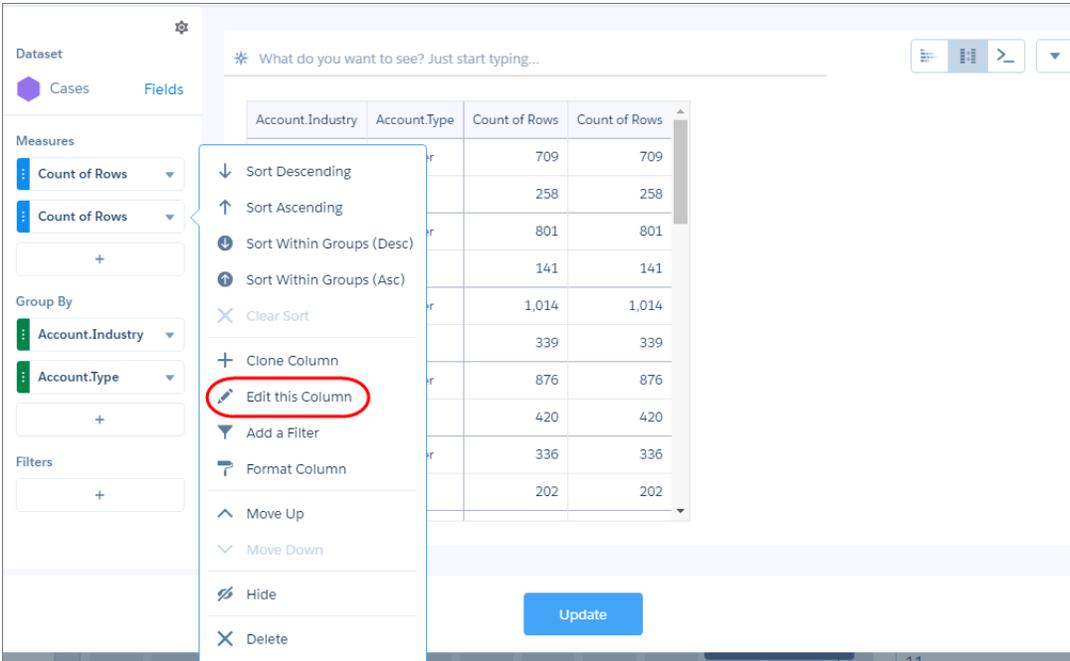
5. Select **Sliding Window** under Calculation, select **Sum** as the function, remove the start and end values for the window, select the **Account.Industry** as the reset group, and then click **Apply**. This calculation combines the counts for both account types for each industry.
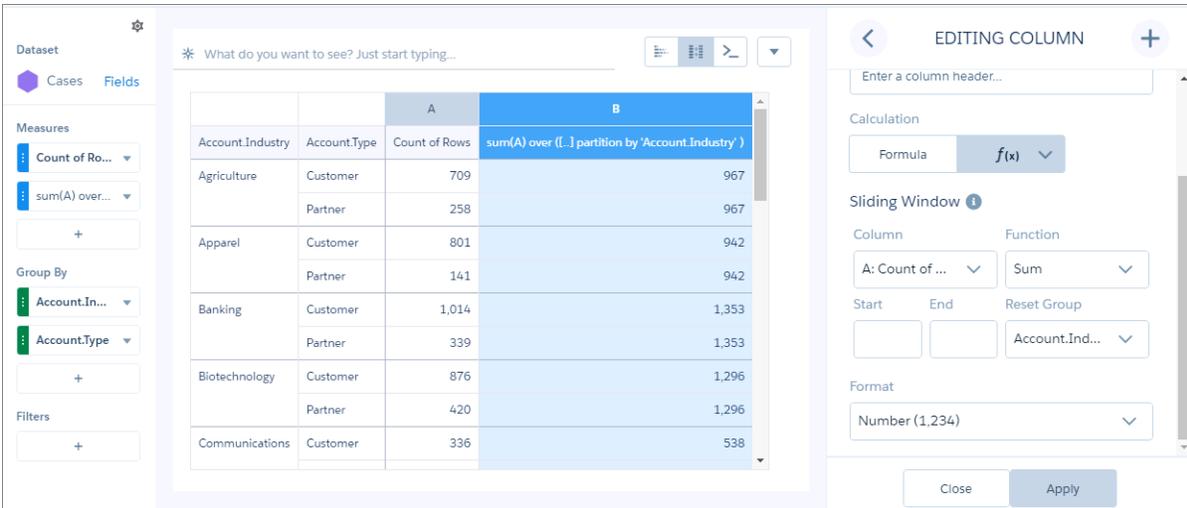
6. To sort the results based on this new total, select the dropdown arrow for this measure and select **Sort Descending**.
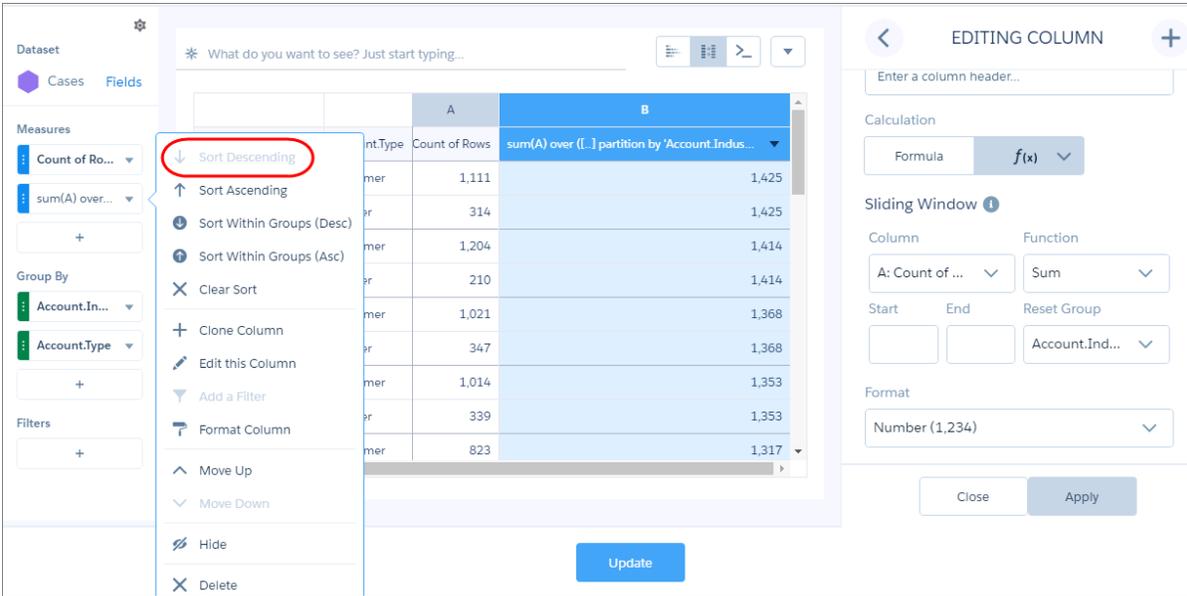
7. To view the results as a stacked bar or column chart, select and select the chart.

When you view the query for the chart in the query editor, you see the following SAQL query.

```
q = load "case95";
result = group q by ('Account.Industry', 'Account.Type');
result = foreach result generate q.'Account.Industry' as 'Account.Industry',
q.'Account.Type' as 'Account.Type', count(q) as 'A';
result = group result by ('Account.Industry', 'Account.Type');
result = foreach result generate 'Account.Industry', 'Account.Type', sum(A) as 'A',
sum(sum(A)) over ([..] partition by 'Account.Industry' ) as 'B';
result = order result by ('B' desc);
result = limit result 2000;
```

8. To hide the second measure that we created for sorting purposes, click the dropdown arrow for that measure, and then select **Hide**.
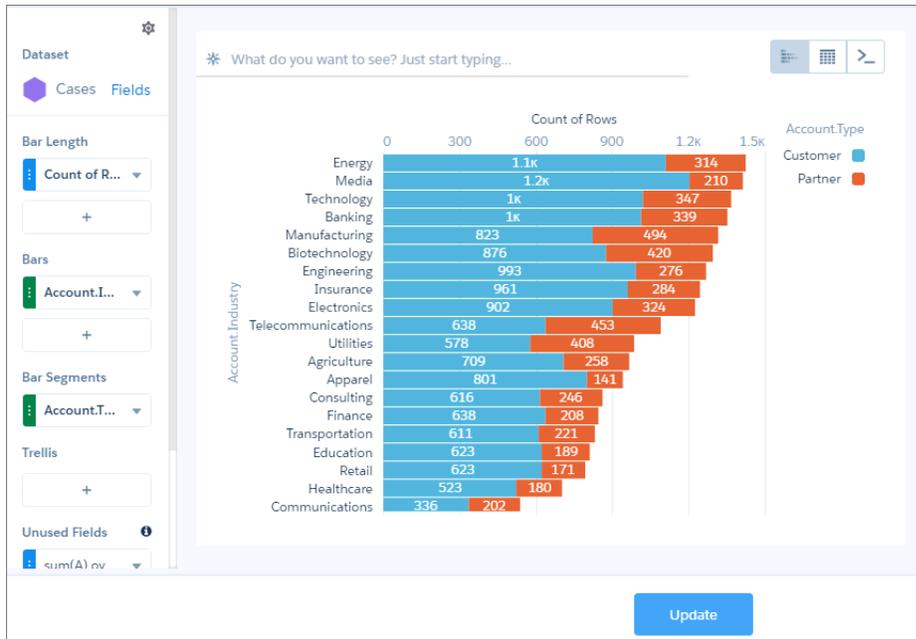
The hidden measure appears under Unused Fields in the left pane.

9. To apply the changes, click **Update**.

Create a Custom Query with User-Defined Values

Use custom queries to apply user-defined values to dimensions, measures, and limits of other queries. For example, you can create a toggle widget that filters case records based on the following user-defined values: High Priority Cases, Medium Priority Cases, and Low Priority Cases. To enable these custom values to filter your data, you map them to your data using a binding.

1. To open the queries panel, click an empty space in the dashboard canvas in the dashboard designer.
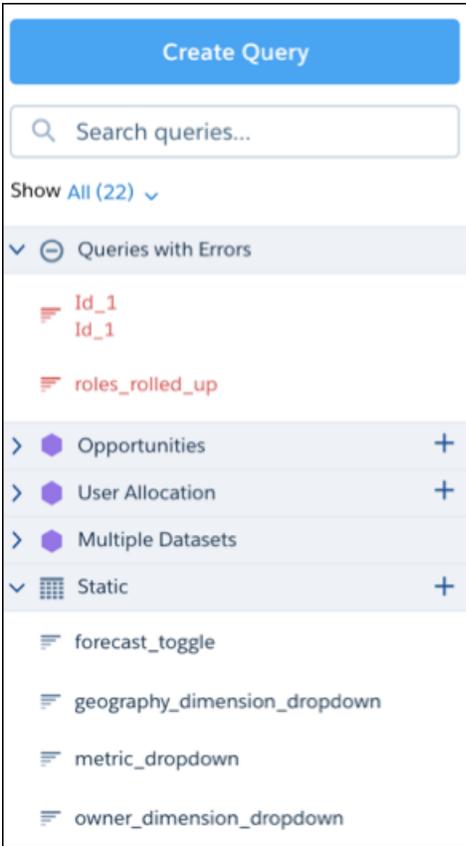
2. Click **Create Query**.
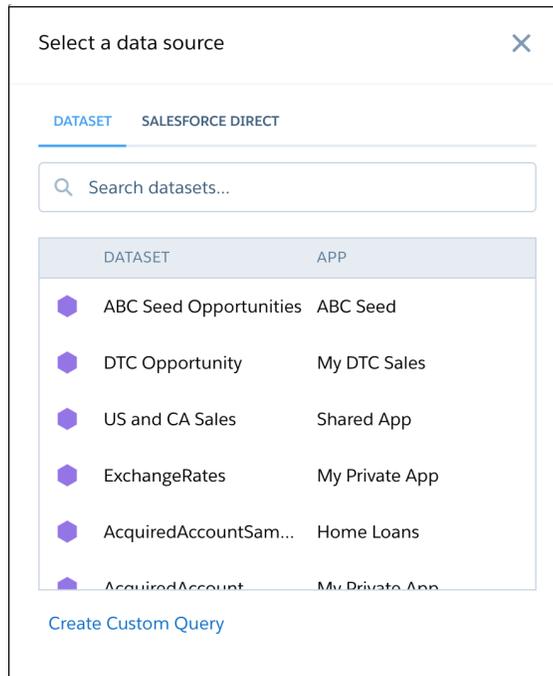
3. Click **Create Custom Query**.

4.
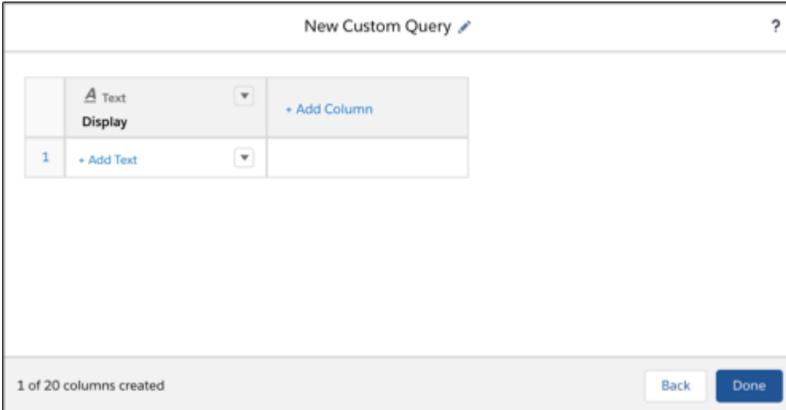
Click [pencil icon] to change the query label.

Tableau CRM creates the query ID from the label. After you create the query, you can't change the query ID. Tableau CRM refers to queries by their ID so dashboards don't break if you change the labels.
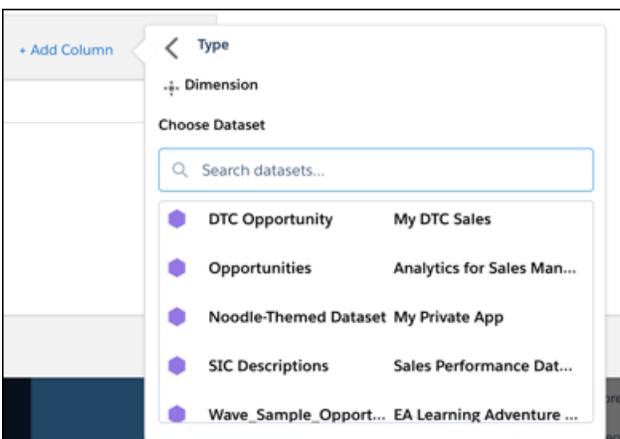
5. Click **+Add Column** to add a dimension, measure, text, or number as a column to your custom query.

    a. For a dimension or measure, choose a dataset.
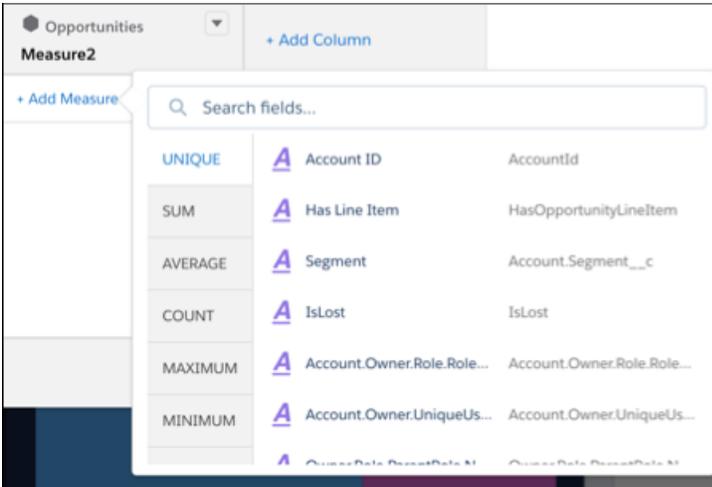
6. Click **+Add** to add rows to your custom query.

   a. For a measure, choose a function. For a dimension or measure, choose a field.

   > **Note:** Each measure column in a custom query creates two values in the JSON. One value for the function (sum, average, etc.) and one for the selected measure field. Use both values when creating a compact-form binding on a measure.
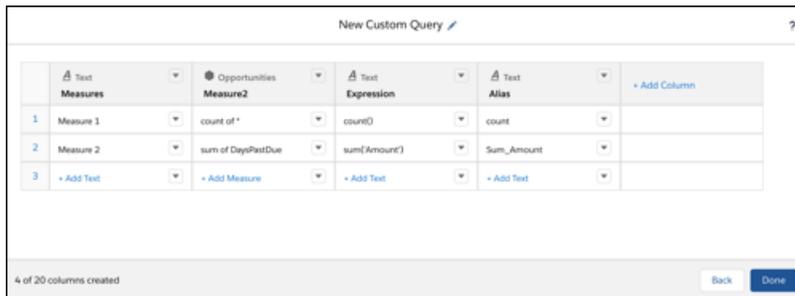
7. Add up to 20 columns and click **Done**.

   > **Note:** To create SAQL bindings, your custom query requires two additional text columns. One column determines the value of the field and the other column is for the field label.

   Tableau CRM adds the query to the query panel, making it available for use by other widgets in the dashboard.

8.

   To make changes, find the query in the query panel, click , then **Edit**.
   You can change the query name, add and remove columns, add row and column values, reorder columns, and pick values from a dataset.

   > **Note:** To use a custom query for linking, make the column the first column in the dialog.

9. To apply this custom query to a list or toggle widget:

   a. Drag the list or toggle widget from the widget toolbar to the dashboard canvas.

   b. Drag the custom query on top of the widget.
   The widget shows the custom display values as selection options.

c. To change the widget and query properties, select the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook any properties, expand any collapsed sections.

d. To customize the appearance of the widget, set the widget properties.

e. To preview your changes to the dashboard, click 👁 .

f. Save the dashboard.

If you applied the custom query to a list or toggle widget, you can see the widget in the dashboard. To specify how a selection in this query affects other widgets, bind the values of the custom query to other widgets. For example, create a binding that filters the other widgets based on a value in the list or toggle widget. For more information about binding queries, see the *Analytics Cloud Interactions Developer Guide*.

SEE ALSO:

[Query Properties for Tableau CRM Dashboards](#)

[Widget Properties for Tableau CRM Dashboards](#)

[Query Properties for Tableau CRM Dashboards](#)

Find a Query

You can find all queries defined in the dashboard in the query panel. The panel organizes queries into categories and provides a search bar to help you locate your queries faster.

The queries panel categorizes queries by dataset name and query type. It also contains a special category for erroneous queries. Categories and queries are listed in alphabetical order. You can collapse each category.
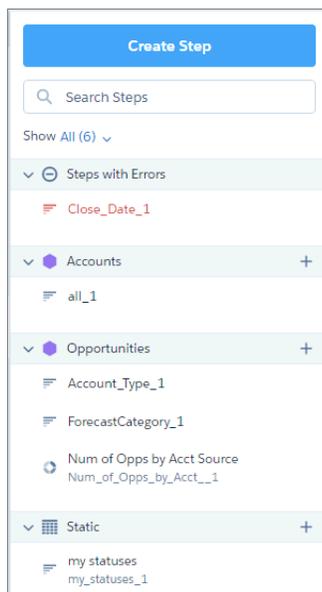
Hover over the query and click the down arrow to view actions that you can perform on the query. For instance, in the Queries with Errors section, click **View Error** to understand what's wrong with the query.
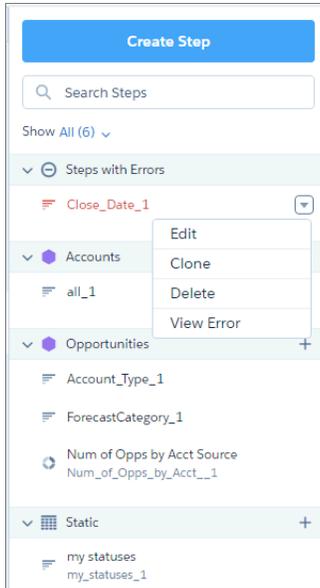
Edit a Query

When you edit a query, you can change the details, like the query label. These changes affect all widgets that use the query. You can also change the visualization properties, like chart type. Depending on the method that you use to edit the query, Tableau CRM handles visualization changes differently.

You can use the following methods to change a query.

Method	Handling Visualization Changes
Edit the query from the query panel	Tableau CRM saves the visualization changes in the query, but doesn't apply them to existing widgets that use the query.
Double-click the widget	Tableau CRM saves the visualization changes in the query and applies them to the widget. Tableau CRM doesn't apply them to other widgets that use the query.

Consider the following limitations before editing a query.

- You can edit only queries with types `aggregate`, `aggregateflex`, `saql`, and `grain`. If you edit an `aggregate` or `aggregateflex` that contains a SAQL query in the `pigql` JSON attribute, Tableau CRM converts the query type to `saql`. To learn about the implications of converting between these query types, see Considerations When Converting Query Types .
- You can't edit a query that contains a binding or that's used in a global filter widget created before Winter '18.

1. Open the dashboard in edit mode.
2. Use one of the following methods to edit the query.
   a. To edit the query from the queries panel, hover over the query, click the down arrow, and then click **Edit** in the query actions menu.
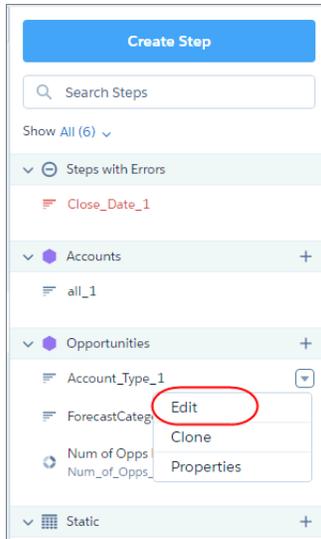
a. Double-click the widget in the dashboard.

You can also select the widget and click ![edit icon] in the widget actions menu.
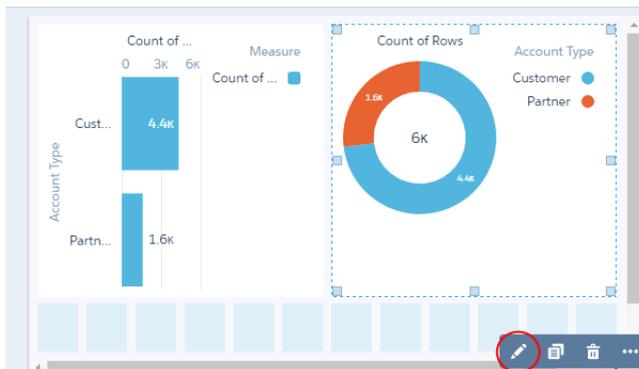
3. Make your changes.

You can change the following details:

- Query label (1)
- Query (2)
- Visualization properties (3)
- Chart type and table type (4)
- SAQL query (5)
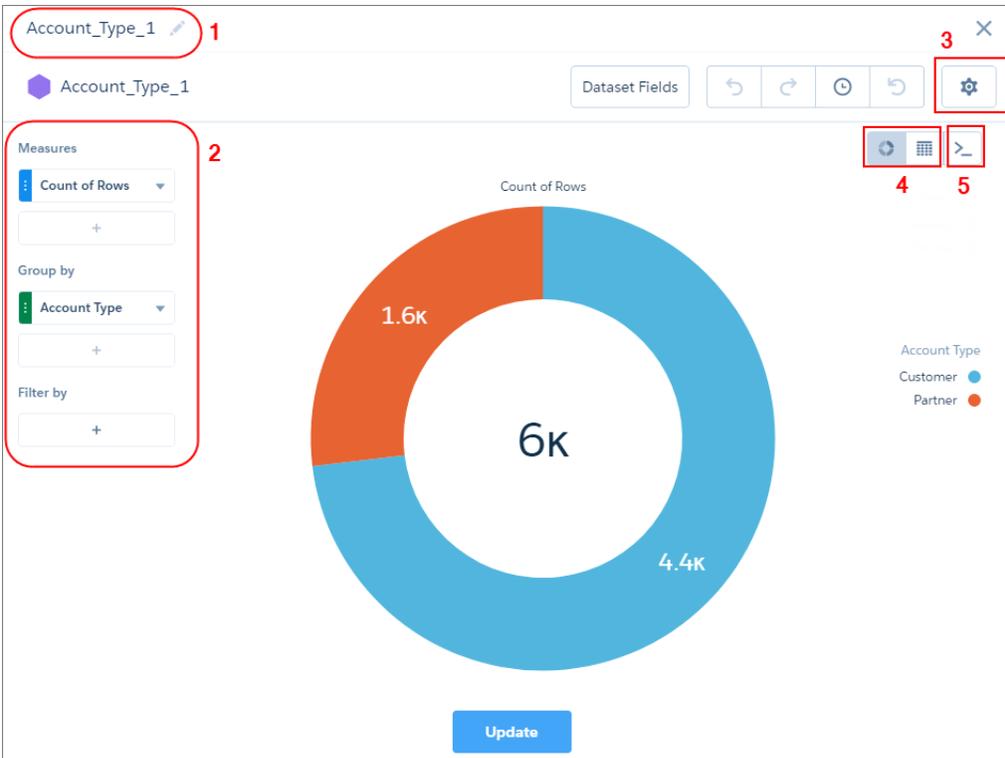
> 📝 **Note:** For information about **Dataset Fields**, go View and Configure Dataset Columns.

4. Click **Update**.

5. To save your changes, save the dashboard.

Considerations When Converting Query Types
Tableau CRM treats query types differently. Before you convert an `aggregate` or `aggregateflex` query type to `saql`, review the following implications.

Considerations When Converting Query Types

Tableau CRM treats query types differently. Before you convert an `aggregate` or `aggregateflex` query type to `saql`, review the following implications.

Unlike the `aggregateflex` query type, the `saql` query type provides limited support when the actual number of query results exceeds the limit that a query can return.

- When performing a search in a list widget, a `saql` query type doesn't find values that aren't returned in the first batch of query results.
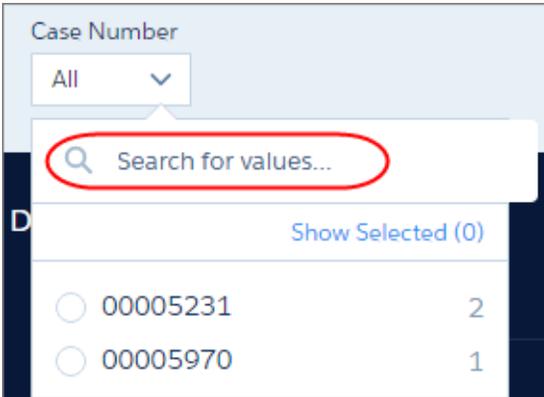
- Tableau CRM doesn't apply initial selections for a `saql` query type if the selected value isn't returned in the first batch of query results. For example, you set an initial selection on the Account Name field. When you open the dashboard, assume the corresponding widget shows only 2,000 of 100,000 accounts. If the initial selection isn't one of the 2,000 returned accounts, Tableau CRM doesn't apply the initial selection to the dashboard.

For more information about how to use the `limit` JSON parameter to configure the maximum number of results that each query type can return, see steps JSON.

Clone a Query

Clone a query to quickly create a query based on an existing one. Cloning is useful when you can reuse an existing query with just a few tweaks.

📝 **Note:** You can't clone a SAQL query for a single compare table column or a custom query with user-defined values.

1. To open the queries panel, click an empty space in the dashboard canvas in the dashboard designer.
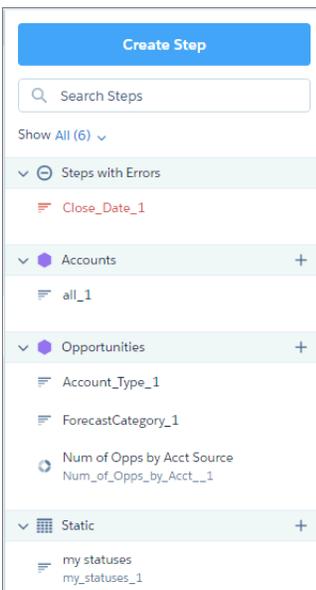
📝 **Note:** You can't apply grayed-out queries to widgets. But you can clone them and view their properties.

2. To show all queries, select **All** in the Show field.

3. Hover over the query that you want to clone, and then click the down arrow.
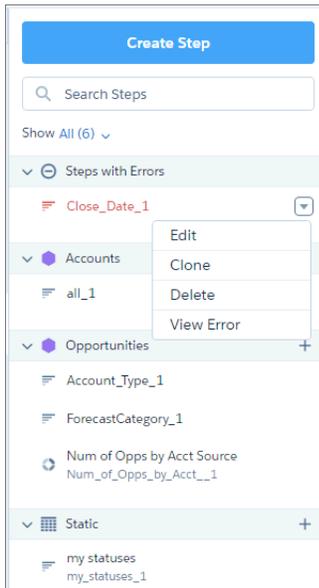
4. Select **Clone**.

   The query opens in the explorer, where you can easily modify it. If the query was created using SAQL, the clone opens in the SAQL editor instead.

5. Edit the query label.

   The default label has "Clone of " prepended to the original query label

6. Make your changes to the clone.

7. Click **Done**.

   Tableau CRM adds the query to the query panel, making it available for use by other widgets in the dashboard.

8. Save the dashboard.

   📝 Note: When you clone a query and edit its SAQL, the resulting JSON doesn't always reflect the functions applied to measures. After you run and save the SAQL, the function is changed to `count`. For example, if the original query contained this `measures` array:

```
"query": {
        "measures": [
        ["max","DailyActiveUsers"]
        ]
      }
```

   and the SAQL was edited, run, and saved, the updated JSON contains this `measures` array instead:

```
"query": {
        "measures": [
        ["count","*","max_DailyActiveUsers"]
        ]
      }
```

   This discrepancy has no effect on the data viewed in the dashboard.

Modify the Query Results Based on the Dashboard Viewer

You can modify a query based on the user viewing the dashboard. For example, you can filter opportunity records to show only those opportunities owned by this user.

You can specify user tokens in any query parameter in `saql`, `soql`, and `aggregateflex` query types. The syntax to specify a token is: `!{user.<User object field name>}`. Valid tokens are: `user.Id`, `user.name`, `user.roleid`, and `user.rolename`.

💡 **Tip:** Because names aren't always unique, consider using IDs when specifying a user token.

This `saql`-type query adds the `user.Id` token as a query filter so that opportunity owners viewing the dashboard see only their opportunities.

```
"My_Opportunities_1": {
    "type": "saql",
    "label": "My Opportunities",
    "query": "q = load \"opportunity\";\nq = filter q by 'OwnerId' == \"!{user.Id}\";\nq
= group q by 'Account.Type';\nq = foreach q generate 'Account.Type' as 'Account.Type',
count() as 'count';\nq = order q by 'count' desc;\nq = limit q 1000;",
    "datasets": [
        {
            "id": "0FbB000000017QxKAI",
            "name": "opportunity",
            "label": "Opportunities",
            "url": "/services/data/v41.0/wave/datasets/0FbB000000017QxKAI"
        }
    ],
    "useGlobal": true,
    "isGlobal": false,
    "broadcastFacet": true,
    "receiveFacet": true,
    "selectMode": "single",
    "numbers": [],
    "groups": [],
    "strings": [],
    "visualizationParameters": {
        ....
    },
},
```

SEE ALSO:

Optimize a Query

Queries can slow down your dashboard. Run a performance check on the queries to ensure that they're running optimally. Tableau CRM identifies query bottlenecks and provides recommendations to improve query performance.

1. Open the dashboard in view mode.

   Because Tableau CRM analyzes query performance for the current layout only, ensure that you're viewing the dashboard in the right layout. Performance can differ from one layout to another because layouts can have different widgets and interactions. If the dashboard contains multiple pages, Tableau CRM provides performance analysis for the current page.

2. Click the drop-down arrow for the widget that uses the query that you want to analyze.
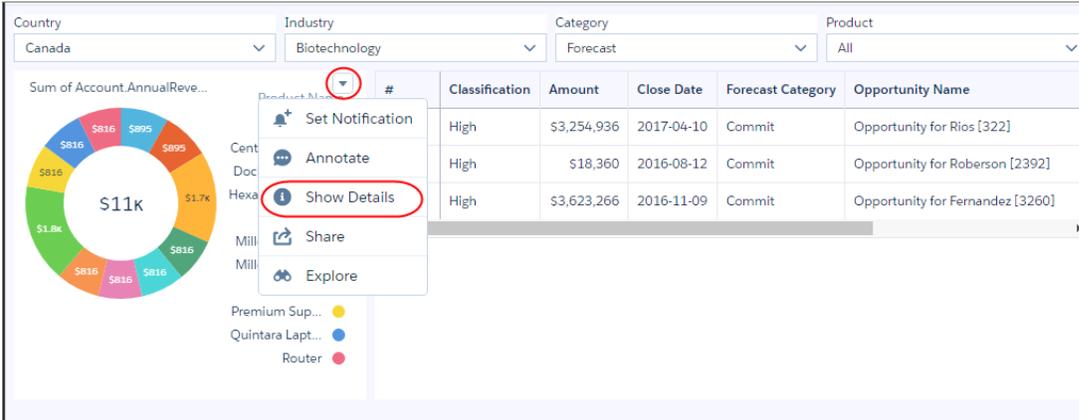
3. Select **Show Details** for the widget.

> 💡 Tip: To see the Show Details option for a list widget, enable the "Show explorer icon" widget property. The Show Details option doesn't show for link, image, container, global filter panel, and text widget types because they don't have queries.

The query details summarize different parts of the query, like measures, filters, groupings, and the dataset on which its built.
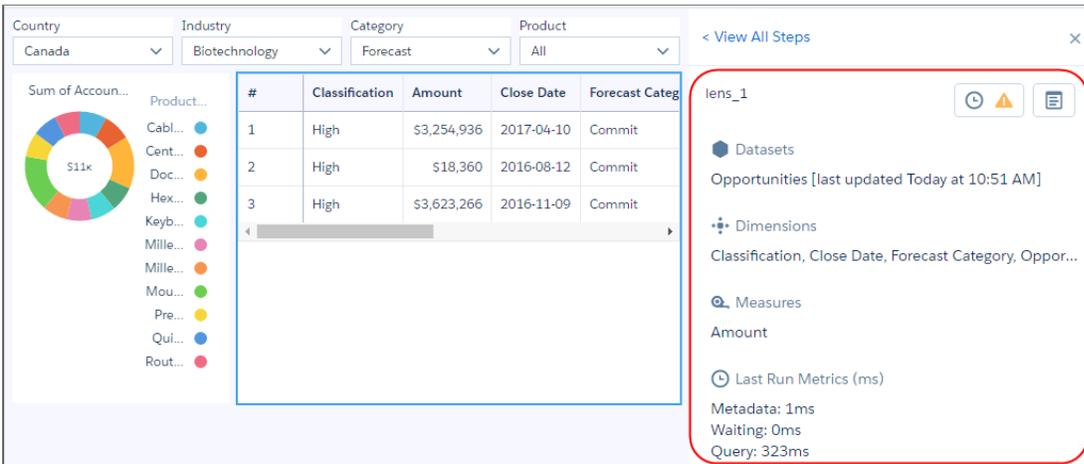
The Last Run Metrics section identifies other factors that contribute to the total time required to populate the results of a query.

Last Run Metric	Description
Metadata	Number of milliseconds needed to retrieve the metadata for the datasets involved in the query. The metadata fetch only occurs the first time the query runs.
Waiting	Number of milliseconds needed to resolve bindings to construct the query.
Query	Number of milliseconds to run the query. Note that a run might execute the query or use cached results.

If Tableau CRM identifies a performance issue with the query, it shows a warning icon (⚠) next to the View Performance Details button (🕐), as the previous screenshot shows.

4. To view the performance issues and suggestions for resolving them, click 🕐 .

The Performance Factors section shows all factors by which the query is evaluated. Ignore the factors with a green checkmark (✓) because these issues don't apply to the query.
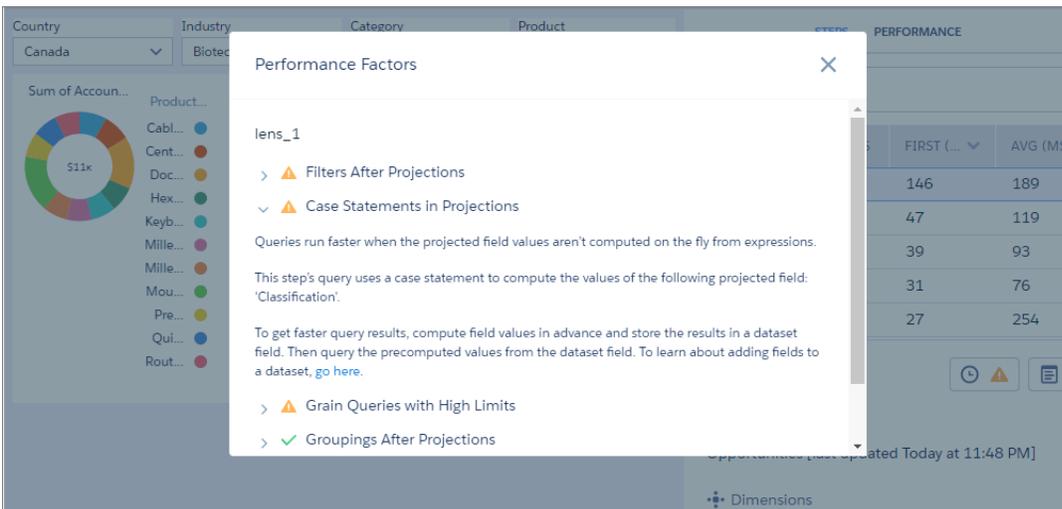
5. Expand each factor with a warning icon to learn how to improve the query's performance.

Considerations for Queries with Multiple Datasets

When a query has multiple datasets, the query results are formatted using the XMD of the first loaded dataset. If the query is faceted, it receives filters for all of its datasets, but broadcasts selections only from its first loaded dataset.

For more information about formatting results using the XMD, see the Extended Metadata (XMD) Reference.

SEE ALSO:

Faceting Queries with Multiple Datasets

Troubleshoot Unexpected Query Results

If you have questions about a query's results, review the query details to learn how Tableau CRM determines the results. Compare the original query to the final query executed by Tableau CRM. If the query is faceted or the dashboard contains global filters, Tableau CRM applies extra filters to query. If the query contains a binding, use the bindings tracer to verify the binding results.

1. Open the dashboard in view mode.

2. Click the drop-down arrow for the widget that uses the query that you want to analyze.
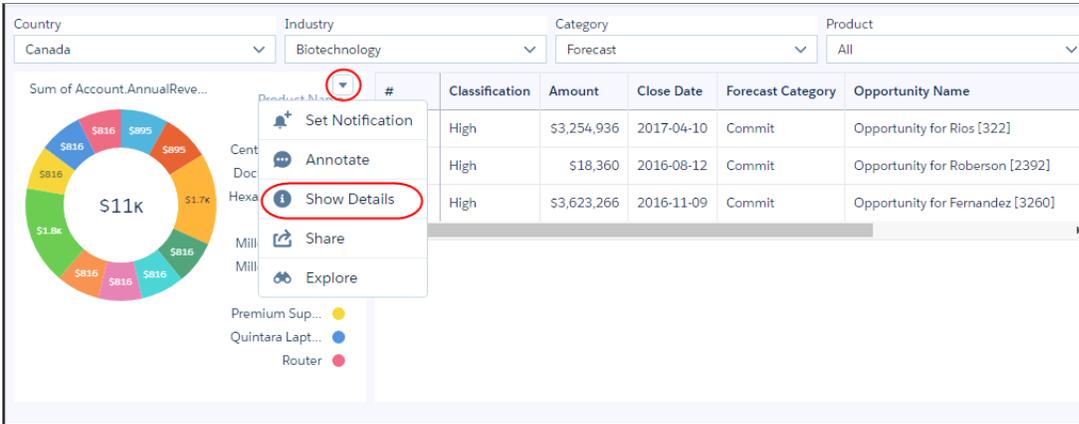
3. Select **Show Details** for the widget.

> 💡 Tip: To see the Show Details option for a list widget, enable the "Show explorer icon" widget property. The Show Details option doesn't show for link, image, container, global filter panel, and text widget types because they don't have queries.
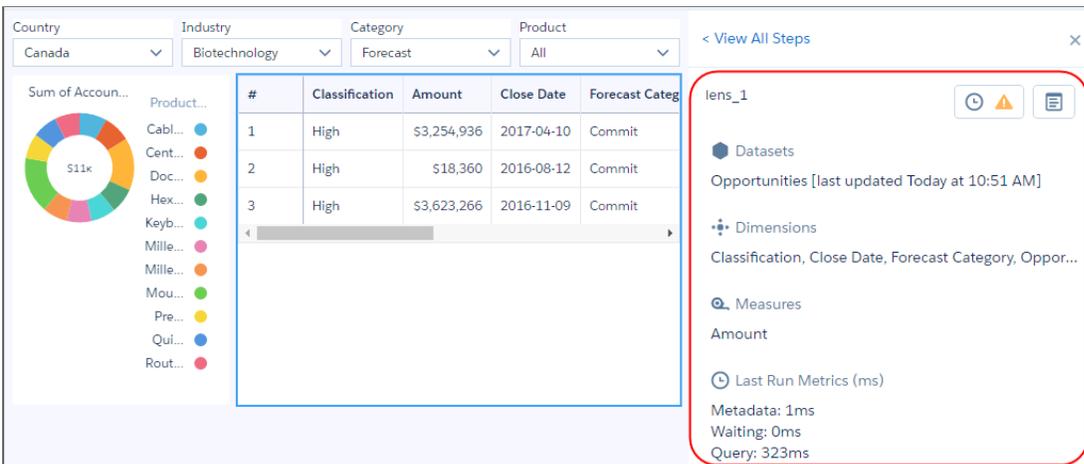
4. To compare the original query against the final one, click 📄 .

Tableau CRM modifies queries based on interactions with other widgets. It adds filters based on faceting and global filters, and can modify the query based on bindings. The final query in the following example has three filters added to the original query.

Original Query

```
1  q = load "opportunity";
2  q = foreach q generate (case when 'Amount' < 5000000 then "High" when 'Amount' < 20000000 then "Med" else
       "#Low" end) as 'Classification', 'Amount' as 'Amount', 'CloseDate' as 'CloseDate',
       'ForecastCategoryName' as 'ForecastCategoryName', 'Name' as 'Name', 'Owner.Name' as 'Owner.Name',
       'StageName' as 'StageName';
3  q = filter q by 'Classification' == "High";
4  q = limit q 10000;
```

Final Query

```
1  q = load "opportunity";
2  q = filter q by 'Account.BillingCountry' in ["Canada"];
3  q = filter q by 'ForecastCategory' in ["Forecast"];
4  q = filter q by 'Account.Industry' in ["Biotechnology"];
5  q = foreach q generate case when ('Amount' < 5000000) then "High" when ('Amount' < 20000000) then "Med"
       else "#Low" end as 'Classification', 'Amount' as 'Amount', 'CloseDate' as 'CloseDate',
       'ForecastCategoryName' as 'ForecastCategoryName', 'Name' as 'Name', 'Owner.Name' as 'Owner.Name',
       'StageName' as 'StageName';
6  q = filter q by 'Classification' == "High";
7  q = limit q 10000;
```

5. If the query has a binding, review the bindings trace to understand how Tableau CRM resolves the binding.

BINDINGS TRACE	RESULT
∨ coalesce(cell(Numbers.selection, 0, "value"), cell(Numbers.result, 0, "value"))…	5
∨ coalesce(*, *).asObject()	5
∨ coalesce(cell(*, *, *), cell(*, *, *))	5
cell(Numbers.selection, 0, "value")	null
cell(Numbers.result, 0, "value")	5

The easiest way to understand the bindings trace is to start from the bottom. In this example, you can see the results of both `cell` binding functions. You can also see that the `coalesce` function uses the results from the second `cell` function. The final result of the entire binding, which appears at the top, shows 5.

Query Properties for Tableau CRM Dashboards

Query properties define the query that returns results to display in the widget. These properties also specify how the widget behaves and interacts with other widgets in the Tableau CRM dashboard. The properties apply to dashboards created in the dashboard designer only.

Property	Description
Display Label	Query label. Use to provide a descriptive label for the query.
Broadcast selections as facets	Controls whether the query's selections are broadcasted as facets. For more information about faceting, see Filter the Dashboard Results with Faceting .
Apply filters from faceting	Controls whether the query listens for broadcasted facets and applies them as filters.

Property	Description
Apply global filters	Apply the global filters to this query. This option is only available for queries on datasets.
Selection Type	Indicate whether dashboard viewers can make one or multiple selections in the associated widget. You can also specify whether selections are required. If a selection is required, set an initial selection in the Initial Selections query property. If a selection is required but not set, Tableau CRM selects the first value in the list. 📝 Note: Selection types are ignored when you open the widget in an explorer lens, or in iOS or Tableau CRM mobile apps.
Initial Selections	Show the initial selections that are applied when the dashboard opens. For information about setting initial selections, see Set Initial Selections in the Dashboard .

Make the Dashboard Widgets Interactive

Tableau CRM dashboards have unique features that allow you to make the widgets interactive. For example, widgets in the dashboard can be filtered to show only results for the region that's selected in a list widget. Or, when the value of a number widget can change to red when it falls below a threshold.

Watch a Demo: ▶ Build Interactive Tableau CRM Dashboards (English Only)

Filter the Dashboard Results with Faceting
Faceting is the simplest and most common way to specify interactions between widgets. Widgets can be linked simply by enabling faceting in their associated queries. When the underlying queries are faceted, selections made on chart, date, list, range, and toggle widgets automatically filter other widgets in the dashboard. Use targeted faceting to include or exclude specific queries from your faceting selections.

Create Interactions with Clicks
Interactions, previously called bindings, are the most powerful way to specify relationships between different queries on your dashboard. Creating the right syntax for an interaction can be complex. With the Advanced Interaction Editor, create, edit, and preview your interactions with clicks.

SEE ALSO:

Manage Queries for Widgets

Filter the Dashboard Results with Faceting

Faceting is the simplest and most common way to specify interactions between widgets. Widgets can be linked simply by enabling faceting in their associated queries. When the underlying queries are faceted, selections made on chart, date, list, range, and toggle widgets automatically filter other widgets in the dashboard. Use targeted faceting to include or exclude specific queries from your faceting selections.

For example, when you select a rep in the Sales Rep list widget, Tableau CRM filters the dashboard's table widget based on the rep. Use targeted faceting to exclude a total open pipeline number widget so the number isn't changed when selecting an individual Sales rep.

By default, queries from the same dataset are faceted together. To facet queries between different data sources—like different datasets or custom queries —first connect them. Queries with a blending (cogrouping) of multiple datasets only facet widgets based on the first dataset specified in the `datasets` field in the dashboard JSON.

1. Click ✏ to edit the dashboard.

2. Select the widget that uses the query.
   The panel shows the widget properties.

3. Click **Query** to view the query properties.
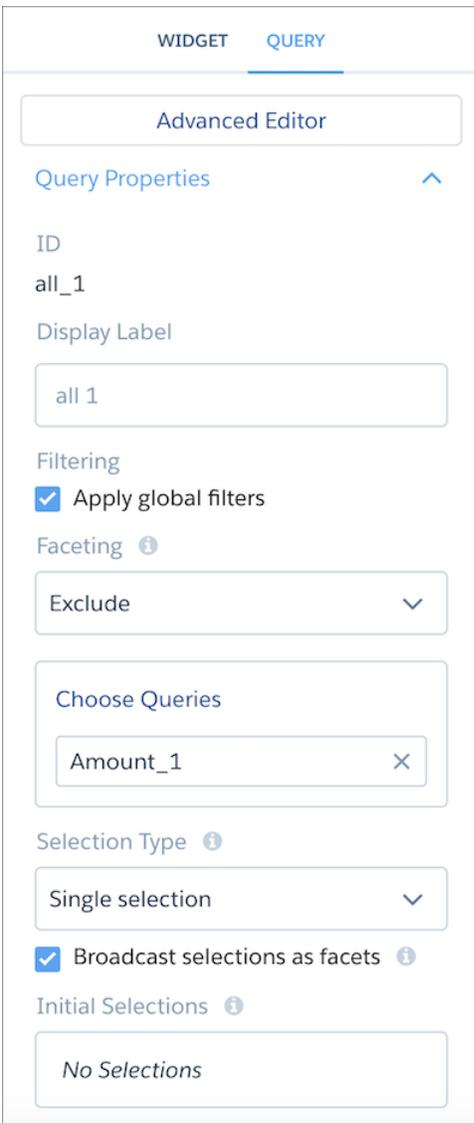
4. In the Faceting section of the query panel, select All, Include, Exclude, or None.

   For Include or Exclude, tap Choose Queries to select from the list of all queries on your dashboard.

   📝 Note: For **All**, external filters are applied to the query. For any other selection, queries do not receive external filters. Global filters are not external filters.

5. **Broadcast selections as facets** is enabled by default.

   Controls whether the query's selections are broadcasted as facets.

   > **Note:** Keep the following considerations in mind when configuring faceting for queries.
   >
   >   - The **Broadcast selections as facets** appears as read only for the `apex` and `soql` query types because they don't support faceting.
   >
   >   - **Broadcast selections as facets** appears as read only for the `aggregate`, `grain`, and `static` query types because they don't support these options. To change the faceting behavior for these query types, edit the `isFacet` property in the dashboard JSON. For a list of the supported faceting JSON properties for each query type, see the *Analytics Cloud Dashboard JSON Reference*.
   >
   >   - For **Include**, faceting applies to only the selected queries. For **Exclude**, faceting applies to all queries except those selected.

6. To save the changes, save the dashboard.

   [Configure Cross-Dataset Faceting with Connected Data Sources](#)
   By default, queries from the same dataset are faceted—automatically filtering each other. To facet queries from different datasets or even custom queries in the dashboard designer, connect them using common fields between them.

   [Faceting Queries with Multiple Datasets](#)
   You can use a query with multiple datasets in a widget and then facet the query. The query receives filters from all of its datasets, but broadcasts selections from its first loaded dataset.

SEE ALSO:

   [Add Selection-Based Filter Widgets to Enable Users to Filter the Results](#)

Configure Cross-Dataset Faceting with Connected Data Sources

By default, queries from the same dataset are faceted—automatically filtering each other. To facet queries from different datasets or even custom queries in the dashboard designer, connect them using common fields between them.

> **Note:** You can't connect custom queries that specify date or measure ranges.

1. Click ✏️ to edit the dashboard.

2. Click ⋯ and then select **Connect Data Sources**.

3. Click **New Connection**.

4. Enter a name for the connection.

5. Click **Choose Data Source 1** to select the first data source. For each data source, select a dataset or a custom query used in the dashboard.

6. Choose one field from the data source so that Tableau CRM can match records between the specified sources.

   📝 Note: Faceting only works on the fields used to define the connection. For example, the connection in the following screenshot is based on the StageName field. As a result, the StageName list widget is faceted to the bar chart. If a selection is made in the Account.Type list widget, faceting doesn't happen—the results are not filtered. To enable faceting on the Account.Type field, define another connection based on this field using the same data sources.
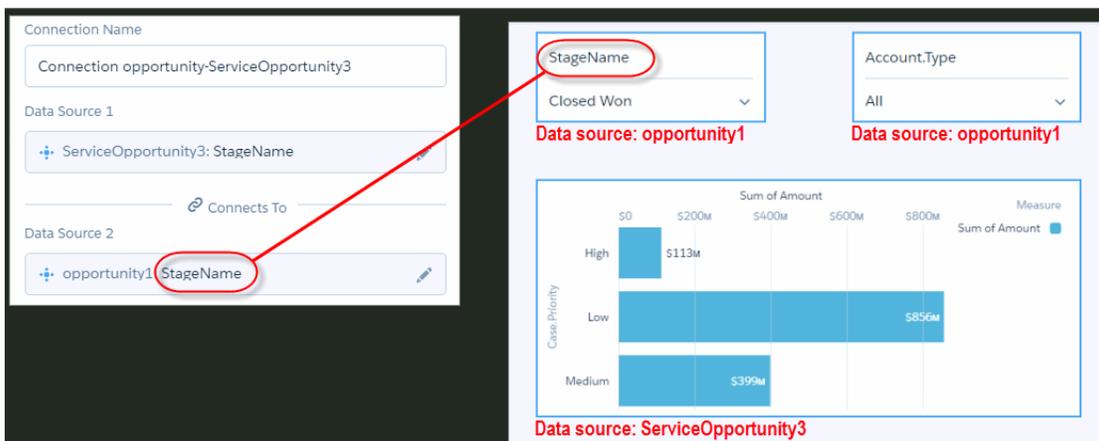
7. Click **Choose Data Source 2** to select the second source, and then choose a field.

8. To add another data source, click **Add Data Source**. This option appears after you connect two data sources.

   You can connect two or more data sources.

9. Click **Save**.

Faceting Queries with Multiple Datasets

You can use a query with multiple datasets in a widget and then facet the query. The query receives filters from all of its datasets, but broadcasts selections from its first loaded dataset.

For example, suppose that the Opportunities dataset has the fields Account_Owner, Region, and Amount. The Quota dataset has the fields Name and Quota.

```
quo = load "Quota";
opt = load "Opportunities";
att = cogroup opt by 'Account_Owner', quo by 'Employee';
att = foreach att generate opt.'Account_Owner' as 'Owner', sum(opt.'Amount')/sum(quo.'Quota')
 as 'percent attained', opt.Region as 'Region'.
```

A widget that uses this query receives filters for the following selections:

- Opportunities.Name
- Opportunities.Amount
- Opportunities.Region
- Quota.Name
- Quota.Quota

A widget that uses this query broadcasts selections for Quota.Name and Quota.Quota.

Create Interactions with Clicks

Interactions, previously called bindings, are the most powerful way to specify relationships between different queries on your dashboard. Creating the right syntax for an interaction can be complex. With the Advanced Interaction Editor, create, edit, and preview your interactions with clicks.

> **Note:** Dashboard builders must place each interaction (binding) in the right place in the JSON or SAQL for an individual widget or query. To learn more, see the Tableau CRM Bindings Developer Guide.

1.

Click **Advanced Editor** in the widget properties panel or ⊞ to launch the Advanced Interaction Editor.

2. Click **Select a query…** and choose from any query in the dashboard.

Queries are searchable by name.

3. Click **Select data…**.

   a. Select Row, Column, or Cell.

   b. Select the desired options.

   Use the **Preview** button to see details of the query without leaving the Interaction Editor.

   c. Click ← .

4. Select **Result** or **Selection** for your interaction.

   Result is the default interaction type.

   To learn more about interaction types, see Result Binding and Selection Binding.

5. Click **More Options** to set a default value or change the data serialization function.

   To learn more, see Data Serialization Functions.

6.
   To edit the interaction manually, click [✎]. This step is optional.

   > **Note:** Once you click the [✎] button, you can't return to the original editor without resetting the entire Advanced Interaction
   > Builder. Any interaction you've already created is removed once you reset.

   a. Add functions to your interaction and specify the source.

7. **Copy** your interaction.

8. Paste the interaction into the JSON or SAQL.

9. Click **Save**.

Dynamically Modify the Query, Initial Selection, and Widgets with Bindings

Compared to facets, bindings are more programmatic, and provide more ways to specify interactions. Bindings can dynamically
modify a query, set an initial filter selection, or change a widget's appearance. For example, you can change the measure in a query
based on a selection in a custom query. You can initially filter a region-based dashboard based on the country of the logged-in user.
Or, you can change the color of the metric in a number widget based on whether its value is high, medium, or low. Each binding is
unidirectional, meaning it affects other queries, but other queries don't affect it.

Dynamically Modify the Query, Initial Selection, and Widgets with Bindings

Compared to facets, bindings are more programmatic, and provide more ways to specify interactions. Bindings can dynamically modify
a query, set an initial filter selection, or change a widget's appearance. For example, you can change the measure in a query based on
a selection in a custom query. You can initially filter a region-based dashboard based on the country of the logged-in user. Or, you can
change the color of the metric in a number widget based on whether its value is high, medium, or low. Each binding is unidirectional,
meaning it affects other queries, but other queries don't affect it.

For more information about bindings, see *Analytics Cloud Interactions Developer Guide*

Set Initial Selections and Global Filters in the Dashboard

Set the initial selections and global filters that appear when the dashboard first opens. To analyze the results from a different angle, the
dashboard viewer can change the initial selections and, if configured, global filters while viewing the dashboard.

Set Initial Selections in the Dashboard

If the Tableau CRM dashboard contains selection-based widgets (chart, date, list, range, or toggle), you can set initial selections that
are applied when the dashboard first opens. For example, you can select a bar in a bar chart and set that as the initial selection. If
the underlying query for the bar chart is faceted, other widgets with faceted queries are filtered based on this initial selection. While
viewing the dashboard, a user can change the selection in the chart to filter the results differently or remove the selection and filters
altogether.

If the Tableau CRM dashboard contains global filters, you can set the initial filters that are applied when the dashboard first opens. If a global filter is unlocked, the dashboard viewer can change the filter while viewing the dashboard. Global filters apply to all queries in the dashboard unless a query is configured to ignore these filters.

Set Initial Selections in the Dashboard

If the Tableau CRM dashboard contains selection-based widgets (chart, date, list, range, or toggle), you can set initial selections that are applied when the dashboard first opens. For example, you can select a bar in a bar chart and set that as the initial selection. If the underlying query for the bar chart is faceted, other widgets with faceted queries are filtered based on this initial selection. While viewing the dashboard, a user can change the selection in the chart to filter the results differently or remove the selection and filters altogether.

If a widget's query is configured to require a selection, set the initial selection. If you don't, Tableau CRM selects the first value when the dashboard opens.

Tableau CRM ignores the selection if it can't find the initial selection value in the query results when opening the dashboard. This issue can occur when the underlying query for the selection widget has a `saql` query type and not all results are returned. (This issue doesn't happen for other query types.) For more information about query limits for a `saql` query type, see saql Step Type Properties.

1. While editing the dashboard, click ⸱⸱⸱ and then select **Pick Initial Selections**.
   Tableau CRM previews the dashboard.

2. Make selections in chart, date, list, range, or toggle widgets.

   You can also select a row in a compare table.

3. Click **Done**.
   Tableau CRM saves the selections as start values for the widget queries. You can view the initial selections in the query properties.

You can set initial filter selections to show information that's relevant to the logged in user. For example, you can filter a sales dashboard such that each sales rep views only their sales opportunities and performance. You can set initial filter selections based on the user ID, user name, role ID, or role name of the logged in user.

SEE ALSO:
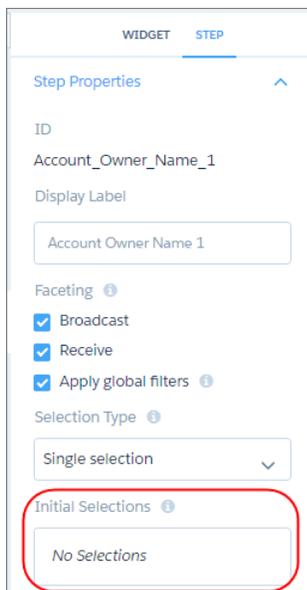
Dynamically Set Initial Selections Based on the Dashboard Viewer

You can set initial filter selections to show information that's relevant to the logged in user. For example, you can filter a sales dashboard such that each sales rep views only their sales opportunities and performance. You can set initial filter selections based on the user ID, user name, role ID, or role name of the logged in user.

You can set initial selections based on the following Salesforce user tokens: `user.id`, `user.name`, `user.rolename`, and `user.roleid`. At runtime, Tableau CRM retrieves the values of these tokens from Salesforce environment variables—these values don't come from datasets.

💡 **Tip:** To set the initial selection based on other user attributes of the logged-in user, like their country, use a binding based on a user token. For more information, see the Analytics Cloud Interactions Developer Guide.

👁 **Example:** When the dashboard opens, you want it to dynamically filter charts based on the name of the currently logged in user.
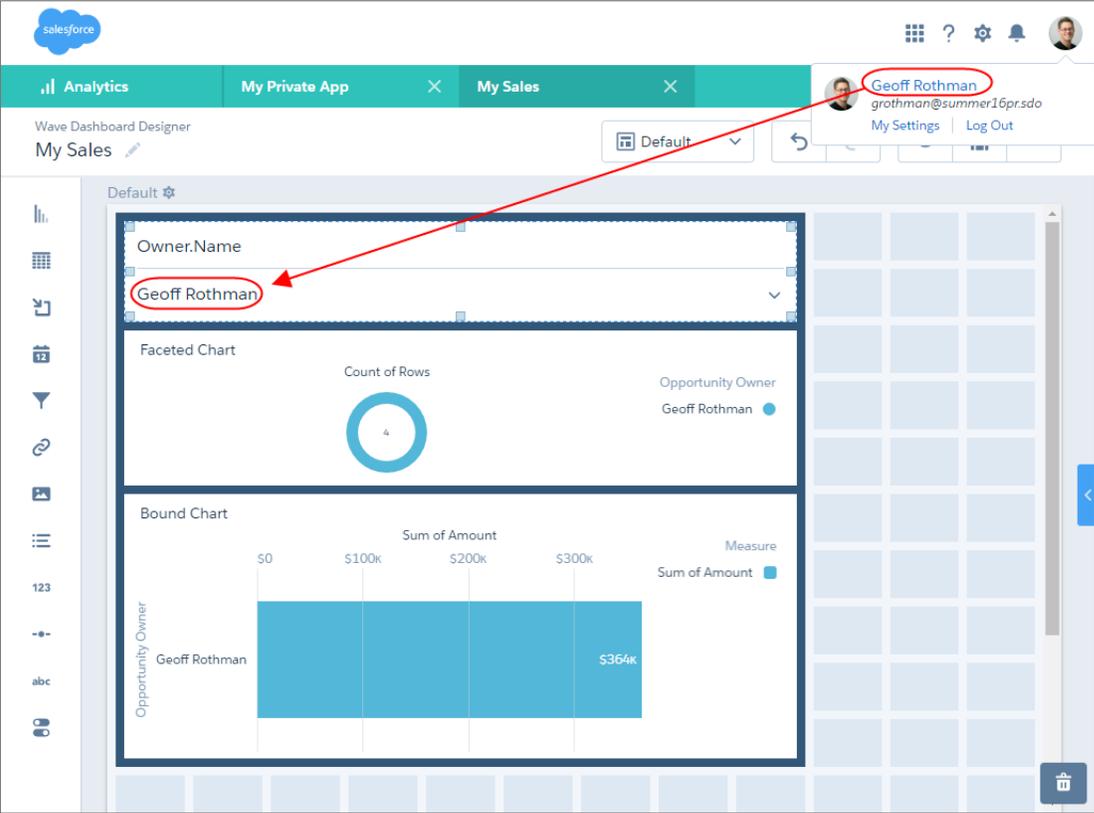
To add the filter, create a list widget and then, in the dashboard JSON, add an initial filter selection based on the `user.name` token in the widget's query (`Owner_Name_1`). This selection filters the following charts:

- Donut chart because its query (`Owner_Name_3`) is faceted with query `Owner_Name_1`.
- Bar chart because its query (`Owner_Username_3`) contains a binding based on query `Owner_Name_1`.

See the bold text in the following dashboard JSON to see how to set the initial filter, as well as facet and bind the queries.

```
{
    "label": "My Sales",
    "state": {
        "gridLayouts": [
        …
        ],
        "layouts": [],
        "steps": {
            "Owner_Name_1": {
                "datasets": [
                    {
                        "id": "0FbB00000000pNNKAY",
                        "label": "Opportunities",
                        "name": "opportunity",
                        "url": "/services/data/v38.0/wave/datasets/0FbB00000000pNNKAY"

                    }
                ],
                "isFacet": true,
                "isGlobal": false,
                "query": {
                    "measures": [
                        [
                            "count",
                            "*"
                        ]
                    ],
                    "groups": [
                        "Owner.Name"
                    ]
                },
                "selectMode": "multi",
                "start": [
                    "!{User.Name}"
                ],
                "type": "aggregateflex",
                "useGlobal": false,
                "visualizationParameters": {
                    "options": {}
                }
            },
            "Owner_Name_3": {
                "datasets": [
                    {
                        "id": "0FbB00000000pNNKAY",
                        "label": "Opportunities",
                        "name": "opportunity",
```

```
                            "url": "/services/data/v38.0/wave/datasets/0FbB00000000pNNKAY"

                        }
                    ],
                    "isFacet": true,
                    "isGlobal": false,
                    "query": {
                        "measures": [
                            [
                                "count",
                                "*"
                            ]
                        ],
                        "groups": [
                            "Owner.Name"
                        ],
                        "order": [
                            [
                                -1,
                                {
                                    "ascending": false
                                }
                            ]
                        ]
                    },
                    "selectMode": "single",
                    "type": "aggregateflex",
                    "useGlobal": true,
                    "visualizationParameters": {
                        "visualizationType": "hbar",
                        "options": {}
                    }
                },
                "Owner_Username_3": {
                    "datasets": [
                        {
                            "id": "0FbB00000000pNNKAY",
                            "label": "Opportunities",
                            "name": "opportunity",
                            "url": "/services/data/v38.0/wave/datasets/0FbB00000000pNNKAY"

                        }
                    ],
                    "isFacet": false,
                    "isGlobal": false,
                    "query": {
                        "measures": [
                            [
                                "sum",
                                "Amount"
                            ]
                        ],
                        "groups": [
                            "Owner.Name"
```

```
                    ],
                    "filters": [
                        [
                            "Owner.Name",
                            "{{column(Owner_Name_1.selection,
[\"Owner.Name\"]).asObject()}}",
                            "in"
                        ]
                    ],
                    "order": [
                        [
                            -1,
                            {
                                "ascending": false
                            }
                        ]
                    ]
                },
                "selectMode": "multi",
                "type": "aggregateflex",
                "useGlobal": true,
                "visualizationParameters": {
                    "visualizationType": "hbar",
                    "options": {}
                }
            }
        },
        "widgetStyle": {
            "backgroundColor": "#FFFFFF",
            "borderColor": "#E6ECF2",
            "borderEdges": [],
            "borderRadius": 0,
            "borderWidth": 1
        },
        "widgets": {
            "listselector_2": {
                "parameters": {
                    "compact": false,
                    "expanded": true,
                    "exploreLink": false,
                    "instant": true,
                    "measureField": "count",
                    "step": "Owner_Name_1",
                    "title": "Owner.Name"
                },
                "type": "listselector"
            },
            "chart_2": {
                "parameters": {
                    "autoFitMode": "fit",
                    "showValues": true,
                    "legend": {
                        "showHeader": true,
                        "show": true,
```

```
                    "position": "right-top",
                    "inside": false
                },
                "axisMode": "sync",
                "visualizationType": "hbar",
                "exploreLink": true,
                "title": {
                    "label": "Bound Chart",
                    "align": "left",
                    "subtitleLabel": ""
                },
                "trellis": {
                    "enable": false,
                    "type": "x",
                    "chartsPerLine": 4
                },
                "measureAxis2": {
                    "showTitle": true,
                    "showAxis": true,
                    "title": ""
                },
                "measureAxis1": {
                    "showTitle": true,
                    "showAxis": true,
                    "title": ""
                },
                "theme": "wave",
                "step": "Owner_Username_3",
                "dimensionAxis": {
                    "showTitle": true,
                    "showAxis": true,
                    "title": ""
                }
            },
            "type": "chart"
        },
        "chart_4": {
            "parameters": {
                "visualizationType": "pie",
                "exploreLink": true,
                "title": {
                    "label": "Faceted Chart",
                    "align": "left",
                    "subtitleLabel": ""
                },
                "step": "Owner_Name_3",
                "theme": "wave",
                "legend": {
                    "show": true,
                    "showHeader": true,
                    "inside": false,
                    "position": "right-top"
                },
                "trellis": {
```

```
                        "enable": false,
                        "type": "x",
                        "chartsPerLine": 4
                    },
                    "showMeasureTitle": true,
                    "showTotal": true,
                    "inner": 70
                },
                "type": "chart"
            },
            "container_1": {
                "type": "container",
                "parameters": {
                    "documentId": "",
                    "fit": "original",
                    "alignmentX": "left",
                    "alignmentY": "top"
                }
            }
        }
    },
    "datasets": [
        {
            "id": "0FbB00000000pNNKAY",
            "label": "Opportunities",
            "name": "opportunity",
            "url": "/services/data/v38.0/wave/datasets/0FbB00000000pNNKAY"
        }
    ]
}
```

For information about editing the dashboard JSON, see the Analytics Cloud Dashboard JSON Reference.

Set Initial Global Filters in the Dashboard

If the Tableau CRM dashboard contains global filters, you can set the initial filters that are applied when the dashboard first opens. If a global filter is unlocked, the dashboard viewer can change the filter while viewing the dashboard. Global filters apply to all queries in the dashboard unless a query is configured to ignore these filters.

1. While editing a dashboard, select a global filter, and then click **Pick Initial Values** (1).
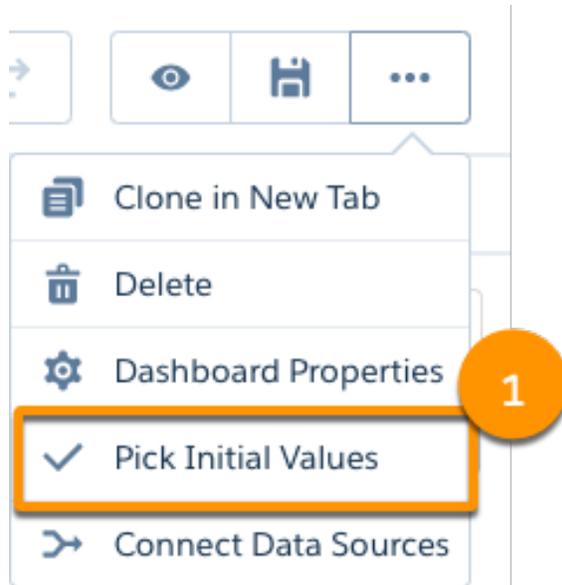
2. Choose the initial filter value for each global filter, then click **Apply**.

3. Click **Done**, and save the dashboard.

SEE ALSO:

Add a Global Filter Panel Widget to Filter Data from the Dashboard

Generate Unique Tableau CRM Dashboard Layouts for Different Devices

After you add widgets to the dashboard, optimize the layout for each device on which the dashboards can be viewed. For example, you can remove widgets from a mobile phone layout to reduce the dashboard size for the smaller screen. You can also move widgets around in one layout and it doesn't affect the other layouts.

Watch a Demo: ▶ Optimize Wave Dashboards for Devices with Layouts (English Only)

When you generate a new layout, Tableau CRM adds the widgets from the currently selected layout to the new layout. If the current layout contains multiple pages, Tableau CRM also copies the pages to the new layout.

📝 Note: Although you can add all chart types to a mobile layout, some aren't supported on mobile devices. For more information, see Tableau CRM mobile limitations.

1. Open in the dashboard designer dashboard.

2. Because each layout can contain different widgets, open the layout that contains the widgets that you want to include in your new layout.

3. From the Layouts menu, select **Manage Layouts**.

4. Select the layout template, and enter a unique layout name.

Each layout template comes with predefined layout properties that you can change.

5. Click **Generate**.

The layout properties appear for the new layout. If needed, click ⚙️ to show the Layout panel.

6. In the Layout panel, expand the following sections and change the default properties, if needed.

General

Set the layout name, designer grid settings, and background color.

Device

Specify information about the devices that can use this layout. For more information about how Tableau CRM uses these properties to choose the right layout, see Rules for Choosing a Layout for a Device.

Background Image

To apply a background image to the entire dashboard when this layout is used, enter the details about the background image. You don't have to include a background image.

The designer previews the dashboard and layout based on the layout property settings. The designer updates the preview in real time so that you can see how your changes affect the display. If you shrink the maximum dashboard width, the designer shows you how the dashboard fits when using this layout. If needed, the designer rearranges the widgets to fit the new size.

7. Rearrange the widgets, if needed.

8. To hide a widget from the layout, select the widget, and click 🗑 .
   If you hide a widget, it appears in the Unused Widgets section of the layout properties.

9. To add an unused widget to the layout, drag the widget from the Unused Widgets section to the canvas.

10. To create a widget, drag the widget from the widget toolbar to the canvas.

> ✏️ **Note:** Tableau CRM adds the widget only to the current layout. To add the widget to another layout, open the other layout and drag the widget from the "Unused Widgets" section of the layout properties.

11. To save your layout changes, save the dashboard.

12. To delete a layout, in the Layouts menu, click **Manage Layouts**, and then click ✕ next to the layout.

> **Note:** You can't delete the layout that's currently open. To delete the current layout, switch to another one first.

[Layout Properties for Tableau CRM Dashboards](#)

Layout properties specify the layout name, designer grid settings, background settings, and requirements for devices that can use this layout. The properties apply only to dashboards created in the dashboard designer.

[Rules for Choosing a Layout for a Device](#)

If multiple layouts are defined for a dashboard, Tableau CRM chooses the optimal layout when displaying the dashboard on a device. To determine the optimal layout, Tableau CRM uses the device properties specified for each layout.

Layout Properties for Tableau CRM Dashboards

Layout properties specify the layout name, designer grid settings, background settings, and requirements for devices that can use this layout. The properties apply only to dashboards created in the dashboard designer.

General Properties

These general properties apply to the specified layout.

Property	Description
Name	Name of the layout.
Columns	Number of columns in the dashboard designer grid. Increase the number of columns to increase the precision in the layout.
Row Height	Height of the rows in the dashboard designer grid. Values are **Normal** and **Fine**. When using the **Fine** option, the vertical spacing is set to 0 and can't be changed. Tableau CRM determines the default sizes of widgets based on number of columns and the row height to ensure that the widgets are readable and retain their relative sizes.
Cell Spacing	Horizontal and vertical spacing (in pixels) between cells in the dashboard designer grid.
Maximum Dashboard Width	Maximum width (in pixels) that the dashboard can use. If needed, Tableau CRM rearranges the existing dashboard widgets based on this setting in the layout.
Background Color	You can set the background color of the following areas. **Grid** Color of the dashboard designer grid. **Gutter** Color of the gutter that appears when the device's screen is wider than the dashboard's maximum width.

Device Properties

Tableau CRM uses these values to choose the optimal layout for the device accessing the dashboard.

Property	Description
Screen Width	Width of the devices supported by this layout. Specify the minimum and maximum width (in pixels).
Orientation	Orientation of the devices supported by this layout.
Platform	Platform of the devices supported by this layout. Select one of the following values: **iOS** For iOS-specific mobile devices that access Tableau CRM apps from the Tableau CRM mobile app. **Android** For Android-specific mobile devices that access Tableau CRM apps from the Tableau CRM mobile app. **All** For all other devices, including mobile devices that access Tableau CRM apps from a web browser or the Salesforce mobile app. Basically, if you access a Tableau CRM app using any method other than the Tableau CRM mobile app, you must set this option to **All**.

Background Image Properties

Property	Description
Image	The file name of the image. The image must be available in the gallery of the Tableau CRM app that contains the dashboard. This property applies to background images created during or after Spring '17.
Image ID	15-character ID of the image. The image must be uploaded to the Documents tab of Salesforce with the Externally Available Image option enabled. This property applies to background images created before Spring '17.
Image Scale	Indicate how to fit the image in the background.
Image Alignment	Horizontal and vertical alignment of the background image.

Rules for Choosing a Layout for a Device

If multiple layouts are defined for a dashboard, Tableau CRM chooses the optimal layout when displaying the dashboard on a device. To determine the optimal layout, Tableau CRM uses the device properties specified for each layout.

Tableau CRM uses the following rules when determining which layout to use for a device.

1. A layout is eligible for use when the device accessing the dashboard meets all the device properties set in the Layout panel.

2. If more than one layout is eligible, the one with the most device properties set is used. If there's a tie, the most recently defined layout is used.

3. If no layouts are eligible with the device, the first defined layout is used.

Optimize Dashboard Performance

Before you finalize the dashboard, run a performance check on the dashboard and its queries to ensure that everything is running optimally. The dashboard inspector identifies different types of bottlenecks, like query issues and redundant queries, and provides recommendations to improve performance. Because dashboard layouts can contain different widgets (and queries), run the inspector on each layout. If a dashboard contains multiple pages, run the inspector on each page. The inspector provides results only for the current page.

1. While browsing the app, open the dashboard in view mode.

   The dashboard inspector is not available while running the app, editing the dashboard, or previewing an embedded dashboard outside of Tableau CRM.

2. To open the dashboard inspector, click ⋯ and then **Dashboard Inspector**.
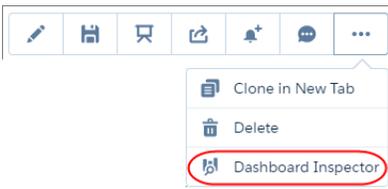
3. To review the dashboard-level performance, open the Performance tab in the dashboard inspector.
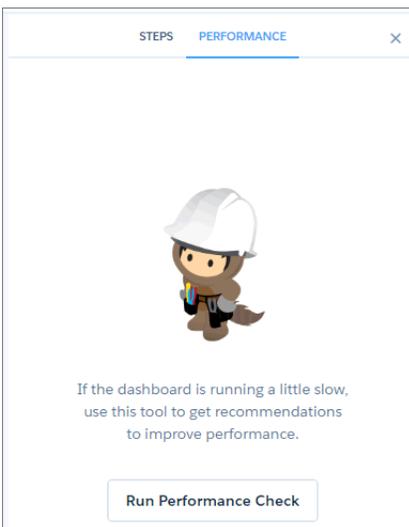
4. Click **Run Performance Check**.
   The Performance Factors section shows all factors by which the dashboard is evaluated.

> **Note:** If your dashboard contains multiple pages, the dashboard inspector shows results for the current page being viewed.

5. If a warning icon (⚠) appears next to a factor, expand the factor to learn how to fix the issue to improve the dashboard's performance.

Ignore the factors with a green checkmark (✓) because these issues don't apply to the dashboard.

6. To analyze the performance of all queries in the dashboard, click the Queries tab.

The table lists all queries that ran when the dashboard loaded. If the dashboard contains multiple pages, the table lists the queries that ran for the current page. Each time you switch pages, newly run queries are added to the table, and previously run queries, which are no longer used, are listed as unused.

The table shows the following metrics for each query.

Metric	Description
# Runs	Number of times the query ran. Queries may run multiple times because of faceting or bindings. This number includes runs that execute the query or use cached results.
First (MS)	Number of milliseconds to run the query for the first time. The first run typically, but not always, executes the query.
Avg (MS)	Average number of milliseconds to run the query. Only runs that execute the query are included in this calculation.

7. Select a query in the table to view additional details about the query and the last run.

   For more information about query performance, see Optimize a Query .

   ✎ Note: Prior to Summer '19, query level information was only available if the query ran successfully. In Summer '19 and later, when a query has errors and fails to execute, open the Dashboard Inspector (1) and select the query with errors (2). Use the View More Details button (3) for more information about bindings syntax errors, query level errors, and the query that was run.
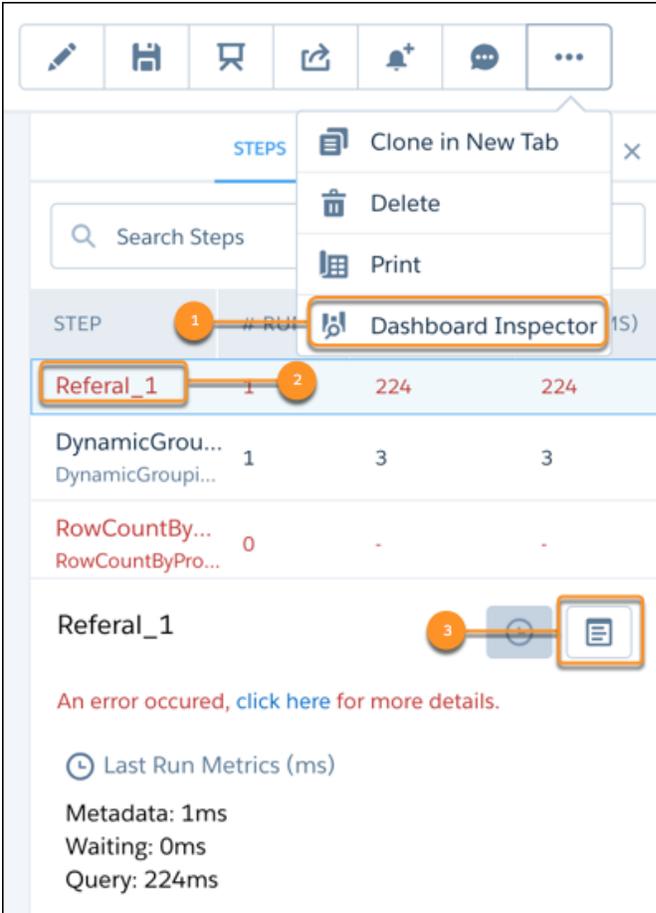
Keyboard Shortcuts for Building Tableau CRM Dashboards and Lenses

You can do some basic actions from your keyboard.

Lens Keyboard Shortcut	Description
L	Clip to dashboard designer
S	Save
N	Clone in new tab
Ctrl+E (Windows) Cmd+E (macOS)	Open JSON Editor

Dashboard Keyboard Shortcut	Description
Tab	Focus on next item in the dashboard.
Shift+Tab	Focus on previous item in the dashboard.

Dashboard Keyboard Shortcut	Description
Spacebar	Select a widget to move it when editing the dashboard.
Arrow keys	Move the selected widget when editing the dashboard.
C	Share the dashboard when viewing it.
E	Toggle between previewing or editing the dashboard.
S	Save the dashboard.
X	Redo the change.
Z	Undo the latest change.
Delete	Delete the selected widgets while editing the dashboard.
Drag+Shift	Reflow other widgets down when dragging a widget in the dashboard canvas. Note: Drag the widget before you press **Shift**.
Ctrl+Click (Windows) Cmd+Click (macOS)	Select multiple widgets while editing the dashboard.
Ctrl+A (Windows) Cmd+A (macOS)	Select all widgets in the canvas while editing the dashboard.

JSON Editor Keyboard Shortcut	Description
Ctrl+E (Windows) Cmd+E (macOS)	View dashboard with changes to JSON.
Ctrl+3 (Windows) Cmd+3 (macOS)	Disregard changes and load the original JSON.
Ctrl+D (Windows) Cmd+D (macOS)	Delete line
Alt+Shift+Down (Windows) Cmd+Option+Down (macOS)	Copy lines down
Alt+Shift+Up (Windows) Cmd+Option+Up (macOS)	Copy lines up
Alt+Down (Windows)	Move lines down

JSON Editor Keyboard Shortcut	Description
Option+Down (macOS)	
Alt+Up (Windows) Option+Up (macOS)	Move lines up
Alt+Delete (Windows) Ctrl+K (macOS)	Delete to end of line
Alt+Backspace (Windows) Cmd+Delete (macOS)	Delete to start of line
Ctrl+Delete (Windows)	Delete word to the right
Ctrl+A (Windows) Cmd+A (macOS)	Select all
Shift+Left (Windows) Shift+Left (macOS)	Select character to the left
Shift+Right (Windows) Shift+Right (macOS)	Select character to the right
Ctrl+Shift+Left (Windows) Option+Shift+Left (macOS)	Select word to the left
Ctrl+Shift+Right (Windows) Option+Shift+Right (macOS)	Select word to the right
Shift+Home (Windows) Shift+Cmd+Left (macOS)	Select to start of line
Shift+End (Windows) Shift+Cmd+Right (macOS)	Select to end of line
Shift+Up (Windows) Shift+Up (macOS)	Select up
Shift+Down (Windows) Shift+Down (macOS)	Select down

JSON Editor Keyboard Shortcut	Description
Shift+PageUp (Windows)	Select page up
Shift+PageDown (Windows)	Select page down
Ctrl+Shift+Home (Windows) Cmd+Shift+Up (macOS)	Select to start
Ctrl+Shift+End (Windows) Cmd+Shift+Down (macOS)	Select to end
Ctrl+Shift+D (Windows) Cmd+Shift+D (macOS)	Duplicate selection
Ctrl+Shift+P (Windows)	Select to matching bracket
Left (Windows) Left, Ctrl+B (macOS)	Go to left
Right (Windows) Right, Ctrl+F (macOS)	Go to right
Ctrl+Left (Windows) Option+Left (macOS)	Go to word left
Ctrl+Right (Windows) Option+Right (macOS)	Go to word right
Up (Windows) Up (macOS)	Go line up
Down (Windows) Down, Ctrl+N (macOS)	Go line down
Alt+Left, Home (Windows) Cmd+Left, Ctrl+A (macOS)	Go to line start
PageUp (Windows)	Go to page up
PageDown (Windows)	Go to page down

JSON Editor Keyboard Shortcut	Description
Ctrl+Home (Windows) Cmd+Up (macOS)	Go to start
Ctrl+End (Windows) Cmd+Down (macOS)	Go to end
Ctrl+L (Windows) Cmd+L (macOS)	Go to line
Ctrl+P (Windows)	Go to matching bracket
Ctrl+F (Windows) Cmd+F (macOS)	Find
Ctrl+H or Ctrl+F twice (Windows) Cmd+Option+F (macOS)	Replace
Ctrl+K (Windows) Cmd+G (macOS)	Find next
Ctrl+Shift+K (Windows) Cmd+Shift+G (macOS)	Find previous

SAQL Editor Keyboard Shortcut	Description
Ctrl+D (Windows) Cmd+D (macOS)	Delete line
Alt+Shift+Down (Windows) Cmd+Option+Down (macOS)	Copy lines down
Alt+Shift+Up (Windows) Cmd+Option+Up (macOS)	Copy lines up
Alt+Down (Windows) Option+Down (macOS)	Move lines down
Alt+Up (Windows)	Move lines up

SAQL Editor Keyboard Shortcut	Description
Option+Up (macOS)	
Alt+Delete (Windows) Ctrl+K (macOS)	Delete to line end
Alt+Backspace (Windows) Cmd+Delete (macOS)	Delete to line start
Ctrl+Delete (Windows) Option+Delete (macOS)	Delete word right
Ctrl+A (Windows) Cmd+A (macOS)	Select all
Shift+Left (Windows) Shift+Left (macOS)	Select left
Shift+Right (Windows) Shift+Right (macOS)	Select right
Ctrl+Shift+Left (Windows) Option+Shift+Left (macOS)	Select word left
Ctrl+Shift+Right (Windows) Option+Shift+Right (macOS)	Select word right
Shift+Home (Windows)	Select to line start
Shift+End (Windows)	Select to line end
Shift+Up (Windows) Shift+Up (macOS)	Select up
Shift+Down (Windows) Shift+Down (macOS)	Select down
Shift+PageUp (Windows)	Select page up
Shift+PageDown (Windows)	Select page down

SAQL Editor Keyboard Shortcut	Description
Ctrl+Shift+Home (Windows) Cmd+Shift+Up (macOS)	Select to start
Ctrl+Shift+End (Windows) Cmd+Shift+Down (macOS)	Select to end
Ctrl+Shift+D (Windows) Cmd+Shift+D (macOS)	Duplicate selection
Ctrl+Shift+P (Windows)	Select to matching bracket
Left (Windows) Left, Ctrl+B (macOS)	Go to left
Right (Windows) Right, Ctrl+F (macOS)	Go to right
Ctrl+Left (Windows) Option+Left (macOS)	Go to word left
Ctrl+Right (Windows) Option+Right (macOS)	Go to word right
Up (Windows) Up, Ctrl+P (macOS)	Go line up
Down (Windows) Down, Ctrl+N (macOS)	Go line down
Alt+Left, Home (Windows) Cmd+Left, Ctrl+A (macOS)	Go to line start
Alt+Right, End (Windows) Cmd+Right(macOS)	Go to line end
PageUp (Windows)	Go to page up
PageDown (Windows) Ctrl+V (macOS)	Go to page down

SAQL Editor Keyboard Shortcut	Description
Ctrl+Home (Windows) Cmd+Up (macOS)	Go to start
Ctrl+End (Windows) Cmd+Down (macOS)	Go to end
Ctrl+L (Windows) Cmd+L (macOS)	Go to line
Ctrl+P (Windows)	Go to matching bracket
Ctrl+H or Ctrl+F twice (Windows) Cmd+F (macOS)	Find
Ctrl+H (Windows) Cmd+Option+F (macOS)	Replace
Ctrl+K (Windows) Cmd+G (macOS)	Find next
Ctrl+Shift+K (Windows) Cmd+Shift+G (macOS)	Find previous

Restore a Previous Version of a Dashboard

Tableau CRM uses version history to back up dashboard versions when you edit them so you can restore a previous version.

Here's how to restore a previous version of a dashboard.

1. Click the [···] button at upper right.

2. Select the **Version** tab to see a list of previous versions of the dashboard.
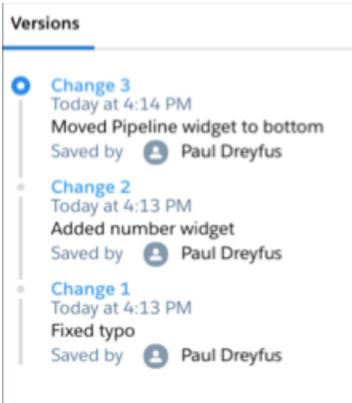
3. Click the links (**Change 1**, **Change 2**, and so on) to view previous versions. The text under the links provides a brief description of the version if one was added to the **Version History** field the last time someone saved the dashboard.

4. When you see the one you want to restore, click the disk icon and then click **Save**. It's a good idea to enter a brief description of what's unique about this version in the **Version History** field for future reference.

⚠️ Warning: When you restore a previous version, the dashboard may not run as expected if it depends on a dashboard, dataset, or lens that's been deleted. If you get an error when you load a restored version of a dashboard, try restoring older versions until you get one that works. You can use that one as a starting point to rebuild the dashboard with errors.

For details, see Backup and Restore Previous Versions of Tableau CRM Assets with History API.

Collaborate on a New Dashboard Version Behind the Scenes (Pilot)

Tableau CRM users can add themselves to the list of dashboard publishers so they can edit and test new versions of the dashboard. Other Tableau CRM users continue to see a live version until a publisher makes the draft the live version.

📝 Note: We provide the publisher feature to selected customers through a pilot program that requires agreement to specific terms and conditions. To be nominated to participate in the program, contact Salesforce. Pilot programs are subject to change, and we can't guarantee acceptance. The feature isn't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features.

To enable Tableau CRM users to add themselves to a dashboard publisher list, give them Editor or Manager access to the app that contains the dashboard. Set access level to an app and all its contents when you share it with users.

1. Add Yourself to the List of Dashboard Publishers

    Tableau CRM users with Editor or Manager access to the dashboard can add themselves to the list of dashboard publishers.

2. Collaborate on and Publish a Draft Version of a Dashboard

    If you are listed as a publisher, you can review and edit a previous dashboard version. Then save it as the draft version for other publishers to review and publish it as the live version that all Tableau CRM users view.

Add Yourself to the List of Dashboard Publishers

Tableau CRM users with Editor or Manager access to the dashboard can add themselves to the list of dashboard publishers.

1.
Click the [···] button at upper right.

2. Select **Version History**.

3. Open the **Publishers** tab and click **Add yourself as a Publisher**.

To remove yourself from the publisher list, click the x to the right of your name in the publisher list.

Collaborate on and Publish a Draft Version of a Dashboard

If you are listed as a publisher, you can review and edit a previous dashboard version. Then save it as the draft version for other publishers to review and publish it as the live version that all Tableau CRM users view.

Make sure you've opened Version History for the dashboard you want to work on and that you've made yourself a Publisher.

Select the **Versions** tab to see a list of all versions of the dashboard. The version at the top of the list is the live version that all Tableau CRM users see. In the default state, when you open, edit, and save any version of the dashboard, it becomes the live version that other users view. It also becomes the newest version at the top of the list.

Here's how to work on a draft version behind the scenes.

1. In the **Versions** tab, click the triangle to the right of the version link at the top of the list and select **Make Live**. It's now marked **Live** in the **Versions** tab.

2. Open any version of the dashboard (including the live version). Edit and save it.

3. Click the ⎡ ⋯ ⎤ icon and select **Version History**.

4. The version you edited and saved now appears at the top of the list as a new version. It's marked **Draft**, which means that only other publishers can view it. Other users in the org continue to view the version marked **Live**. Anytime you edit and save the dashboard, that version appears at the top of the list and is marked **Draft**.

5. Other publishers can open the dashboard, click **Version History**, and view the version of the dashboard marked **Draft**.

6. Once you decide the draft version is ready for other users, click the triangle to the right of the version link and select **Make Live** to publish it. The published version becomes the live version that other Tableau CRM users view.

Embed and Customize Tableau CRM

Extend Tableau CRM everywhere throughout your business. The Tableau CRM visualizations you've built are more powerful when you share them across your Salesforce experience by integrating them into custom pages, Visualforce pages, Experience Cloud sites, and more. In addition, custom menus in lenses and dashboards allow you to perform common Salesforce actions directly from Tableau CRM.

Embed Dashboards Everywhere
Embed Tableau CRM dashboards in every Salesforce experience. From embedded dashboards, users can explore and click to linked assets. If you set them up with a Share icon, embedded dashboards offer Post to Feed and Download sharing options.

Enable Actions for Tableau CRM Lenses and Dashboards
Let users take action on insights directly from Tableau CRM. Add actions to charts and tables in lenses and dashboards.

Integrate Tableau CRM into Salesforce with an Analytics Tab
Access your Tableau CRM home and run Tableau CRM apps from a tab within Salesforce Classic or Lightning Experience.

Embed Dashboards Everywhere

Embed Tableau CRM dashboards in every Salesforce experience. From embedded dashboards, users can explore and click to linked assets. If you set them up with a Share icon, embedded dashboards offer Post to Feed and Download sharing options.

Compare the options for embedding dashboards.

User Interface	Where	How	Component
Lightning Experience only	Home, record, and app home pages	Component in the Lightning App Builder	Wave Dashboard (wave:waveDashboard)
Salesforce Classic only	Record detail page	Component in the enhanced page layout editor	Wave Analytics Asset
Lightning Experience and Salesforce Classic	Any page	Visualforce page component	wave:dashboard
Salesforce App	Lightning app page	Lightning App Builder	Wave Dashboard (wave:waveDashboard)
Experience Cloud sites	Experience Builder page or Visualforce page	Component in a Experience Builder–based template or Salesforce Tabs + Visualforce template	Wave Dashboard (forceCommunity:waveDashboard) or wave:dashboard

Depending on the device and the embedding option, these actions can be taken from embedded Tableau CRM dashboards.

- Save and manage dashboard views
- Post to Chatter
- Download images
- Set notifications
- Subscribe to widgets
- Access global filters
- Link to other Tableau CRM dashboards
- Take action in Salesforce through record actions menus
- Collaborate on dashboards with annotations (available for Experience Cloud site users only)
- Link to the native app ![icon], whether it's Tableau CRM Studio on desktop or Tableau CRM on mobile (available for Tableau CRM users but not Experience Cloud site users)

Embedded Tableau CRM dashboards have these limitations.

- Dashboards can't be created, updated, or deleted.
- The app navigation menu isn't available.
- Full-screen presentation mode isn't available.
- Sharing options don't include Give Access, Get URL, printing, or posting the link to a dashboard.
- Annotations and conversational exploration aren't available.
- The charts don't include hover-over information (tooltips).
- Some filters, such as date selectors, are difficult to set.

📝 **Note:** For limitations of embedded dashboards viewed in the Salesforce mobile app, see View Embedded Dashboards on Mobile Devices.

Embed Tableau CRM Dashboards in Lightning Pages

Add Tableau CRM dashboards and dashboard components to Lightning home pages, record pages, and app home pages to provide interactive visualizations of your data. Users can drill in and explore the dashboard within the frame on the Lightning page or in a Tableau CRM window.

Share Tableau CRM in Experience Cloud Sites

With Tableau CRM in Experience Cloud sites, external users can view apps shared with them via Tableau CRM dashboards embedded in their site. You can embed dashboards using Experience Builder or Visualforce pages.

Add a Tableau CRM Dashboard to a Visualforce Page

By embedding a Tableau CRM dashboard into a Visualforce page, you can provide an interactive presentation of your data. Users can drill in and explore the dashboard within the frame on the Visualforce page or in a new window in Analytics Cloud.

Embed Tableau CRM Dashboards in Salesforce Classic Pages

Add a Tableau CRM dashboard to a detail page layout. On an account detail page, for example, you can include a dashboard of service issues associated with the account. Users can drill in, apply filters, and explore in the dashboard as if they were viewing it in a Tableau CRM window.

View Embedded Dashboards on Mobile Devices

Add Tableau CRM dashboard components to Lightning app pages. Embedded Tableau CRM dashboards can responsively select layouts optimized for any device.

Allow Trusted Sites for Embedded Dashboards

Embed Tableau CRM in any web page or web-based app. By specifying your trusted sites in the Tableau CRM allowlist, you can include embedded Tableau CRM dashboards in websites and apps outside of Salesforce servers.

Filter and Selection Syntax for Embedded Dashboards

To specify types of filters or dashboard selections that cannot be created using the Filter Builder, enter JSON strings in the Filter String box of the embedded Tableau CRM dashboard attributes.

Embed Tableau CRM Dashboards in Lightning Pages

Add Tableau CRM dashboards and dashboard components to Lightning home pages, record pages, and app home pages to provide interactive visualizations of your data. Users can drill in and explore the dashboard within the frame on the Lightning page or in a Tableau CRM window.

Go to an existing Lightning Experience page, select the wheel icon at upper right, then click **Edit Page**. Or, go to Lightning App Builder, click **New**, and follow the prompts to create a new Lightning page. Then, follow these instructions to embed a dashboard in the page.

1. Select **Einstein Analytics Dashboard** from the list of components.

2. From the Dashboard dropdown list, select the dashboard to display.

   > 📝 **Note:** (For Developers) In the code, the developerName attribute is used for the dashboard name. If configuring the component programmatically, you can use either the the developerName attribute or the dashboardId attribute. The developerName can be requested through the API. The dashboardId is the 18-character code beginning with 0FK that is found in the URL when viewing a dashboard.

3. From the layout dropdown list, select the type of screen where your Lightning page will be viewed.

4. Configure the attributes of the embedded Analytics Cloud dashboard:

- **Height**—Specify the height of the dashboard, in pixels. The default is 300.

- **Show Sharing Icon**—Add the Share icon on the dashboard. When this option is selected, users can click the icon to open the Share dialog, where they can post to Chatter and download images and data. To show the Share icon, the minimum dashboard height is 612 pixels. The default is false.

- **Show Title**—Control the visibility of the dashboard title. When this option is selected, the dashboard's title appears above the dashboard. The default is true.

- **Show Header**—Control the visibility of the dashboard header. When this option is selected, the dashboard is displayed with a header bar that includes the Open in Analytics icon, the date and time that the dashboard's data was updated, and the dashboard views menu. Note that the header bar also appears if either Show Sharing Icon or Show Title is selected.

- **Open Dashboard**—Specify where the dashboard opens when users click the open icon. The default behavior is to open the dashboard in Tableau CRM Studio. Other options include opening the dashboard in the Salesforce Lightning Analytics tab or disabling the open icon.

- **Open Links in New Windows**—Specify where links from the dashboard to other assets are opened. When this option is selected, links open in new windows. When this option isn't selected, links open in the same window. The default is true.

- **Hide On Error**— Control whether or not users see a dashboard that has an error. When this option is selected, if the dashboard has an error, it won't appear on the page. When this option isn't selected, the dashboard appears but doesn't show any data. An error can occur when a user doesn't have access to the dashboard or it has been deleted. The default is false.

- **Enable Notifications**— Allow users with permission to set conditions and get notified about updates to dashboard widgets. And when Show Header is true, this option also surfaces the notifications icon in the dashboard header which opens the notifications management panel for editing and deleting notifications. If you enable Show Header, it's recommended that the dashboard is at least 400 pixels in width to accommodate the management panel.

- **Enable Subscriptions**— Allow users with permission to subscribe for periodic email updates on dashboard widgets. And when Show Header is true, this option also surfaces the subscriptions icon in the dashboard header which opens the subscriptions management panel for editing and deleting subscriptions. If you enable Show Header, it's recommended that the dashboard is at least 400 pixels in width to accommodate the management panel.

- **Dashboard Filters**—Add selections or filters to the embedded dashboard at runtime. You can filter dataset fields by variables or specified values. Specify filters either by using the point-and-click Filter Builder **(record pages only)** or by entering JSON strings in the Filter String box. For examples of using JSON strings in the Filter String box, see Filter and Selection Syntax for Embedded Dashboards.

  Use the Filter Builder to set up a dashboard to show the contextual data for each record page in which it appears. Click **Filter Builder** and **+ Add Dashboard Filter**. Select a dataset, dataset field, and operator. Then click **Select** to specify a field in an sObject or custom object.
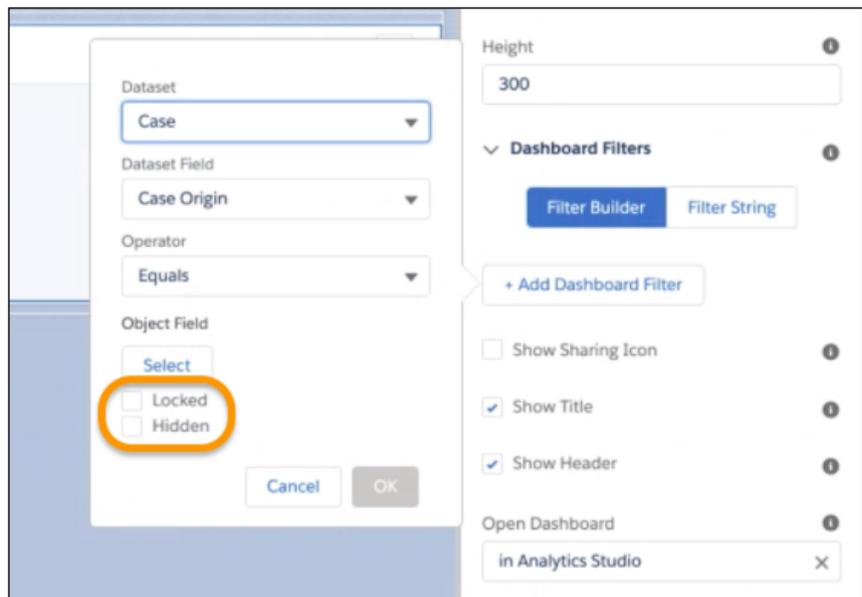
Prevent users from exposing more data by locking or hiding filters. With the **Locked** option, the Filters menu appears in the embedded dashboard header, but users can't modify or delete the filters. With the **Hidden** option, the Filters menu doesn't appear in the embedded dashboard header.

When you click **OK**, the filter appears in the configuration panel.
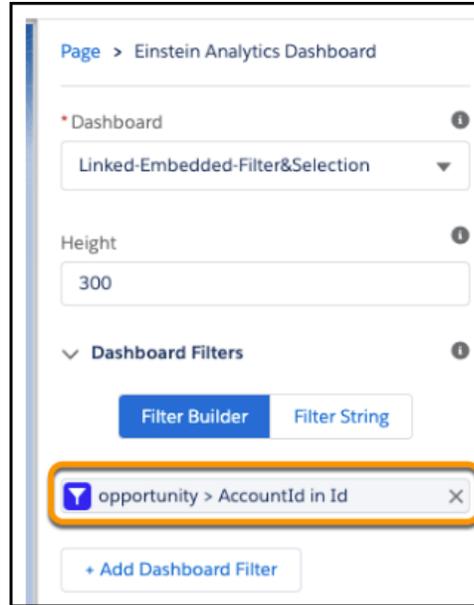
Click an existing filter to edit it in the Filter Builder, or click **X** to remove it. To view or edit the filter JSON, click **Filter String**.
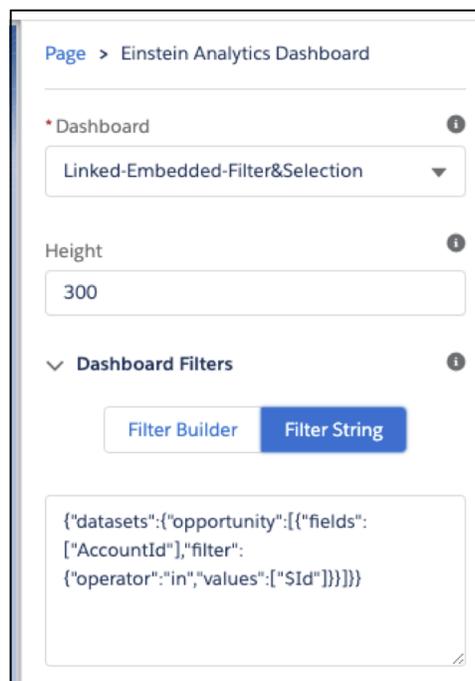
Note: For the Filter Builder, the dataset fields must be dimensions (not dates or measures), and the object field values must be tokenizable (not static values). A filter can't be viewed or edited in the Filter Builder if it was created or edited in the Filter String text box. If a filter or selection can't be applied, the attribute is ignored and the dashboard appears with all its data and no selection.

Note: When setting up locked or hidden filters for embedded dashboards, keep in mind the following considerations.

- Locked and hidden filters are applied on the dashboard. If a filter makes a field visible, users see the field even when you've restricted access to filters with one of these options.
- Locked and Hidden options aren't automatically applied in linked dashboards or lens explorations. The options are applied only on the dashboard where specified.
- Locked and Hidden options are applied to a dashboard when embedded or when opened in Tableau CRM Studio or Analytics tab.

For information about working with the Lightning App Builder, start on the **Lightning App Builder** overview page.

Share Tableau CRM in Experience Cloud Sites

With Tableau CRM in Experience Cloud sites, external users can view apps shared with them via Tableau CRM dashboards embedded in their site. You can embed dashboards using Experience Builder or Visualforce pages.

> 📝 **Note:** Only users with a Customer Community Plus, Partner Community, or Lightning External Apps Plus license can use this feature. This feature isn't supported for guest users.

Experience Cloud site users can view dashboards embedded in their site, and they can explore lenses, facets, and links in the embedded dashboards. Site users can't access the Tableau CRM home page or create, update, or delete any Tableau CRM asset. They also can't upload data. This table summarizes differences between internal Tableau CRM users and external site users.

	Tableau CRM Users	Site Users
Access, Explore, and Facet Dashboards and Lenses	✅	✅
Create, Update, and Delete Dashboards and Lenses	✅	
Share Dashboards and Lenses (post to Chatter and download images and data*)	✅	✅
Save Dashboard Views**	✅	✅
Access the Tableau CRM Studio or Tableau CRM Mobile App	✅	
Access Tableau CRM Prebuilt Apps	✅	
Analyze Large Datasets	✅	✅
Benefit from Security Predicates	✅	✅
Annotate Dashboard Widgets*	✅	✅
Subscribe to Lenses and Dashboard Widgets*	✅	
Set and Track Notifications	✅	
Open Salesforce Records	✅	✅
Perform Record-level Actions from Custom Action Menus	✅	✅
Perform Bulk Actions from Customized Widget Menus	✅	✅
View Charts based on a Timeseries Query	✅	

* Requires user permission. ** Requires setting enabled in Setup.

Here are the high-level steps for setting up Tableau CRM for Experience Cloud sites.

1. Enable `Share Analytics with Communities` and set up site members. See Enable Tableau CRM for Experience Cloud Sites.

2. Create Tableau CRM dashboards and save them in a Tableau CRM app that you can share with your site.

3. From the Tableau CRM app containing your dashboards, select **Share**.

4. In the Share dialog, select **Enable sharing with Communities**.
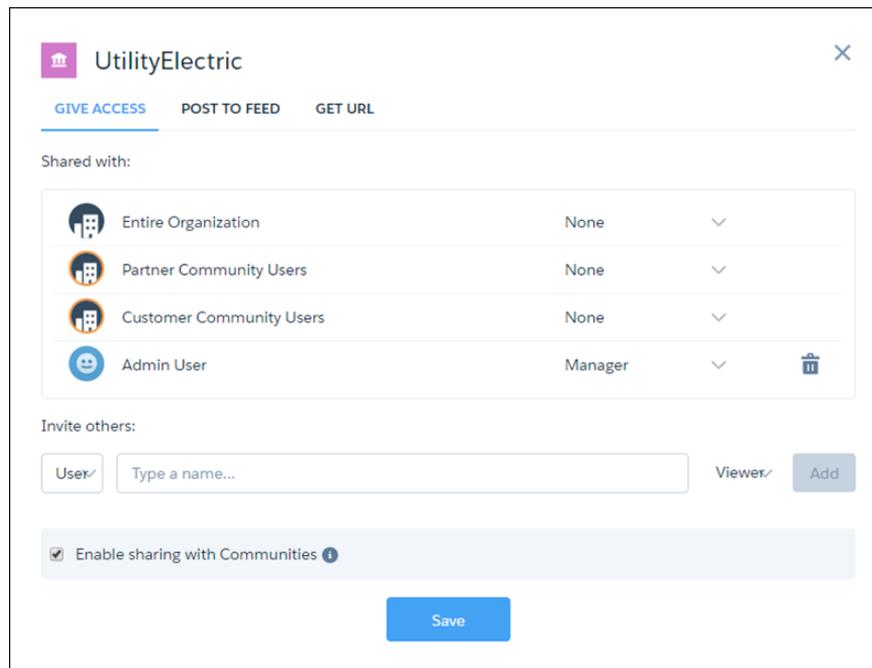
When an app is enabled for sharing with external users, a message appears to let you know at a glance. You can easily check and change the sharing status of apps.
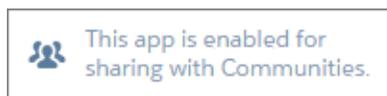
5. Share the app with site partners and customers by entering their names and clicking **Add**. Site members in the sharing list are highlighted in orange to distinguish them as external users, and Viewer is the only access available for them.

6. Embed your Tableau CRM dashboards using either Experience Builder or Visualforce.

Note: Apex steps aren't supported for dashboards embedded in sites.

Note: Security is enforced at these levels.

- By role, group, or user, via the Sharing dialog.
- At the Tableau CRM app level, via the **Enable sharing with Communities** checkbox.
- At the org setup level, via security predicates.

📝 **Note:** Access to the Tableau CRM REST API is limited to the GET method with these endpoints:

- /wave/folders/<folder ID> (Access to folders shared with the site)
- /wave/dashboards/<dashboard ID or API Name> (Access to dashboards belonging to a folder shared with the site)
- /wave/lenses/<lens ID or API Name> (Access to lenses belonging to a folder shared with the site)
- /wave/datasets/<dataset ID> (Access to datasets belonging to a folder shared with the site)

[Use Experience Builder to Embed Tableau CRM Dashboards in Experience Cloud Sites](#)

Add analytics via the Tableau CRM Dashboard component available in Experience Builder–based templates.

[Use Visualforce to Embed Tableau CRM Dashboards in Experience Cloud Sites](#)

Share analytics with external users via Tableau CRM dashboards embedded in their site.

[Collaborate on Tableau CRM Dashboards in Experience Cloud Sites with Annotations](#)

Foster an environment of open communication sparked by Tableau CRM dashboards embedded in Experience Cloud sites. With annotations, partners and customers can converse in Chatter feeds on individual dashboard widgets. Annotations are available whether dashboards are embedded via Visualforce or the Experience Builder.

[Take Action in Salesforce from Experience Cloud Sites](#)

From Tableau CRM in Experience Cloud sites, you can open Salesforce records and perform record-level actions and bulk actions. You set up an action menu on a dataset or a bulk action on a table widget. Then your site users can open records and take action in Salesforce directly from the Tableau CRM dashboard, without leaving your site.

Use Experience Builder to Embed Tableau CRM Dashboards in Experience Cloud Sites

Add analytics via the Tableau CRM Dashboard component available in Experience Builder–based templates.

To add Tableau CRM to your Experience Cloud site, follow the high-level steps in Share Tableau CRM in Experience Cloud Sites. When you're ready to embed a dashboard, use the following procedure.

1. Select the **Tableau CRM Dashboard** component in Experience Builder.

2. From the Dashboard dropdown list, select the dashboard to display. Dashboards appear in the list, and are visible in sites, when **Enable sharing with Communities** is selected in an app's Share dialog.

   > 📝 **Note:** (For Developers) In the code, the dashboard name is represented with the dashboard ID. The dashboardId is the 18-character code beginning with 0FK that is found in the URL when viewing a dashboard.

3. Configure the attributes of the embedded Tableau CRM dashboard.

* **Height**—Specify the height of the dashboard, in pixels. The default is 300.

* **Filter**—Add selections or filters to apply to the dashboard when it appears on the page at runtime. You can filter dataset fields by variables or specified values. Specify filters by entering JSON strings in the Filter box. For example JSON strings, see Filter and Selection Syntax for Embedded Dashboards.

* **Show Sharing Icon**—Add the Share icon on the dashboard. When this option is selected, users can click the icon to open the Share dialog, where they can post to Chatter and download images and data. To show the Share icon, the minimum dashboard height is 612 pixels. The default is false.

* **Show Title**—Control the visibility of the dashboard title. When this option is selected, the dashboard's title appears above the dashboard. The default is true.

* **Show Header**—Control the visibility of the dashboard header. When this option is selected, the dashboard is displayed with a header bar. The header bar includes the Open in Tableau CRM icon, the date and time that the dashboard's data was updated, and the dashboard views menu. The header bar also appears if either Show Sharing Icon or Show Title is selected.

* **Open Links in New Windows**—Specify where links from the dashboard to other assets are opened. When this option is selected, links open in new windows. When this option isn't selected, links open in the same window. The default is true.

* **Hide On Error**—Control whether users see a dashboard that has an error. When this option is selected, if the dashboard has an error, it doesn't appear on the page. When this option isn't selected, the dashboard appears but doesn't show any data. An error can occur when a user doesn't have access to the dashboard or it has been deleted. The default is false.

For information about working with Experience Builder, start with Customize Sites with Experience Builder.

EDITIONS

Available in: Salesforce Classic and Lightning Experience.

Available for an extra cost in: **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in: **Developer** Edition.

USER PERMISSIONS

To create, customize, or activate a site:
* Create and Set Up Experiences

  AND is a member of the site they're trying to update

To modify Tableau CRM settings:
* Analytics Growth permission set license with Manage Analytics permission

To view, explore, and share embedded Tableau CRM dashboards for users with an Experience Cloud license:
* Analytics for Communities permission set license with View Analytics on Communities pages permission

Use Visualforce to Embed Tableau CRM Dashboards in Experience Cloud Sites

Share analytics with external users via Tableau CRM dashboards embedded in their site.

1. To add Tableau CRM to your site, follow the high-level steps in Share Tableau CRM in Experience Cloud Sites.

2. Select the Visualforce page where you'll embed Tableau CRM, or create a new page. See Defining Visualforce Pages.

3. Optionally, you can add your page to a Visualforce tab. For more information, refer to Create Visualforce Tabs.

4. When you're ready to embed a dashboard, see Add a Tableau CRM dashboard to a Visualforce Page.

Collaborate on Tableau CRM Dashboards in Experience Cloud Sites with Annotations

Foster an environment of open communication sparked by Tableau CRM dashboards embedded in Experience Cloud sites. With annotations, partners and customers can converse in Chatter feeds on individual dashboard widgets. Annotations are available whether dashboards are embedded via Visualforce or the Experience Builder.

📝 **Note:** Only users with a Customer Community Plus, Partner Community, or Lightning External Apps Plus license can use this feature.

If you haven't already set up Tableau CRM for Experience Cloud, see Share Tableau CRM in Experience Cloud Sites. To set up annotations, verify that Chatter is enabled for your community users. Also, enable feed tracking on Tableau CRM Assets. (For more information, see Enable Annotations on Dashboard Widgets.)

To create annotations, click [icon] to open your embedded dashboard in Tableau CRM. Then select **Annotate** from the dropdown menu on a widget.

In the annotations panel, you can type messages and @mention people in your org who have access to the dashboard. You can also share screenshots of the current state of the dashboard. Tableau CRM dashboard annotations are natively integrated with Chatter, and comments on dashboards also appear as Chatter posts. If users click the **Follow** button, they see all comments on the dashboard. From Chatter, they can click the dashboard image to navigate directly to the annotation on the dashboard in Tableau CRM, where they can add to the conversation. For more information, see Collaborate with Dashboard Annotations.

The visibility of posts depends on the type of user that makes the original post. For security, external site users can't see annotations originated by your org's internal users (unless the external user is @mentioned). But if an internal user replies on an annotation started by an external user, then all users can see all posts. For more information, see the `visibility` property in the *Connect REST API Developer Guide*.

Take Action in Salesforce from Experience Cloud Sites

From Tableau CRM in Experience Cloud sites, you can open Salesforce records and perform record-level actions and bulk actions. You set up an action menu on a dataset or a bulk action on a table widget. Then your site users can open records and take action in Salesforce directly from the Tableau CRM dashboard, without leaving your site.

📝 **Note:** Only users with a Customer Community Plus, Partner Community, or Lightning External Apps Plus license can use this feature.

Custom Action Menus

Let users create, update, and interact with Salesforce records that underlie the Tableau CRM dashboards they see in their site. Action menus can include object-specific and global actions defined in Salesforce, such as creating cases, updating accounts, and posting to Chatter.

Add an action menu to any dataset field, except a measure. You can set up the menu to include any actions available for the field. When an action menu is set up, a dropdown arrow appears in charts, chart legends, and table cells. Records that users access then open in new tabs within the site. To set up action menus, see Perform Actions on a Salesforce Record from Tableau CRM.

Bulk Actions

You can also configure bulk action menu options to let users perform actions on multiple Salesforce records with only one click. For example, using the dashboard to filter a set of leads in a table, a user could create a campaign for those leads by selecting the bulk action in the table's menu.

With Tableau CRM bulk actions, you can perform an action on all records shown in a table. You define the bulk action in a Visualforce page that is enabled for Experience Builder sites. You can also configure the Visualforce page to display the results of the action or show an interactive form. When the bulk action is set up, users can access it in the table's menu. For setup details, see Perform Bulk Actions on Multiple Salesforce Records from Tableau CRM.

Considerations

Be aware of the following considerations for Salesforce actions in Tableau CRM in Experience Cloud sites.

- Action menus are accessible in dashboards embedded via either Experience Builder–based templates or Salesforce Tabs + Visualforce templates.
- Tableau CRM in Experience Cloud isn't supported on mobile.
- To access a record from a dashboard's action menu, the user must have access to its objects and fields.
- Users see only actions available to their profile.
- Bulk actions are available only in values tables.
- Visualforce pages and components used in bulk actions must be made available for use in Experience Builder sites.
- Bulk action queries on Visualforce pages require the site prefix in the query URL, and the site prefix must start with a slash (/).

Add a Tableau CRM Dashboard to a Visualforce Page

By embedding a Tableau CRM dashboard into a Visualforce page, you can provide an interactive presentation of your data. Users can drill in and explore the dashboard within the frame on the Visualforce page or in a new window in Analytics Cloud.

To add a dashboard to your Visualforce page, configure the `<wave:dashboard>` component. You can add one dashboard per Visualforce page. However, the dashboard can include links to other dashboards or Tableau CRM assets.

Here are two examples. In these examples, the dashboard is identified by the `dashboardId`, which is an 18-character code beginning with 0FK. You can find the `dashboardId` at the end of the URL when viewing a dashboard. An alternative way to identify the dashboard is by its `developerName`, which can be requested through the API.

```
<apex:page sidebar="true" standardController="Opportunity">

  <wave:dashboard dashboardId="0FKB000000006Y7OAI"
       showTitle="true"
       height="800px"
       openLinksInNewWindow="true"
       filter="{'datasets' : {'opportunity': [ {'fields':['Id'], 'selection':
['{!Opportunity.Id}']}]}}"/>

</apex:page>
```

```
<apex:page standardController="case">

    <wave:dashboard dashboardId="0FKxx000000000uGAA"
         height="1000px"
         showTitle="false"
         showSharing="true"
         openLocation="HIDE"
         openLinksInNewWindow="false"
         hideOnError="true"
         filter="{'datasets' : {'ServiceCase4': [ {'fields':['Id'], 'selection':
['{!case.Id}']},  {'fields':['OwnerId'], 'filter': { 'operator':'in', 'values':
```

```
['{!case.OwnerId}', '{!case.Product__c}']}}]}}"/>

</apex:page>
```

- The `showTitle` attribute gives you control over the visibility of the dashboard title. If set to true, the dashboard's title is included above the dashboard. If set to false, the dashboard appears without a title. The default is true.

- The `showSharing` attribute lets you include the Share icon on the dashboard. If the icon is present, users can click to open the Share dialog, where they can post to Chatter and download images and data. If set to true, and the dashboard is sharable, then the dashboard shows the Share icon. If set to false, the dashboard doesn't show the Share icon. The default is false. To show the Share icon, the minimum dashboard height is 612 pixels.

- The `showHeader` attribute lets you control the visibility of the dashboard header. If set to true, the dashboard is displayed with a header bar. The header bar includes the open dashboard icon, the date and time that the dashboard's data was updated, and the dashboard views menu. If set to false, the dashboard appears without a header bar. The default is true. The header bar automatically appears when either `showSharing` or `showTitle` is true.

- The `openLocation` attribute lets you control where the open icon on the dashboard header opens the dashboard. The default value is `OPEN_IN_STUDIO`, which opens the dashboard in the Tableau CRM Studio. The other options are `OPEN_IN_TAB`, which opens the dashboard in the Salesforce Lightning Analytics tab and `HIDE`, which disables the open icon.

- With the `openLinksInNewWindow` attribute, you can specify where links from the dashboard to other assets are opened. If set to true, links open in new windows. If set to false, links to other assets open in the same window. The default is true.

- The `hideOnError` attribute gives you control over whether a Tableau CRM asset appears if there is an error (such as the user doesn't have access to the dashboard or it has been deleted). When set to true, if the dashboard has an error, it won't appear on the page. When set to false, the dashboard appears but doesn't show any data. The default is false.

- The `enableNotifications` attribute lets users with permission to set conditions and get notified about updates to dashboard widgets. When `showHeader` is true, this option also surfaces the notifications icon in the dashboard header which opens the notifications management panel for editing and deleting notifications. If you enable Show Header, it's recommended that the dashboard is at least 400 pixels in width to accommodate the management panel. The default is false.

- The `enableSubscriptions` attribute lets users with permission to subscribe for periodic email updates on dashboard widgets. When Show Header is true, this option also surfaces the subscriptions icon in the dashboard header which opens the subscriptions management panel for editing and deleting subscriptions. If you enable Show Header, it's recommended that the dashboard is at least 400 pixels in width to accommodate the management panel. The default is false.

- The `width` attribute specifies the width of the dashboard, in pixels or percent. Pixel values are simply the number of pixels, for example, 500. Percentage values specify the width of the containing HTML element and must include the percent sign, for example, 20%. If the specified width is too large for the device, the maximum device width is used.

- With the `filter` attribute, you can either select or filter on dataset fields at run time. For example, this attribute can be used to set up the embedded dashboard to show only the data that's relevant for the record being viewed by linking data fields in the dashboard to an object's fields. Filters are configured with JSON strings.

  Use this syntax for filtering on dimensions.

```
filter="{'datasets' : {'dataset1': [ {'fields': ['field1'], 'selection': ['!value1',
'!value2']}, {'fields': ['field2'], 'filter': { 'operator': 'operator1', 'values':
['!value3', '!value4']}}]}}"
```

  Use this syntax for filtering on measures.

```
filter="{'datasets' : {'dataset1': [ {'fields': ['field1'], 'selection': ['!value1',
'!value2']}, {'fields': ['field2'], 'filter': { 'operator': 'operator1', 'values':
[[value3]]}}]}}"
```

For example:

```
filter="{'datasets' : {'Sales': [ {'fields':['Region', 'Stage'], 'selection': [ ['West',
 'Prospecting'], ['North', 'Qualification'] ]}]}}"
```

Refer to the table to understand the notation for the `filter` attribute.

💡 **Tip:** This is a new syntax as of Spring '17. The previous syntax, noted here, is still supported.

Notation	Description	Notes
datasets	Tableau CRM dataset	Where to find it: On the Tableau CRM home page, select Edit on the dataset. The System Name is in the left panel of the edit page for the dataset. (If your org has namespaces, include the namespace prefix and two underscores before the dataset system name.)
fields	Dimension or Measure in the Tableau CRM dataset	Where to find it: Click the Explore icon to open the widget. Then select **Show SAQL** from the Options menu. For dimension names, look for "group by" statements. For measure names, look for statements with functions such as "sum" or "stddev."
values	Field in the Salesforce object, or a specific value	Where to find it: In Setup, find the object you want, and select Fields. Use the **Field Name** (also known as the API Name). For custom fields, use the name with "__c" at the end.
selection	With the selection option, the dashboard is shown with all its data, and the specified dimension values are highlighted	The selection option can be used alone or with the filter option. Selection takes dimension values only. To use this option, the dashboard must include a list, date, or toggle widget that groups by the specified dimension. If a selection specifies a value that doesn't exist, or the dashboard doesn't include a list, date, or toggle widget that groups by the specified dimension, then the selection input is ignored and the dashboard appears with all its data and no selection.
filter	With the filter option, the dashboard is shown with only filtered data	The filter option can be used alone or with the selection option. Filter takes dimension or measure values.
operator	Use with the filter option	Supported operators for dimensions: in, not in, matches Supported operators for measures: ==; !=; >= ; > ; <= ; <

✏️ **Note:** In Visualforce, values must have the format `object.field`.

✏️ **Note:** To filter on a field that contains special characters, use Visualforce's JSENCODE function in the filter to replace special characters with encoded values.

For example, this syntax applies the name of the account in the Visualforce page as a selection on the dashboard:

```
filter="{'datasets': {'Account':[{'fields':['Name'], 'selection':['{!Account.Name}']}]}}"
```

This example syntax applies the name of the account in the Visualforce page as a filter so that the dashboard shows only the filtered data:

```
filter="{'datasets': {'Account': [{'fields':['Name'], 'filter':{'operator': 'matches',
 'values':['{!Account.Name}']}}]}}"
```

📝 **Note:** The preceding syntax is new in Spring '17. The previous syntax continues to be supported, and it works the same as the new selection option. For reference, here's the previous syntax:

```
filter="{ 'datasetSystemName1': {'field1': ['!value1']}, 'datasetSystemName2': {'field1':
 ['!value1', '!value2'],'field2': ['!value3']} }"
```

For example:

```
filter="{'opportunity': {'Id': ['{!Opportunity.Id}']}}"
```

and

```
filter="{'ServiceCase4': {'Id': ['{!case.Id}'], 'OwnerId' : ['{!case.OwnerId}'],
 'Product__c' : ['{!case.Product__c}'], 'Reason' : ['{!case.Reason}']}}"
```

For more information about embedding Tableau CRM, see the *Visualforce Developer Guide Standard Component Reference*.

Embed Tableau CRM Dashboards in Salesforce Classic Pages

Add a Tableau CRM dashboard to a detail page layout. On an account detail page, for example, you can include a dashboard of service issues associated with the account. Users can drill in, apply filters, and explore in the dashboard as if they were viewing it in a Tableau CRM window.

EDITIONS

Available in: Salesforce Classic (not available in all orgs)

Available in: **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

USER PERMISSIONS

To customize page layouts:
- Customize Application

To view embedded Tableau CRM dashboards:
- Tableau CRM Growth permission set license with Use Analytics permission
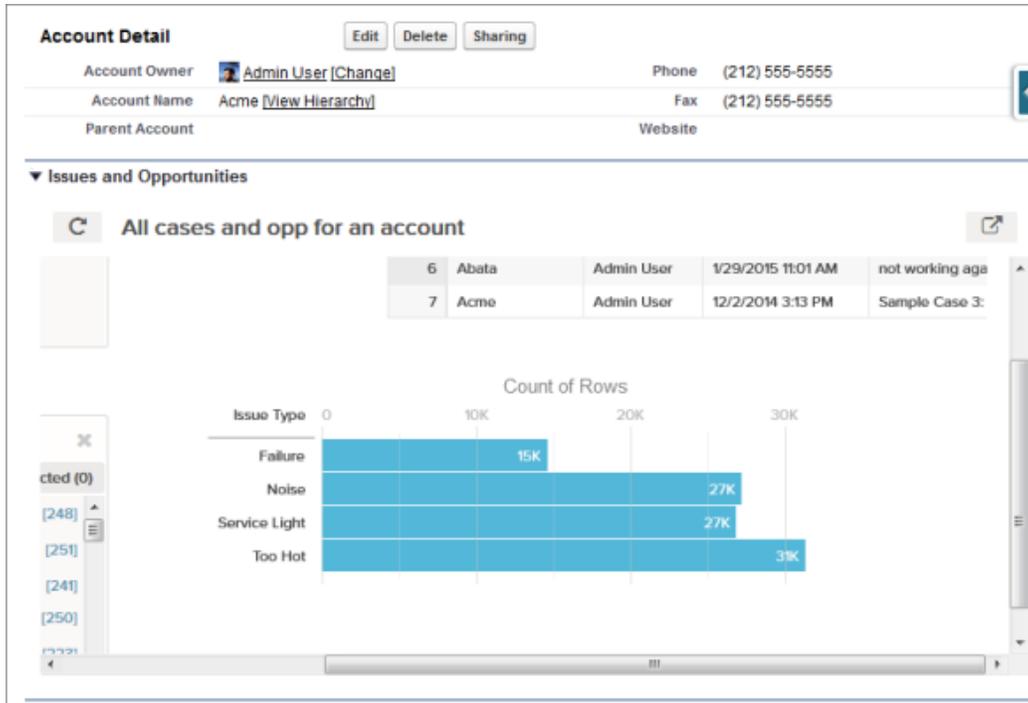
1. In the enhanced page layout editor, select Tableau CRM Assets in the left column of the palette.
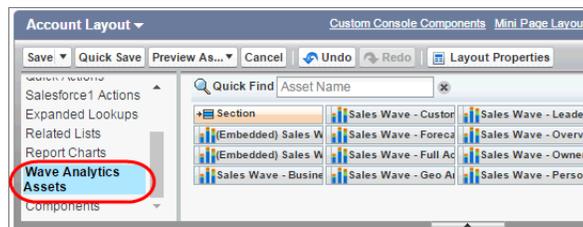
2. Drag an item from the list of available dashboards to a detail section on the page layout.

3. After the dashboard is positioned in the layout, you can change properties such as height by double-clicking the element or clicking the wrench icon next to it ( ).

   Additional properties of Analytics Cloud Assets:

   - The `Show Title` checkbox gives you control over the visibility of the dashboard title.
   - The `Show Sharing Icon` option lets you include the Share icon on the dashboard. If the icon is present, users can click to open the Share dialog, where they can post to Chatter and download images and data. To show the Share icon, the minimum dashboard height is 612 pixels.
   - The `Show Header` option lets you control the visibility of the dashboard header. If selected, the dashboard is displayed with a header bar that includes the Open in Tableau CRM icon, the date and time that the dashboard's data was updated, and the dashboard views menu. The header bar will also appear if either Show Sharing Icon or Show Title is selected.
   - The `Hide on error` checkbox gives you control over whether the Analytics Cloud asset appears if there is an error (such as the dashboard can't be found).
   - Dashboards include the date and time when they were last refreshed.

To set up the dashboard to show only the data that's relevant for the record being viewed, use field mapping. Field mapping links data fields in the dashboard to the object's fields. For example, the field mapping here applies the name of the account on the record page as a filter so that the dashboard shows only information related to that account name:

Refer to the table to understand the notation for adding the field mapping property.

💡 **Tip:** This is a new syntax as of Spring '17. The previous syntax, noted here, is still supported.

Notation	Description	Notes
datasets	Tableau CRM dataset	Where to find it: On the Tableau CRM home page, select Edit on the dataset. The System Name is in the left panel of the edit page for the dataset. (If your org has namespaces, include the namespace prefix and two underscores before the dataset system name.)
fields	Dimension or Measure in the Tableau CRM dataset	Where to find it: Click the Explore icon to open the widget. Then select **Show SAQL** from the Options menu. For dimension names, look for "group by" statements. For measure names, look for statements with functions such as "sum" or "stddev."
values	Field in the Salesforce object, or a specific value	Where to find it: In Setup, find the object you want, and select Fields. Use the **Field Name** (also known as the API Name). For custom fields, use the name with "__c" at the end.
selection	With the selection option, the dashboard is shown with all its data, and the specified dimension values are highlighted	The selection option can be used alone or with the filter option. Selection takes dimension values only. To use this option, the dashboard must include a list, date, or toggle widget that groups by the specified dimension. If a selection specifies a value that doesn't exist, or the dashboard doesn't include a list, date, or toggle widget that groups by the specified dimension, then the selection input is ignored and the dashboard appears with all its data and no selection.

Notation	Description	Notes
filter	With the filter option, the dashboard is shown with only filtered data	The filter option can be used alone or with the selection option. Filter takes dimension or measure values.
operator	Use with the filter option	Supported operators for dimensions: in, not in, matches
		Supported operators for measures: ==; !=; >= ; > ; <= ; <

The field mapping follows this format:

```
{
  "datasets": {
 "dataset1": [
  {"fields": ["field1"],
   "selection": ["$selection1", "$selection2"],
   "filter": {"operator": "operator1", "values": [["$value1, $value2"], ["$value3",
"$value4"]]}
  },
            {"fields": ["field2", "field3"],
   "selection": ["$selection3", "$selection4"]
  }
 ],
 "dataset2": [
  {"fields": ["field4"],
   "filter": {"operator": "operator2", "values": ["$value5"]}
  }
 ]
   }

}
```

📝 Note: Use this syntax for filtering on dimensions.

```
"filter": {"operator": "operator1", "values": ["$value1"]}
```

Use this syntax for filtering on measures.

```
"filter": {"operator": "operator1", "values": [[value1]]}
```

For example, if the dashboard has a dataset named Service with the dimensions Account and Industry, this field mapping would show the dashboard with all its data but the record's account ID and industry would be highlighted:

```
{
  "datasets": {
 "service": [
  {"fields": ["AccountId"],
   "selection": ["$Id"]
  },
            {"fields": ["Industry"],
   "selection": ["$Industry"]
  }
 ]
```

```
  }
}
```

With this field mapping, the dashboard would appear with just the record's account ID and industry data:

```
{
  "datasets": {
 "service": [
  {"fields": ["AccountId"],
   "filter": {"operator": "matches", "values": ["$Id"]}
  },
             {"fields": ["Industry"],
   "filter": {"operator": "in", "values": ["$Industry"]}
  }
 ]
  }
}
```

Here's another example with two datasets and a combination of selections and filters. This field mapping will show the dashboard with only data that is not in ExistingOpps but is in TX, AL, and DE, with revenue that is greater than or equal to 1000000. In addition, the dashboard will show Retail1, Wholesale1, and TX highlighted.

```
{
  "datasets": {
 "Opportunities": [
  {"fields": ["OpptyName"],
   "selection": ["$Retail1", "$Wholesale1"],
   "filter": {"operator": "not in", "values": ["$ExistingOpps"]}
  },
             {"fields": ["State", "Country"],
   "selection": ["$TX"],
   "filter": {"operator": "in", "values": [["$TX"], ["$AL", "$DE"]]}
  }
 ],
 "Accounts": [
  {"fields": ["Revenue"],
   "selection": [],
   "filter": {"operator": ">=", "values": [[1000000]]}
  }
 ]
  }
}
```

📝 Note: The above syntax is new in Spring '17. The previous syntax continues to be supported, and it works the same as the new selection option. For reference, here's the previous syntax:

```
{
    "dataset1": {
        "field1": ["$value1"]
    },
    "dataset2": {
        "field2": ["$value2"],
        "field3": ["$value3"]
    }
}
```

For example, if the dashboard shows data from a dataset named Service, with the dimensions Account and Industry, the field mapping would be defined like this:

```
{
    "service": {
        "AccountId": ["$Id"],
        "Industry": ["$Industry"]
    }
}
```

Be aware of these limits and limitations:

- You can add one dashboard per page layout.
- Analytics Cloud dashboards aren't supported in the original page layout editor. If you open and then save a layout with a Tableau CRM dashboard in the original page layout editor, the dashboard is deleted.

For information about working with the enhanced page layout editor, see Customizing Page Layouts with the Enhanced Page Layout Editor.

View Embedded Dashboards on Mobile Devices

Add Tableau CRM dashboard components to Lightning app pages. Embedded Tableau CRM dashboards can responsively select layouts optimized for any device.

Dashboards built in the dashboard designer are responsive to any screen size. If you've created a tablet, phone, or custom layout in the dashboard designer, the dashboard automatically optimizes to fit your screen. For more information, see Generate Unique Tableau CRM Dashboard Layouts for Different Devices .

When your dashboard has the right layout, embed it via the Lightning App Builder. For the Salesforce mobile app, only Lightning app pages are supported. See Lightning Pages for more information.

Understanding the following considerations can help you when planning for Tableau CRM dashboard use in the Salesforce mobile app.

- You can access embedded dashboards in the Salesforce mobile app from record layouts, within a Lightning page tab, and via custom links and buttons. Viewing embedded dashboards on mobile browsers isn't supported.
- Dashboards with multiple scrolling charts aren't optimized for mobile browsers.
- Users can view saved dashboard views on mobile, but the menu for managing saved views isn't accessible due to space constraints.
- When adding list selectors to dashboards, make them wide enough to render on mobile.
- Dashboard links that are set to open in a new window open the native Tableau CRM mobile app.
- Widget dropdown menus aren't supported, which means Explore, Share, Details, Annotate, Set Notification, and (if configured) bulk actions aren't available.
- Sharing actions aren't available.
- Record-level actions and bulk actions on tables aren't supported.
- Global filters can't be adjusted.
- Charts don't include hover-over information (tooltips).
- Some filters, such as range selectors, aren't optimized for mobile touch behavior.

Allow Trusted Sites for Embedded Dashboards

Embed Tableau CRM in any web page or web-based app. By specifying your trusted sites in the Tableau CRM allowlist, you can include embedded Tableau CRM dashboards in websites and apps outside of Salesforce servers.

First, add a Tableau CRM dashboard component to your site, using Visualforce embedding, Lightning App Builder, or Lightning Out. This code sample shows how to add a Tableau CRM dashboard component via Lightning Out.

```
$Lightning.use("wave:waveApp", function() {  $Lightning.createComponent("wave:waveDashboard",
   {    dashboardId: "0FKxx00000006bOGAQ"  }, "app");}, url, accessToken)
```

Second, set up an allowlist using the Content Security Policy (CSP). See Set Up an Allowlist of Trusted Sites for Embedded Tableau CRM Dashboards .

The allowlist for embedded analytics includes the following by default:

- *.salesforce.com:*
- *.force.com:*
- Your org's company-specific My Domain or custom domain.

Filter and Selection Syntax for Embedded Dashboards

To specify types of filters or dashboard selections that cannot be created using the Filter Builder, enter JSON strings in the Filter String box of the embedded Tableau CRM dashboard attributes.

📝 **Note:** Tokens can be used to pass values from the parent page. If the token resolves to multiple values, the resolved values are properly reflected in the resulting filter string.

Filter Syntax

Use this syntax for filtering on dimensions. Example values syntax for measures and dates for both filters and selections follow further down.

```
{"datasets" : {
  "dataset1": [
    {
      "fields": ["field1"],
      "filter": {
          "operator": "in",
          "values": ["$value1", "$value2"]
        }
      }
    ]
  }
}
```

In this example, the dashboard is filtered by the account name on the record page and shows only data related to that account name.

```
{"datasets": {
  "account":[
    {
      "fields":["Name"],
      "filter": {
        "operator": "matches",
        "values":["$Name"]
        }
      }
```

```
    ]
  }
}
```

Selection Syntax

With the selection option, the dashboard shows with all its data, and the specified dimension values are highlighted. The selection option can be used alone or with the filter option. Selection takes dimension values only. To use this option, the dashboard must include a query (step) of the specified id that groups by the specified group name. If a selection specifies a value that doesn't exist or if the query itself is unused (for instance, not attached to any widget), then the selection input is ignored and the dashboard appears with all its data and no selection.

This example applies the account name as a selection. With the selection option, the dashboard displays all of its data, and the specified values appear highlighted or selected within the dashboard.

```
{"steps" : {
  "Account_Step":
    {
      "metadata": {"groups": ["Name"]},
      "values": ["$Name"]
    }
  }
}
```

Additional Filter and Selection Syntax Examples

Use this syntax for filtering or selecting with measures values.

```
{
  ...
  "values": [[value1]]
  ...
}
```

Use this syntax for filtering or selecting with relative date range values.

```
{
  ...
  "values": [["year", 0], ["year", 2]]
  ...
}
```

Use this syntax for filtering or selecting with open-ended relative date range values.

```
{
  ...
  "values": [null, ["year", 2]]
  ...
}
```

Use this syntax for filtering or selecting with absolute date range. Use epoch times for this range.

```
{
  ...
  "values": [[1581033600000, 1585353600000]]
  ...
}
```

Filter and Selection Syntax Reference Tables

Refer to these tables to understand the notation for the filters and selection syntax.

Notation	Description	Notes
`datasets`	The Tableau CRM datasets. Datasets have `fields` and a `filter`, described in the Datasets Notation table.	Where to find it: On the Tableau CRM home page, select **Edit** on the dataset. The System Name is in the left panel of the edit page for the dataset. If your org has namespaces, include the namespace prefix and two underscores before the dataset system name.
`steps`	The step names to reference for the selection. Steps have `metadata` and `values`, described in the Steps Notation table.	Where to find it: On the Tableau CRM home page, select **Edit** on the dashboard. Select the widget to apply the selection on. The step name is **ID** value on the **Query** tab.

Datasets Notation	Description	Notes
`fields`	A list of Dimensions or Measures in the Tableau CRM dataset.	Where to find it: Click the Explore icon to open the widget. Then select **Show SAQL** from the Options menu. For dimension names, look for "group by" statements. For measure names, look for statements with functions such as "sum" or "stddev."
filter	The filter to apply to the dashboard data. Filters have `operators` and `values`, described in the Filter Notation table.	The filter takes dimension or measure values, described in the next table.

Filter Notation	Description	Notes
`operator`	The filter operator.	Supported operators for dimensions: in, not in, matches
		Supported operators for measures: ==; !=; >= ; > ; <= ; <
values	The values to use for the filter. These can be statically specified or inserted dynamically from the page context using the $ prefix like $>FieldName>.	Where to find it: In Setup, find the object you want, and select **Fields**. Use the Field Name, also known as the API Name. For custom fields, use the name with "__c" at the end.

Steps Notation	Description	Notes
`metadata`	The step metadata.	Any `groups` listed for the grouping must be referenced in the `values`.
`values`	The list of step values to use for the selection. These can be statically specified or inserted dynamically	Where to find it: In Setup, find the object you want, and select **Fields**. Use the Field Name, also known

Steps Notation	Description	Notes
	from the page context using the $ prefix like $>FieldName>.	as the API Name. For custom fields, use the name with "__c" at the end.

Prevent users from exposing more data by locking or hiding filters. With the **Locked** option, the Filters menu appears in the embedded dashboard header, but users can't modify or delete the filters. With the **Hidden** option, the Filters menu doesn't appear in the embedded dashboard header.

Locked and hidden options can only be used for filters and not for selections. Add the options in the filter's JSON with a Boolean value. For example, `"locked":false,"hidden":true`

When setting up locked or hidden filters for embedded dashboards, keep in mind the following considerations.

- Locked and hidden filters are applied on the dashboard. If a filter makes a field visible, users see the field even when you've restricted access to filters with one of these options.

- Locked and Hidden options aren't automatically applied in linked dashboards or lens explorations. The options are applied only on the dashboard where specified.

- Locked and Hidden options are applied to a dashboard when embedded or when opened in Tableau CRM Studio or Analytics tab.

> ✏️ **Note:** The above syntax became available in Spring '17. The previous syntax remains supported, and it works the same as the new selection option. The new syntax is recommended for use, as it has better performance. For reference, here's the previous syntax:

```
{ "datasetSystemName1": {"field1": ["$value1"]}, "datasetSystemName2": {"field1":
["$value1", "$value2"],"field2": ["$value3"]} }
```

For example:

```
{"opportunity": {"AccountId': ["$Id"]}}
```

Enable Actions for Tableau CRM Lenses and Dashboards

Let users take action on insights directly from Tableau CRM. Add actions to charts and tables in lenses and dashboards.

You can enable object-specific actions and global actions that you previously defined on Salesforce objects directly from Tableau CRM. Lens and dashboard viewers can invoke these actions to create, update, and interact with Salesforce records. For example, you can enable actions that let users create cases, update accounts, and post to Chatter.

After enabling actions for a specific dimension, viewers can choose to view all available actions for the dimension when they click records in that dimension column.

> ✏️ **Note:** Selecting a Salesforce record action in a Tableau CRM dashboard opens the record in a new Lightning Experience tab. To open in a new browser tab, select the `Open Salesforce records in new browser tabs` setting in Setup.

Perform Actions on a Salesforce Record from Tableau CRM

As you gain insights into your business from Tableau CRM, you sometimes perform actions on an underlying Salesforce record. No need to exit Tableau CRM. You can invoke a Salesforce action on the record directly from Tableau CRM. You can also open web pages, such as search the web for the company name listed in a Tableau CRM lens or dashboard.

Perform Bulk Actions on Multiple Salesforce Records from Tableau CRM

Sometimes, you need to perform the same action on a group of records. With Tableau CRM bulk actions, you can perform an action on all records shown in a table in Tableau CRM.

Configure Mass Quick Actions on Multiple Salesforce Records from Tableau CRM Dashboards

Sometimes users want to perform the same quick action on a list of records. With Tableau CRM mass actions, users can save time by performing a quick action on up to 100 records at once from a step in Tableau CRM.

Perform Actions on a Salesforce Record from Tableau CRM

As you gain insights into your business from Tableau CRM, you sometimes perform actions on an underlying Salesforce record. No need to exit Tableau CRM. You can invoke a Salesforce action on the record directly from Tableau CRM. You can also open web pages, such as search the web for the company name listed in a Tableau CRM lens or dashboard.

You can add an action menu to any dataset field, except a measure. These actions appear in Tableau CRM chart legends and table cells.

Tableau CRM supports the following object and global actions defined in Salesforce:

- Create and update actions for object records.
- Log a Call actions.
- Custom actions to invoke flows, Lightning components, and Visualforce Pages. Canvas apps aren't supported.
- Standard Chatter actions except for File.

Actions assigned to a page layout for the User object aren't supported. Actions are only available for the local org, and are not supported for multi-org records. Actions are configured from a dataset or from the table chart view in Explorer.

1. Configure actions from a dataset

    a. Edit the dataset.
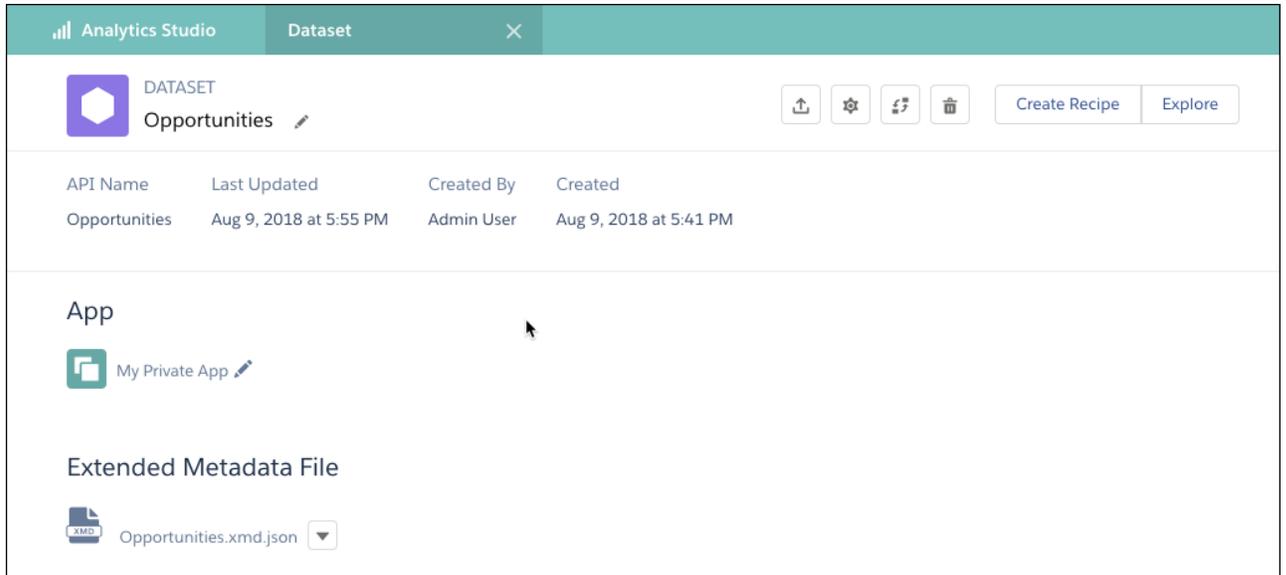
b. Click the Configure Actions button (⚙).

c. In the left pane, select the dataset field where you want the action menu to appear.
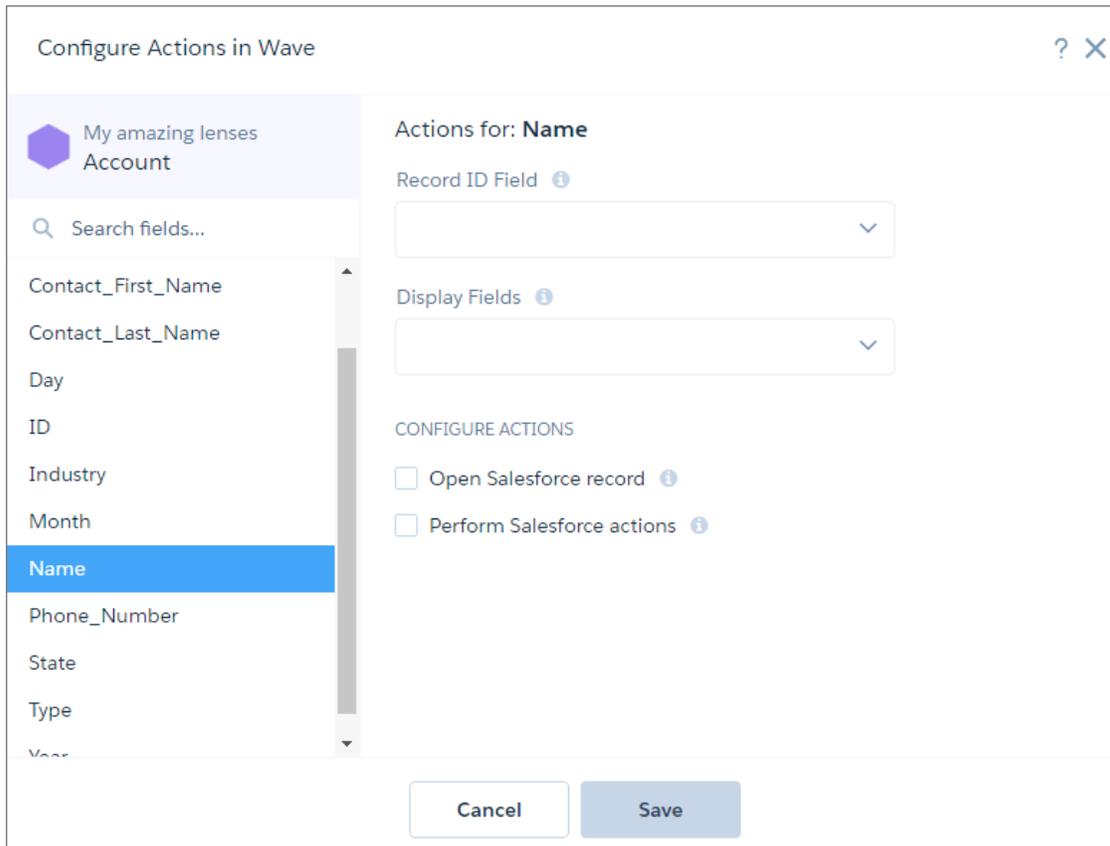
2. Configure actions from a table.

From the table view in Explorer, from the dimension column's action menu, select **Configure Actions**.
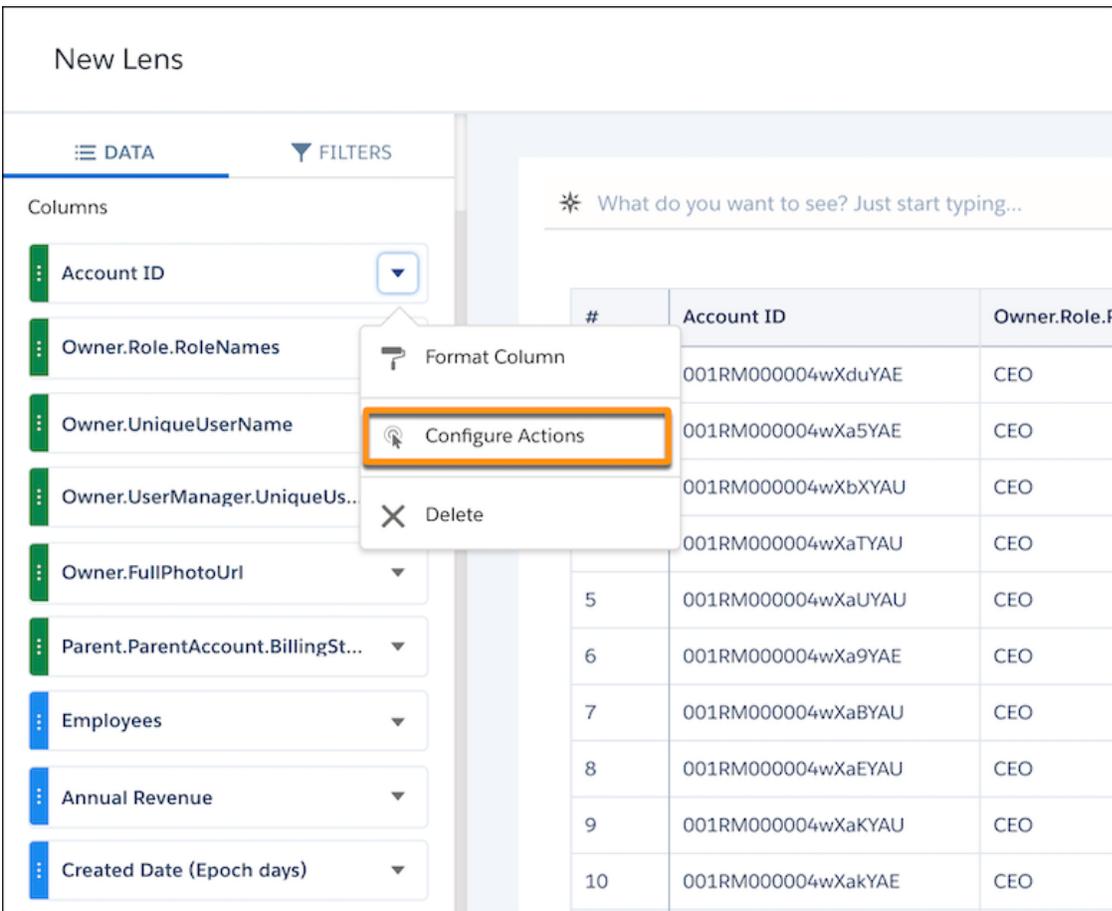
3. In the Record ID Field, select the dataset field that contains the record ID of the object that you want to perform the action on.

For instance, if the dataset has Opportunity ID and Account ID fields, select Account ID to perform actions on the account records. Tableau CRM uses the ID to identify the Salesforce object and record to perform the action on.

> 📝 Note: In values tables, Tableau CRM uses the ID from the table row to perform the action on if the record ID field used in the Configure Actions setup is included in the query.

4. In Display Fields, choose which dataset fields to show users. An action can apply to multiple Salesforce records. Specifying the dataset fields narrows the choices and helps users pick the correct record to perform the action on.

For example, if you enable actions on opportunity records, you can show the opportunity name, account name, and opportunity owner name to help users pick the right opportunity.

5. To let users open a Salesforce record or URL, select **Open Salesforce record**, choose the action in the Open field, and then enter a tooltip. Users see the tooltip when they hover over the action in a Tableau CRM lens or dashboard.

Choose one of the following actions.

Action	Description
Salesforce Record	Opens the Salesforce record. Tableau CRM searches for the record in the same org where it's running.
Salesforce Record (Multiple Orgs)	Opens the Salesforce record from another org. If the dataset contains records from multiple orgs, select this option. In the Org ID Field, select the dataset field that contains the org ID. To enable Tableau CRM to connect to each org, select **Manage Orgs** and specify the URL for each org.
URL	Opens a URL that uses HTTP or HTTPS protocols.
	Use the following syntax to pass a dataset field as a parameter in the URL:
	```{{row.<dataset_field_dev_name>}}```
	For example, ```http://www.google.com?q={{row.AccountName}}```
	If you don't specify the URL, Tableau CRM uses the org's URL. For example, you can specify the following parameter to open the Account record in Salesforce:
	```/{{row.AccountId}}```
	Tableau CRM retrieves the Account ID from the AccountId field in the dataset record.

6. To enable users to perform actions on a Salesforce record, select **Perform Salesforce actions** and then choose the actions. The list of actions varies based on the object identified by the Record ID Field.

Option	Description
All actions	Adds all actions defined in the page layout for this object, including new ones created later.
Choose actions	Select from the list of currently available actions. 📝 Note: The list contains a superset of all actions defined in all page layouts for the Salesforce object. Actions are only available for the local org, and are not supported for multi-org records.

☑ Perform Salesforce actions ⓘ

   ◯ All actions

   🔘 Choose actions

Each user can only see actions assigned to their page layout.

Salesforce Actions

🔍

☐	LABEL	API NAME
☐	📅 New Event	NewEvent
☐	📋 New Task	NewTask

📝 Note: A Tableau CRM user sees only the actions that you select and that are assigned to their page layout. If the predefined actions are overridden for the Salesforce mobile app and Lightning Experience in a page layout, Tableau CRM displays only the actions that are added to both the Quick Actions in the Salesforce Classic Publisher and the Salesforce Mobile and Lightning Experience Actions sections of the layout.

7. Click **Save**.

Each week, opportunity owners analyze their opportunities using a Tableau CRM dashboard. Sometimes, they must update an opportunity. To help them perform the update without going back and forth between Tableau CRM and Sales Cloud, you decide to set up Tableau CRM actions. To let Tableau CRM dashboard viewers update an opportunity record from Tableau CRM, you add the Update Opportunity action to the Opportunity Name field in a Tableau CRM dataset. The first step is to set up the action in Salesforce and then add the action to the Tableau CRM dataset field.

Record Actions Example

Each week, opportunity owners analyze their opportunities using a Tableau CRM dashboard. Sometimes, they must update an opportunity. To help them perform the update without going back and forth between Tableau CRM and Sales Cloud, you decide to set up Tableau CRM actions. To let Tableau CRM dashboard viewers update an opportunity record from Tableau CRM, you add the Update Opportunity action to the Opportunity Name field in a Tableau CRM dataset. The first step is to set up the action in Salesforce and then add the action to the Tableau CRM dataset field.

To implement this record-level action, perform the following tasks.

1. Create the Salesforce Action and Add It to an Object

   This example adds the predefined Update a Record action to the Opportunities object.

2. Apply the Action to the Object's Page Layout

   Add the action to the page layout for the object. Users whose profiles are assigned to that layout can see the action in Salesforce. Page layouts control the layout and organization of content, including custom actions, on record detail pages. Each record detail page can have multiple page layouts to give different users a unique experience.

3. Add the Action to the Tableau CRM Dataset Field

   When you add the Update Opportunity action to the Opportunity Name field in the Opportunities dataset, you specify which dimension shows the action menu. You also specify the dataset field that contains the Salesforce record ID. The ID identifies both the Salesforce object and the record.

Create the Salesforce Action and Add It to an Object

This example adds the predefined Update a Record action to the Opportunities object.

1. In Setup, in the Quick Find box, enter the name of the object that you want to add actions to. For this example, enter `Opportunities`, then select **Buttons, Links, and Actions**.

2. Click **New Action** to see which types of actions are available.

3. From the Action Type dropdown list, select the action to add. In this case, select **Update a Record**.

4. Enter a label and description for the action.

5. Click **Save**.

   The next page shows the default fields that you can update. For the example, add the **Probability (%)** and **Type** fields.

6. Drag **Probability (%)** and **Type** below the Stage field in the workspace. To select multiple fields, hold down **Ctrl** (for Windows) or **Cmd** (for macOS).

7. Click **Save**.

You added the Update Opportunity action to the Opportunities object. Next, you add it to a page layout to determine which users can see the action.

Apply the Action to the Object's Page Layout

Add the action to the page layout for the object. Users whose profiles are assigned to that layout can see the action in Salesforce. Page layouts control the layout and organization of content, including custom actions, on record detail pages. Each record detail page can have multiple page layouts to give different users a unique experience.

In this example, you add the Update Opportunity action to the opportunity page layout that's assigned to the Standard User profile. You can use these same techniques to add actions to any Salesforce page layout and user profile.

1. Go to Setup in Salesforce, and enter the name of the object in the Quick Find box. For the example, enter `Opportunities`. Then select **Page Layouts** to see what's available.

2. If multiple page layouts are defined for the object, click **Page Layout Assignment** to see which page layout is assigned to each user profile.

3. Edit the page layout assigned to the profile for the users for whom you want to access the action.

For this example, edit the Opportunity Layout, which is assigned to the Standard User profile. Users who are assigned the other opportunity page layouts can't see this custom action, unless you apply the action to those page layouts.

4. To open the layout, click the **Opportunity Layout** link next to a profile. It doesn't matter which profile.

5. In the palette, click **Quick Actions** to select the actions to add to the layout.
The available actions appear to the right.

6. Drag the **Update Opportunity** custom action to the existing actions in the Quick Actions in the Salesforce Classic Publisher section.
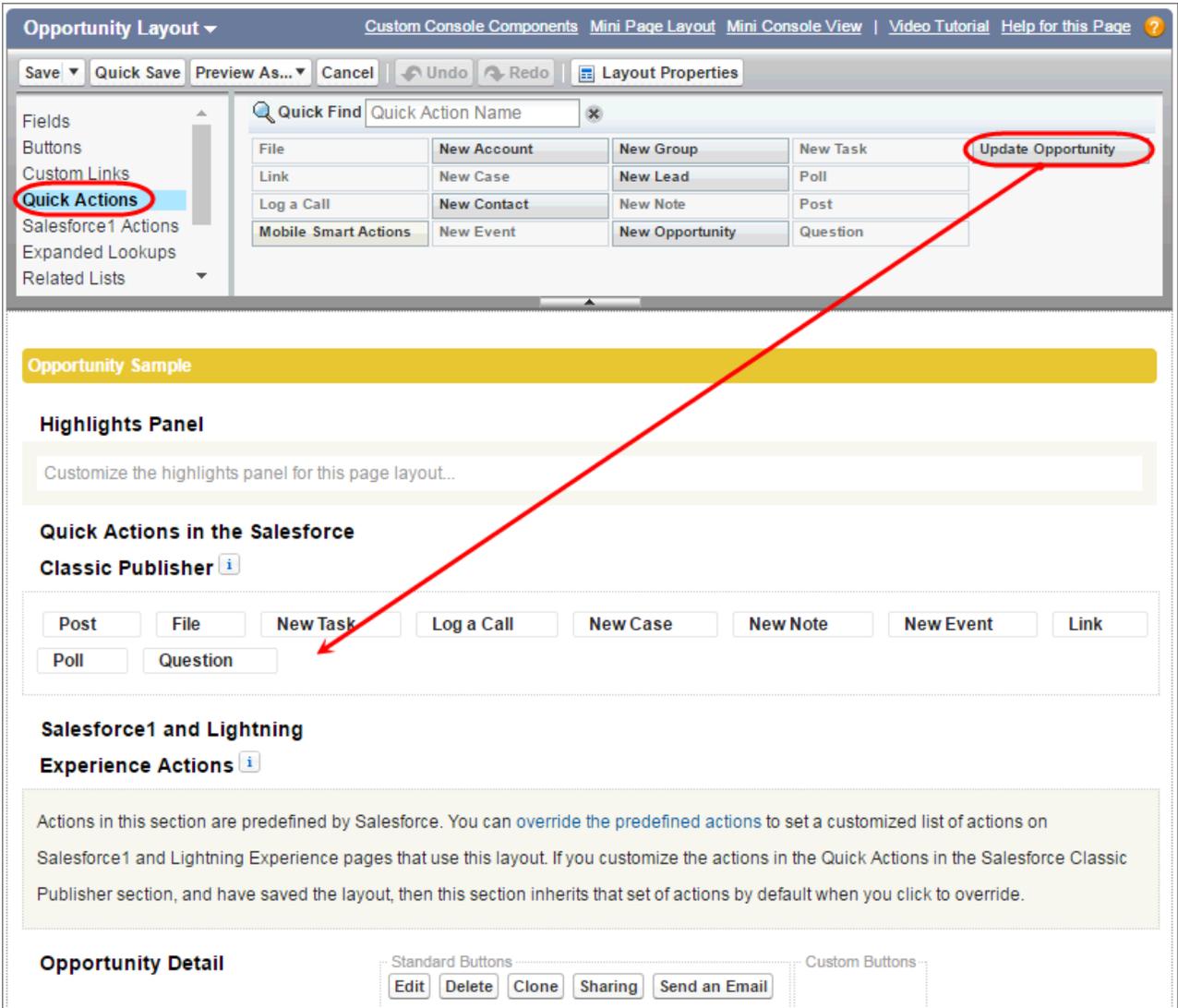
7. If the predefined actions are overridden in the Salesforce app and Lightning Experience Actions section, you must also drag the action to that section.

   If the predefined actions are overridden, Analytics shows only the actions that appear in both the Quick Actions in the Salesforce Classic Publisher and Mobile & Lightning Actions sections.

8. Click **Save**.

This action is now available on the Opportunity details page for all users assigned the Opportunity Layout.
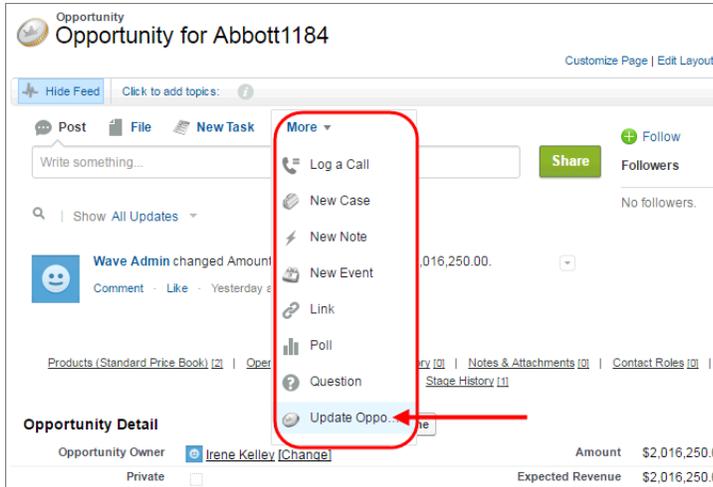
> **Note:** If a page layout is based on a record type that certain users can't access, actions for that type aren't available to them.

Add the Action to the Tableau CRM Dataset Field

When you add the Update Opportunity action to the Opportunity Name field in the Opportunities dataset, you specify which dimension shows the action menu. You also specify the dataset field that contains the Salesforce record ID. The ID identifies both the Salesforce object and the record.

1. Edit the dataset.

2.
Click the Configure Actions button (⚙ .

3. In the left pane, select the dataset field where you want the action menu to appear.
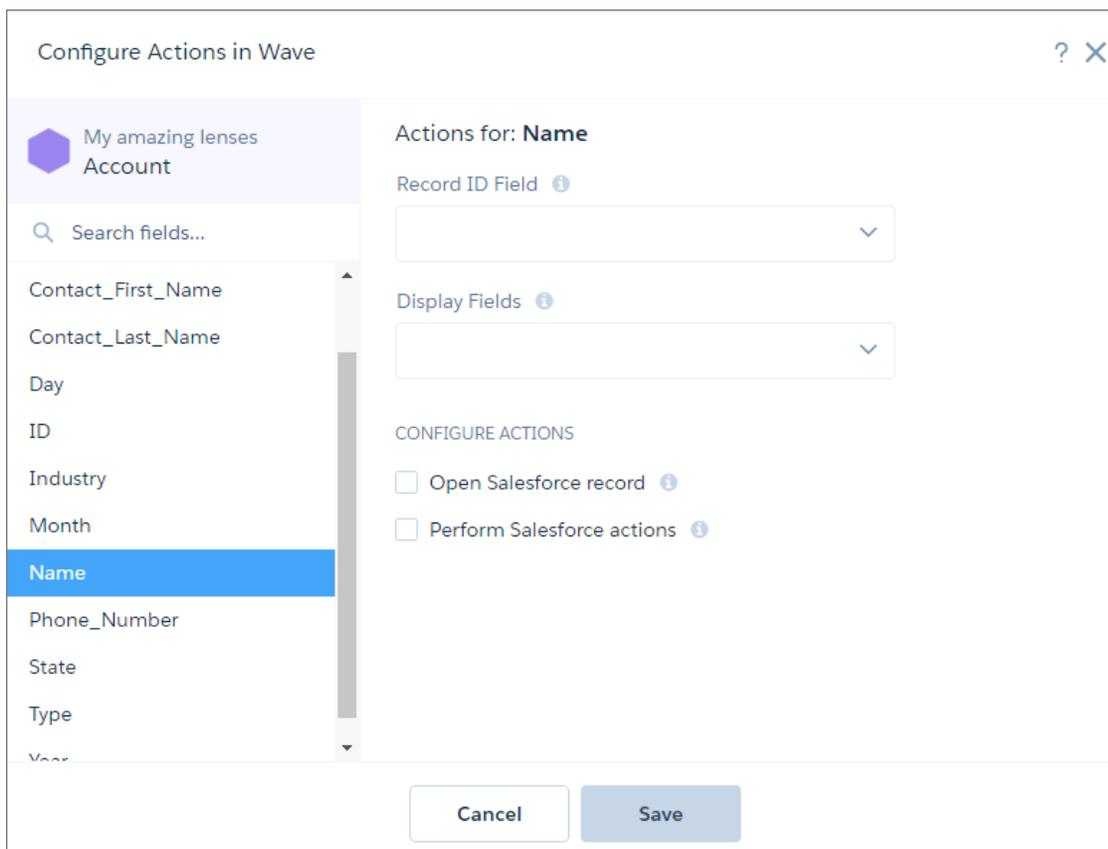
4. In the Record ID Field, select **Opportunity Id**, which is the dataset field that contains the record ID of the Opportunity object.

5. In Display Fields, select the dataset fields to show if multiple records match the action. Users can review this information to determine which Salesforce record to perform the action on. To help users identify the correct opportunity record, show Opportunity ID, Opportunity Name, Opportunity Owner, and Account Name.

6. To enable users to perform actions on a Salesforce record, select **Perform Salesforce actions** and then select **Choose actions**.

7. Select **Update Opportunity**.
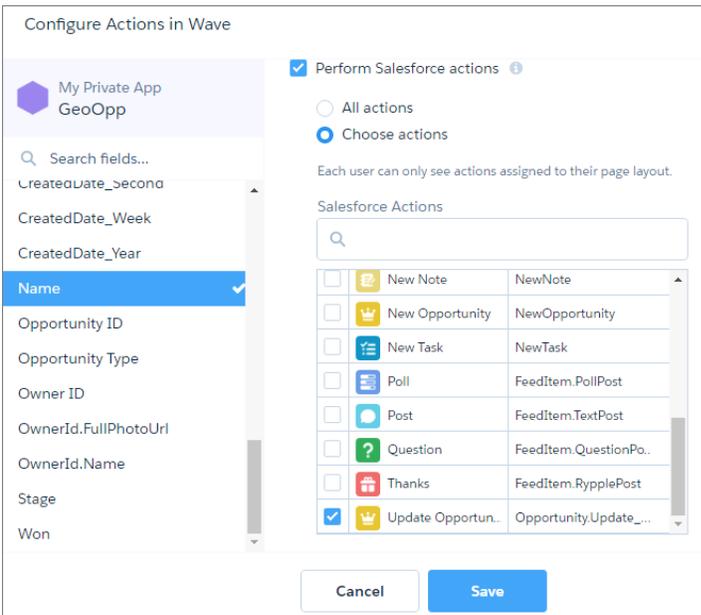
8. Click **Save**.

Perform Bulk Actions on Multiple Salesforce Records from Tableau CRM

Sometimes, you need to perform the same action on a group of records. With Tableau CRM bulk actions, you can perform an action on all records shown in a table in Tableau CRM.

You define the bulk action in a Visualforce page. The Visualforce page can make calls to Apex methods. You can also configure the Visualforce page to display the results of the action or show an interactive form. For example, you can create a form that allows the dashboard viewer to change the account owner of each opportunity shown in the table widget.

Before we get into the details about how to create a bulk action, review the following high-level process.

1. Tableau CRM passes the SAQL query used to populate the table widget to the Visualforce page.

> 📝 **Note:** The SAQL query that Tableau CRM passes usually contains a limit. To return more results, modify the `limit` statement in the SAQL query.

2. You write custom Visualforce and, if needed, Apex code to extract the SAQL query.

> 💡 **Tip:** When issuing a Tableau CRM API request for the SAQL query, if most of your logic is in Javascript on the client side, then you should use AJAX from Javascript. If most of your logic is in Apex, then use HTTP from your Apex class.

3. Your code executes the SAQL query against the Tableau CRM Query API.

4. The Tableau CRM API returns the results of the query—dataset rows—to the Visualforce page.

5. You write more custom code that performs actions on the returned dataset rows. For example, you can perform a bulk edit, bulk create, or integrate with third party.

📝 **Note:** Bulk actions only work on steps that query datasets. If the table widget is built on a step with type `static` or `soql`, the action doesn't work.

Bulk Actions Example

You want to add the "Create Opportunities" option in a table widget's action menu. This action must create an opportunity for each account shown in a table widget.

Bulk Actions Example

You want to add the "Create Opportunities" option in a table widget's action menu. This action must create an opportunity for each account shown in a table widget.

📝 **Note:** This widget-level action menu is different from the record-level action menu that you can access from a record.

To keep the user informed of what is happening, the action must also display a Visualforce page that shows the progress of the following tasks.

1. The Visualforce page retrieves a list of accounts from the SAQL query used to populate the table widget. If the table widget's results were previously filtered because of selections in other dashboard widgets, only the filtered records are returned.

2. The Visualforce page creates an opportunity for each retrieved account.

3. The Visualforce page shows the opportunity created for each account.

To implement this bulk action, perform the following tasks.

1. Define Action Behavior with Apex Controller Class and Methods

   Define an Apex controller class and methods that generate a list of accounts shown in the table widget, create an opportunity for each account, and return the progress of each task.

2. Create a Visualforce Page That Executes the Action

   The Visualforce page calls the methods defined in the Apex controller class to invoke the action and provide a status on the tasks.

3. Assign Bulk Action to Table Widget

   Configure the table widget properties to expose the bulk action defined in the Visualforce page.

Define Action Behavior with Apex Controller Class and Methods

Define an Apex controller class and methods that generate a list of accounts shown in the table widget, create an opportunity for each account, and return the progress of each task.

1. From setup, enter *Apex Classes* in the Quick Find box.

2. Select **Apex Classes**.

3. Click **New**.

4. Add the following code.

```
public class CreateOpportunitiesController {
    public string query{get; set;}

/* To determine the records to perform the bulk action on, extract the SAQL query */
    public PageReference init() {
        query = ApexPages.currentPage().getParameters().get('query');
        return null;
    }

/* Takes the account records from the SAQL query, creates an opportunity for each account,
 and then returns a map between account ID and new opportunity name. Note: Account.Name
 and AccountId referenced below refer to the dataset field names. Update them to match
 your dataset fields. */
    @RemoteAction
    public static Map<String, String> create(List <Map<String, String>> accountRecords)
{
        Map<String, String> result = new Map<String, String>();


        List<Opportunity> opps = new List<Opportunity>();
        for (Map<String, String> accountRecord : accountRecords) {

            String name = accountRecord.get('Account.Name') + ' - Sprint Review - 12/2';

             String accountId = accountRecord.get('AccountId');
             result.put(accountId, name);
             Opportunity opp = new Opportunity(
/* You can set different fields from the Opportunity object than those listed below,
```

```
like Amount. */
                Name = name,
                Type ='New Customer',
                AccountId = Id.valueOf(accountId),
                CloseDate=Date.valueOf('2016-12-31'),
                StageName='Prospecting');
            opps.add(opp);
        }
        insert opps;
        return result;
    }
}
```

5. Click **Save**.

Create a Visualforce Page That Executes the Action

The Visualforce page calls the methods defined in the Apex controller class to invoke the action and provide a status on the tasks.

1. From setup, enter *Visualforce* in the Quick Find box.

2. Select **Visualforce Pages**.

3. Click **New**.

4. Enter the Visualforce page label and description.

5. Replace the existing markup with this code.

```
<apex:page controller="CreateOpportunitiesController" action="{!init}" showheader="false"
 sidebar="false" standardStylesheets="false" title="Create Opportunities" >
    <apex:stylesheet
value="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css"/>

/* Use jQuery to process API calls */
    <apex:includeScript value="https://code.jquery.com/jquery-3.1.0.min.js"/>
    <style>
        th {
            width: 50%;
```

```
        }
        h4 {
            font-size: 24px;
        }
        table {
            font-size: 20px;
            width: 100%;
        }
    </style>
    <div class="container-fluid">

/* Add HTML table to the Visualforce page that shows the accounts, the opportunity
creation status, and then finally the new opportunity name */
        <h4 id="message">Querying Accounts...</h4>
        <table name="results" id="results" data-role="table" class="table-bordered
table-striped table-responsive">
            <thead><tr><th>Account</th><th>Opportunity</th></tr></thead>
            <tbody></tbody>
        </table>
    </div>

    <script>
        $(function() {
            $.ajaxSetup({
                headers: {"Authorization": 'Bearer {!$Api.Session_ID}'}
            });

            setTimeout(executeQuery, 1000);
        });

/* Executes the SAQL query and displays the resulting accounts. Note: Account.Name and
 AccountId referenced below refer to the dataset field names. Update them to match your
 dataset fields. */
        function executeQuery() {
            var query = {};
            query.statements = "{!JSENCODE(query)}";
            var queryObj = {query: query.statements};
            $.ajax({
                type: 'POST',
                url: '/services/data/v39.0/wave/query',
                data: JSON.stringify(queryObj),
                contentType: 'application/json',
                success: function(data) {
                    $('#message').html('Creating Opportunities...');
                    var record = null;
                    var row = null;
                    $('#results tbody').empty();

                    for (var i = 0; i < data.results.records.length; i++) {
                        record = data.results.records[i];
                        row = $('<tr>');
                        row.append($('<td>').html(record['Account.Name']));
                        row.append($('<td class="' + record.AccountId +
'">').html('Creating...'));
```

```
                    $('#results tbody').append(row);
                 }

                 setTimeout(function() {createOpportunities(data.results.records);},
 1000);
             },
          });
       }

/* Calls the Apex controller method that creates opportunities for each account and
returns the opportunity name for each account to the HTML table. */
       function createOpportunities(accountRecords) {
           CreateOpportunitiesController.create(accountRecords, function(result, event)
 {
               console.log(result);
               if (event.status) {
                   for (var i = 0; i < accountRecords.length; i++) {
                       $('td.' +
accountRecords[i].AccountId).html(result[accountRecords[i].AccountId]);
                   }
                 $('#message').html(accountRecords.length + ' Opportunities Created');

               }
               else {
                   $('#message').html('Error: ' + event.message);
               }
           });
       }
    </script>
</apex:page>
```

6. Click **Save**.

Assign Bulk Action to Table Widget

Configure the table widget properties to expose the bulk action defined in the Visualforce page.

1. Open the Tableau CRM dashboard that contains the table widget.

2. Edit the dashboard.

3. Select the table widget.

4. In the widget properties, select **Show custom action**.

5. Enter the following widget properties.

Property	Description
Custom Action Label	Label for the action in the action menu.
Visualforce Page Name	API name of the Visualforce page.
Visualforce Namespace Prefix	Optional. Namespace prefix specified for the Visualforce page.

6. Save the dashboard.

Configure Mass Quick Actions on Multiple Salesforce Records from Tableau CRM Dashboards

Sometimes users want to perform the same quick action on a list of records. With Tableau CRM mass actions, users can save time by performing a quick action on up to 100 records at once from a step in Tableau CRM.

To use mass quick actions in Tableau CRM, configure mass actions for your selected Salesforce object. See Set Up a Mass Quick Action for configuration details.

After you configure mass quick actions in your org, you can add a mass action link widget for your dashboard users.

Your dashboard must contain a step for the configured object, with a query that returns the record ID.

> **Note:** Any step works with the mass quick action link, but the table chart provides the best visibility to the list of objects to perform a quick action on. It's also best practice to include filtering that reduces the list of actionable records to 100 records or less. If more than 100 records are returned in the table query, the mass quick action is performed only on the first 100 actions.

In the Tableau CRM Studio, clicking a mass action link widget in a dashboard opens the action window for a user to perform quick actions on a list of records. The list of records is returned by step query from a table or chart and can be filtered using global filters or selection facets from other charts.

1. Drag the link widget to the dashboard canvas.

2. To change the widget properties, select the widget.
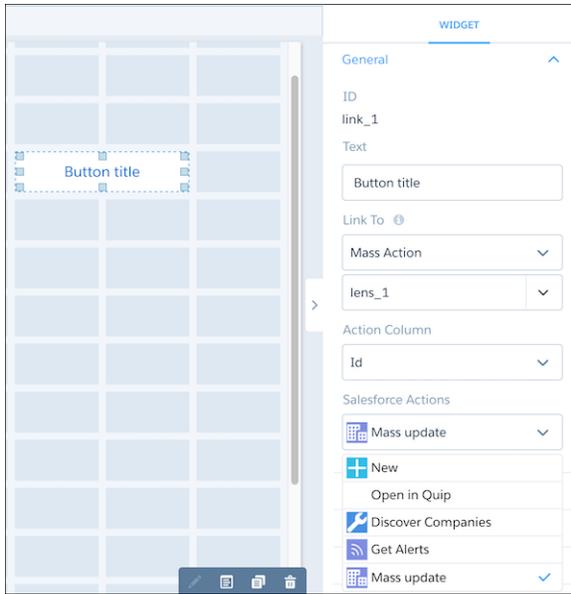
3. In the **Text** field, enter the label that appears on the link widget.

> 📝 **Note:** It's best practice to make the label match the selected action so the user understands what action the link performs.

4. In the **Link To** field, select **Mass Action** and then the step to perform the mass action on.

5. In the **Action Column** field, select the record ID. If the selected step query doesn't return the record ID, the action column appears empty. Select the record ID for the mass quick action to function correctly.

6. In the **Salesforce Actions** field, select the mass quick action to perform when your link is clicked. The action menu is only populated when mass quick actions are configured for the List View layout of the object.

7. To customize the appearance of the widget, set the widget properties.

8. To preview your changes to the dashboard, click 👁 .

9. Save the dashboard.

Integrate Tableau CRM into Salesforce with an Analytics Tab

Access your Tableau CRM home and run Tableau CRM apps from a tab within Salesforce Classic or Lightning Experience.

The Tableau CRM tab is available in new Tableau CRM-enabled orgs.
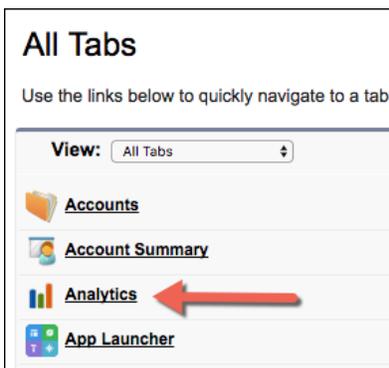
In existing orgs, set up the Tableau CRM tab as you would any new tab.

The Tableau CRM tab can include these features:

- Annotations, notifications, and full-screen presentation
- Sharing and downloading options
- Links to other Tableau CRM assets
- Record actions menu
- Link to Tableau CRM Studio (available for Tableau CRM users but not Experience Cloud sites users)

Customize Onboarding with In-Dashboard Instructional Content

Drive adoption and engagement with specialized educational resources right where users work. With the widget-specific Learn option, you can provide videos and webpages that help users get the most out of each dashboard and its charts.

Customized onboarding content can help users to understand and interpret the data, make informed decisions, and act based on the latest information. Selecting **Learn** in the widget dropdown menu opens the in-app, resizable, draggable onboarding frame, loaded with the content you specify.
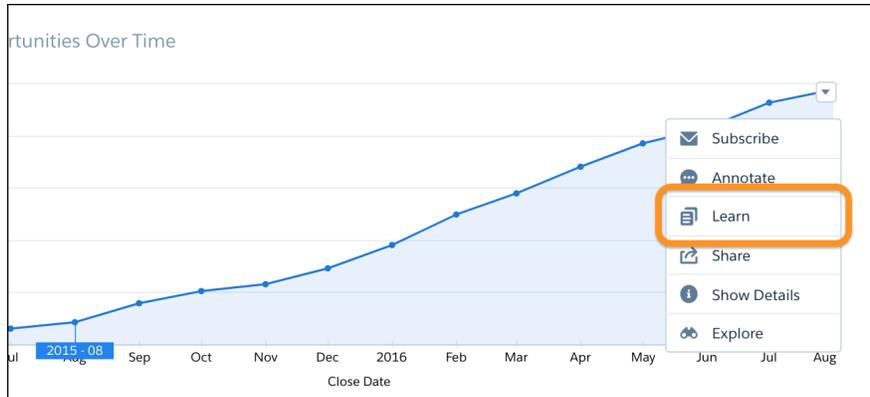
To set up customized onboarding, follow these steps.

1. In the dashboard designer, select the widget where you want to add the Learn option.

2. In the widget properties panel, open the Onboarding section.

3. Enter a title for the frame that presents your resource.

4. Enter the URL for your video or webpage. See considerations below.

5. Click **Preview your content** to see what your users will experience.

6. (Optional) Add custom onboarding for multiple charts to create a step-by-step learning progression through the dashboard.

When you have at least one widget with onboarding, the Learning Resources icon appears. Click the icon to open and close the Learning Resources panel. When the panel is open, hover on a learning resource to highlight the chart that goes with it. Each learning resource is numbered and the numbers appear on the charts for easy reference.
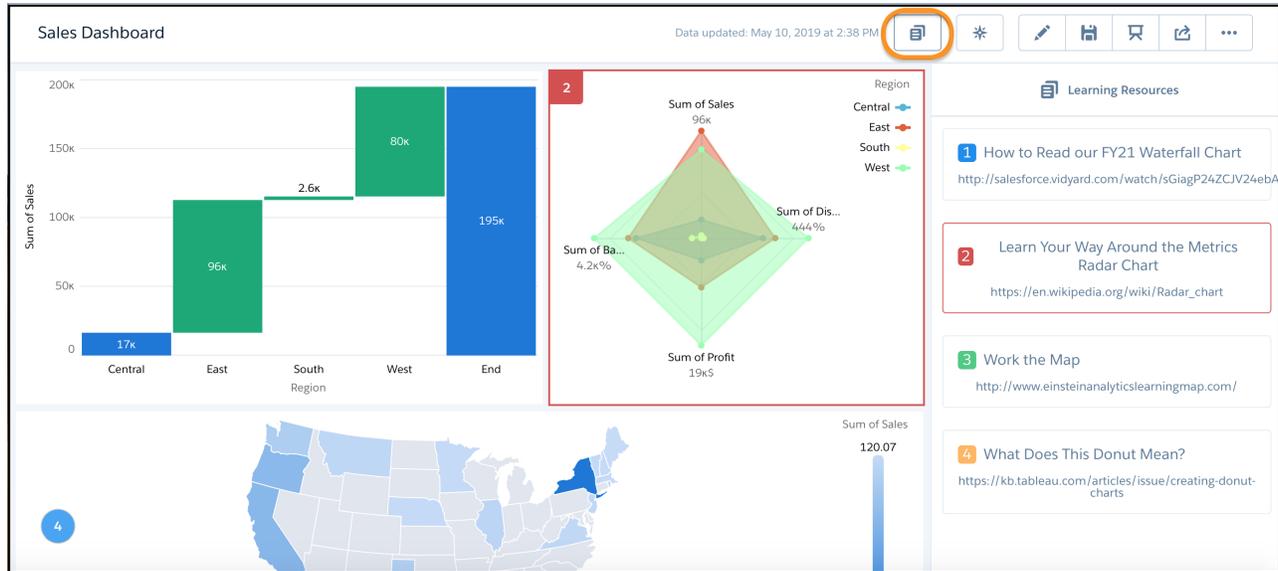
Take note of these considerations for dashboard widget onboarding.

- Provide the URL to a site that doesn't prevent embedding.
- The Onboarding property loads content from any URL into a Tableau CRM inline frame. If the URL allows execution within an inline frame, then all content at the URL becomes accessible to dashboard viewers. Salesforce is not responsible for third-party URLs.
- If your resource is secured, instruct users to authenticate so that the resource can appear in the onboarding frame.
- For best results with YouTube, provide your video's embed URL. To get the embed URL in YouTube, click **Share** and then **Embed**. The URL must have this format: *https://www.youtube.com/embed/xxxxxxxxx*.
- When designing web content for widget onboarding, keep in mind that while the onboarding frame allows linking to other webpages, browser controls such as Back and Refresh aren't accessible.

Customize Tableau CRM Dashboards using JSON

Configure advanced features in Tableau CRM dashboards using JSON.

The easiest way to build dashboards in Tableau CRM is to use the designer. However, if needed, you can further customize dashboards by editing their JSON files. The JSON defines the components of the dashboard and how they interact.

Modify a dashboard's JSON file to perform advanced customization tasks that can't be accomplished in the designer's user interface, like:

- Manually set up bindings to override the default faceting behavior and specify the relationships between the steps that aren't bound by default.
- Set query limits.
- Specify columns for a values table.

For complete information about Dashboard JSON, see the *Tableau CRM Dashboard JSON Reference*.

Format Measures and Display Elements with Tableau CRM Extended Metadata (XMD)

Set up your data and customize dashboard elements with Tableau CRM Extended Metadata.

Extended metadata (XMD) enables you to customize the formatting of dataset fields and their values in Tableau CRM dashboards and lenses. If you modify the XMD for a dataset, every UI visualization that uses the dataset shows the modified format.

You can customize the following with XMD:

- Format measures. Example: Show the decimal and grouping separators for currency.
- Add prefixes and suffixes to measures. Example: Show the percent symbol (%) after each percentage.
- Multiply measures by a factor. Example: Multiply by 100 to convert a decimal to a percent.
- Change display labels for dimensions and measures.
- Customize colors in charts based on field values.
- Define the first day of the week for the calendar year.
- Add action menus to dimensions that let dashboard viewers invoke actions from lenses and dashboards.

For complete information about Dashboard JSON, see the *Tableau CRM Extended Metadata (XMD) Reference*.

Create and Share Tableau CRM Apps to Give Business Users a Big Data Picture

Tableau CRM apps are purpose-built sets of analyses and answers that tell a story about a specific area of your business.

With Tableau CRM apps, you can provide curated paths through your data, plus powerful tools for spontaneous, deep explorations. After creating dashboards, lenses, and datasets, you can organize them in apps to present dashboards in relevant order, and then share apps with appropriate groups.

Build your own Tableau CRM apps that contain datasets, dataflows, recipes, lenses, dashboards—the assets you create in Tableau CRM Studio. Apps can also include Einstein Discover stories. Or, select a prebuilt app from Salesforce, such as Sales Analytics or Service Analytics, as a starting point for your Tableau CRM solutions. Either way, Tableau CRM apps let you bundle Analytics assets and share them with other people in your organization.

App-Level Sharing
Analytics Cloud apps are like folders, allowing users to organize their own data projects—both private and shared—and control sharing of datasets, lenses, and dashboards.

Create an App
Create an app to organize and contain any combination of lenses, dashboards, and datasets.

Set Up Navigation Through Your App
Customize app navigation so users can get right to the dashboards and lenses first that are targeted for them.

Share an App
To enable others to see a lens, dashboard, or dataset, one way to share is by sharing the app it's in.

Control Who Sees What in an App with Asset Visibility (Beta)
App Managers and Editors can use the Asset Visibility feature to control visibility of select assets. Asset visibility is limited for users that are assigned 'Viewer' sharing access. Assets remain visible to app Managers and Editors regardless of the asset's status.

Delete an App
If you have Manager access to an app, you can delete it. Deleting an app permanently removes all of its lenses, dashboards, and datasets from Analytics Cloud.

Tableau CRM Migration, Packaging, and Distribution
Migrate Tableau CRM assets using change sets, bundle them together in managed packages, distribute and track packages through AppExchange and the License Management App, and use the metadata API to manage customizations for your org.

App-Level Sharing

Analytics Cloud apps are like folders, allowing users to organize their own data projects—both private and shared—and control sharing of datasets, lenses, and dashboards.

All Analytics Cloud users start off with Viewer access to the default Shared App that's available out of the box; administrators can change this default setting to restrict or extend access. Each user also has access to a default app out of the box, called My Private App, intended for personal projects in progress. The contents of each user's My Private App aren't visible to administrators, but dashboards and lenses in My Private App can be shared.

All other apps created by individual users are private, by default; the app owner and administrators have Manager access and can extend access to other users, groups, or roles.

Here's a summary of what users can do with Viewer, Editor, and Manager access.

Action	Viewer	Editor	Manager
View dashboards, lenses, and datasets in the app 📝 Note: If the underlying dataset is in a different app than a lens or dashboard, the user must have access to both apps to view the lens or dashboard.	X	X	X
See who has access to the app	X	X	X
Save contents of the app to another app that the user has Editor or Manager access to	X	X	X
Save changes to existing dashboards, lenses, and datasets in the app (saving dashboards requires the appropriate permission set license and permission)		X	X
Change the app's sharing settings			X
Rename the app			X
Update asset visibility in an app		X	X
Delete the app			X

🛑 **Important:** When users are deactivated, they lose share and delete access to all apps they manage. To avoid "stranding" an app, be sure that manager access is assigned to at least one active user BEFORE deactivating the user who's the manager of the app.

Create an App

Create an app to organize and contain any combination of lenses, dashboards, and datasets.

1. On the home page, click **Create**.

2. Select **App**.

3. Click **Create Blank App**, and then click **Continue**.
   To start from a template, see Create Apps from Tableau CRM Templates on page 1391.

4. Enter the name of your app, and click **Create**.
   Your app is created and displayed in a new tab. You can now save lenses, dashboards, and datasets in the app.

5. Click the **Details** sub tab.

6. Add a description of the app's purpose and contents so that your colleagues understand how to use the data.

   a. Click the pencil icon in the description field.

   b. Type a description.
      The description is automatically saved.

7. Change the app's default icon to make it easier to distinguish from other apps.

   a. Click the default icon.

   b. Select an icon from the gallery.
      The new icon is automatically saved.

> ⓘ **Important:** When users are deactivated, they lose share and delete access to all apps they manage. To avoid "stranding" an app, be sure that manager access is assigned to at least one active user BEFORE deactivating the user who's the manager of the app.

Set Up Navigation Through Your App

Customize app navigation so users can get right to the dashboards and lenses first that are targeted for them.

1. In Tableau CRM Studio, select the app. Then click **Run App**.

2. Open the navigation list by clicking the disclosure triangle ▼ next to the app name.

3. If you're the manager of an app, you see the **Edit List** button. Click it to set up the app.
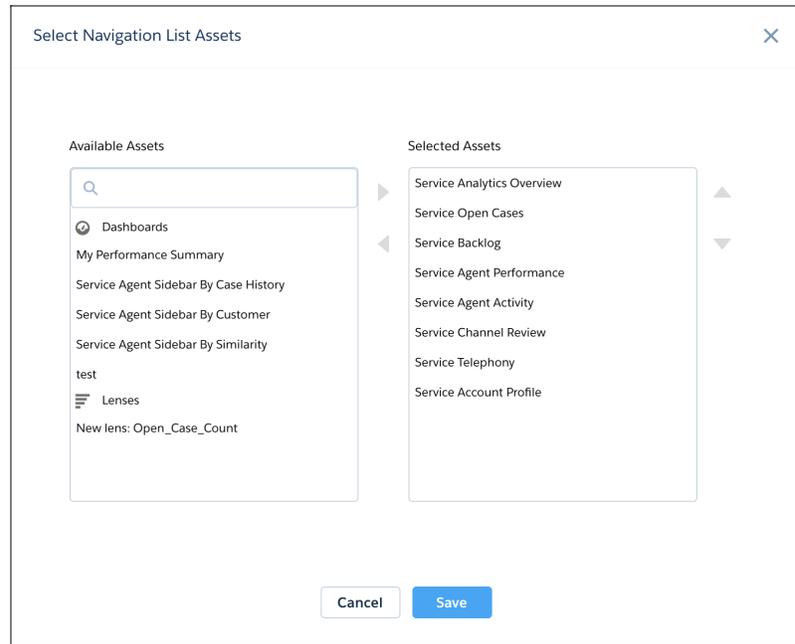
4. Change the presentation order by sliding the Selected Assets up or down. To add an available dashboard or lens to the navigation list, select it and click the right arrow. To hide an asset from the navigation list, select it and click the left arrow.

5. To retain the navigation for your app, click **Save**.
   Share the app with colleagues or customers. When they run the app, it starts with the first asset in your navigation list.

Share an App

To enable others to see a lens, dashboard, or dataset, one way to share is by sharing the app it's in.

1. On the app page, click the **Share** button.

2. On the Give Access tab:

   a. Choose whether you're sharing the app with a user, group, or role.

   b. Start typing the name and select from the suggested matches.

   c. Choose the level of sharing access: Viewer, Editor, or Manager.

   d. Click **Add**.

   e. Click **Save**, then click **Done**.

🛇 **Important:** When users are deactivated, they lose share and delete access to all apps they manage. To avoid "stranding" an app, be sure that manager access is assigned to at least one active user BEFORE deactivating the user who's the manager of the app.

Control Who Sees What in an App with Asset Visibility (Beta)

App Managers and Editors can use the Asset Visibility feature to control visibility of select assets. Asset visibility is limited for users that are assigned 'Viewer' sharing access. Assets remain visible to app Managers and Editors regardless of the asset's status.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To share an app:
- Use Analytics and Manager access to the app

📝 **Note:** As a beta feature, Asset Visibility is a preview and isn't part of Tableau CRM under your master subscription agreement with Salesforce. Use this feature at your sole discretion, and make your purchase decisions only on the basis of generally available products and features. Salesforce doesn't guarantee general availability of this feature within any particular time frame or at all, and we can discontinue it at any time. This feature is for evaluation purposes only, not for production use. It's offered as is and isn't supported, and Salesforce has no liability for any harm or damage arising out of or about it. All restrictions, Salesforce reservation of rights, obligations concerning Tableau CRM, and terms for related Non-Salesforce Applications and Content apply equally to your use of this feature. You can provide feedback and suggestions for Asset Visibility in the Trailblazer Community.

Contact your Salesforce account representative to enable Asset Visibility in your org.

1. Hide an asset in an app.

   a. Open the app and locate the asset to update.

   b. Click the row level action drop-down menu.

   c. Click **Hide** to change the asset's visibility to hidden.

   Hidden assets display a transparent state giving app managers an at-a-glance view of an asset's status. Asset status is also available in the asset tooltip.
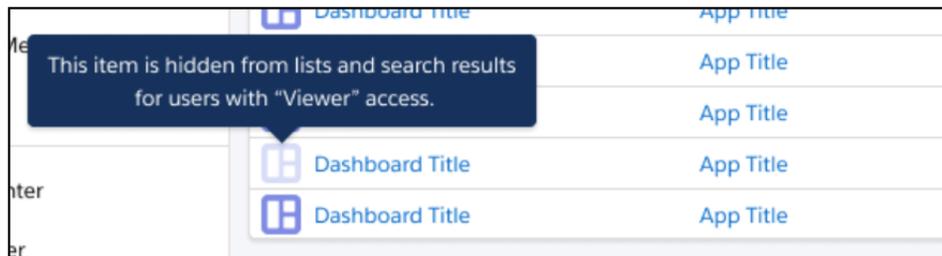
2. Show an asset in an app.

   a. Open the app and locate the asset to update.

   b. Click the row level action drop-down menu.

   c. Click **Show** to change the asset's visibility to visible.

As an app manager, the asset visibility settings go beyond the shared app and impact other areas of Tableau CRM for app members with Viewer status. Consider the following before updating asset status.

Analytics Studio

- Hidden assets aren't available in Tableau CRM Home, which includes Recent Dashboards, Favorites, Created by Me, Shared with Me.

- Hidden assets can't be browsed, seen in All Items, or any other list.

- Hidden assets don't appear in a search's Recents listing.

- Hidden assets don't appear in search results, including pop-over and the full search results page.

- URLs associated with a hidden dashboard or lens are blocked.

- Users can access hidden dashboards that are embedded (included in a Salesforce page or report).

- Users can add items from a hidden dashboard to a watchlist if the dashboard is embedded.

- Users receive a "Resource not found" error while navigating to a hidden source dashboard from watchlist or notification.

- Users can define notifications and subscriptions using items from a hidden dashboard that's embedded.

Analytics Tab

- Hidden assets aren't available in Recents.

- Hidden assets don't appear in a search's most recently used (MRU) pop-over.

- Access is blocked to hidden assets in Salesforce global search results, including pop-over and the full search results page.

Tableau CRM Mobile

- Hidden assets aren't accessible at all for users with Viewer status.

Salesforce Mobile

- Users can access hidden dashboards that are embedded in a Salesforce page.

Delete an App

If you have Manager access to an app, you can delete it. Deleting an app permanently removes all of its lenses, dashboards, and datasets from Analytics Cloud.

1. Open the app, and then click **Delete**.

   ⚠ **Warning:** You can't recover a deleted app.

   If applicable, Analytics Cloud lists the lenses and dashboards outside of your app that will be affected. When the datasets belonging to your app are deleted, any lens or dashboard that references those datasets becomes unusable.
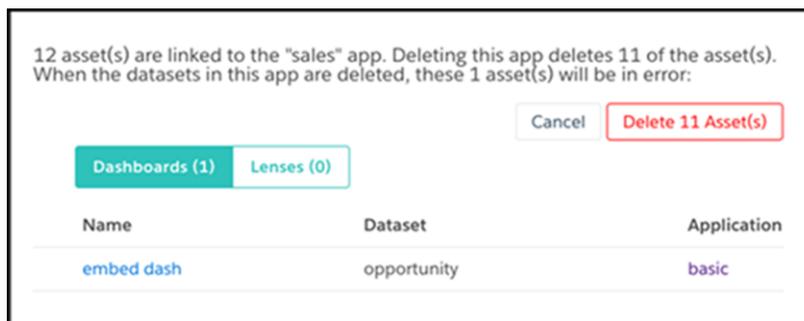
   ⓘ **Important:** Only lenses and dashboards to which you have access appear in the list. If you don't have access to them, lenses and dashboards aren't listed even though they have references to datasets in your app.

2. Click **Delete Asset(s)**.

Tableau CRM Migration, Packaging, and Distribution

Migrate Tableau CRM assets using change sets, bundle them together in managed packages, distribute and track packages through AppExchange and the License Management App, and use the metadata API to manage customizations for your org.

Migrate Tableau CRM Assets with Change Sets

Use change sets to move customized Tableau CRM assets between orgs that have a deployment connection. For example, create a Tableau CRM app containing dashboards, lenses, datasets, dataflows in your Sandbox org, then migrate the app assets to your production org once testing is complete.

Tips for Migrating Tableau CRM Assets with an Ant Script

If you use the Salesforce Ant Migration Tool to migrate Tableau CRM assets between orgs, consider the following tips.

Package Tableau CRM Assets in Managed Packages

You can create managed packages of Tableau CRM assets, including Tableau CRM apps, dashboards, lenses, datasets, recipes, dataflows, and user XMD. Use packages to distribute those assets to other users or organizations, including those outside your company.

Distribute Tableau CRM Assets Through AppExchange

Once your managed package is certified through our security review, you can make it available to your customers by uploading it to AppExchange. You can specify your package release type, control major and minor version numbers, and specify licensing options through the Licensing Management Application.

Use the Metadata API for Tableau CRM Assets and User XMD

Tableau CRM provides full support for the Metadata API, which can be used to retrieve, deploy, create, update, or delete customizations for your organization.

Migrate Tableau CRM Assets with Change Sets

Use change sets to move customized Tableau CRM assets between orgs that have a deployment connection. For example, create a Tableau CRM app containing dashboards, lenses, datasets, dataflows in your Sandbox org, then migrate the app assets to your production org once testing is complete.

💡 **Tip:** Remember, this topic describes how to migrate Tableau CRM assets—such as lenses, customized dashboards, dataset containers, and so on. It does not discuss migrating Salesforce or Tableau CRM data from org to org. Where this topic refers to a Tableau CRM dataset, what actually gets migrated is the container; the dataset metadata.

To migrate Tableau CRM assets:

1. Create and test your Tableau CRM assets in your source org, such as Sandbox.

2. From Setup in your source org, select **Outbound Change Sets** and create a new one.

3. Click **Add** in the Change Set Components section, then use the **Component Type** drop-down list to select and add your Tableau CRM assets to the change set. Each asset type is prefixed with "Analytics." Note that you must add dependent dataflows manually. That is, if you add a dataset to a change set, the associated dataflows are not automatically picked up.

Important: Analytics components include Analytics applications, dashboards, dataflows, datasets, lenses, recipes, and user XMD. As you package Analytics components, keep these tips and best practices in mind.

- Analytics unmanaged packages, as opposed to managed packages, are considered a developer-only feature and aren't supported for general-purpose distribution. While Analytics unmanaged packages work as expected within the constraints of Salesforce unmanaged packages, they aren't subject to as much testing as managed packages. Unmanaged packages come without many of the safeguards of managed packages, and are intended for developers familiar with their limitations. Also refer to the relevant topic in the ISV Guide.

- Before a recipe is available for packaging, you must create a dataset with the recipe. The related dataflow must be added to the package along with the recipe for deployment to succeed.

- Analytics Admin permissions are required to create a package but not for deployment, which requires only Salesforce admin permissions.

- There's no spidering between datasets and dataflows, meaning there's no dependency following. When packaging both, they must be added manually. If they aren't, an error appears during deployment. The same is true for change sets—when packaging both datasets and dataflows, add them manually.

- When you package a data flow, source and security predicates aren't included in the package.

- Because views are user-specific, they aren't included when you package the dashboard.

- If you migrate dashboards manually using JSON copy and paste, any conditional formatting, widget-specific number formats, and measure labels on blended queries are lost. To retain these formats and labels in the migrated dashboard, include the Analytics Dataset Metadata component type when packaging your change set.

- The Winter '18 release contains a beta version of Apex steps, which lets developers include custom Apex functionality in a dashboard to access Salesforce platform features that aren't inherently supported in Analytics. If you include dashboards in a package, Apex steps aren't included—migrate Apex classes separately.

- Before the Spring '17 release, images didn't render when deploying a dashboard that used an image widget that referenced image files not available on the target org. There were two workarounds: Manually upload the images, or add a folder containing the images to the package. As of the Spring '17 release, images are packaged with the dashboard, and references between dashboards are maintained. You can't delete a dashboard that is referenced in a link. Either re-create the image, or link the widgets in the dashboard in the source org. Then repackage or fix the link issues in the target org.

- Take care when packaging dataflows. Invalid schema overrides and unsupported or illegal parameters are removed. For example, `Type = dim` is no longer supported. Use `Type = text` instead. Comments in JSON are removed. Nodes can appear in a different order.

4. Click **View/Add Dependencies** to ensure all dependent assets and appropriate permission sets and profile settings are added.

5. Click **Upload** and select your target org, such as Production. Make sure that the target org allows inbound connections. The inbound and outbound orgs must have a deployment connection.

6. From **Setup** in your target org, select **Inbound Change Sets** and find the one you uploaded from your source org.

7. Validate and deploy the change set, making sure to assign the right users to any permission sets and profiles you included. Your Tableau CRM assets will be available in the target org.

For details on using change sets to migrate configuration changes between orgs, view the Release Management: Deploying Changes Using Change Sets (Salesforce Classic) video.

SEE ALSO:

Change Sets

Change Sets Best Practices

Upload Outbound Change Sets

Deploy Inbound Change Sets

Tips for Migrating Tableau CRM Assets with an Ant Script

If you use the Salesforce Ant Migration Tool to migrate Tableau CRM assets between orgs, consider the following tips.

If you migrate a dashboard configured with conditional formatting, you must manually add the corresponding asset XMD to the project manifest (package.xml) before deployment to the target org. This step is required because the asset XMD contains the conditional formatting settings for the dashboard and is not included in the package.xml file, by default.

In the following sample package.xml, the first `<types>` node specifies the dashboard and the second node specifies the associated asset XMD.

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
<types>
    <members>MyDashboard_DevName</members>
    <name>WaveDashboard</name>
</types>
<types>
    <members>MyDashboard_DevName</members>
    <name>WaveXmd</name>
</types>
```

```
<version>43.0</version>
</Package>
```

For the asset XMD `<types>` node, `<members>` specifies the dashboard dev name and `<name>` specifies the metadata type for an asset XMD.

If you don't add the asset XMD to package.xml, the dashboard migrates without conditional formatting. If needed, you can reapply the conditional formatting to the dashboard in the target org.

Package Tableau CRM Assets in Managed Packages

You can create managed packages of Tableau CRM assets, including Tableau CRM apps, dashboards, lenses, datasets, recipes, dataflows, and user XMD. Use packages to distribute those assets to other users or organizations, including those outside your company.

A package is a container for something as small as an individual component or as large as a set of related apps. Packages can be distributed to other Salesforce users and organizations, including those outside your company. While packages come in two forms—unmanaged and managed—Tableau CRM supports only managed packages, which must be created from a Developer Edition organization.

Why only managed packages? There are some key advantages, especially around licensing and managing licenses. Using the AppExchange and the License Management Application (LMA), partners, ISVs, and developers can sell and manage user-based licenses for the app. Managed packages are also upgradeable, meaning you don't have to start over; you can incrementally upgrade.

To package Tableau CRM assets:

1. From Setup in your Developer Edition org, select **Packages** and click **New** to create a managed package.

2. Click **Add** in the Components tab, then use the **Component Type** drop-down list to select and add your Tableau CRM assets to the package. Each component type is prefixed with "Analytics". Note that you must add dependent dataflows manually. That is, if you add a dataset to a package, the associated dataflows are not automatically picked up.

   > ⛔ Important: Analytics components include Analytics applications, dashboards, dataflows, datasets, lenses, recipes, and user XMD. As you package Analytics components, keep these tips and best practices in mind.
   >
   > - Analytics unmanaged packages, as opposed to managed packages, are considered a developer-only feature and aren't supported for general-purpose distribution. While Analytics unmanaged packages work as expected within the constraints of Salesforce unmanaged packages, they aren't subject to as much testing as managed packages. Unmanaged packages come without many of the safeguards of managed packages, and are intended for developers familiar with their limitations. Also refer to the relevant topic in the ISV Guide.
   >
   > - Before a recipe is available for packaging, you must create a dataset with the recipe. The related dataflow must be added to the package along with the recipe for deployment to succeed.
   >
   > - Analytics Admin permissions are required to create a package but not for deployment, which requires only Salesforce admin permissions.
   >
   > - There's no spidering between datasets and dataflows, meaning there's no dependency following. When packaging both, they must be added manually. If they aren't, an error appears during deployment. The same is true for change sets—when packaging both datasets and dataflows, add them manually.
   >
   > - When you package a data flow, source and security predicates aren't included in the package.
   >
   > - Because views are user-specific, they aren't included when you package the dashboard.
   >
   > - If you migrate dashboards manually using JSON copy and paste, any conditional formatting, widget-specific number formats, and measure labels on blended queries are lost. To retain these formats and labels in the migrated dashboard, include the Analytics Dataset Metadata component type when packaging your change set.
   >
   > - The Winter '18 release contains a beta version of Apex steps, which lets developers include custom Apex functionality in a dashboard to access Salesforce platform features that aren't inherently supported in Analytics. If you include dashboards in a package, Apex steps aren't included—migrate Apex classes separately.

- Before the Spring '17 release, images didn't render when deploying a dashboard that used an image widget that referenced image files not available on the target org. There were two workarounds: Manually upload the images, or add a folder containing the images to the package. As of the Spring '17 release, images are packaged with the dashboard, and references between dashboards are maintained. You can't delete a dashboard that is referenced in a link. Either re-create the image, or link the widgets in the dashboard in the source org. Then repackage or fix the link issues in the target org.

- Take care when packaging dataflows. Invalid schema overrides and unsupported or illegal parameters are removed. For example, `Type = dim` is no longer supported. Use `Type = text` instead. Comments in JSON are removed. Nodes can appear in a different order.

3. Click **View Dependencies** to ensure all dependent assets and appropriate permission sets and profile settings are added.

4. Once packaged, prepare your apps for distribution.

SEE ALSO:

    Package and Distribute Your Apps

    Introduction to the License Management App

Distribute Tableau CRM Assets Through AppExchange

Once your managed package is certified through our security review, you can make it available to your customers by uploading it to AppExchange. You can specify your package release type, control major and minor version numbers, and specify licensing options through the Licensing Management Application.

To distribute a Tableau CRM managed package:

1. Submit your package for a security review.

   Before a package can be listed on AppExchange, it must meet established standards. Since packages containing Tableau CRM assets can potentially contain standard Salesforce objects too, they are subject to the same security requirements as all other AppExchange listings.

2. When your package is ready, click **Upload**, then fill in the required information.

3. Add version specifics, including the name (usually a description and date) and the version number.

4. Choose the release type—*Managed - Beta* or *Managed - Released*—and an appropriate version number.

   The *Managed - Beta* type is for testing, usually with a subset of customers, and allows changes that the *Managed - Released* type does not.

5. Add a description for your AppExchange listing.

6. Optionally, add URLs for release notes and instructions, a password (if you wish to keep the package private), and any other objects required.

7. Click **Upload**.

📝 **Note:** Deployment requires Salesforce admin permissions.

Use the AppExchange and the License Management Application (LMA) to manage user-based licenses. You can request the application through your partner community site.

Use the Metadata API for Tableau CRM Assets and User XMD

Tableau CRM provides full support for the Metadata API, which can be used to retrieve, deploy, create, update, or delete customizations for your organization.

Every time your Salesforce organization is customized, its metadata is modified. Editing page layouts, creating a custom field, or adding a Tableau CRM dataflow are all metadata updates. Metadata is the information that describes the configuration of your organization and of your Tableau CRM assets. It's data about your data. For example, the metadata type `WaveLens` includes properties that describe the lens—what dataset it uses, the label of the dashboard, what visualization widget type it uses (pivottable, stackvbar, heatmap, and so on).

You can create your own custom metadata types, which enable you to create your own setup objects whose records are metadata rather than data. Rather than building apps from data records in custom objects or custom settings, you can create custom metadata types and add metadata records, with all the manageability that comes with metadata: package, deploy, and upgrade.

While the Metadata API is the foundation on which change sets and packaging is built, developers can use the Metadata API to manipulate their Tableau CRM assets. For example, your application could retrieve metadata about your dashboards and back it up to (and restore from) some repository, effectively providing version control.

You can package and deploy user XMD too. User XMD is a JSON document that allows an application to override certain metadata for a dataset. You can govern settings to control how the data is presented—for example, colors, formats, and labels. To deploy the overridden dataset metadata, you must package the user XMD attributes. Metadata API support allows migration tools to migrate datasets with user XMD between orgs—for example, you can migrate user XMD overrides from sandbox to production. Partners can deploy their application-specific user XMD with their AppExchange package. They retain control of the user XMD and can update it with future package versions. Refer to the WaveXmd metadata type in the Metadata API Developer Guide.

The following Tableau CRM metadata types are supported in the Metadata API:

- WaveApplication
- WaveDashboard
- WaveDataflow
- WaveDataset
- WaveLens
- WaveRecipe
- WaveXmd
- WaveTemplateBundle

Deploy Tableau CRM Prebuilt Apps

Tableau CRM templated apps give you a fast way to get started with your analytics journey. Create an app from a template to get ready-made dashboards and KPIs that your team can use out of the box or customize to meet your needs.

Tableau CRM templated apps can speed up your organization's time-to-value with Tableau CRM. Instead of building visualizations yourself, create templates that do the heavy lifting for you. You can create an app from a template with just a few clicks or answer questions in a handy configuration wizard. Templated apps can also be installed automatically when you install managed packages. Tableau CRM takes care of the rest, creating datasets and dashboards designed and built by Salesforce based on our years of experience helping companies manage customer interactions.

Many apps are designed for desktop and mobile devices. These apps come with dashboards that you can embed in Salesforce pages to give users direct access to business intelligence where they do their everyday work. And you can drill deeper into key aspects of your business by customizing to meet your specific needs.

To get started, see generic instructions for creating apps from templates and specifics for each template, and how to monitor auto-installed apps by following these links.

Create Apps from Tableau CRM Templates: Start Here

Follow these general procedures when you create Apps from any Tableau CRM Template. Before you create an app, see the help for the Tableau CRM Template you want to use for requirements and instructions specific to that template.

Monitor, Update, and Delete Auto-Installed Apps

Templated Tableau CRM apps are automatically installed in Salesforce orgs via managed packages. Use the Auto-Installed App setup page to monitor the status of auto-install requests, view logs to troubleshoot auto-install issues, and update and delete auto-installed apps.

The Einstein Accuracy Analytics Template

The Einstein Accuracy Analytics app helps you monitor how well Einstein Discovery models predict actual outcomes over time.

Adoption Analytics Template

Create an app from the Adoption Analytics template for ready-made insight into how your team uses Analytics Cloud apps, dashboards, lenses, and datasets.

Analytics for Retail Banking Template

Analytics for Retail Banking is part of a suite of Tableau CRM apps for Financial Services Cloud customers. Its dashboards visualize all the metrics and key performance indicators (KPIs) personal bankers require to grow client relationships.

Appointment Analytics Template

Appointment Analytics brings Lightning Scheduler appointments data into Tableau CRM to help you deliver the best service possible to your customers.

Approval Analytics Template

Create an app from the Approval Analytics Template to increase your visibility into approval processes. Managers and team leaders can use the app to view approval history, understand trends, identify bottlenecks, and take action to streamline the process.

B2B Commerce Analytics Template

The B2B Commerce Analytics template gives you a fast way to apply the power of Tableau CRM to your B2B ecommerce data.

B2B Marketing Analytics App

B2B Marketing Analytics provides advanced analytics for the data-driven marketer, letting you explore both your marketing and sales data in one place. With powerful dashboards that consolidate Pardot and Sales Cloud data, you can quickly explore data, understand the impact of marketing on revenue, and take instant action to drive marketing results.

Event Monitoring Analytics App

The Event Monitoring Analytics App integrates with event monitoring and setup audit trail data to give you insights into your user and org behavior. The app is a built-in way to explore your monitoring data in Salesforce. App creation is easy and with its prebuilt dashboards and datasets, you can start exploring right away. This app helps you drill into your org's data and swiftly identify suspicious behavior, slow page performance, and poor user adoption.

Campaign Analytics Template

The Campaign Analytics template uses the power of Tableau CRM to show how your marketing campaigns impact the bottom line.

Change Analytics Template

Use Change Analytics to visualize field history data in any Salesforce object and learn how, when, and where members of your team change data.

Consumer Banking Starter Analytics Template

Consumer Banking Starter Analytics is part of a suite of Tableau CRM apps for Financial Services Cloud customers. It gives you a quick way of creating an analytics solution personal bankers can use to move clients up the value chain.

Design Style Guide Template

Use the Design Style Guide template to create an app with ready-made dashboard, charts, and widget templates that you can use to get started fast with your own apps.

Tableau CRM for Financial Services

Tableau CRM for Financial Services gives financial advisors and personal bankers a comprehensive customer intelligence solution based on the Tableau CRM platform. It includes Einstein Discovery, which lets you create automated artificial intelligence models on any dataset without writing code.

Field Service Analytics

The Field Service Analytics app brings the power of Analytics to Field Service Lightning on any device that supports Tableau CRM.

Fundraising Analytics Template

The Fundraising Analytics template lets you create an app that brings the power of Analytics Cloud to data from the Salesforce Nonprofit Success Pack.

Fundraising Performance Analytics Template

Nonprofits can measure the impact of fundraising work with a precise picture of donor retention and attrition. The app also provides actionable insight into donor gains and losses, donation amount gains and losses, and other key performance indicators.

The Analytics for Healthcare Template

Use actionable insights from Tableau CRM for Health Cloud apps to drive intelligent patient engagement, improve care effectiveness, and manage patient risk.

The Tableau CRM for Insurance Template

Analytics for Insurance powers agents with practical insights on their sales performance and enables them to be more efficient in sales execution. App visualizations segment the customer base and provide insights on upsell/cross-sell opportunities so agents can grow their written premiums. Managers get insights on their team's performance and what makes the top performers different so they can coach their team members.

Lead Trending Analytics Template

Create an app from the Lead Trending Analytics Template to increase your visibility into your team's lead conversion process. Sales operations can get instant insight into how quickly the team converts leads and can identify bottlenecks in the conversion process.

The Tableau CRM for Manufacturing Template

The Tableau CRM for Manufacturing lets account managers visualize all aspects of their business to keep them on top of sales agreements, orders, and contracts.

Analytics for Mortgage Template

Analytics for Mortgage enables loan officers and managers to drive increased mortgage sales by helping prioritize the customer leads and mortgage applications to focus on. Process loans faster by highlighting the mortgage applications that have the highest value, applications that have been open the longest, and applications that are missing documents.

Multi Org Sales Analytics Template

Use the Multi Org Sales Analytics template to create a scaled-down version of the Sales Analytics that pulls in data from all your connected orgs.

Patient Risk Stratification Template

The Patient Risk Stratification template creates an app that lets healthcare coordinators identify high-risk patients to make sure they get the extra care they need.

Pipeline Analytics Template

Use Pipeline Analytics to plug your pipeline snapshot data into a ready-made waterfall dashboard. You get instant insight into how the pipeline changes between two snapshots.

Pricing Analytics Template

With the Pricing Analytics template, Salesforce CPQ customers can be sure they're pricing their deals correctly. It recommends the ideal price for every quote and helps you identify potentially underpriced deals.

Public Sector Case Analytics Template

Public Sector Case Analytics lets public sector service organizations measure their effectiveness and identify immediate actions to improve case work performance.

Quoting Analytics Template

Salesforce CPQ customers: Create an app using the Quoting Analytics template to get instant insights into your configure-price-quote (CPQ) processes.

Sales Analytics

Sales Analytics brings the power of Analytics to Sales Cloud on any device that supports Analytics. With intuitive visualizations based on your Salesforce data, Sales Analytics lets you move from insight to action quickly and helps you turn data into smarter sales.

Einstein Discovery for Sales Analytics

Use the Einstein Discovery for Sales Analytics template to apply the power of Einstein Discovery predictive analytics to your Sales Cloud data.

Service Analytics

The Service Analytics template gets you started fast with Analytics and provides a clear path through your Service Cloud data on any device. Whether you're a service manager or agent, you get everything you need in one place to uncover key data insights to help you grow your business.

Salesforce Analytics for Veeva Template

For Veeva CRM customers only: Salesforce Analytics for Veeva assembles key metrics in one place to give you insight into pharmaceutical reps processes and effectiveness.

Snapshot Analytics Template

Create an app using the Snapshot Analytics template to trend data from any existing Tableau CRM dataset or Salesforce object.

Revenue Operations Analytics

Use the Revenue Operations Analytics template to help sales teams gain performance insights to build stronger pipelines, improve forecast accuracy, and generate more revenue.

Social Case Analytics Template

The Social Case Analytics template creates an app that provides ready-made insights into team performance on each social channel.

Subscription Analytics Template

Salesforce CPQ customers: Create an app from the Subscription Analytics template to track your renewal business.

Predict Client Churn Risk for Wealth Management Analytics

Predict Client Churn Risk for Wealth Management Analytics allows financial advisors to intelligently predict customer churn. Using the power of Einstein Discovery, the app prescribes corrective actions to help minimize occurrences of churn.

Predict Likelihood to Add Assets for Wealth Management Analytics

Predict Likelihood to Add Assets for Wealth Management Analytics allows financial advisors to intelligently forecast the likelihood of adding assets to accounts. Using the power of Einstein Discovery, the app prescribes actions on how to increase the chances to grow your account assets.

Analytics for Wealth Management Template

Analytics for Wealth Management a comprehensive analytics solution that's part of a suite of Tableau CRM apps for Financial Services Cloud customers. Its extensive sets of dashboards apply the power of Tableau CRM to all significant data and KPIs from the Financial Services Cloud. .

Wealth Starter Analytics Template

Creating an app from the Wealth Starter Analytics template gives Financial Services Cloud customers a quick way to apply the power of Tableau CRM to data about their book of business.

Create Apps from Tableau CRM Templates: Start Here

Follow these general procedures when you create Apps from any Tableau CRM Template. Before you create an app, see the help for the Tableau CRM Template you want to use for requirements and instructions specific to that template.

Before creating an app, set up your org to use the Analytics Cloud platform and Tableau CRM templated apps. See Set Up the Tableau CRM Platform.

💡 **Tip:** For best results, follow the steps in the order shown.

1. Create and Share an App from a Tableau CRM Template

   Follow these generic steps to create and share an app from the Tableau CRM template of your choice.

2. How to Answer Tableau CRM Template Custom Wizard Questions

   Some Tableau CRM templates include a custom configuration wizard, which asks you a series of questions about your data. Follow these general guidelines to get the best result. Help for each template with a custom wizard provides details for that template.

3. Schedule Data Sync and the Dataflow for an Tableau CRM Template

   The app creation process includes a dataflow that imports the latest data to Tableau CRM. Schedule the sync and dataflow to be rerun every day to ensure that your app uses up-to-date data.

4. Set Field Level Security to Enable Creation of an Tableau CRM Template

   Before creating an app from a template, you may have to give the Analytics Integration User access to all fields used in the app.

5. Reconfigure a Tableau CRM App

   To restore deleted or altered dashboards or change wizard settings, reconfigure an existing app.

6. Upgrade a Tableau CRM App

   Take advantage of a Tableau CRM app's latest features by upgrading when we release a new version.

Create and Share an App from a Tableau CRM Template

Follow these generic steps to create and share an app from the Tableau CRM template of your choice.

> ⊘ **Important:** Before following these steps, read the help for the template you want to create. Each template has specific org setup and data requirements. Make sure that your org meets those requirements before trying to create the app.

USER PERMISSIONS

To create and manage Tableau CRM apps:
- Manage Analytics Templated Apps

1. Navigate to **Analytics Studio**.

2. Click **Create**, then select **App**.
   Alternatively, click **Template Gallery** to see app template overviews, get ideas for your dashboards, and select the best template for your use case. The gallery has categorization tags for quick reference and a Recommended section with the most popular templates.

3. Select a template. You may have to scroll to find it. Then click **Continue**.

   If the template you choose includes a configuration wizard, it opens now. The help for your template includes instructions about the wizard.

   Answer the questions on each page of the wizard. The first page asks you to choose between creating a new app or using answers of an existing app if you've created an app from that template already. See the help for the template for instructions about using the wizard. If you don't see a page with questions, you're almost done creating your app.

   Some wizards perform a compatibility check of your org's data. If so, you see a wizard page that shows each stage of the org check. If it uncovers any issues, you see error messages with instructions about how to address them. Fix the issues and try app creation again.

4. Name your app and click **Create**.

View the status of app creation on the next page. The process takes a minute or two. Once it's complete, refresh the page to see your app.

> ✏ **Note:** If you see an error saying the Analytics Integration User does not have access to selected fields, edit Salesforce field-level security so the Integration User has the required access and create the app again.

Now that you've created the app, share it with users in your organization. You can only share it with users assigned the Tableau CRM Plus User permission set.

1. Open your app if it's not already open. If you've navigated away from Tableau CRM Studio, go back to it, select **All Items**, find your app, and click it.

2.
   Click the Share icon [icon] at upper right.

3. In the next screen, use the search field under **Invite others:** to find other users in your org.

4. Select whether you want to make the selected user a Viewer, Editor, or Manager of the app.

   > ⊘ **Important:** Users with the "Use Analytics Templated Apps" permission and Editor or Manager access to the app can create, edit, and delete assets in the app.

5. Click **Add**, then click **Save**.

How to Answer Tableau CRM Template Custom Wizard Questions

Some Tableau CRM templates include a custom configuration wizard, which asks you a series of
questions about your data. Follow these general guidelines to get the best result. Help for each
template with a custom wizard provides details for that template.

- **Most questions provide answers in pick lists showing fields from Salesforce objects.**
  The lists include standard Salesforce fields and any custom fields you've set up on an object.
  Answer these questions by selecting from the fields shown. Most questions of this type let you
  choose only one field, and some let you choose multiple fields.

- **You can choose a field from an object only once.** After you select a field, it's no longer
  available as an answer to other questions.

- **Other questions are yes/no, provide a set of options, or require you to enter text.**

- **Questions marked with an asterisk (*) require answers.**

- **Using the default answers results in a useful set of dashboards.** If you're not sure what to select, use the answer that's preselected.
  Some questions do not have preselected answers. In those cases, Tableau CRM reminds you to make a selection. If you're not happy
  with the result, you can delete the app and create it again.

- **You control only parts of the app with wizard settings, such as the filters used in dashboards.** We built the app to provide
  immediate value without much work on your part.

- **Only some answers make sense given the question's context.** The answer pick-lists contain many fields, but only some make
  sense for your app. For example, it's unlikely that you'd filter Accounts data by Photo URL, even though the Accounts object includes
  a Photo URL field.

🚫 **Important:** Fields that aren't available to the Analytics Integration User aren't visible in the wizard. See Schedule Data Sync and
the Dataflow for an Tableau CRM Template to give the Integration User access to all the fields you want your app to use.

Schedule Data Sync and the Dataflow for an Tableau CRM Template

The app creation process includes a dataflow that imports the latest data to Tableau CRM. Schedule the sync and dataflow to be rerun
every day to ensure that your app uses up-to-date data.

1. In Tableau CRM Studio, click the wheel icon at upper right and select **Data Manager**. Or, click the **Data Manager** link in the left-hand
column.

2. First, schedule the sync. Select the **Connect** tab on the left.

> ✏️ **Note:** If you can't see the Connect tab, you need to enable data sync in your org. See Enable Data Sync and Connections on
> page 693.

3. Click the arrow ▾ to the far right of **SFDC_LOCAL**, which is the name of the connection your app uses. From the menu that appears,

select **Schedule**.

4. Set a time for running the data sync. It's best to select a time outside normal working hours so the sync and dataflow don't interrupt
business activities. Then click **Save**.

5. Next, schedule the dataflow. Select the **Dataflows & Recipes** tab on the left.

6. Look for the dataflow that contains the name of your app, and click the triangle ⏷ to the far right.

7. Select **Schedule**, then check the box next to **Event-based**. You see a message telling you that the dataflow runs after the data sync—exactly what you want.

8. Click **Save**.

The sync and dataflow for your app now runs ever day at the time you set.

🛑 Important: For the Multi Org Sales Analytics Template only: Your app uses multiple connections. Be sure to schedule sync for each of them. The app uses only a single dataflow. Schedule the dataflow to run as instructed starting in Step 5.

Set Field Level Security to Enable Creation of an Tableau CRM Template

Before creating an app from a template, you may have to give the Analytics Integration User access to all fields used in the app.

You may have to set Salesforce field-level security to enable the Tableau CRM Integration User to see all fields used in the app you create. Integration users run the dataflow, and if they don't have proper field-level security permissions, the dataflow can fail.

Follow these steps in Lightning Experience.

1. Go to Salesforce Setup, enter `object` in the **Quick Find/Search** box, and hit **Enter**.

2. Select **Object Manager**.

3. Enter the name of the object whose field-level security you need to edit in the **Quick Find / Search** box, and hit enter.

4. Select the object you need to edit, then select **Fields & Relationships**.

5. Select the field you need to edit, then select **Set Field-Level Security**.

6. Look for the Analytics Cloud Integration User, check the box(es) for the required fields under **Visible**, and click **Save**.

7. Repeat the Steps 5 and 6 for all fields you want to use.

8. Refresh your browser cache.

Follow these steps in Salesforce Classic.

1. **Step 1.** Go to Salesforce Setup and enter the name of the object whose field-level security you need to edit in the **Quick Find/Search** box and hit **Enter**.

2. **Step 2.** Click the name of the object.

3. **Step 3.** The next window shows all the fields for the object. Go to the one(s) where you need to edit field-level security.

4. **Step 4.** Look for the Analytics Cloud Integration User, check the box(es) for the required fields under Visible, and click Save.

5. **Step 6.** Repeat Steps 2 through 5 for all objects with fields you want to use.

6. **Step 7.** Refresh your browser cache.

You can now create an app from the template.

<div style="border:1px solid">

USER PERMISSIONS

To create and manage Tableau CRM apps:
- Manage Analytics Templated Apps

- Edit Analytics Dataflows

</div>

Reconfigure a Tableau CRM App

To restore deleted or altered dashboards or change wizard settings, reconfigure an existing app.

⛔ **Important:** Reconfiguring your app gets rid of dashboard customizations, including fields or objects added to the dataflow. It also deletes actions you enable, or changes to security settings and dashboard colors and labels. If you've made any customizations, save copies of your dataflow definition file or dashboards. Then copy them into your reconfigured app.

1. Navigate to Tableau CRM Studio and open the app. Make sure you're viewing the app's landing page and that you can see the app name at upper left, with a list of dashboards below.

2. Click the **Reconfigure app** link below the app name. If you see **Upgrade** instead of **Reconfigure**, you need to upgrade to the new version of the app before you can use the reconfigure feature. See Upgrade a Tableau CRM App.

3. Review the next screen carefully. It warns you that reconfiguring overwrites app customizations. If you're comfortable overwriting customizations, check the box and click **Continue**. If not, click **Back** or the **X** in the upper right corner.

4. Complete the app creation process described in Create and Share an App from a Tableau CRM Template.

5. At the end of the process, you're asked to confirm that you want to reconfigure your app. This gives you one more chance to make sure you're comfortable overwriting customizations. If you are, click **OK**. Tableau CRM creates a new version of your app.

⛔ **Important:** If you delete your app, Tableau CRM doesn't retain the answers you selected when you created that version of the app, and you have to start the creation process all over again. If you wish to reuse settings from the app, keep it until you've completed recreating the app and then delete it.

Upgrade a Tableau CRM App

Take advantage of a Tableau CRM app's latest features by upgrading when we release a new version.

The banner at the top of your app home page tells you that we've released a new version. It also provides a link to more information about the release. There's also a link inviting you to start the upgrade process in the left-hand column of the home page, just below the app name.

Here's how to upgrade your app.

1. Click either the "What's new" link in the banner or the "New version available" link in the left-hand column of your app's home page.

2. You can read information about the new version on top of the page that opens. At the bottom of the page, find the buttons: **Upgrade current app** and **Create new app**.

3. See descriptions of what the buttons do by hovering over them. If you click **Upgrade current app**, you overwrite your current app and all its assets, replacing it with an app based on the new version. This also deletes any customizations, including any new fields or object you've added to the dataflow or changes you've made to security settings and dashboard labels and colors. If you click **Create new app**, you create a second copy of your current app based on the new version. Decide which option suits your circumstances, and click the appropriate button.

4. ⚠️ **Warning:** If you've customized your app, click **Create new app** instead of **Upgrade current app** to make an app copy based on the new version. This preserves the current version and any customizations, which you can then manually copy into the new version of the app.

   If you click **Upgrade current app**, you see a screen warning you that the upgrade option overwrites the current app and gets rid of any customizations you've made to it. If you're OK with that, check the box and click **Continue**. If you're not, click **Back** to return to the previous screen. Clicking **Continue** takes you to the configuration wizard. Skip ahead to step 6.

5. If you click **Create new app**, you're taken to the configuration wizard. You don't see a warning, since you're not overwriting your current app, which is preserved with any customizations you've made to it.

6. The configuration wizard is preloaded with the settings you chose last time you used the wizard. You can either keep those settings or change them. Go through each page of the wizard.

7. Once you complete the wizard, Tableau CRM shows a screen that indicates which assets are impacted by the upgrade. Review the screen to see how many datasets, dashboards, and lenses will be changed, deleted, or added by the upgrade. It also shows how changed assets will be modified—whether the change is to data or appearance. On the same screen, you have the option to download a file with code for changes made to the app. If you've customized the app, download and save the file so you can copy and paste customization code into the upgraded version.

8. *Upgrade option only*: If you're upgrading your app, click **Upgrade current app**. Remember, this overwrites any customizations you've made. You can also click **Back** to go back into the configuration wizard and change your selections or click the **X** in the upper-right corner to cancel.

9. *Create option only*: If you're creating a new app, name your app something different from the current version and click **Create new app**. This option saves your current app and all its customizations. You can also click **Back** to go back into the configuration wizard and change your selections or click the **X** in the upper-right corner to cancel.

If the link below the app name says "Reset app," you're using the latest version and don't need to upgrade.

Monitor, Update, and Delete Auto-Installed Apps

Templated Tableau CRM apps are automatically installed in Salesforce orgs via managed packages. Use the Auto-Installed App setup page to monitor the status of auto-install requests, view logs to troubleshoot auto-install issues, and update and delete auto-installed apps.

From Setup, enter `Auto-Installed Apps` in the `Quick Find` box, and then select **Auto-Installed Apps**.
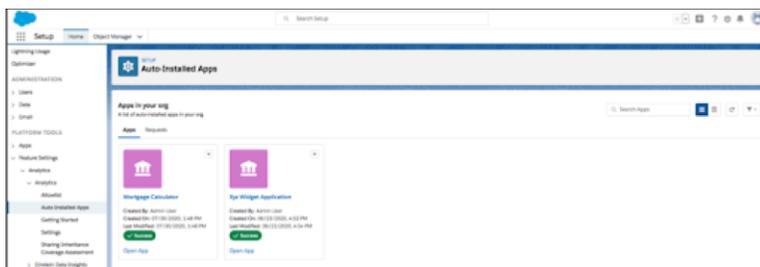
The main page has two tabs, **Apps** and **Requests**, with **Apps** as the default tab when the page is opened.

If there are no installed apps or auto-install requests in your org, the page shows that no apps or requests are available.
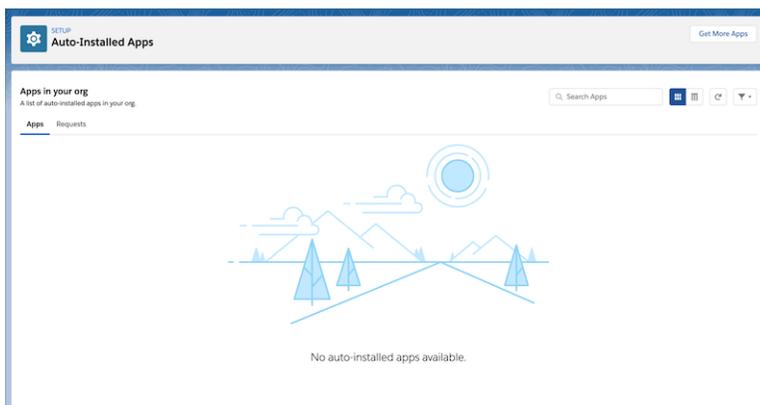
EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

1. Auto-Installed Apps

   Monitor and interact with the installed apps in your org on the Apps tab.

2. Auto-Install Requests

   Monitor and interact with the auto-install requests in your org on the Requests tab.

Auto-Installed Apps

Monitor and interact with the installed apps in your org on the Apps tab.
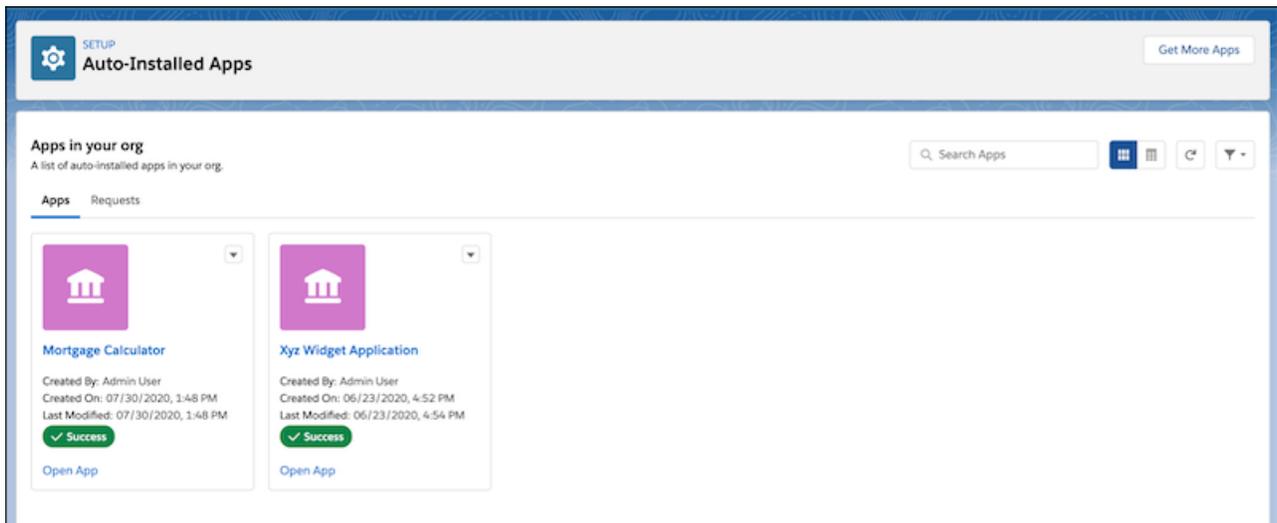
The default view of your install apps is the tile view. You can also view the installed apps in a list view.

The information for each installed app includes:

Attribute Name	Description
Name	The name of the installed app.
Type	The type of auto-install request last processed for this app. The values are: `App Create`, `App Update`, `App Delete`, and `Start Dataflow`. This value is only visible in the apps list view.
Status	The status of the most recent auto-install request for the installed app. The values include: `Success`, `Failed`, `In Progress`, `Not Started`, and `Canceled`
Reason	The failed reason of the installed app. For successful requests, the value is `N/A`. For failed requests, the failure reason is displayed.
Created By	The user who created the installed app.
Created On	The date and time the installed app was created.

Open App

   This link is only visible to users with access to Tableau CRM Studio and only in the apps tile view. To open the installed app in Tableau CRM Studio, click the **Open App** link on the app tile.

App Actions

Each app has an action menu accessible from the dropdown list icon [dropdown icon]. The app actions include:

- `Open` - opens the installed app details. These details include: all auto-install requests for the installed app; the most recent app logs for the installed app; and the details of the auto-install request described here.
- `Delete App` - deletes the installed app.
- `View Log` - displays the auto-install request log.
- `Try Again` - runs the last auto-install request for this installed app again.
- `Cancel` - displays only when the auto-install request status is `In Progress`. Use this action to cancel long running requests or when you do not want app creation/upgrade to complete.

Search Apps

Use the search bar to search installed apps by app name.

Refresh Page

To update the statuses of the installed apps, click the Refresh icon [refresh icon].

Auto-Install Requests

Monitor and interact with the auto-install requests in your org on the Requests tab.

The information for each installed app includes:

Attribute Name	Description
Name	The name of the auto-install request.
Type	The type of auto-install request. The values are: `App Create`, `App Update`, `App Delete`, and `Start Dataflow`.
Status	The status of the auto-install request. The values include: `Success`, `Failed`, `In Progress`, `Not Started`, and `Canceled`
Reason	The failed reason of the auto-install request. For successful requests, the value is `N/A`. For failed requests, the failure reason is displayed.
Created By	The user who created the auto-install request.
Created On	The date and time the auto-install request was created.

Attribute Name	Description
App	The name of the auto-installed app.

Request Actions

Each auto-install request has an action menu accessible from the dropdown list icon [▼]. The request actions include:

- `Open` - opens the associated installed app details. These details include: all auto-install requests for the installed app; the most recent app logs for the installed app; and the details of the auto-install request described here.
- `Delete App` - deletes the installed app. associated with the auto-install request.
- `View Log` - displays the auto-install request log.
- `Try Again` - runs the auto-install request for this installed app again.
- `Cancel` - displays only when the auto-install request status is `In Progress`. Use this action to cancel long running requests or when you do not want app creation/upgrade to complete.

Clean Stale Requests

To clean the list of auto-install requests of stale requests, click the **Clean Stale Requests**. A stale auto-install request is a request that doesn't have an installed app. Stale requests occur when app creation fails or when the installed app is deleted, either manually or by another auto-install request

Search Requests

Use the search bar to search auto-install requests by app name.

Filter Requests

Use the filter dropdown list [▼▾] to filter the auto-install request list by request status.

Refresh Page

To update the statuses of the auto-install requests, click the Refresh icon [↻].

View Auto Request Details

After opening the auto-install request details from the **Open** action, you see the **Details** view.
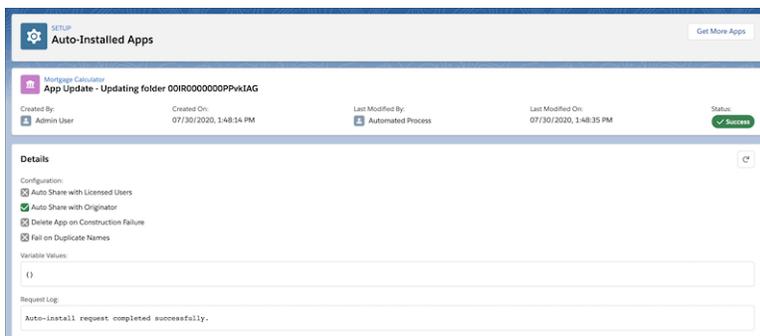
This view includes the auto-install request name, the request creation date and user, the user and date of the last modification, and the request status.

The auto-install request details describe the configuration settings and the variable values used for app creation and the request log. These details provide information necessary for troubleshooting requests and installed apps.

The Einstein Accuracy Analytics Template

The Einstein Accuracy Analytics app helps you monitor how well Einstein Discovery models predict actual outcomes over time.

> **Note:** Create the Accuracy Analytics app using the Einstein Discovery Model Manager. See Analyze Prediction Accuracy with the Einstein Accuracy Analytics App.

Adoption Analytics Template

Create an app from the Adoption Analytics template for ready-made insight into how your team uses Analytics Cloud apps, dashboards, lenses, and datasets.

The app you create from the Adoption Analytics template provides datasets and prebuilt dashboards showing how your team uses Tableau CRM assets—apps, dashboard, and lenses.

The app includes a dataset called *WaveChangeEA* with data from logs that track how often users open Tableau CRM apps, dashboards, and lenses. The dataset also includes mobile usage data. The dataset stores data for the amount of time you select in the configuration wizard when you create the app, from 1 to 365 days. App visualizations are based on the data from the dataset. You can also use it to create your own visualizations to meet your organization's unique needs.

> **Tip:** Follow the steps in the order shown to assure success creating an app from the Adoption Analytics template.

USER PERMISSIONS
To create and manage the Adoption Analytics app:
• Adoption Analytics Templates and Apps
• Manage Analytics Templated Apps
To use the Adoption Analytics app:
• Use Analytics Templated Apps

1. Get Your Org Ready to Create the Adoption Analytics App

   Complete a few special setup procedures before creating an app from the Adoption Analytics template.

2. Create and Upgrade the Adoption Analytics App

   To create an app from the Adoption Analytics template, start from the Tableau CRM Studio home page. To upgrade to the latest version of the app, start from the Adoption Analytics home page.

3. Schedule the Dataflow for the Adoption Analytics App

   Creating the Adoption Analytics app also creates a dataflow that imports the latest Tableau CRM asset log data to Tableau CRM. Schedule the Adoption Analytics dataflow to be rerun twice a day to assure that your app uses up-to-date data.

4. Use the Adoption Analytics App

   The Adoption Analytics app includes two dashboards that visualize how your team uses Tableau CRM apps, dashboards, and lenses. It also includes two lenses that show deleted assets and users previously assigned Tableau CRM permission set licenses (PSLs).

Get Your Org Ready to Create the Adoption Analytics App

Complete a few special setup procedures before creating an app from the Adoption Analytics template.

> **Note:** Make sure you and all app users have an Tableau CRM Plus license or the Tableau CRM Growth license.

Enable **View Event Log Data in Analytics Apps**

1. Go to Setup.

2. Enter *Event* in the **Quick Find** (search) box.

3. Click **Event Monitoring Settings**.

4. Set the toggle next to **View Event Log Data in Analytics Apps** to Enabled if it isn't already.

This signals to Salesforce to store data about the use of Analytics apps, dashboards, lenses, and datasets in event logs. If you have the Tableau CRM Plus license, your app includes Analytics log data. If you also have the Event Monitoring license, your app includes Event Monitoring bundle data.

Enable **Extract Adoption Analytics Metadata via Dataflow**

1. In Setup, enter `Analytics` in the **Quick Find** (search) box.

2. Click **Settings**.

3. Check the box next to **Extract Adoption Analytics Metadata via Dataflow** if it isn't checked already.

4. Click **Save**.

Adoption Analytics Limitations

These limitations apply to Adoption Analytics.

- Datasets created for the app count against row limits for your Tableau CRM license.
- Includes only Tableau CRM logs. For any Salesforce logs, use Event Monitoring.
- The first time you use Adoption Analytics, it may take 24 hours for data to collect in event logs before it can be registered in the app's datasets and dashboard.
- Salesforce stores event log data for 30 days.
- Tableau CRM stores event log data in a dataset for up to 365 days, which counts against your license's row limit.

Create and Upgrade the Adoption Analytics App

To create an app from the Adoption Analytics template, start from the Tableau CRM Studio home page. To upgrade to the latest version of the app, start from the Adoption Analytics home page.

Create an App from the Adoption Analytics Template

1. Navigate to Tableau CRM Studio.

2. Click the **Create** button in the upper right corner, the select **App**.

3. Click **Create App from Template**.

4. Locate locate the **Adoption Analytics** tile, select it, and click **Continue**.

5. Take a quick look at the preview page, and click **Continue** to open the configuration wizard.

6. If the wizard asks if you want to create a new app or use settings from an existing app, make a selection and click **Continue**.

7. The wizard asks you to choose the number of days of data you'd like Tableau CRM to store for viewing in the Adoption Analytics app. Choose a number from 1 to 365, depending on your policy about data retention. Click **Looks good, next**.

   📝 Note: To exclude any user's data from being retained, add that user's 18-character Salesforce ID to the `GDPR_Legal_Compute` node in the dataflow.

8. Give your app a name that's easy for you and users in your org to remember, and click **Create**.

Upgrade to the Latest Version of Adoption Analytics

1. Navigate to the home page for the app.

2. Click the **New version available** link near the top of the page, just below the app name.

3. Read the release note for information about the new version, then click **Looks good, upgrade**.

4. The wizard asks you to choose the number of days of Tableau CRM data you'd like to view in the Adoption Analytics app. Choose a number from 1 to 365, and click Looks good, next.

5. The next page warns you that upgrading the app overwrites any customizations you've made to the current version. It also gives you the options to create a new app or upgrade the current version.

    a. **Create new app**. Select this option to create a new app and retain the current version. Tableau CRM collects log data starting when you create it and updating it based on future asset use. This option is recommended if you've made extensive customizations to your current app and want to retain them.

    b. **Upgrade current app**. Select this option to overwrite the current version. Tableau CRM retains the historical log data in the WaveChangeEA dataset that you're currently using to track Tableau CRM adoption.

6. If you selected **Create new app**, give your app a name that's easy for you and users in your org to remember, and click **Create New**.

7. If you selected **Upgrade current app**, read the warning and check the box if you're sure you want to overwrite the current version. Then click **Upgrade app**.

Schedule the Dataflow for the Adoption Analytics App

Creating the Adoption Analytics app also creates a dataflow that imports the latest Tableau CRM asset log data to Tableau CRM. Schedule the Adoption Analytics dataflow to be rerun twice a day to assure that your app uses up-to-date data.

Set the dataflow to run at least an hour after the time the app's WaveChange dataset gets refreshed with the latest logs. This usually happens overnight. Here's how to check the time.

1. In Tableau CRM Studio, click **All Items** in the left-hand column.

2. Open the **Datasets** tab.

3. Scroll to the **WaveChange** dataset and look for the time in the **Data Refreshed** column.

Now, set the time to run the dataflow.

1. In Tableau CRM Studio, click the wheel icon [⚙] at upper right and select **Data Manager**.

2. Select **Dataflows**.

3. Look for the name of your app, and click the triangle to the far right.

4. Select **Schedule**, then make the following selections.

Setting	Selection
Schedule Mode	Time-based
Schedule by	Hour
Start at	A time at least one hour after the WaveChange dataset is refreshed. See the previous set of steps.
Run every	12 hours
Days of the week	Every day

Setting	Selection
Stop queuing at a specific time	Uncheck

5. Click **Save**.

Use the Adoption Analytics App

The Adoption Analytics app includes two dashboards that visualize how your team uses Tableau CRM apps, dashboards, and lenses. It also includes two lenses that show deleted assets and users previously assigned Tableau CRM permission set licenses (PSLs).

Dashboards

- **Analytics Adoption**. Shows how each member of your team uses Tableau CRM apps, dashboards, and lenses. Filter by individual team member, department, division, and other groupings. You can also filter by asset type. Also view the most popular assets, unused assets, and users who don't use any assets.

  Metric Calculations

  – **Trend chart.** Number of views by asset type by day.

  – **Percent calculations**. Number of users with Tableau CRM permission set licenses who interact with asset type at least once. Calculation for the number is (Selected period / Number of users with Tableau CRM permission set licenses) X 100.

  – **DAU**. Daily active users. Number of unique users / Number of days with data in the dataset.

  – **MAU**. Monthly active users. Number of unique users interacting with Tableau CRM assets at least once in the last 30 days.

- **Dataset Linkage**. Shows use of datasets by Tableau CRM assets, mapping apps, dashboards, and lenses to the datasets where they get data.

Lenses

- **Deleted Assets in Logs**. Lists assets that have been deleted from Tableau CRM Studio.
- **Users Previously Assigned EA PSL**. Shows users previously assigned a Tableau CRM PSL who no longer have the PSL.

Analytics for Retail Banking Template

Analytics for Retail Banking is part of a suite of Tableau CRM apps for Financial Services Cloud customers. Its dashboards visualize all the metrics and key performance indicators (KPIs) personal bankers require to grow client relationships.

> **Note:** Analytics for Retail Banking is included with Tableau CRM for Financial Services, which is available at an extra charge for customers with Financial Services Cloud Basic or Standard licenses and the FSCAnalyticsPlus (Tableau CRM for Financial Services) license. See Deploy Tableau CRM for Financial Services for complete deployment instructions.

Appointment Analytics Template

Appointment Analytics brings Lightning Scheduler appointments data into Tableau CRM to help you deliver the best service possible to your customers.

Lightning Scheduler is about making sure you schedule the right resources at the right times to meet your customers' service needs. Appointment Analytics gives you the tools to find out how well that's working. The app's overview dashboard focuses on your team's appointment volume, while its Service Resources Performance dashboard analyzes your reps' performance. Together, they help you determine if your service team is applying the appropriate resources to meet customer need.

> ⊘ **Important**: See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Appointment Analytics app.

Org Requirements

Your org requires the following before you can create an app from the Appointment Analytics template:

- Make sure that Lightning Scheduler is set up in your org. See Set Up Lightning Scheduler.
- The Service Appointment object must contain at least one record.

Use Appointment Analytics

Open the app.

1.
   From the app picker ⠿ , select **Tableau CRM Studio** to open the Tableau CRM home page.

2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

Click **Run App** to open the app to the Appointment Overview dashboard.
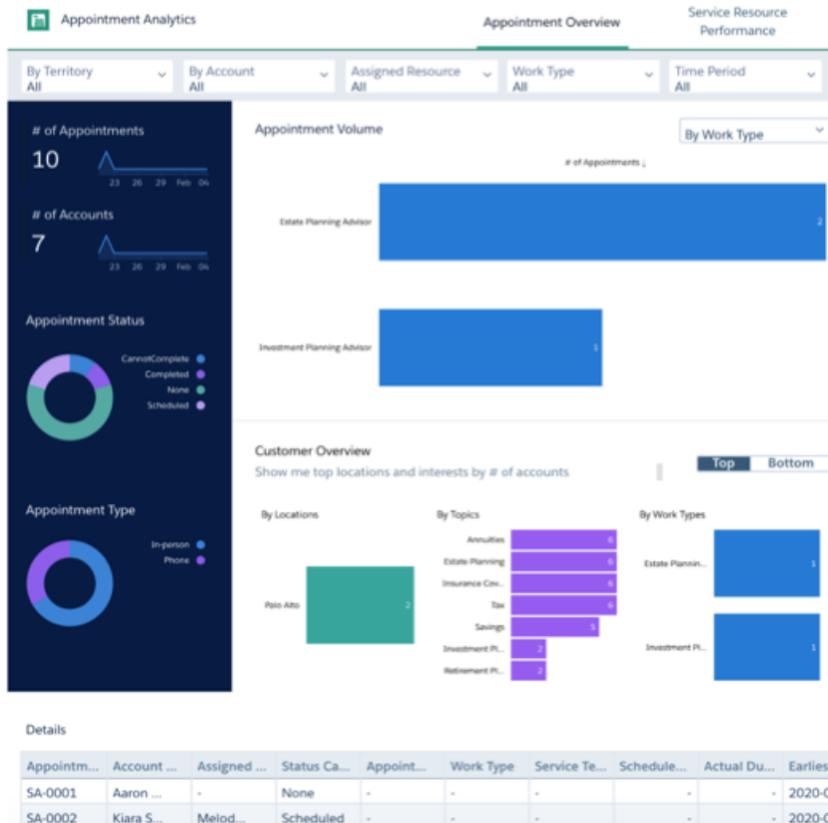
It links to the app's other dashboard, Service Resource Performance. Here's what they contain:

Appointment Overview Dashboard

Shows Lightning Scheduler appointment volume data. The left-hand column displays top-level metrics: total number of appointments and accounts and the status of appointments and appointment type. On the right, view volume by types of resource and get an overview of customer according to their locations, topics, and types. Scroll further down to see details about every appointment. Adjust the menus along the top to view your data by specific location, account, service resource, type, and date.

Service Resource Performance

Shows how your resources (agents) perform during service appointments so you can see if you're assigning resources appropriately. See the average duration of appointments for your agents. View duration for all agents or for one or more agents by selecting their names from the Resources filter at the top. Also view scheduled versus actual duration, numbers of appointment by type, and average duration by territories. See details about each appointment in the chart at the bottom.

Approval Analytics Template

Create an app from the Approval Analytics Template to increase your visibility into approval processes. Managers and team leaders can use the app to view approval history, understand trends, identify bottlenecks, and take action to streamline the process.

The Approval Analytics template creates an app that visualizes the history of your approval data. You can see the entire history of the approval process in a single dashboard. It includes the number of approvals and rejection, pending approvals, who's approved what, and how long approvals have taken. Use the app to understand trends, identify bottlenecks, and streamline approvals.

⊘ **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Approval Analytics app.

Org Requirements

Your org requires the following before you can create an app from the Approval Analytics template:

- Make sure you and all app users, including the Analytics Integration User, have the Tableau CRM Growth license.

- Your org needs to have defined an approval process and at least one object with a record submitted for approval.

- Set Salesforce field-level security to enable the Analytics Integration User to see all fields used in the app. See Set Field Level Security to Enable Creation of an Tableau CRM Template.

Use Approval Analytics

Open the app.

1. From the app picker ⦂⦂⦂ , select **Tableau CRM Studio** to open the Tableau CRM home page.

2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

Click **Run App** to open its dashboard. Use the chart at the top to get an overview of your approval process data. You can see the following:

- Number of approvals pending, approved, and rejected.

- Approvals in each stage of the process. Depends on how your org defines stages. Examples include, for example, approved, reassigned, and removed.

- Number of approvals created by month.

Charts in the middle of the dashboard show you key approval metrics, including approval time by approver, process name, and object. Use the metrics to understand who's taking longer to make approvals and which processes are running faster or slower. You can also quickly see the objects where approvals take the longest.

Scroll down to see a list of all approval records. Take action on a record by hovering over its ID and clicking the disclosure triangle to the right. That opens the Actions Menu from the record. From there, post to Chatter, create a task or an event, or perform another action to move the process along.

B2B Commerce Analytics Template

The B2B Commerce Analytics template gives you a fast way to apply the power of Tableau CRM to your B2B ecommerce data.
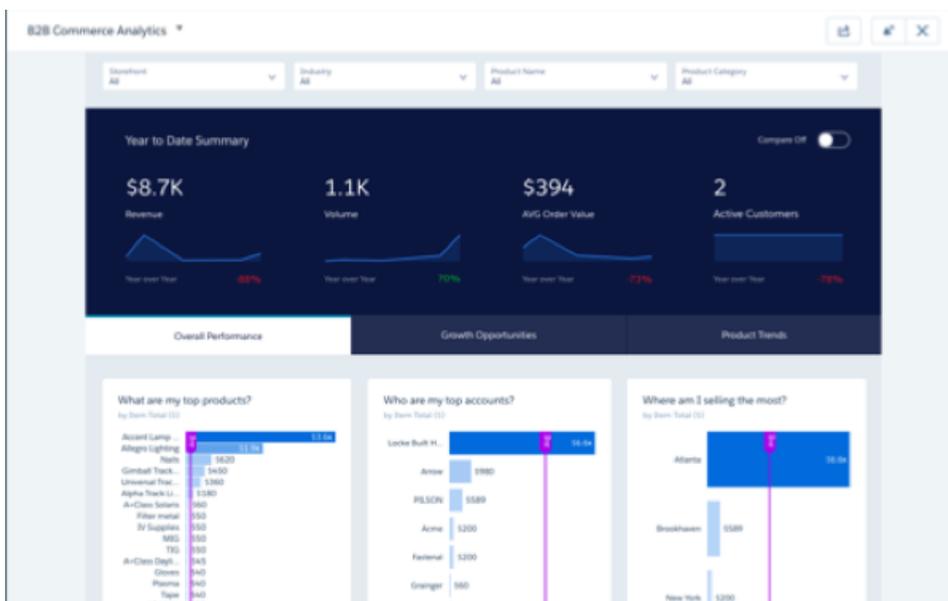
⊘ **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about the B2B Commerce Analytics app.

Salesforce B2B Commerce (formerly CloudCraze) customers: Create an app from the template to empower your team with instant insights into your ecommerce data. Track orders, see who's buying what products, learn year-to-date and year-over-year sales revenue, and more—all in a single dashboard.

Org Requirements

Your org requires the following before you can create an app from the B2B Commerce Analytics template:

- Make sure you and all app users have Tableau CRM Growth license.
- B2B Commerce Analytics is only for customers who have installed the Salesforce B2B Commerce (CloudCraze) managed package.

Use the B2B Commerce Analytics App

Open the app.

1.

From the app picker ⠿ , select **Tableau CRM Studio** to open the Tableau CRM home page.

2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

Click **Run App**. The charts at the top of the dashboard show year-to-date summaries for revenue, volume, average order value, and active customers. In the upper right corner **Switch Compare Off** to **Compare On** to change chart layout from left-to-right to top-to-bottom to more easily compare month-by-month performance changes. Filter data in the dashboard by using the **Storefront**, **Industry**, **Product Name**, and **Product Category** menus along the top.

Charts along the bottom provide details.

- **Overall Performance** shows top products, accounts, and geographies.
- **Growth Opportunities** helps you uncover accounts with resell and upsell potential by showing whitespace. Choose the criteria for the whitespace evaluation using the menus along the left. Default shows revenue for accounts (X [top] axis) purchasing products (Y [left] axis).
- **Product Trends** shows fast- and slow-growing areas of your business. Choose the criteria using the menus along the top.

Selected Metric Calculations

Year to Date Summary Chart

All metrics based on shipped orders.

- **Revenue**. Total of order line item amount.
- **Volume**. Total of order line item quantity.
- **AVG Order Value**. Total of order line item amount / Number of orders.
- **Active Customers**. Unique number of customers year-to-date who placed at least one order.
- **Year over Year**. Percent increase or decrease compared to previous year.

Overall Performance Chart

- **What are my top products?** Order item/product = Amount (price * quantity).
- **Who are my top accounts?** Account tied to order; Order item = Amount (price * quantity).
- **Where am I selling the most?** Ship to City on Account (tied to order) object; Order item = Amount (price * quantity).

B2B Marketing Analytics App

B2B Marketing Analytics provides advanced analytics for the data-driven marketer, letting you explore both your marketing and sales data in one place. With powerful dashboards that consolidate Pardot and Sales Cloud data, you can quickly explore data, understand the impact of marketing on revenue, and take instant action to drive marketing results.

📝 Note: Help for the B2B Marketing Analytics app currently resides within the Salesforce Pardot Knowledge Base. Go to the B2B Marketing Analytics overview page to get started.

Event Monitoring Analytics App

The Event Monitoring Analytics App integrates with event monitoring and setup audit trail data to give you insights into your user and org behavior. The app is a built-in way to explore your monitoring data in Salesforce. App creation is easy and with its prebuilt dashboards and datasets, you can start exploring right away. This app helps you drill into your org's data and swiftly identify suspicious behavior, slow page performance, and poor user adoption.

> 📝 **Note:** As part of Event Monitoring, you also get the Event Monitoring Analytics App. Use this app to upload and access only the data provided to you as part of your subscription. Please prevent your users from using the app to upload or access any other data. Salesforce sometimes monitors such usage. The Event Monitoring Analytics App is available in English only. If you have a Tableau CRM Platform license, you can access dashboards in the Event Monitoring Analytics app. Hourly event log file integration with the Event Monitoring Analytics app is unavailable. Data is refreshed once a day in the app.

Event Monitoring log files aren't a system of record for user activity. They are a source of truth, but aren't durable. During Salesforce site switches, instance refreshes, or unplanned system outages, data loss can occur.

For example, if Salesforce moves your production org instance, your event log files might have a gap in data. Salesforce makes commercially reasonable efforts to preserve event log file data integrity and avoid data loss. When Salesforce performs a site switch or instance refresh, it uses an automated process to replicate event logs.

1. Set Up Salesforce Permissions for the Event Monitoring Analytics App

   Set up your org to use the Event Monitoring Analytics App by enabling Tableau CRM, assigning permission set licenses, and then creating and assigning permission sets.

2. Create and Share the Event Monitoring Analytics App

   After you've enabled the correct permissions, the setup for the Event Monitoring Analytics App is simple.

3. Upgrade the Event Monitoring Analytics App

   Take advantage of the latest Event Monitoring Analytics features by upgrading your app every time we release a new version.

4. Schedule the Event Monitoring Analytics Daily Dataflow

   Schedule a daily dataflow before you start exploring with Event Monitoring Analytics.

5. Event Monitoring Analytics App Prebuilt Dashboards

   The Event Monitoring Analytics App comes with a set of prebuilt dashboards so that you can start exploring your data right away. Data from your org is automatically loaded into your Event Monitoring Analytics App datasets when it's generated.

6. Get to Know Event Monitoring Analytics App Terminology

   To make the best use of the Event Monitoring Analytics App, it's helpful to understand the metrics and terms used in the app.

Set Up Salesforce Permissions for the Event Monitoring Analytics App

Set up your org to use the Event Monitoring Analytics App by enabling Tableau CRM, assigning permission set licenses, and then creating and assigning permission sets.

> ✏️ **Note:** As part of Event Monitoring, you also get the Event Monitoring Analytics App. Use this app to upload and access only the data provided to you as part of your subscription. Please prevent your users from using the app to upload or access any other data. Salesforce sometimes monitors such usage. The Event Monitoring Analytics App is available in English only. If you have an Tableau CRM Growth license, you can access dashboards in the Event Monitoring app.

Each Event Monitoring Analytics Apps license provides access to Event Monitoring Analytics to up to 10 users. The table shows data storage limits for the app. If you require more data, you can purchase Analytics Cloud - Additional Data Rows, which entitles you to an additional 100 million rows.

Table 6: Event Monitoring Analytics App Data Storage Limits

License	Limit
Event Monitoring Analytics Apps	50 million rows when used without Tableau CRM Growth license. Provides access to up to 10 users.
Analytics Cloud - Additional Data Rows	100 million rows.

> ⛔ **Important:** Event Monitoring Analytics App license data storage limits are contractual, not technical. Licensee agrees to strictly monitor the total number of data rows used by the Tableau CRM app to ensure successful dataflow updates. If your org has only one Tableau CRM app, and you have more than the licensed number of rows across all datasets, dataflows will fail. If you have multiple Tableau CRM apps (such as Event Monitoring Analytics and Service Analytics), data storage limits are pooled across the Tableau CRM apps and any app can consume storage without error.

The sample data for a particular event remains in the app until it's overwritten with the data from an actual event. Event log file data is available for the past 30 days. If you have a Developer Edition org, it's available for the last 24 hours.

Your org can use the Event Monitoring Analytics App with or without the Tableau CRM platform. The Event Monitoring Analytics app is available for free to any org with the paid Event Monitoring license. The Event Monitoring Analytics Apps license enables the following permissions.

Table 7: Event Monitoring Analytics App User Permissions

Permission	Function
Use Analytics Templated Apps	Enables access to all Tableau CRM templated apps, including the Event Monitoring Analytics App. ⚠️ **Warning:** This permission allows users to access event monitoring data without requiring the View Event Log Files and API Enabled user permissions. It also allows users to edit app assets if they have Editor or Manager rights to the app. Use caution when assigning it.
Access Event Monitoring Analytics Templates and Apps	Gives admins the ability to create Event Monitoring Analytics apps.

Permission	Function
Manage Analytics Templated Apps	Gives admins the ability to create templated Tableau CRM apps for users in their orgs.
Edit Analytics Dataflows	Enables admins to upload and download JSON for existing data flows.

Event Monitoring App Setup Process

For best results, follow the steps for setting up the Event Monitoring Analytics app in the order shown.

1. Event Monitoring Analytics App Permission Set License and User Permissions
   The Event Monitoring Analytics Apps permission set license enables the user permissions required to explore data with and manage the Event Monitoring Analytics app.
2. Identify Event Monitoring Analytics App User Types
   Identify Event Monitoring Analytics app user types and the tasks they perform to ensure that the setup process meets your team's event monitoring analytics needs.
3. Enable Analytics Cloud and Event Monitoring Integration with Tableau CRM
   After Salesforce provisions you with the Event Monitoring Analytics Apps license, enable Tableau CRM and Event Monitoring.
4. Select User Permissions and Assign Users to Event Monitoring Analytics App Permission Sets
   Give your users access to Event Monitoring Analytics features by assigning one or more permission sets based on the capabilities they need.

Event Monitoring Analytics App Permission Set License and User Permissions

The Event Monitoring Analytics Apps permission set license enables the user permissions required to explore data with and manage the Event Monitoring Analytics app.

Each Event Monitoring app user needs an Event Monitoring Analytics Apps permission set license. The Event Monitoring Analytics Apps permission set license enables the following permissions.

Table 8: Event Monitoring Analytics Apps Permissions

User Permission	What It Enables
Use Analytics Templated Apps	General access to all Tableau CRM templated apps, including the Event Monitoring Analytics app.
Access Event Monitoring Analytics Templates and Apps	The ability to create Event Monitoring Analytics apps.
Manage Analytics Templated Apps	The ability to create templated Tableau CRM apps for users in their orgs.
Edit Analytics Dataflows	Edit, delete, and use remote connections; add and remove connected objects; run and schedule data sync; create, edit, delete, run, schedule, and monitordataflows and recipes. Use discretion when assigning this user permission because it enables access to all Salesforce object data to which the Integration User has access. See Salesforce Data Access in Tableau CRM.

Identify Event Monitoring Analytics App User Types

Identify Event Monitoring Analytics app user types and the tasks they perform to ensure that the setup process meets your team's event monitoring analytics needs.

The Event Monitoring Analytics Apps license includes the permissions required by users in your organization to administer Event Monitoring Analytics and explore event data. Set up Event Monitoring Analytics by considering the different ways people can access Event Monitoring Analytics features and categorizing users into types. Here are a few examples.

In most cases, Event Monitoring Analytics addresses the needs of two basic types of people on your team:

- Users, who view Event Monitoring Analytics dashboards and datasets.
- Administrators or managers, who access all Event Monitoring Analytics functionality to create apps, dashboards, and datasets, and otherwise customize the Event Monitoring app experience.

Accordingly, when you set up Event Monitoring Analytics, creating two basic permission sets probably meets the needs of most users in your organization:

- A "view" permission set that contains the permissions needed by most basic-level users when they access Event Monitoring Analytics.
- A "manage" permission set that's reserved for a select few administrators and managers and contains permissions to use all Event Monitoring Analytics features.

In the detailed Event Monitoring Analytics app setup instructions that follow, we show you how to create and assign these two permission sets. Or, you can use the standard permission sets we provide.

Enable Analytics Cloud and Event Monitoring Integration with Tableau CRM

After Salesforce provisions you with the Event Monitoring Analytics Apps license, enable Tableau CRM and Event Monitoring.

Enable the Tableau CRM Platform

1. In the Salesforce Setup menu, under Administer, select **Analytics Cloud | Getting Started**.
2. Click **Enable Analytics**.

Enable Event Monitoring

1. In Setup, enter `Event` in the Quick Find box, then select **Event Monitoring Settings**.
2. Select **View Event Log Data in Tableau CRM Apps**.

Select User Permissions and Assign Users to Event Monitoring Analytics App Permission Sets

Give your users access to Event Monitoring Analytics features by assigning one or more permission sets based on the capabilities they need.

After enabling Analytics Cloud and Event Monitoring Integration with Tableau CRM, identify users who will access the Event Monitoring Analytics app and assign them to a permission set. You can create any permission set you need to meet the needs of your users. For convenience, two standard permission sets are provided.

USER PERMISSIONS

To create permission sets:
- Manage Profiles and Permission Sets

Standard Permission Sets	For
Event Monitoring Analytics Apps Admin	Users who create and customize Event Monitoring Analytics apps and data flows
Event Monitoring Analytics Apps User	Users who view Event Monitoring Analytics apps

Let's begin selecting the user permissions in the Event Monitoring Analytics Apps Admin standard permission set.

1. From Setup, enter `Permission Sets` in the Quick Find box, then select **Permission Sets**.

2. Select the Event Monitoring Analytics Apps Admin permission set.

3. Select the feature permissions to enable for your permission set. Use `Find Settings` to search for them quickly.Select **Use Analytics Templated Apps**, **Access Event Monitoring Analytics Templates and Apps**, **Manage Analytics Templated Apps**, and **Edit Analytics Dataflows**. Make sure to select all four permissions.

4. Click **Save**. You can now assign the permission set to users. See Assign Permission Sets to a Single User.

5. Repeat the steps for the Event Monitoring Analytics Apps User standard permission set. However, for this permission set, select only the **Use Analytics Templated Apps** user permission, because your end users need only view access. If they need more access, you can create your own permission set by cloning and customizing the standard permission set.

> ⚠ Warning: Assign the Event Monitoring Analytics Apps Admin permission set carefully, because it lets users create, edit, and delete the app. Assign it only to users who administer or manage the app.

SEE ALSO:

   Assign Permission Sets to a Single User

   Create Permission Sets

Create and Share the Event Monitoring Analytics App

After you've enabled the correct permissions, the setup for the Event Monitoring Analytics App is simple.

If you haven't enabled permissions and event log file integration, complete the steps in Enable Analytics Cloud and Event Monitoring Integration with Tableau CRM on page 1412.

> 📝 Note: Because event log files have a 24-hour generation period, you won't see your data in the app for the first day after you create it. The app includes some sample data so you can preview your dashboards in the meantime. The sample data for a particular event remains in the app until it's overwritten with the data from an actual event. Event log file data is available for the past 30 days. If you have a Developer Edition org, it's available for the last 24 hours.

1. Log in to Salesforce.

2. Select Analytics Cloud from the App Picker.

3. Click **App**, select **Event Monitoring Analytics App**, and then click **Continue.**

4. Give your app a name.

5. Click **Create**. The app creation process can take a while. You can check the progress by selecting **Data Manager** from the gear menu at the top of the page. If any part of your app creation fails, you must recreate the app.

6. After your app is ready to go, you can share it with other users in your org. Remember that only users with the Manage Analytics Templated Apps and Access Event Monitoring Analytics Templates and Apps permissions can see the app. To share your app, hover your mouse over the app tile on the Analytics Cloud home screen and click **Share**. You can also share from the app itself.

USER PERMISSIONS

To use the Event Monitoring Analytics App:
- Use Analytics Templated Apps

To create and manage Event Monitoring Analytics App:
- Access Event Monitoring Analytics Templates and Apps
- Manage Analytics Templated Apps
- Edit Analytics Dataflows

Upgrade the Event Monitoring Analytics App

Take advantage of the latest Event Monitoring Analytics features by upgrading your app every time we release a new version.

The banner at the top of your app home page tells you that we've released a new version. It also provides a link to more information about the release. There's also a link inviting you to start the upgrade process in the left column of the home page, just below the app name.

Here's how to upgrade your app.

1. Click either the **What's new** link in the banner or the **New version available** link in the left column of your app's home page.

2. You can read information about the new version on top of the page that opens. At the bottom of the page, find the buttons: **Upgrade current app** and **Create new app**.

3. See descriptions of what the buttons do by hovering over them. If you click **Upgrade current app**, you overwrite your current app and all its assets, replacing it with an app based on the new version. Upgrading also deletes any customizations, including any new fields or object that you've added to the dataflow or changes you've made to security settings and dashboard labels and colors. If you click **Create new app**, you create a second copy of your current app based on the new version. Decide which option suits your circumstances, and click the appropriate button.

4. ⚠ Warning: If you've customized your app, click **Create new app** instead of **Upgrade current app** to make an app copy based on the new version. Creating a new app preserves the current version and any customizations, which you can then manually copy into the new version of the app.

   If you click **Upgrade current app**, you see a screen warning you that the upgrade option overwrites the current app and gets rid of any customizations you've made. If you're OK with overwriting customizations, check the box and click **Continue**. If you're not, click **Back** to return to the previous screen. Clicking **Continue** takes you to the configuration wizard. Skip ahead to step 6.

5. If you click **Create new app**, you're taken to the configuration wizard. You don't see a warning, since you're not overwriting your current app, which is preserved with any customizations you've made to it.

6. The configuration wizard is preloaded with the settings you chose last time you used the wizard. You can either keep those settings or change them. Go through each page of the wizard.

7. Once you complete the wizard, Tableau CRM shows a screen that indicates which assets are impacted by the upgrade. Review the screen to see how many datasets, dashboards, and lenses will be changed, deleted, or added by the upgrade. It also shows how changed assets will be modified—whether the change is to data or appearance. On the same screen, you have the option to download a file with code for changes made to the app. If you've customized the app, download and save the file so you can copy and paste customization code into the upgraded version.

8. Upgrade option only: If you're upgrading your app, click **Upgrade current app**. Remember, this overwrites any customizations you've made. You can also click **Back** to go back into the configuration wizard and change your selections or click the **X** in the upper-right corner to cancel.

9. Create option only: If you're creating a new app, name your app something different from the current version and click **Create new app**. This option saves your current app and all its customizations. You can also click **Back** to go back into the configuration wizard and change your selections or click the **X** in the upper-right corner to cancel.

If the link below the app name says "Reset app," you're using the latest version and don't need to upgrade.

Schedule the Event Monitoring Analytics Daily Dataflow

Schedule a daily dataflow before you start exploring with Event Monitoring Analytics.

When you create Event Monitoring Analytics, the creation process includes a dataflow that imports the latest Event Monitoring data to Tableau CRM. You can schedule the dataflow to be rerun every day to assure that your app uses up-to-date Salesforce data. Salesforce generates and uploads event log files in the early morning, usually at 3:00 AM in the time zone where your Salesforce instance (like NA7) is located. Schedule the dataflow for at least a couple of hours after that so the app has the latest data and the dataflow doesn't interrupt your use of the app. We recommend running the dataflow between 8:00 and 9:00 AM to make sure that you start your day with refreshed dashboards.

> **Note:** The Event Monitoring Analytics dataflow runs only once when you create the app. Schedule it to run daily so the app uses the latest event data.

1. Go to Tableau CRM in the Lightning Platform menu (top right of the Salesforce window) if you're not already there. Open the Data Manager by clicking the gear icon at the upper right of the screen.

2. Select Dataflow view from the menu at the top left of the Data Manager screen.
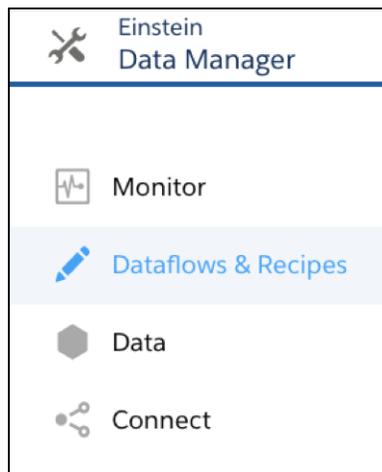
3. Find the app you created; you may have to scroll down the page. Open the menu on the far right of the screen next to the app icon and name. Select **Schedule**, and set a time for the dataflow. Select a time outside normal work hours so the dataflow doesn't interrupt business activities. Click **Save**.

Event Monitoring Analytics App Prebuilt Dashboards

The Event Monitoring Analytics App comes with a set of prebuilt dashboards so that you can start exploring your data right away. Data from your org is automatically loaded into your Event Monitoring Analytics App datasets when it's generated.

Most of the dashboards represent a single event type on the EventLogFile object. The EventLogFile object reference contains detailed information about each event type. Let's get a quick idea about what each dashboard does.

> **Note:** Each dashboard listed has a corresponding dataset. The Event Monitoring Analytics App also supports the following datasets that don't have dashboards: Bulk API, Apex Callout, Concurrent Long-Running Apex Limit, Apex REST API, Apex SOAP, Apex Trigger, Apex Unexpected Exception, and Console.

Analytics Adoption

Corresponds to the Wave Change, Wave Interaction, and Wave Performance event types. This dashboard shows Tableau CRM usage and performance information.

Apex Executions

Corresponds to the Apex Execution event type. This dashboard lets you track trends in Apex code executions and performance.

API

Corresponds to the API Event event type. This dashboard gives you information about both your users' API usage and API performance in your org. You can see how often each object is being used, how fast each object is being processed, and what methods are being invoked on that object.

Dashboards

Corresponds to the Dashboard event type. This dashboard helps you track dashboard adoption and performance.

Files

Corresponds to the Content Transfer event type. When users in your org perform content transfers (downloads, uploads, or previews), they show up on this dashboard. You can also track file adoption.

Lightning Adoption

Corresponds to the Lightning Interaction and Lightning Page View event types. Use this dashboard to see how users interact with Lightning Experience on the desktop and mobile devices.

Lightning Performance

Corresponds to the Lightning Error, Lightning Interaction, Lightning Page View, and Lightning Performance event types. Use this dashboard to optimize performance and user interactions with Lightning Experience and the Salesforce mobile app.

Login-As

Corresponds to the Login As event type. This dashboard lets you see which admins are using the login-as feature and on which user accounts.

My Trust

The My Trust dashboard gives you an overall idea of what kind of events are taking place in your org over time. It also shows the average speed of these transactions. The dashboard corresponds to the following event types: Apex Execution, API, Content Transfer, Dashboard, Lightning Page View, Login As, Login, Report, Report Export, REST API, and Visualforce, all correlated by User IDs. For the My Trust dashboard to work, add all datasets to your app in the Configuration Wizard. This could impact your row utilization, depending on the number of events in your org.

Page Views (URIs)

Corresponds to the URI event type. This dashboard lets you see which pages users are accessing in the Salesforce desktop app.

Report Downloads

Corresponds to the Report Export event type. This dashboard lets you see which users are downloading your reports and where they're downloading them from.

Reports

Corresponds to the Report event type. This dashboard shows you trends in reporting as well as which users are running specific reports. You can also find out which reports are having performance issues.

RestAPI

Corresponds to REST API event type. This dashboard shows you trends in REST API usage and which endpoints are seeing the most traffic. You can also view information about the IP ranges issuing the requests and which methods are being called.

Setup Audit Trail

Corresponds to the Setup Audit Trail page in Setup. Use this dashboard to see the changes your users are making in the Setup area.

User Logins

Corresponds to the Login event type. This dashboard shows login trends by user and information about where and how users are accessing your org.

Visualforce Requests

Corresponds to the Visualforce Request event type. Here you can see trends in Visualforce adoption and page performance.

Get to Know Event Monitoring Analytics App Terminology

To make the best use of the Event Monitoring Analytics App, it's helpful to understand the metrics and terms used in the app.

Term	Description
APT	Average page time. An aggregate metric across multiple transactions.
EPT	Experienced page time. The timing of a page transaction as seen by the user.
Event	A user-initiated interaction in Salesforce. For example, a login, the firing of an API call, or the download of a report. Events are related to the EventLogFile sObject.

Campaign Analytics Template

The Campaign Analytics template uses the power of Tableau CRM to show how your marketing campaigns impact the bottom line.

Use the Campaign Analytics template to create an app that shows how campaigns translate to revenue. Its Campaign Influence dashboard uses Salesforce data to paint a picture of how your campaigns help to drive business.

You get actionable insight into the return-on-investment of your marketing spend. You can compare performance over time, drill into your regional top performers, and highlight successful campaigns. Also review channels to see which ones are most effective. And use insights from the app to be reallocate marketing dollars to where they can have the most impact.

> 🛑 **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Campaign Analytics app.

USER PERMISSIONS

To create and manage the Campaign Analytics app:
- Manage Analytics Templated Apps

To use the Campaign Analytics app:
- Use Analytics Templated Apps

Org Requirements

Your org requires the following before you can create an app from the Campaign Analytics template:

- Make sure you and all app users have the Tableau CRM Growth license.
- Create at least 1 row of data in the customizable Campaign Influence object.
- Set Salesforce field-level security to enable the Analytics Integration User to see the fields you want to analyze. During app creation, Tableau CRM checks your org's field-level security and lets you know if you have to edit it.

Use Campaign Analytics

Open the app.

1.
From the app picker ⠿ , select **Tableau CRM Studio** to open the Tableau CRM home page.

2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

The app opens to its Campaign Influence dashboard, which shows a wealth of data visualizations about your campaigns. Use it to do the following:

- Slice and dice campaign metrics such as revenue share and average influence by campaign type, parent campaign, region or any segmentation you might have for campaigns and their related accounts.

- Drill down into a particular campaign type to view its revenue share over time.

- See campaign impact on various regions.

- Take action on campaigns with insufficient return-on-investment.

- Control campaign costs.

Change Analytics Template

Use Change Analytics to visualize field history data in any Salesforce object and learn how, when, and where members of your team change data.

Create an app from the Change Analytics template to bring your historical data to life. Put it to work as is on data from any Salesforce object that uses field history to gain valuable insights about how your team uses an object. Or customize it to meet your exacting specifications.

Its prebuilt dashboard shows which fields change the most, the most common ways fields change, duration between changes, and other key change metrics for the selected object. Customize the dashboard to behave the way you want it to or to add KPIs. Or open the template's dataset to create your own explorations—for example visualizing change trends using a Sankey diagram—and save them into your own dashboards.

> ⓘ **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Change Analytics app.

Org Requirements

Your org requires the following before you can create an app from the Change Analytics template:

- Make sure you and all app users have the Tableau CRM Growth license.
- The object you analyze must have fields with history tracking enabled and valid history records.
- Set Salesforce field-level security to enable the Analytics Integration User to see the fields you want to analyze.

The Change Analytics Configuration Wizard

Answer the three questions in the Change Analytics wizard when you create the app.

1. Question 1: Select an object. Only objects with history tracking enabled appear here.
2. Question 2: Select up to eight fields to track. Only fields from the object selected in Question 1 with valid history records appear here. You can select up to eight fields. Hold down the Shift or Control keys to select multiple fields, depending on your operating system.
3. Question 3: Select up to eight fields. Includes all fields from the object selected in Question 1. The app doesn't track history for these fields, but they do appear in the app's datasets and dashboard filters.

Use Change Analytics

The Change Analytics app includes one prebuilt dashboard called *Change Analytics*. It includes visualizations showing how your team changes the selected object and fields. It also includes two datasets named after the selected object, like so: *{Object_name}* and *{Object_name}History*.

Consumer Banking Starter Analytics Template

Consumer Banking Starter Analytics is part of a suite of Tableau CRM apps for Financial Services Cloud customers. It gives you a quick way of creating an analytics solution personal bankers can use to move clients up the value chain.

> 📝 **Note:** Consumer Banking Starter Analytics is included with Tableau CRM for Financial Services, which is available at an extra charge for customers with Financial Services Cloud Basic or Standard licenses and the FSCAnalyticsPlus (Tableau CRM for Financial Services) license. See Deploy Tableau CRM for Financial Services for complete deployment instructions.

Design Style Guide Template

Use the Design Style Guide template to create an app with ready-made dashboard, charts, and widget templates that you can use to get started fast with your own apps.

The Design Style Guide template gives you a starting point for your own Tableau CRM app design and development. The app's dashboard, chart, and widget templates follow the internal Salesforce design guidelines. Brand the app with your own color, then apply the designs to your own data to build your own app.

> ⛔ **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Design Style Guide app.

Use the Design Style Guide Configuration Wizard

The Design Style Guide configuration wizard requires that you select a color for your app. Click the disclosure triangle next to the color square on the left of the wizard page and select a color. You can also manually enter values for Hex, R, G, and B. Then click **Done**.

> 💡 **Tip:** Select a color that matches your color scheme you use for your company or your brand ID.

Use Design Style Guide

Open the app.

1. From the app picker ⣿ , select **Tableau CRM Studio** to open the Tableau CRM home page.

2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

Click **Run App** to open its dashboard, which includes three tabs:

- **Styles**. Shows you the styles used in the app, including colors, typography, and widgets.

- **Widgets**. Includes a wide variety of dashboard widgets that follow the Tableau CRM app design guidelines. Base your own widgets on the examples here.

- **Template**. An example Tableau CRM dashboard you can use as the basis for your own dashboards.
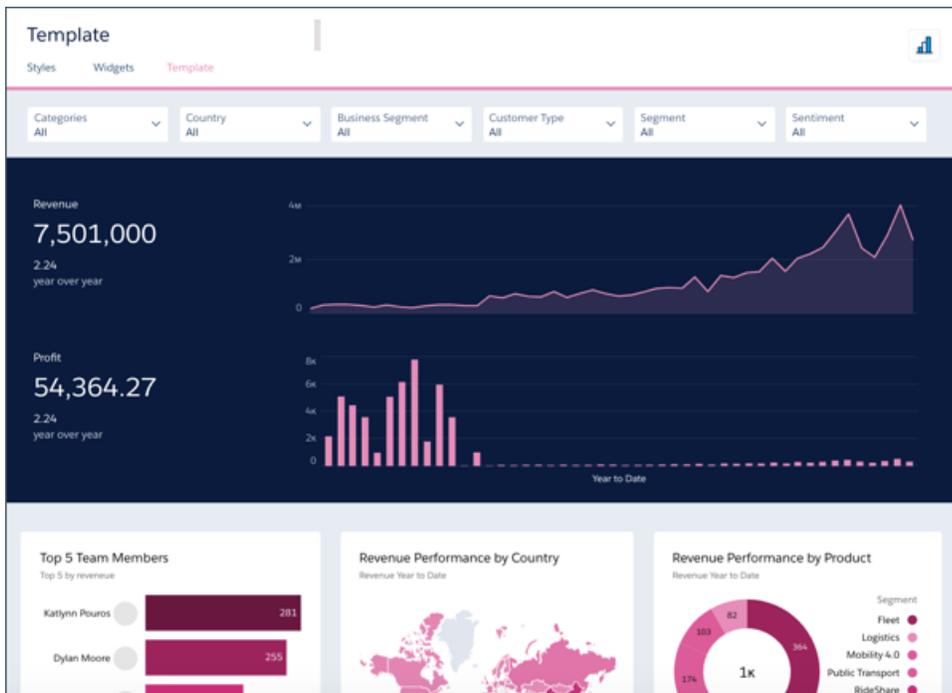
Design Style Guide includes two datasets based on CSV files to populate the examples in the app. Add your own data to the app, then reuse the examples to build your own app. See Build Tableau CRM Dashboards on page 1198.

Create Your Own Beautifully Designed Apps

Here are some tips for putting Design Style Guide to use in your own app designs:

1. When you create the app, select the color pattern that most closely matches your brand or corporate identity.

2. In the app's dashboard, select the container or widget of your choice and move it to a new page in the app you're building. From there, you can add metrics and other Tableau CRM Studio features using dashboard designer.

3. Create a new layout by saving a new copy of the dashboard in your own app and moving widgets around. The template created them with the color pattern of your choice, so they'll already reflect your brand.

Tableau CRM for Financial Services

Tableau CRM for Financial Services gives financial advisors and personal bankers a comprehensive customer intelligence solution based on the Tableau CRM platform. It includes Einstein Discovery, which lets you create automated artificial intelligence models on any dataset without writing code.

> 📝 Note: Tableau CRM for Financial Services is available at an extra charge for customers with Financial Services Cloud Basic or Standard licenses and the FSCAnalyticsPlus (Tableau CRM for Financial Services) license. See *Deploy Tableau CRM for Financial Services* in the Financial Services Cloud Administrator Guide for complete deployment instructions.

Field Service Analytics

The Field Service Analytics app brings the power of Analytics to Field Service Lightning on any device that supports Tableau CRM.

With intuitive visualizations based on your field service data, Field Service Analytics provides just-in-time analytics for field service managers and dispatchers. You get clear data insights to improve every on-site experience. Field Service Analytics empowers managers with answers to help their technicians deliver personalized, actionable customer service.

> 💡 **Tip:** Follow the steps in the order shown to get started with Field Service Analytics. If you haven't used Tableau CRM before, learn more about it from the *Analytics Documentation*.

1. About the Field Service Analytics App

   Learn the benefits of Field Service Analytics before you create and use the app to help manage field service operations.

2. Field Service Analytics Prebuilt Dashboards

   The Fields Service Analytics app includes prebuilt dashboards to accelerate Field Service data exploration.

3. Set Up and Create the Field Service Analytics App

   Set up Field Service Analytics app permissions, make sure your org is enabled to use Field Service, and assign field level security correctly so you can create and share the app.

About the Field Service Analytics App

Learn the benefits of Field Service Analytics before you create and use the app to help manage field service operations.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To use Tableau CRM templated apps:
- Use Analytics Templated Apps

To use Field Service Analytics:
- Access Service Cloud Analytics Templates and Apps

To create and manage Tableau CRM apps:
- Manage Analytics Templated Apps
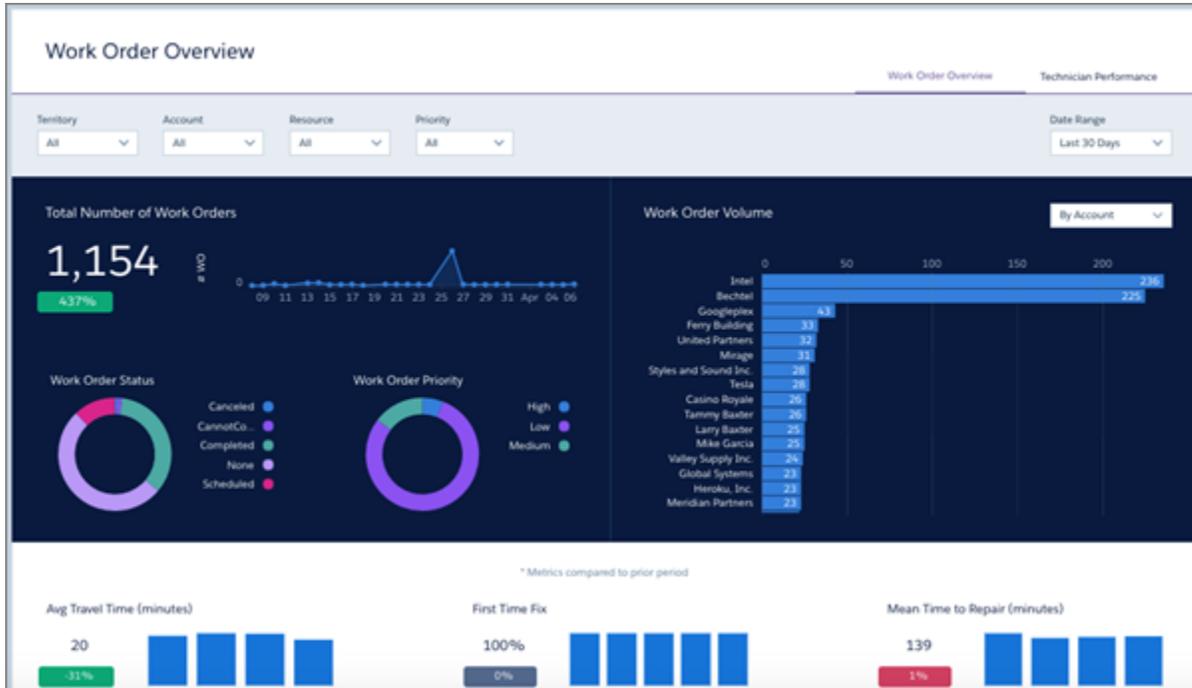
Salesforce created the Field Service Analytics app to help service managers analyze customer service data their teams generate when they use Field Service. The app is bundled with the Service Analytics.

🛑 **Important:** Your org must be enabled to use Field Service. Tableau CRM datasets are based on Field Service's new object model.

We've designed Field Service Analytics to give you just-in-time analytics to optimize the outcome of your field service business. Service managers get a new level of insight into their field service data that they can use to maximize the efficiency of their technical service agents.

Using the app, service managers can access insights to improve every on-site service visit. You gain visibility into mobile workforce utilization, first-time-fix rate, and travel time. Problem detection lets you optimize field technician agent productivity. Performance analysis visualizations help your team prioritize and resolve work orders faster to improve customer satisfaction. Armed with actionable analytics, your agents can deliver personalized, effective customer service.

Tableau CRM automatically creates the app for you based on your Field Service data. Select the Field Service Analytics template, click Create, and Tableau CRM does the rest. You get actionable insights fast from your field service data using the intuitive Tableau CRM interface. And you can drill deeper into key aspects of service engagements and technician performance by customizing Field Service Analytics around your business needs.

📝 **Note:** Your organization can use Field Service Analytics with or without the Tableau CRM platform by purchasing a Service Analytics license. Each Salesforce Analytics license includes a Service Analytics license.

Field Service Analytics Prebuilt Dashboards

The Fields Service Analytics app includes prebuilt dashboards to accelerate Field Service data exploration.
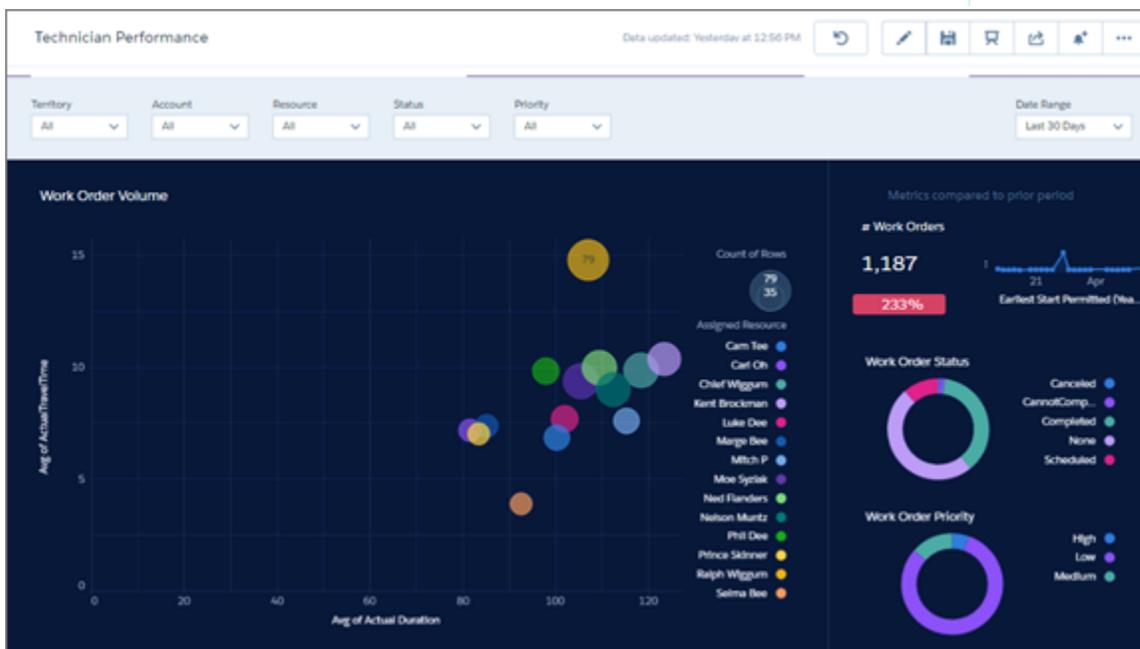
The prebuilt dashboards in the Fields Service Analytics app contain key performance indicators that help you get value from your Salesforce field service data—fast. The dashboards help you understand key elements of customer service work order status and field team performance. You can prioritize customer service engagements, make technician utilization decisions, and quickly drill down to isolate problems and take action. Quickly review first-time fix rates, utilization, SLA compliance, and travel-time analysis. Start from a territory overview of all accounts, products, and teams and zoom into the technician and work order level.

The following table guides you through Field Service Analytics dashboards. You can also explore further on your own at any point. To learn more about exploring data in Tableau CRM, see Explore and Visualize Your Data.

Table 9: Field Service Analytics Dashboards

Dashboard Name	Contents	Target User Role
Work Order Performance	Service managers start here for a summary of KPIs. Review work order volume, average work time, average travel times, and first-time fix rate. Also review technician agent data, including the number of work orders per technician, their planned and actual travel times, and utilization rates.	Service manager/dispatcher
Technician Performance	Focuses on each technician's work order volume. Review agent-by-agent work order prioritization, travel time, first-time fix rates, mean time to repair, due date and SLA violations.	Service manager
Service Resource	Shows time spent by an individual technician on each type of job so managers can assess technicians' efficiency. Also shows average number of jobs completed by the technician for a specific date range, the worker's first-time fix rate, and actual versus estimated travel times to job sites. Embeddable in technicians' pages so they can monitor their own efficiency KPIs.	Service manager and technician
Service Territory	Service managers can see the types of jobs completed in their territories—for example, HVAC, electrical, or plumbing—to ensure that their team members have the skills they need. They can also see the daily average number of jobs completed, technician utilization rate, and actual versus estimated technician travel times.	Service manager

Set Up and Create the Field Service Analytics App

Set up Field Service Analytics app permissions, make sure your org is enabled to use Field Service, and assign field level security correctly so you can create and share the app.

Set Up Field Service Analytics App Permissions

The Service Analytics license provides access to the Field Service Analytics app. To set up permissions for Field Service Analytics, follow the detailed instructions starting at Set Up Permissions for the Service Analytics App.

Make Sure That Field Service Is Enabled in Your Org

🛑 **Important:** Important: Your org must be enabled to use Field Service and its new standard object model to use Field Service Analytics.

1. Go to Salesforce Setup.

2. Enter **Field Service** in the Quick Find/Search box. If **Field Service Settings** appears in Setup, click it.

3. In the center window, under Field Service Settings, check to see if the box for **Enable Field Service** is checked.

If you're unsure that your org is set up to use Field Service, contact your Salesforce representative.

Assign Field Level Security

Before you create Field Service Analytics, set Salesforce field-level security to enable the Analytics Cloud Integration User to see all fields you'd like your app to include. Integration users run the dataflow, and if they don't have proper field-level security permissions, the dataflow can fail. For instructions, see Create and Share the Service Analytics App, Step 2.

Create Field Service Analytics

Once you've completed the previous setup processes, create Field Service Analytics.

1. Go to Analytics Cloud.

2. Click Create in the upper right corner.

3. Select the Field Service Analytics template.

4. Name your app and click Create.

Tableau CRM creates your app. This may take a few minutes while Tableau CRM runs a new dataflow and creates the app's dashboards. Refresh your browser, and click the app you just created to see the dashboards.

🛑 **Important:** When users are deactivated, they lose share and delete access to all apps they manage. To avoid "stranding" an app, be sure that manager access is assigned to at least one active user BEFORE deactivating the user who's the manager of the app.

Share Field Service Analytics

Now that you've created the app, share it with users in your organization. You can only share it with users who have the "Use Analytics Templated Apps" and "Access Service Cloud Analytics Templates and Apps" permissions enabled.

1. Open your app if it's not already open. If you've navigated away from Tableau CRM Studio, go back to it, select **All Items**, find your app, and click it.

2. Click the Share icon [icon] at upper right.

3. In the next screen, use the search field under **Invite others:** to find other users in your org.

4. Select whether you want to make the selected user a Viewer, Editor, or Manager of the app.

   (!) Important: Users with the "Use Analytics Templated Apps" permission and Editor or Manager access to the app can create, edit, and delete assets in the app.

5. Click **Add**, then click **Save**.

Fundraising Analytics Template

The Fundraising Analytics template lets you create an app that brings the power of Analytics Cloud to data from the Salesforce Nonprofit Success Pack.

Fundraising Analytics gives you a fast way to get started analyzing the success of your fundraising with Tableau CRM. Optimized for organizations who use the Salesforce Nonprofit Success Pack, the app brings Salesforce donation data from accounts, contacts, and opportunities (donations) into Analytics Cloud. You can visualize your donation performance out-of-the-box with its prebuilt dashboards. Or customize them and create your own explorations using the app's datasets to meet your own requirements.

   (!) Important: See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Fundraising Analytics app.

Org Requirements

Your org requires the following before you can create an app from the Fundraising Analytics template:

- Make sure you and all app users have the Tableau CRM Growth license.

- Data stored in the Salesforce Opportunities, Accounts objects and at least one task and one event. The Salesforce Nonprofit Starter Pack uses the Opportunities object for donation data. In the app, *opportunities* are called *donations*.

- Set Salesforce field-level security to enable the Analytics Integration User to see all fields used in the app. See Set Field Level Security to Enable Creation of an Tableau CRM Template

The Fundraising Analytics Configuration Wizard

Fundraising Analytics includes a configuration wizard. Follow these steps to use it.

1. **Org check page.** Analytics Cloud checks your org for the data needed to create the app. When the check is done, click **Looks good, next**, and move to the next page. If Analytics Cloud detects an issue, it shows an error message. Follow the instructions in the message to fix the problem, and try creating your app again.

2. **Basic and custom create options page.**

- Use basic create the first time you create Fundraising Analytics or when you want to create the app quickly.Analytics Cloud creates the app based on default settings. Select **Basic**, click **Looks good, next**, name your app, and click **Create**. Analytics Cloud takes a few minutes to create your app. You can see its status on the next page. When it's done, refresh your browser to view the app.

- Custom app creation gives you more fine-grained control over Fundraising Analytics features and data. To select it, click **Custom**, then **Looks good, next**, and move to Step 3—the first page of wizard questions. For general guidance on how to use the wizard, see How to Answer Tableau CRM Template Custom Wizard Questions .

3. **Choose additional objects page.** The org check from Step 1 checks your org for objects you can add to your app. Based on your selection, Analytics Cloud adds wizard questions about how you want to use data from the additional objects. Select the ones you want and click **Looks good, next**.

4. **Questions about how Fundraising Analytics uses Accounts data.** Answer the questions on this page about how app dashboards segment customer and geographic data. When you're satisfied with your answers, click **Looks good, next**.

5. **Questions about how Fundraising Analytics uses Opportunities data.** Answer the questions on this page about the opportunity field that contains total amounts and the field that indicates new business. Remember, *opportunities* are called *donations* in your finished app, after it's created. When you're satisfied with your answers, click **Looks good, next**.

6. **Questions to enhance the Analytics Cloud experience.** Answer the questions on this page, as follows, and click**Looks good, next**.

   - Question 1: Lets you control user access to data in your app. Here are the available options:

     - Option 1 enforces Salesforce role hierarchy, which means that users can only see data in Opportunities and Accounts owned by them and their subordinates.

     - Option 2 enables team benchmarking. It lets users see data in Opportunities and Accounts owned by them and their subordinates. It also lets users see data from Opportunities and Accounts owned by others at the same level in the role hierarchy.

     - Option 3 lets all users see all Sales Cloud data regardless of role.

   - Question 2, mandatory: Select how your app handles Account Team information. Defaults to Option 3, which excludes Account Team data. Select Option 1 to add Account Team member data to datasets or Option 2 to add Account Team data and grant record visibility to members of the team.

   - Question 3, mandatory: Select how your app handles Opportunity Team information. Defaults to Option 3, which excludes Opportunity Team data. Select Option 1 to add Opportunity Team member data to datasets or Option 2 to add Opportunity Team data and grant record visibility to members of the team.

   - Question 4, optional. Only appears if your app detects that you segment opportunities by record types. Choose record types and Tableau CRM excludes *unselected* record types from datasets. Leave this question blank to include *all* record types.

7. **Questions about adding data to Fundraising Analytics datasets.** By default, this page asks you 4 questions about adding data from Accounts, Donations (Opportunities), Users, and Activities to the app. If you added objects to the app in Step 3, questions about adding data from those objects appear here. Select the fields you'd like added to app datasets, and click **Looks good, next**. Then name your app, and click **Create**. Analytics Cloud takes a few minutes to create your app. You can see its status on the next page. When it's done, refresh your browser to view the app.

Fundraising Performance Analytics Template

Nonprofits can measure the impact of fundraising work with a precise picture of donor retention and attrition. The app also provides actionable insight into donor gains and losses, donation amount gains and losses, and other key performance indicators.
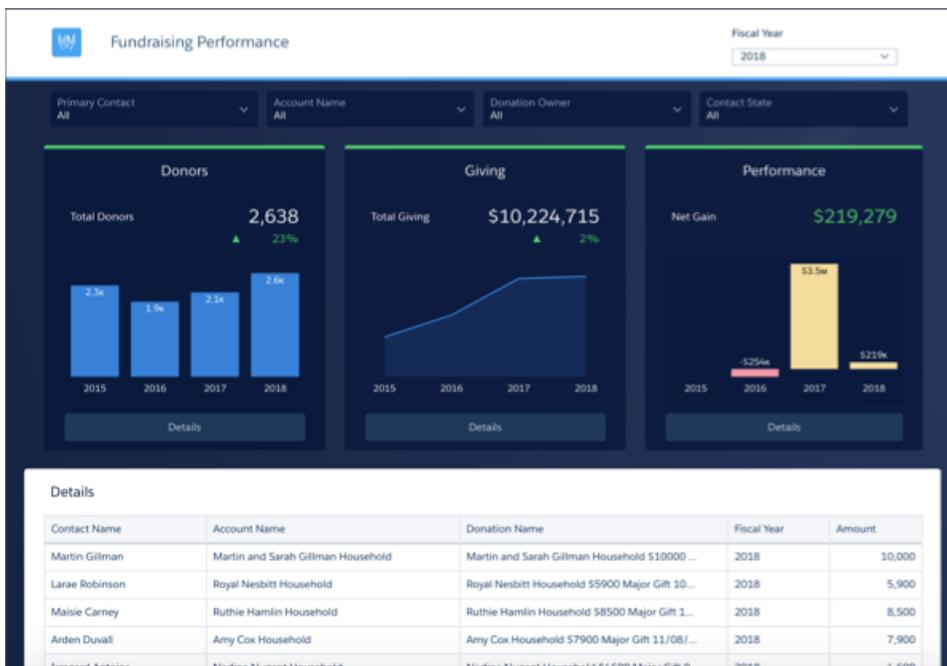
Use the Fundraising Performance Analytics template to create an app targeted to addressing the analytics needs of Salesforce Nonprofit Success Pack users. Its dashboard measures and compares fundraising efforts concisely and uniformly to help you develop strategies to improve future drives.

❗ **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Fundraising Performance Analytics app.

Org Requirements

Your org should include the following to make the best use of the Fundraising Performance Analytics template:

📝 **Note:** You can create the Fundraising Performance Analytics app in orgs with or without the Nonprofit Starter Pack, as long as you use the objects mentioned here.

- At least three years of data in standard objects used in the Nonprofit Success Pack, including the Contact, Account, Opportunity (Donation), and User objects. Use of the Contact object is optional; if Analytics does not detect contacts, the template uses accounts.

- Data should include Opportunities (Donations) with stage type Closed Won, for example, Stage - Closed Won, Awarded, Posted.
- Your org should use the Campaign object Campaign Type field to signify a donation's type of program or segment. That lets you analyze fund-raising results according to different programs, such as direct mail, major giving, and so on.

    For best results, exclude large one-time gifts, which may distort your results.

Use Fundraising Performance Analytics

Open the app.

1.
    From the app picker ⣿ , select **Tableau CRM Studio** to open the Tableau CRM home page.

2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

The app's dashboard shows metrics for donors, giving, and performance. Summary charts across the top display top-level measures. Click **Details** below each chart for more information about each aspect of your fund-raising performance.

- **Donors**
    - Summary shows annual total number of donors and totals for each year.
    - Details include donor and retention rates, growth of number of donors, and donors by type.

- **Giving**
    - Summary shows annual total donations and totals each year.
    - Details include net growth in donations, average gift per donor, average gift, donations over time, and a breakout of the size of donations by category.

- **Performance**
    - Summary shows net gain for the selected period and annual gains for each year.
    - Details include percentage donor churn (donors gained and lost each year), and donations gained and lost broken out by categories.

Key Performance Indicators

Key performance indicators (KPI) are determined as follows:

Total Donors	Total Donors Current Year
Total New Donors	Donor Type - New
Total Recurring Donors	Donor Type - Recaptured, Upgraded, Same, Downgraded
Total Donations	Total Donations Current Year
Total Giving	Total Giving Current Year
Net Growth in Donors	Current Year Total Donors - Last Year Total Donors
Net Growth in Giving	Current Year Total Giving - Last Year Total Giving
Rate of Growth in Donors	(Current Year Total Donors - Last Year Total Donors) / Last Year Total Donors * 100

Rate of Growth in Giving	(Current Year Total Giving - Last Year Total Giving) / Last Year Total Giving * 100
Donor Retention Rate	Total Donors that gave Current Year (Same, Upgraded, Downgraded) / Total Donors Last Year * 100
Donation Retention Rate	Total Retained Donations Current Year (Same, Upgraded, Downgraded) / Total Donations Last Year * 100
Donor Attrition Rate	Total Donors that did not give Current Year (Lapsed New & Repeat) / Total Last Year Donors * 100
Donation Attrition Rate	Total Lost Donations Current Year (Lapsed New & Repeat) / Total Last Year Donations * 100
Donor Participation Rate	Total Donors Current Year (Recaptured, Upgraded, Same) / Total Donors Current Year * 100
Donor Recapture/Reactivation Rate	Total Donors Current Year (Recaptured) / Total Donors Current Year * 100
Donor Churn	Total Gained Donors Current Year (New, Recaptured) - Total Lost Donors Current Year (Lapsed New & Repeat) / Total Donors Current Year * 100
Average Gift $	Total Giving Current Year / Total Gifts Current Year
Average Giving $ per Donor	Total Giving Current Year / Total Donors Current Year
Average Number of Gifts per Donor	Total Gifts Current Year / Total Donors Current Year
Average Gift $ per Donor	Total Donor Average Gift $ / Total Donors Current Year

Gain/Loss Category Definitions

Categories for gains and losses are defined as follows:

- **New.** Donors who first gave during current year.
- **Recaptured.** Previously lapsed donors who gave again in current year.
- **Upgraded.** Donors who gave more in current year than in previous year.
- **Same.** Donors who gave the same amount current and previous years.
- **Downgraded.** Donors who gave less in current year than in previous year.
- **Lapsed new.** New, first-time donors in previous year who did not give in current year.
- **Lapsed repeat.** Other lapsed donors who gave in previous and prior years but not in current year.

The Analytics for Healthcare Template

Use actionable insights from Tableau CRM for Health Cloud apps to drive intelligent patient engagement, improve care effectiveness, and manage patient risk.

Note: Analytics for Healthcare is only for Salesforce Health Cloud users. It requires that you have the Health Analytics Plus add-on license. See Deploy Analytics for Healthcare.

The Tableau CRM for Insurance Template

Analytics for Insurance powers agents with practical insights on their sales performance and enables them to be more efficient in sales execution. App visualizations segment the customer base and provide insights on upsell/cross-sell opportunities so agents can grow their written premiums. Managers get insights on their team's performance and what makes the top performers different so they can coach their team members.

📝 **Note:** Analytics for Insurance is included with Tableau CRM for Financial Services, which is available at an extra charge for customers with Financial Services Cloud Basic or Standard licenses and the FSCAnalyticsPlus (Tableau CRM for Financial Services) license. See Deploy Tableau CRM for Financial Services for complete deployment instructions.

Lead Trending Analytics Template

Create an app from the Lead Trending Analytics Template to increase your visibility into your team's lead conversion process. Sales operations can get instant insight into how quickly the team converts leads and can identify bottlenecks in the conversion process.

<div style="float:right">
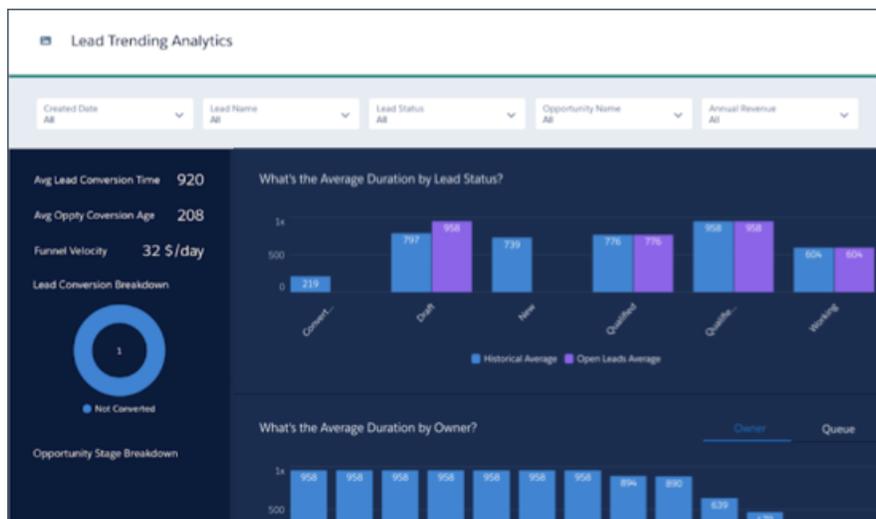
USER PERMISSIONS

To create and manage the Lead Trending Analytics app:
- Manage Analytics Templated Apps

To use the Lead Trending Analytics app:
- Use Analytics Templated Apps

</div>

⛔ **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Lead Trending Analytics app.

Org Requirements

Your org requires the following before you can create an app from the Lead Trending Analytics template:

- Make sure you and all app users have the Tableau CRM Growth license.
- Your org must use lead history and have at least one opportunity linked to a lead. It also must use lead record types.
- Set Salesforce field-level security to enable the Analytics Integration User to see all fields used in the app. See Set Field Level Security to Enable Creation of an Tableau CRM Template.

Use Lead Trending Analytics

Open the app.

1.

From the app picker ⠿ , select **Tableau CRM Studio** to open the Tableau CRM home page.

2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

Use the chart at the top to get an overview of your lead conversion data. You can see the following:

- Leads with the most changes.

- The average change duration, that is the average time leads spend in the process before being converted.

- Average time to close leads.

- What happened to leads after conversion.

Filter the data by lead creation date, industry, conversion date, and stage.

Charts in the middle of the dashboard show lead converted to accounts and contacts.

Scroll down to see details about all your leads. Take action on a record by hovering over the lead name and clicking the disclosure triangle to the right. That opens the Actions Menu from the record. From there, post to Chatter, create a task or an event, or perform another action to move the process along.

Metric Calculations

Left Panel

- **Avg Lead Conversion Time**. Average time between creation of leads and their conversion to opportunities. If leads are not converted to opportunities, average time between creation and current date. Includes all leads in the org.

- **Avg Oppty Conversion Age**.Average time between lead creation and opportunity close date for all closed opportunities, or current time for all open opportunities. Only includes leads that have converted to opportunities.

- **Funnel Velocity**. Dollars closed per day, which helps sales team members determine if there is enough business in the pipeline. Calculated as the number of (Converted leads * Average conversion rate * Average deal size) / Average conversion time. Average conversion rate equals Number of converted leads / Number of leads. Average conversion time equals time difference between lead creation date and opportunity creation date.

Center Bar Charts

- **What's the Average Duration by Lead Status?** Average number of hours leads have remained in the designated status (for example, *Qualified* or *New*). Includes all leads with changed status.

- **What's the Average Duration by Owner?** Average number of hours designated owner has owned leads. Owner can be a user or queue.

The Tableau CRM for Manufacturing Template

The Tableau CRM for Manufacturing lets account managers visualize all aspects of their business to keep them on top of sales agreements, orders, and contracts.

> 📝 **Note:** Tableau CRM for Manufacturing is only for Salesforce Manufacturing Cloud users. It requires that you have the Manufacturing Analytics Plus add-on license. See Deploy Tableau CRM for Manufacturing.

Analytics for Mortgage Template

Analytics for Mortgage enables loan officers and managers to drive increased mortgage sales by helping prioritize the customer leads and mortgage applications to focus on. Process loans faster by highlighting the mortgage applications that have the highest value, applications that have been open the longest, and applications that are missing documents.

📝 **Note:** Analytics for Mortgage is included with Tableau CRM for Financial Services, which is available at an extra charge for customers with Financial Services Cloud Basic or Standard licenses and the FSCAnalyticsPlus (Tableau CRM for Financial Services) license. See Deploy Tableau CRM for Financial Services for complete deployment instructions.

Multi Org Sales Analytics Template

Use the Multi Org Sales Analytics template to create a scaled-down version of the Sales Analytics that pulls in data from all your connected orgs.

🛑 **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Multi Org Sales Analytics app.

USER PERMISSIONS

To create and manage the Multi Org Sales Analytics app:
- Manage Analytics Templated Apps

- Manage Einstein Discovery

To use the Multi Org Sales Analytics app:
- Use Analytics Templated Apps
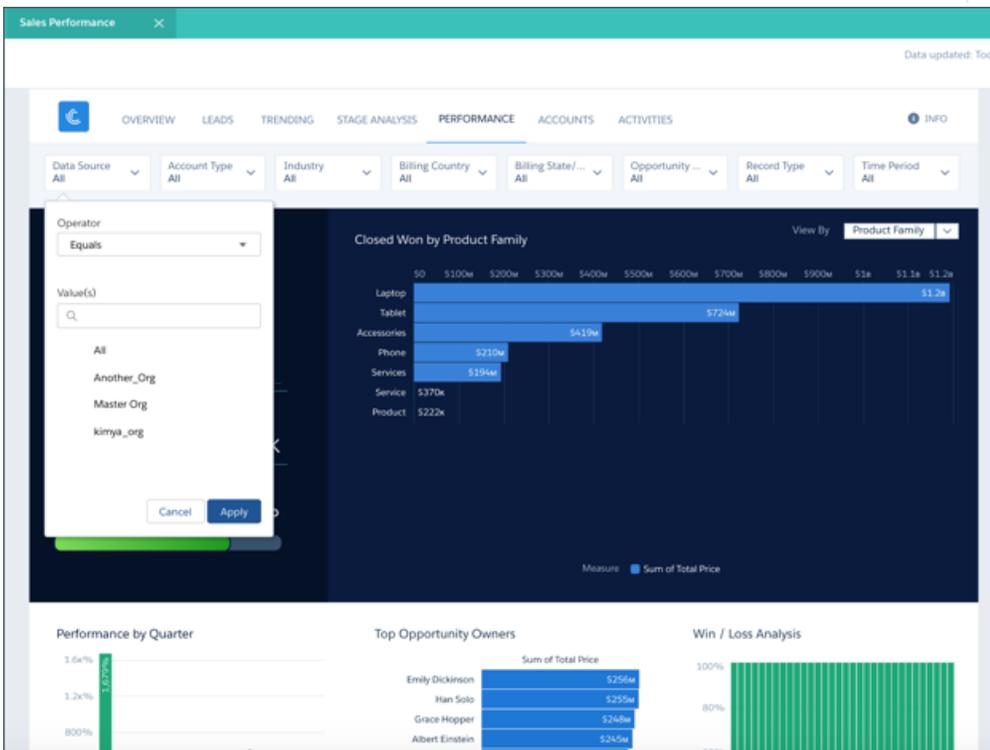
Now you can create a version of the Sales Analytics app that includes data from all the Salesforce orgs you use to manage sales interactions. You get key dashboards that help sales managers unlock the full power of Sales Cloud data—from any org already connected to Tableau CRM. Select the orgs whose data you want to add to the app in the configuration wizard when you create the app. Then, when you use the app's interactive dashboards, choose the org data you want to view.

Note: Multi Org Sales Analytics doesn't support sharing rules or security predicates when you first create the app. Initially, all users can view all data in app dashboards, and you can't restrict access to data by role. You can build a security model for app data by customizing datasets after the template has been created. See Add Row-Level Security with a Security Predicate.

Org Requirements

Your org requires the following before you can create an app from the Multi Org Sales Analytics template:

Note: The org you use to create the Multi Org Sales Analytics app is referred to as the *local org*. Other orgs with Sales Cloud data you add to the app are referred to as *connected orgs*.

- Make sure you and all app users have Tableau CRM Growth license.
- Create connections from the local org to connected orgs before attempting to create the app. Connected orgs must support API version 35 or higher. See Create a Remote Connection on page 620. Or watch the first 1:20 of the video Connect to External Orgs.
- Sales Analytics data requirements apply to Multi Org Sales Analytics. See Sales Analytics Limitations on page 1537. Also, the Multi Org Sales Analytics doesn't enable you to include data from the Campaign object.
- All connected orgs must use the same objects as the local org, and all those objects must contain data. If they don't, app creation may fail.
- Only the local org supports action menus. To add action menus to connected orgs, see Add Actions to Dimensions in the *Analytics Extended Metadata (XMD) Reference*.
- The local and connected orgs are limited to a combined maximum of 100 objects or the Multi Org Sales Analytics dataflow fails.
- Schedule data connections to sync every day at a time before the daily dataflow on page 1393 for the Multi Org Sales Analytics app. It's recommended that you set data connections to sync at least 3 hours before the dataflow. The Multi Org Sales Analytics app doesn't contain the latest data if the daily data syncs don't finish running before the daily app dataflow starts running. See Schedule, Run, and Monitor Data Sync on page 686.
- Set Salesforce field-level security to enable the Analytics Integration User to see the fields that you want to analyze. During app creation, Tableau CRM checks your org's field-level security and lets you know if you have to edit it.

Multi Org Sales Analytics Configuration Wizard

Use the Multi Org Sales Analytics configuration wizard to add data from other connected orgs with Sales Cloud data to your app.

1. In Tableau CRM Studio, click **Create**, then **App**.

2. Select **Create App from Template**. On the Choose an App Template page, find Multi Org Sales Analytics, and select it. Then click **Continue**.

3. That brings you to the Multi Org Sales Analytics page. Have a quick look at a preview of app dashboards, and click **Continue**.

4. The next page requires you to add data from connected orgs to your app. Select the connected orgs you want to use and click **Looks good, next**.

   Important: Be sure that the number of objects from the local org and connected orgs combined doesn't exceed 100 or the dataflow fails.

5. That starts a compatibility check of the local org. It checks to be sure that the data in the org meets the data requirements to create the app. If it determines that the local org doesn't contain the required data, you see error messages with instructions on what to fix. Once the compatibility check finishes running, click **Looks good, next**.

6. The next page lets you choose to add data from the Products, Leads, and Cases objects in your local org to your app. Select the ones you want, then click **Looks good, next**.

> ⛔ **Important:** Be sure that the connected orgs contain data in the selected objects. If they don't, app creation may fail.

7. On the next page, click **Sync Data** to bring in data from the connected orgs. When the progress bar reaches 100%, click **Looks good, next**.

8. **Name your app, and click Continue.**

Tableau CRM creates your app and shows a page with status. If app creation fails, check to see if the sync with connected orgs succeeded.

1. Click the wheel icon at upper right, and select **Data Manager**.

2. To see the status of your data connections, click **Connect** in the left-hand column.

3. Roll over the icons next to the names of the connections to see their status. The icons show green if the data sync succeeds, red if the sync fails.

4. You can try rerunning the sync. Click the arrow to the far right of the connection name and select **Run Data Sync**.

5. Data sync fails for one of two reasons:

   a. The number of objects in your local and selected remote orgs exceeds 100.

   b. One or more of your remote orgs don't contain data in the same objects as the local org.

6. Fix the issue and try creating the app again.

> ⛔ **Important:** The Multi Org Sales Analytics Template uses multiple connections. Be sure to schedule sync for each of them as described in Schedule Data Sync and the Dataflow for an Tableau CRM Template on page 1393. The app uses only a single dataflow. Schedule the dataflow to run as instructed starting in Step 5 of Schedule Data Sync and the Dataflow for an Tableau CRM Template.

Use the Multi Org Sales Analytics App

Open the app.

1. From the app picker ⣿ , select **Tableau CRM Studio** to open the Tableau CRM home page.

2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

Click **Run App**. The app opens to the Company Overview dashboard. Navigate to the app's other dashboards by following the links across the top. The app includes dashboards from the Sales Analytics app's targeted for sales managers. App dashboards function identically to the Sales Analytics dashboards with one addition. Use the **Data Source** menu to select the org data to view in the dashboard.
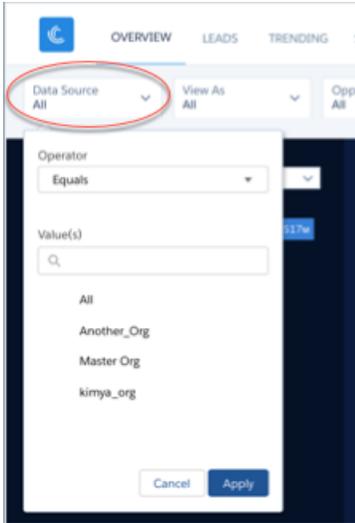
Follow help links from the dashboard to details about using the dashboards and metric calculations. Or, follow the links here.

> **Note:** The following features used in the Sales Analytics are not supported in Multi Org Sales Analytics:
>
> - Collaborative forecasting
> - Opportunity splits
> - Product schedules

Patient Risk Stratification Template

The Patient Risk Stratification template creates an app that lets healthcare coordinators identify high-risk patients to make sure they get the extra care they need.

USER PERMISSIONS

To create and manage the Patient Risk Stratification app and set up custom actions in the app:
- Manage Analytics Templated Apps

To use the Patient Risk Stratification app:
- Use Analytics Templated Apps
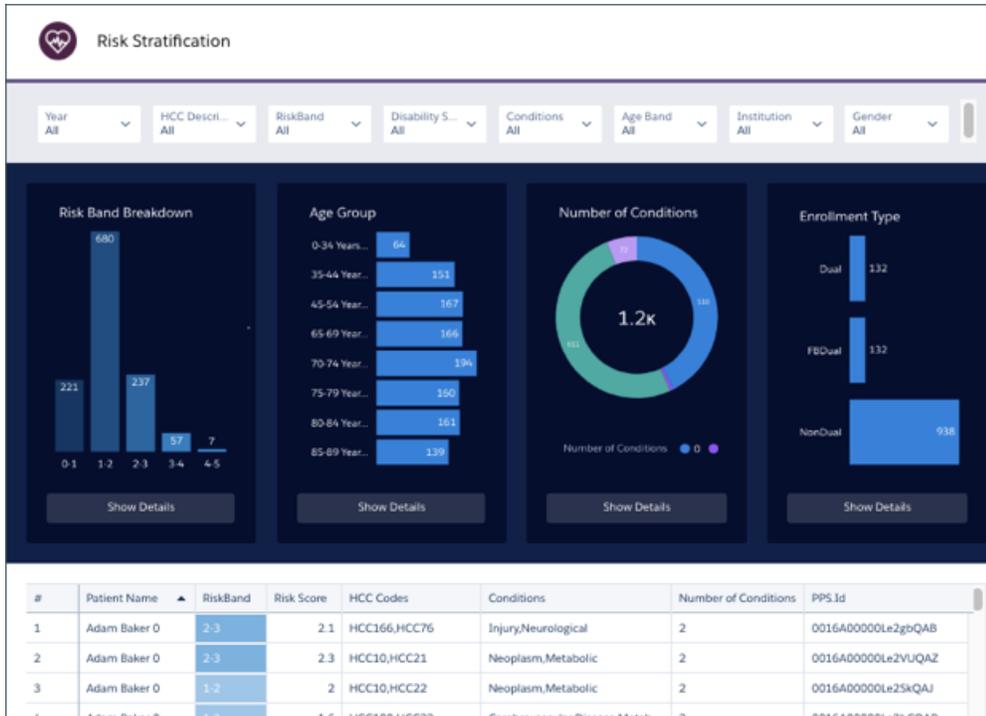
Patient Risk Stratification brings your Health Cloud data into Tableau CRM to help you deliver the right level of care management to high-risk patients. The Patient Care Risk Stratification dashboard lets you drill in to a population. You can see risk score, age, number of conditions, and enrollment type for specific population segments. The metrics help you understand risk for your patients and plan the appropriate level of care to assure best-possible treatment outcomes.

> **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Patient Risk Stratification app.

Org Requirements

Your org requires the following before you can create an app from the Patient Risk Stratification template:

- Make sure you and all app users have the Tableau CRM Growth license.
- Patient Risk Stratification is only for Health Cloud customers who have installed the Health Cloud Risk Stratification package.
- Your org needs to include at least one risk score within a program patient summary record. You also need to use Record Types.
- Set Salesforce field-level security to enable the Analytics Integration User to see all fields used in the app. See Set Field Level Security to Enable Creation of an Tableau CRM Template.

Enable Bulk Actions in the App

After creating the app, enable bulk actions in the table at the bottom of the Risk Stratification dashboard. Bulk actions let you perform an action on multiple Salesforce records at the same time. In this case, you can send a list of high-risk patients to other Health Cloud users.

1. Click the edit icon ![edit icon] at the top of the dashboard.

2. Select the table with the list of patients at the bottom of the dashboard.

3. In the Properties panel at right, select **Widget**.

4. To enable the bulk action, click **Show custom action**. In the **Custom Action Label** field, enter a name for the custom action, for

example *Send List to Health Cloud*.

5. In the **Visualforce Page Name** field, enter *HcWaveListIntegrationPage*. This string refers to a Visualforce page that's included with the Health Cloud package.

6.
Click the Save icon [💾] at the top of the dashboard, then click the **Save** button.

Use Patient Risk Stratification

Open the app.

1.
From the app picker ⚏ , select **Tableau CRM Studio** to open the Tableau CRM home page.

2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

The filters along the top of the dashboard let you select criteria for creating a list of high-risk patients. In the example dashboard shown above, you could select **Diabetes** from the **Conditions** filter and **60-64 Years Old** from the **Age Band** filter to create a list of 60-to-64-year-old patients with diabetes.

The visualizations just below the filters change to show data for the selected group of patients as determined by filter selections. Click the **Show Details** button below each chart to learn more about that group. View the list of patients in the table at the bottom of the dashboard. Click the arrow in the top right corner of the table and select **Send to Health Cloud** to share the list with others in your org.

📝 Note: The action is called something different in your org if you gave it a different name. See the section above, *Enable Bulk Actions in the App*.

Pipeline Analytics Template

Use Pipeline Analytics to plug your pipeline snapshot data into a ready-made waterfall dashboard. You get instant insight into how the pipeline changes between two snapshots.

Use the Pipeline Analytics template to create a waterfall dashboard from snapshotted pipeline data. You can use a dataset created using the Snapshot Analytics template, another snapshotted Salesforce object, a trended pipeline report, or even a snapshot of a CSV file with pipeline data. The template's configuration wizard takes you through a step-by-step process to make it easy to get a cool waterfall chart showing your pipeline. You don't have to manually import data or create a dashboard from scratch in Tableau CRM Studio. Tableau CRM does the work for you.

> 🛑 **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Pipeline Analytics app.

Org Requirements

Your org requires the following before you can create an app from the Pipeline Analytics template:

- Make sure you and all app users have the Tableau CRM Growth license.

- Make sure you and others who create the app have the 'Use Any API Client' permission. Without that permission, you may not see the datasets you want to use for the app in the wizard. To assign that permission, request API Access Control from Salesforce Customer Support. See Restrict Access to APIs with Whitelisted Connected Apps.

- A snapshotted dataset with pipeline data. It's recommended that you use the Snapshot Analytics template to snapshot the dataset, but any other snapshotted dataset can work. The source of your data can be any of the following, as long as it's imported to a Tableau CRM dataset:

  1. Data from the standard Salesforce Opportunities object.

  2. Data from a trended pipeline report.

  3. Data from another standard or custom object.

  4. Data from a CSV or some external tool.

- The dataset you use in the waterfall must include fields that contain the following data or app creation fails.

  - Amount (measure)
  - Is closed (yes/no; boolean)
  - Is won (yes/no; boolean)
  - Unique ID
  - Close date (dimension)
  - Snapshot date

The Pipeline Analytics Configuration Wizard

Answer the questions in the eight questions in the wizard to select a dataset and fields required to create the waterfall.

1. Choose dataset. Choose a snapshotted dataset with your pipeline data. If the dataset doesn't contain pipeline data, the remaining questions don't list the fields required to create the waterfall.

   > 🛑 **Important:** If you don't see the dataset you want to use in the wizard, be sure you have the 'Use Any API Client' permission. See *Org Requirements*, above.

2. Select field that contains total amount.

3. Select the dimension field with opportunity close date.

4. Select the boolean field that indicates if an opportunity is open or closed.

5. Select the boolean field that indicates if an opportunity is closed won or closed lost.

6. Select the field that contains the unique idea for an opportunity.

7. Select the field that contains the date of the snapshot date.

8. Select fields to use as filters in the dashboard. Values in those fields appear as filter options at the top of the dashboard. The dashboard also uses field names as column headers in a values table at the bottom of the dashboard. Select up to six fields

Use Pipeline Analytics
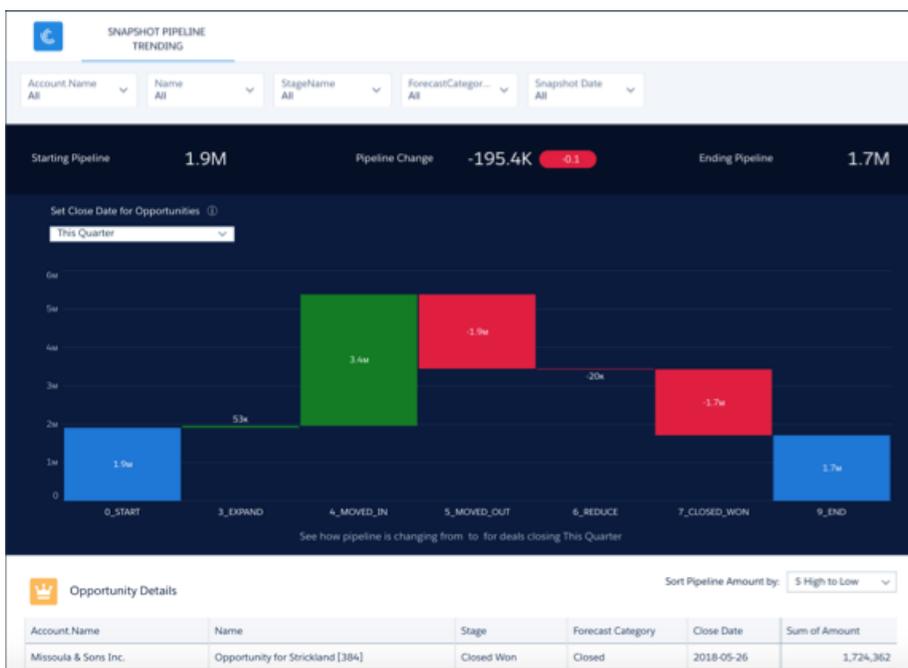
Open the app.

1.
From the app picker ⠿ , select **Tableau CRM Studio** to open the Tableau CRM home page.

2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

Change filter selections in the Pipeline Analytics dashboard to dig deeper into your pipeline. Available filters depend on your selections in the last question of the wizard. Here you can see five of possible six filters, as follows:

• **Account Name**: View pipeline for selected account.

• **Name**: View pipeline by named opportunity.

• **StageName**: View pipeline by sales stage, such as Closed Won or Closed Lost.

• **ForecastCategory**: View pipeline by forecast category, such as Best Case, Commit, or Closed.

- **Snapshot Date**: View pipeline as of a specific date.

If you don't see the filter selections you need to analyze the pipeline, create the app again using different selections in that last wizard question.

Pricing Analytics Template

With the Pricing Analytics template, Salesforce CPQ customers can be sure they're pricing their deals correctly. It recommends the ideal price for every quote and helps you identify potentially underpriced deals.

Salesforce CPQ customers: Use the Pricing Analytics template to create an app that uses Einstein Discovery intelligence to calculate ideal discounts for your quotes. Its dashboard also compares well-priced deals with under-priced deals and shows you potential missed revenue from underpricing. You can also learn about reps who are underpricing deals so you can coach them on the discounting process.

> ⛔ **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Pricing Analytics app.

Org Requirements

> ⛔ **Important:** The Pricing Analytics is only for Salesforce CPQ customers. To enable complete Pricing Analytics functionality, install on page 1785 the Einstein Discovery managed package, and set up your org on page 1593 to use Einstein Discovery.

Your org requires the following before you can create an app from the Pricing Analytics template:

- Make sure you and all app users have Tableau CRM Growth and Salesforce CPQ licenses. To create the app, you also need the Einstein Discovery license.

- Add a custom number field for recommended discounts on the Salesforce CPQ Quote object (API name $SBQQ\_\_QuoteLine\_\_c$). Use a name for the field you can easily remember. You select that field in the template's configuration wizard.

- Set Salesforce field-level security to enable the Analytics Integration User to see the fields that you want to analyze. During app creation, Tableau CRM checks your org's field-level security and lets you know if you have to edit it.

Pricing Analytics Configuration Wizard

Pricing Analytics includes a one-page configuration wizard with two questions. Based on your answers, Einstein Discovery identifies factors that determine your current discount practices and uses them to compute a discount recommendation.

1. Question 1: Indicate the recommended discounts field on the Quote Line Item object ($SBQQ\_\_QuoteLine\_\_c$).

2. Question 2 (optional): Indicate the field that you use to track competitors for opportunities. If you don't track competitors, leave blank.

Enable Pricing Analytics Recommendations

Adding Einstein Discovery intelligence for pricing recommendations requires extra setup after you create the app. First, follow these Einstein Discovery procedures.

1. Tableau CRM runs an Einstein Discovery story during app creation. To be sure that the story is complete, navigate to Tableau CRM Studio.

2. Open the app you created using the Pricing Analytics template.

3. In the Stories section of the app overview page, click **Pricing**. If you see a message, "Einstein is in the lab crunching numbers," the story isn't done running. If you see a chart with **What Happened**, **Why It Happened**, and so on, at the top, the story is complete.

4. Deploy the Einstein Discovery model. See "Deploy a Model to a Salesforce Object" in Einstein Discovery help on page 1784.

5. Initiate a write-back to the Salesforce Quote object. See Connect Einstein Discovery to Your Custom Fields on page 1786.

Next, run the app dataflow.

1. Navigate to Tableau CRM Studio.

2. Click the Wheel icon at upper right and select **Data Manager**.
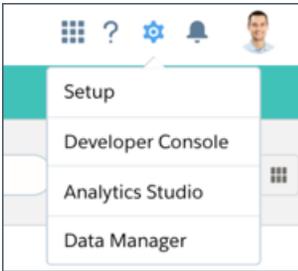
3. Select **Dataflows** and look for the app you created using the Pricing Analytics template.

4. Click the disclosure triangle on the far right, then select **Run Now** to run the dataflow.
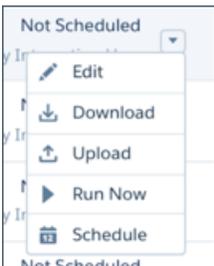
Use Pricing Analytics

Open the app.

1. From the app picker ⣿ , select **Tableau CRM Studio** to open the Tableau CRM home page.

2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

Click **Run App** to open the app's Einstein Pricing Overview dashboard.

The charts at the top of the dashboard show you the amount of revenue at risk from under-priced open opportunities and revenue from under-priced closed opportunities. See the chart at right for sales team members who need coaching on the pricing process. The chart highlights reps who have missed revenue from underpricing.

Scroll down to details chart further down for a list of deals. The **Category** column shows whether a deal is well priced or not. To the right, see the pricing score and recommended discount. These metrics are based on the computed **Pricing Score** field (Recommended Discount / Total Discount). A ratio of more than 1 indicates a well-priced deal. Less than 1 indicates that the discount may be too large.

Take action from the details chart by positioning the cursor over an account name, clicking the disclosure triangle that appears, and opening the Actions menu. Then post to Chatter, kick off a new event or task, or take any other action to help reps correctly price deals.

SEE ALSO:

 Explain, Predict, and Take Action with Einstein Discovery

 Display Einstein Predictions Using Custom Fields (Deprecated)

 Install the Einstein Discovery Managed Package (Deprecated)

Public Sector Case Analytics Template

Public Sector Case Analytics lets public sector service organizations measure their effectiveness and identify immediate actions to improve case work performance.

🛑 **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Public Sector Case Analytics app.
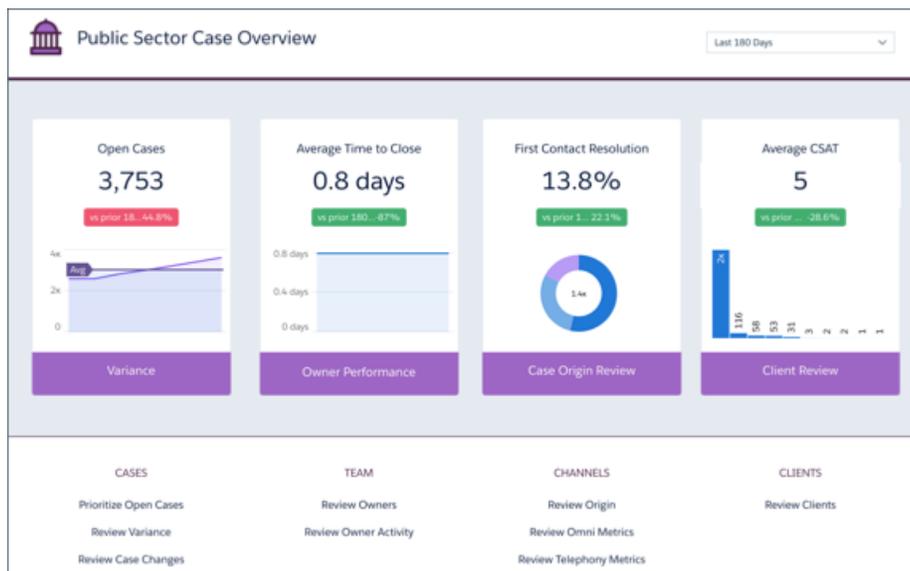
EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer Edition**.

USER PERMISSIONS

To create and manage Public Sector Case Analytics:
- Manage Analytics Templated Apps

To use the Public Sector Case Analytics app:
- Use Analytics Templated Apps

The app's prebuilt dashboards give you instant insight into case status, surfacing the ones that require immediate attention. Visualizations surface case-related activities that are near or past due to help you stay on track delivering client services. View bottlenecks in the process so you can take action to assure effective case management and resolution. And see if you and your team are compliant with service level agreements.

Also, get details about your clients and beneficiaries, trends in client satisfaction, and case history brought to life via interactive visualizations. You can also immediately surface top-performing case managers and understand why they're effective. And identify lower performers so you can help them take steps to become more effective.

Org Requirements

Your org requires the following before you can create an app from the Public Sector Case Analytics template:

- Make sure you and all app users have the Analytics Plus or Analytics Growth licenses.

- Your org must have at least one closed case and one closed task to enable app creation.
- The wizard checks to see if you track data about the following in your org:
  - Customer satisfaction (CSAT)
  - Knowledge
  - Case history
  - Queues
  - Events
  - Business hours
  - Case record types
  - Telephony
  - Omni-Channel
  - Person Accounts

- Set Salesforce field-level security to enable the Analytics Integration User to see all fields used in the app. See Set Field Level Security to Enable Creation of a Tableau CRM Template.
- Tableau CRM looks for your org's fiscal year start date and uses it in app dashboards. To choose a different fiscal start date, use the app's Custom create option.

1. Customize Public Sector Case Analytics with the Configuration Wizard
   Use the configuration wizard to make sure Public Sector Case Analytics reflects how you prefer to view case management data.

2. Use Public Sector Case Analytics Prebuilt Dashboards
   After you create Public Sector Case Analytics, navigate to the app and use its dashboards to measure your team's effectiveness and improve services delivery.

Customize Public Sector Case Analytics with the Configuration Wizard

Use the configuration wizard to make sure Public Sector Case Analytics reflects how you prefer to view case management data.

When you start the app creation process, Public Sector Case Analytics opens the configuration wizard. The wizard guides you through the following steps.

1. Checks your org to be sure it meets minimum data requirements and to detect features that can be added to your app. The results let you know if you have to add data or change Salesforce settings to create the app. It also lets you know about available features.

2. Asks you to choose between basic and custom create options. Basic is intended for first-time app users, while custom lets experienced administrators fine-tune app setup.

3. Choose the basic option, and Analytics Cloud creates the app quickly with default settings.

4. Choose custom and the wizard guides you through the steps to fine-tune your app.

   a. Add features to your app

   b. Answer a series of questions about how you use data in your org.

The following provide details about each using each part of the wizard. Read them in the order shown to get the best results when you create Service Analytics. Click the question mark [?] in the top-right corner of each page of the wizard to see help for that page.

1. Public Sector Case Analytics Wizard Data and Feature Check

   At the start of app creation, Public Sector Case Analytics scans your org. It checks to make sure you can create the app and looks for features and data to add to your app.

2. Choose Public Sector Case Analytics App Creation Options

   After Public Sector Case Analytics checks your org's data and features, choose between basics and custom app creation options.

3. Use the Public Sector Case Analytics Basic Create Option

   Select the basic app creation option when you create Public Sector Case Analytics for the first time or when you want to create the app quickly.

4. Use the Public Sector Case Analytics Custom Create Option

   Custom app creation gives you fine-grained control over Public Sector Case Analytics features and data.

5. Case Metrics Questions, Public Sector Case Analytics Custom Wizard Step 4 of 6

   Answer a series of questions to tell Tableau CRM how you prefer to view metrics about your cases.

6. Data Drill Down Questions, Public Sector Case Analytics Custom Wizard Step 5 of 6

   Tell Public Sector Case Analytics how you prefer to drill into data about cases, including category, type, severity, owner, and reason, and if they're resolved at first contact.

7. Questions About Features Added to Public Sector Case Analytics; Custom Wizard Step 6 of 6

   Provide more specifics about how Public Sector Case Analytics uses data from the features you add to the app.

8. Set Up Public Sector Case Analytics CSAT Metrics: Example

   This scenario provides detail to help you answer questions about CSAT data in Step 6 of the Public Sector Case Analytics configuration wizard.

Public Sector Case Analytics Wizard Data and Feature Check

At the start of app creation, Public Sector Case Analytics scans your org. It checks to make sure you can create the app and looks for features and data to add to your app.

Public Sector Case Analytics checks your org for data and features when you open the configuration wizard. If it finds any issues you need to correct before creating the app, you see messages that tell you what to do. Position your cursor over the tooltip for more information.

Here are details about each phase of the check.

Minimum Requirements

Your org must have at least one closed case and one closed task to enable app creation. If you see an error message, close a case and a task and try creating the app again

Features to Add to Your App

The wizard checks to see if you track data about the following in your org:

- Customer satisfaction (CSAT)
- Knowledge
- Case history
- Queues
- Events

- Business hours
- Case record types
- Telephony
- Omni Channel
- Person Accounts

The tooltip [icon] tells you which are available in your org. Basic app creation automatically adds those features. Custom app creation lets you choose the ones to add.

> [icon] **Note:** CSAT and knowledge data can only be added using custom app creation.

Field-Level Security

The wizard checks if the Analytics Integration User has access to all fields needed to create the app using the basic create option. If the Integration User can't access all fields, you see a warning and the tooltip lists the fields that lack access. Go to Salesforce Setup and provide access to those fields. See Set Field Level Security to Enable Creation of an Tableau CRM Template. Until all fields are available, you can't use the basic create option.

Fiscal Calendar Setting

The wizard looks for your org's fiscal year start date and uses it for Service Analytics. To choose a different fiscal start date, use the app's Custom create option.

Case History

The wizard checks to be sure that your org uses history tracking on the Cases object for case status and owner. If it doesn't, follow the directions in the error message and try creating your app again.

Choose Public Sector Case Analytics App Creation Options

After Public Sector Case Analytics checks your org's data and features, choose between basics and custom app creation options.

Select **Basic** if you're using Public Sector Case Analytics for the first time. It sets up your app quickly based on standard settings determined by the org compatibility check. You get an immediately useful version of the app so you can see how it works. Experiment with it and share it with your team. Based on what you learn, change the standard settings by recreating the app using the custom create option.

Public Sector Case Analytics disables basic app creation if the Analytics Integration User can't access all required fields. If the basic option is disabled, click **Back** to return to the org check and follow the instructions in the field-level security tooltip and error message.

Select **Custom** and use the wizard to make custom settings that reflect the way you and your team prefer to view data. The initial, default settings in the screens that follow are the ones used when you create an app using the basic create option. Custom create lets you vary these settings to meet your team's specific needs. You can choose to add or delete features detected by the org check. You can also make specific choices about the data used in the app's dashboards.

Whether you use basic or custom create, Public Sector Case Analytics runs a final scan of your org's field-level security settings. The scan detects if the Analytics Integration User has access to all data fields required to create the app. If the scan fails, you see a message telling you how to fix the issue. For more information about field-level security settings, see Set Field Level Security to Enable Creation of an Tableau CRM Template

Use the Public Sector Case Analytics Basic Create Option

Select the basic app creation option when you create Public Sector Case Analytics for the first time or when you want to create the app quickly.

The basic create option uses data, features, and default settings detected during the compatibility check that runs when you start app creation.

The wizard asks you a single, required question about the field you use to categorize cases. Choose the name of the field your business uses most to identify and organize cases, service, category, type, or level. Then click **Looks good, next**.

Name your app, and click **Create**.

Basic create can't use client satisfaction and knowledge data. Tableau CRM requires more information about how they're used in your org before creating an app. To add them, use the custom create option, and answer the wizard questions about the data.

Basic create uses the following settings:

- To identify cases by geography, the Billing State/Province field from the Accounts object.
- Fifty (50) seconds as the maximum time agents can keep customers waiting during phone calls.

To vary these settings based on how you use Service Cloud, select the custom create option.

Use the Public Sector Case Analytics Custom Create Option

Custom app creation gives you fine-grained control over Public Sector Case Analytics features and data.

Add Data and Features

The first window in the custom creation process lets you add data and features to your app. By default, Public Sector Case Analytics includes data from the following standard Salesforce objects:

- Cases
- User
- UserRole
- Task

The compatibility check that run at the start of app creation looks for other available data and features. The wizard displays the results on the first custom create window, letting you know which features you use in your org. If you use a feature, this window tells you it's available to your app. If you don't use the feature, Tableau CRM tells you it's not available.

Tableau CRM preselects the features used for basic app creation. You have the option of adding others. Preselected options contain a check mark in the upper right corner. Add other available options to your app by clicking them. Click options with a check mark to deselect them. Here are the options Public Sector Case Analytics lets you add, if available, and details about adding each.

- **Customer Satisfaction (CSAT) Score**. Select to add CSAT data to your app. If you select, Tableau CRM adds questions to the wizard that ask how you track CSAT data. See Questions About Features Added to Public Sector Case Analytics; Custom Wizard Step 6 of 6.
- **Knowledge**. Select to add Salesforce knowledge data to your app. If you select, Tableau CRM adds questions to the wizard that ask how you track knowledge data. See Questions About Features Added to Public Sector Case Analytics; Custom Wizard Step 6 of 6.
- **Business Hours**. Select to add a business hours case duration calculation to datasets. This uses data from the Business Hours field in the Cases object. Be sure that the Analytics Integration User has access to this field by editing field-level security for the Cases object.
- **Case History**. Select to include historical tracking for both case owner and for the field selected in wizard Page 5, Question 4. Data Drill Down Questions, Public Sector Case Analytics Custom Wizard Step 5 of 6.

- **Case Record Types**. Select to add record type for the Cases object. If you select, Public Sector Case Analytics lets you filter cases by record type.

- **Queues**. Select to add queues data from the Group object, which your org uses to assign case ownership. If you select, Public Sector Case Analytics lets you filter cases by user and queues.

- **Events**. Select to add data from the Events object, which your org uses to track case activity. If you select, Public Sector Case Analytics includes events data and combines it with tasks data so you can drill into cases by activities (that is, tasks and events).

- **Telephony**. Select to add telephony data from standard call fields in the Tasks object. Standard fields include call duration, call object identifier, call result, and call type. If you select, Public Sector Case Analytics creates the Telephony dashboard showing data from those fields.

- **Omni-Channel**. Select to add data from Omni-Channel, which your org uses to create and route work items. If you select, Public Sector Case Analytics creates the Omni-Channel dashboard and dataset.

- **Person Accounts**. Select to add data from the Person Accounts object, which your org uses to track clients. If you select Person Accounts, Public Sector Case Analytics lets you filter cases by clients. If you don't, the app extracts clients from the Contacts object and augments Accounts with them.

Case Metrics Questions, Public Sector Case Analytics Custom Wizard Step 4 of 6

Answer a series of questions to tell Tableau CRM how you prefer to view metrics about your cases.

> **Note:** This page is labeled *Step 4 of 6*, even though it's the first page of questions you see after you select custom create. This is the fourth of six steps for creating Public Sector Case Analytics with the custom option.

- Question 1, optional: Asks you to indicate the primary field you use in Salesforce to track case duration, if any. If you don't track case duration, Tableau CRM uses its own formula to calculate case duration based on the date a case was opened and closed. If the case is still open, it uses today's date as part of the calculation. Allows only a single selection. If you add business hours to the app, Tableau CRM includes a business hours case duration calculation.

- Question 2, optional: Asks if you use other (secondary) fields related to case duration. You can select multiple fields, for example, **Duration with client** and **Duration with case manager**.

- Question 3, optional: Asks you to select the field you use to track SLA compliance. Typically, you would track SLA compliance with a custom formula field on the Cases object or the standard Milestone Status field. (Milestone Status field values include compliant, open violation, and closed Violation). If you track SLA compliance using a Boolean custom formula field, you can't select that field from the wizard. In that case, do the following: Select any field and then manually edit the dataflow to include the Boolean field. (See Configure the Dataflow Through the Definition File.) Then, find and replace the field name in Public Sector Case Analytics dashboard JSON (see Analytics Cloud Dashboard JSON Reference).

- Question 4, optional: Asks you to select the field you use to record that a case is resolved on first contact. You would typically track first contact resolution using a Boolean custom formula field on the Cases object or the standard Closed when Created (IsClosedOnCreate) field.

- Question 5, optional: Asks if you'd like to include other Cases object metrics in your app. Choose a numeric field.

Data Drill Down Questions, Public Sector Case Analytics Custom Wizard Step 5 of 6

Tell Public Sector Case Analytics how you prefer to drill into data about cases, including category, type, severity, owner, and reason, and if they're resolved at first contact.

> **Note:** This page is labeled *Step 5 of 6*, even though it's the second page of questions you see after you select custom create. This is the fifth of six steps for creating Public Sector Case Analytics with the custom option.

- Question 1, required. Asks about the field you use to categorize cases. Choose the name of the field you use to identify and organize cases, such as service category, type or level.

> **Note:** This is the only question you see if you use basic app creation.

- Question 2, optional. Asks about the field you use to record clients' type of support. Examples include premium and basic.

- Question 3, required. Asks which field you use to categorize case severity—usually the field you use to prioritize cases.

- Question 4, required. Asks which field you use to track case status of case. You filter and analyze cases in dashboards based on your selection.

- Question 5, required. Asks about the field you use to identify topics, reasons, and closure codes of cases. Your selection helps you understand which kinds of cases your team solves most efficiently.

- Question 6, required. Asks you to choose the field you use to track types of cases so you can filter and analyze cases according to support type.

- Question 7, required. Asks you to select the field used to track channels clients use to open cases, such as phone, email, mobile, or web.

- Question 8, required. Asks you to select the object used to track cases according to geography. Options are **Accounts**, **Contacts** (if not using Person Accounts), and **Cases**. Defaults to **Accounts**, which is used in basic create.

- Question 9, required. Asks you to indicate the field in the object selected in Question 8 to track cases by geography. Defaults to **Billing State/Province**, which is used in basic create.

- Question 10, required. Asks you to indicate your fiscal year start month. Defaults to January; select another month if your fiscal year is different from calendar year. Public Sector Case Analytics supports only standard fiscal periods.

- Question 11, optional: Select dimension fields from the Cases object you haven't selected in previous questions. Adding dimensions lets you drill into dimensions that are important to your organization. Dimensions are qualitative values, such as date, region, and product name.

Questions About Features Added to Public Sector Case Analytics; Custom Wizard Step 6 of 6

Provide more specifics about how Public Sector Case Analytics uses data from the features you add to the app.

> **Note:** This page is labeled *Step 6 of 6*, even though it's the third page of questions you see after you select custom create. This is the last of six steps for creating Public Sector Case Analytics with the custom option.

Most of the questions on this page appear only if you add CSAT and knowledge features to your app. Follow the instructions carefully to assure success.

- Question 1, required. You see this question only if you add CSAT to your app. Asks which object you use to track CSAT. Public Sector Case Analytics defaults to **Cases**. If you use a different object to track CSAT, select it from the pick list and answer the additional questions that appear.

- Question 2, required. You see this question only if you add CSAT to your app. Asks you to select the field from the object selected in Question 1 you use to track CSAT.

  - Question 2a, required. You see this question only if you select an object other than Cases in Question 1. Public Sector Case Analytics has to create a join between that object and Cases for you to see CSAT data in dashboards. (For an example, see Set Up Public Sector Case Analytics CSAT Metrics: Example.) Enter the API name the field where you store case identification information. Usually, that's the **Id** field, in which case you enter `Id`.

    If it's a field other than **Id**, find the API name for the field. Go to **Setup—>Cases—>Fields**. In the top chart showing standard fields, field name column shows the API name. For example **IsSelfServiceClosed** is the API name for the **Closed by Self-Service User** field. The chart further down for custom fields includes an API Name column.

  - Question 2b, required. You see this question only if you select an object other than Cases in Question 1. Determine which field from the object selected in Question 1 to use for the other side of the join with the Cases object. Use the field that contains case identification. Enter the API name for the field here.

- Question 3, required. You see this question only if you added knowledge to your app. Asks about the article type you want to see data about in Public Sector Case Analytics dashboards. Choose only a single article type. To see the names of available article types, go to Salesforce Setup and enter `knowledge` in the Quick Find box.

- Question 4, required. You see this question only if you add telephony to your app. Asks you to set the maximum time in seconds case managers can keep clients waiting during a call. Public Sector Case Analytics assesses case manager effectiveness against their ability to meet this requirement.

- Question 5, required. You always see this question in the wizard. Gives you the option of having Public Sector Case Analytics check to see if data in your org supports the answers you select. We recommend that you select **Yes** and run the check, because app creation fails if your org doesn't contain the right data. Public Sector Case Analytics displays an error message that tells you what to fix. Make any required changes in Salesforce, refresh your browser cache, and come back to the wizard and finish creating your app.

SEE ALSO:

Set Up Public Sector Case Analytics CSAT Metrics: Example

Set Up Public Sector Case Analytics CSAT Metrics: Example

This scenario provides detail to help you answer questions about CSAT data in Step 6 of the Public Sector Case Analytics configuration wizard.

If you track customer satisfaction (CSAT) data, such as the CSAT score for a service case, in an object other than Cases, take particular care in how you answer the questions on this page.

For example, if you store the CSAT score in a custom object, for example "CSAT Survey", select this object in question 1. In Question 2, specify the field that has the CSAT metric you would like to report, for example "CSAT Score." Public Sector Case Analytics needs to relate the custom object specified in Question 1 to the Cases object so dashboards can include data about cases, accounts, and agents by CSAT. This relationship is created through a join.

To specify the relationship between the Cases object and your custom CSAT object, designate join keys from both objects and enter their API names in Questions 3 and 4. On the Cases object, this will typically be the "Id" field. On the example custom object "CSAT Survey," this is the custom field that stores the case ID, which in this scenario is the field with the API name `CaseId__c`.

The following diagram shows the join between the two objects.

Use Public Sector Case Analytics Prebuilt Dashboards

After you create Public Sector Case Analytics, navigate to the app and use its dashboards to measure your team's effectiveness and improve services delivery.

Open the app.

1. From the app picker ⋮⋮⋮ , select **Tableau CRM Studio** to open the Tableau CRM home page.

2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

Click **Dashboards** to see a list of all app dashboards. Public Sector Case Analytics prebuilt dashboards contain best practices that help you get value from your Salesforce data—fast. The dashboards let you manage service cases and forecast and understand key business performance drivers, visualize trends, assign actions. They also help you get fast answers to questions you have about your service business's results.

> 📝 Note: The dashboards and datasets included in your instance of Public Sector Case Analytics can differ. The ones you see depend on how you answer configuration wizard questions when you create the app.

Public Sector Case Overview. Start here. Summarizes key performance indicators (KPIs) about your case load, including open cases in the backlog, agent average time to close, and first contact resolution rate. Helps you gauge effectiveness of service delivery and surface issues requiring further investigation. Also, provides springboard to all other dashboards.

Case Changes. Brings your historical case data to life. View cases with the most changes and see their average duration in each status in the resolution process. Also sort cases according to their owner and see how cases move from one status to another.

Case Owner Activity. Helps determine how well case owners use their time. Shows how much work they do and how their work aligns with case-closing numbers and duration as well as CSAT.

Case Owner Performance. Shows case manager team performance against key activity and client satisfaction metrics. Performance trends and benchmarks help case managers provide direction and drive team success.

Case Origin Review. Shows client satisfaction and activities—duration and volume—on cases by where they originated to help you monitor success of each origin.

Case Variances. Gives complete view of your backlog to show how efficiently your team resolves cases and how quickly your backlog is growing. Helps you prioritize and staff your team accordingly.

Client Review. Provides a full breakdown of a selected client's service history and current backlog and includes client satisfaction data and trends. Identify bottlenecks in case lifecycle with insights into average duration spent in various case statuses.

Customer Satisfaction. You only see this dashboard if you add CSAT data to your app. Highlights CSAT measures so you can identify specific areas to improve service. Orders clients from most to least satisfied and associates case managers, products, and channels with satisfaction.

Knowledge Impact. You only see this dashboard if you add knowledge data to your app. Provides a view into how knowledge articles attached to cases impact CSAT and resolution time. See which case managers used the most articles to determine who needs coaching about how to use knowledge to help resolve service issues.

Knowledge Use. You only see this dashboard if you add knowledge data to your app. Helps you understand how case managers use knowledge articles to help you drive article creation. See which articles are attached the most and the least and which have the most views or votes (high ratings) over time.

Omni-Channel. You only see this dashboard if you add Omni-Channel data to your app. Gives managers insight into case manager utilization based on Omni-Channel work record tracking. Includes incoming case manager work volume, average speed to answer, average handle time, average active time, and other agent utilization metrics.

Prioritize Open Cases. Shows current open case workload to help you prioritize, investigate problematic cases, and view escalations and SLA compliance. Let's you prioritize cases by case reason, priority, or any other classification so you can take appropriate action.

Telephony. You only see this dashboard if you add telephony data to your app. Helps you understand the impact of phone contact with client during case resolution. Filter call volume and duration by inbound or outbound call, the result of the call, and other factors. Also relates call volume to case manager performance.

Quoting Analytics Template

Salesforce CPQ customers: Create an app using the Quoting Analytics template to get instant insights into your configure-price-quote (CPQ) processes.
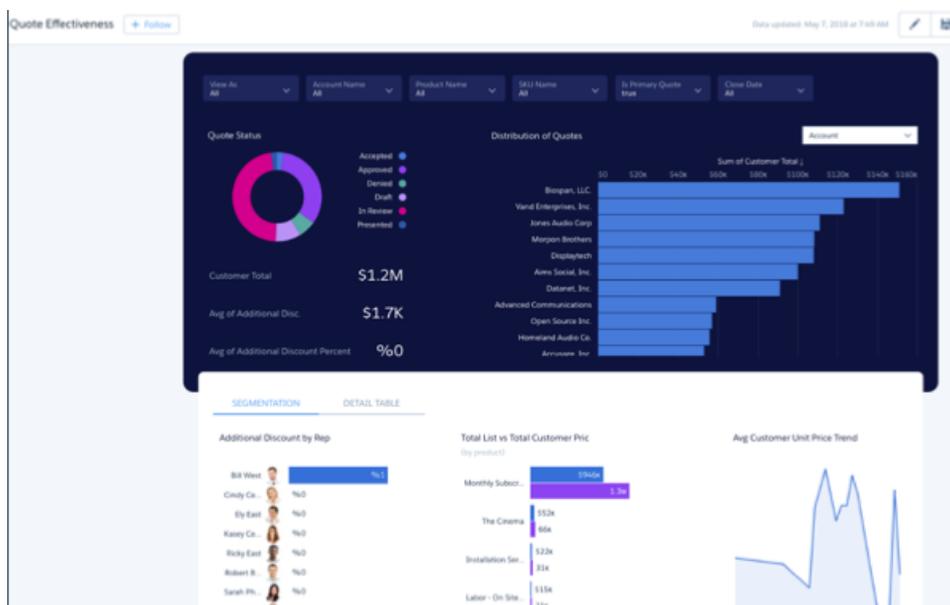
Quoting Analytics gives you a great way to measure your quoting process's effectiveness across teams, regions, and products. Create the app with just a few clicks to get ready-made dashboards that let you view quotes across all your accounts, uncover bottlenecks, and assess reps' discount practices. You can customize them to meet your team's unique requirements. Or quickly create your own explorations using the app's datasets and save them to your own dashboards.

⛔ **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Quoting Analytics app.

Org Requirements

🛑 **Important:** Quoting Analytics can only be used by Salesforce CPQ customers.

Your org requires the following before you can create an app from the Quoting Analytics template:

- Make sure you and all app users have the Tableau CRM Growth and Salesforce CPQ licenses. The Analytics Integration User needs to be assigned the Tableau CRM Growth and the Salesforce CPQ license.
- Your org needs to use at least one Salesforce CPQ Quote record and have quote history enabled.
- Set Salesforce field-level security to enable the Analytics Integration User to see all fields used in the app. See Set Field Level Security to Enable Creation of an Tableau CRM Template.

Sales Analytics

Sales Analytics brings the power of Analytics to Sales Cloud on any device that supports Analytics. With intuitive visualizations based on your Salesforce data, Sales Analytics lets you move from insight to action quickly and helps you turn data into smarter sales.

💡 **Tip:** Follow the steps in the order shown to get started with Sales Analytics. If you haven't used Analytics before, learn more about it from the Analytics Documentation.

1. About Sales Analytics

   Learn the benefits of Sales Analytics before you create and use the app to explore your Sales Cloud data.

2. Sales Analytics Prebuilt Dashboards, Lenses, and Datasets

   Sales Analytics includes prebuilt dashboards, lenses, and datasets to accelerate your data exploration.

3. Set Up Permissions for the Sales Analytics

   Set up your organization to use the Sales Analytics by enabling Tableau CRM and assigning permission sets to users.

4. Create and Share Sales Analytics

   Follow these steps to create Sales Analytics and start uncovering the value of your Salesforce data.

5. Customize Sales Analytics with the Configuration Wizard

   Create Sales Analytics with the configuration wizard so the app reflects how your company prefers to view Salesforce data.

6. Delete Sales Analytics

   Delete apps to start app creation all over or to get rid of apps you no longer use.

7. Upgrade Sales Analytics App

   Take advantage of the latest Sales Analytics features by upgrading your app when we release a new version.

8. Reconfigure Sales Analytics

   To restore deleted or altered dashboards or change wizard settings, reconfigure an existing app.

9. Collaborative Forecasting and Quotas Data in Sales Analytics

   Sales Analytics gives you a choice for how to include your team's quotas depending on whether you use the Sales Cloud Collaborative Forecasts feature to store quotas data.

10. Schedule the Sales Analytics Data Sync and Dataflow

    Schedule a data sync and dataflow to rerun every day to assure that Sales Analytics uses up-to-date data.

11. Integrate Sales Analytics With Salesforce

    Make Sales Analytics more usable by performing a variety of optional integrations and customizations.

12. Understand Sales Analytics Limitations

    Sales Analytics requires that Sales Cloud include specific data and supports a limited set of Salesforce objects.

13. Get to Know Sales Analytics Data Terminology

To make the best use of Sales Analytics, it's helpful to understand the metrics and terms used in the app.

About Sales Analytics

Learn the benefits of Sales Analytics before you create and use the app to explore your Sales Cloud data.
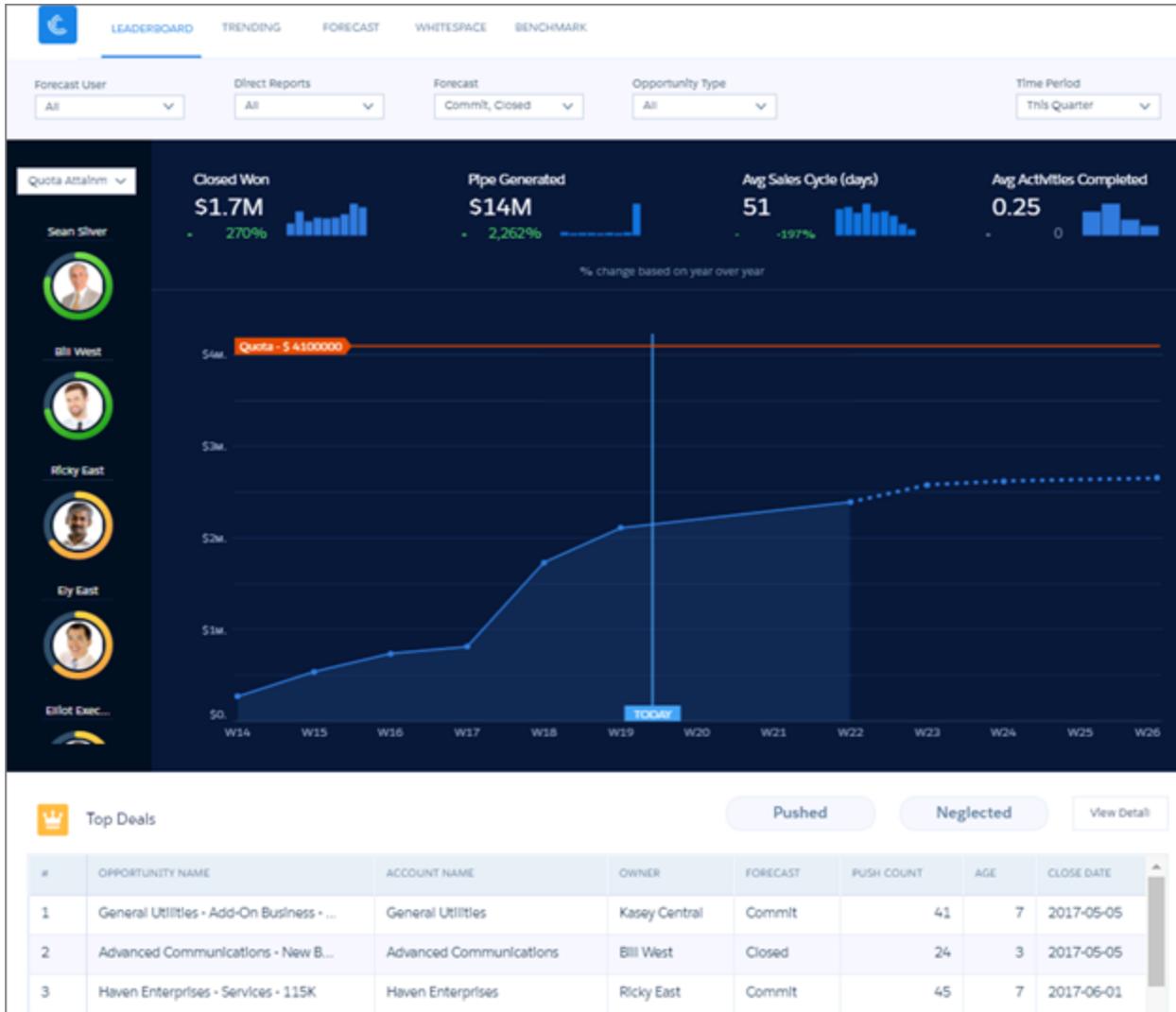
Important: Each Tableau CRM platform license includes a Sales Analytics license. Licenses for individual Tableau CRM apps do not provide the complete capabilities available to Tableau CRM platform license holders. See Understand Sales Analytics Limitations for details.

Salesforce created Sales Analytics to make it easy to analyze Sales Cloud data. Whether you're a sales executive or manager, a rep, or on the operations side, Sales Analytics gives you a new level of insight into your Salesforce data. Start with the Sales Analytics Home dashboard to see the latest wins, win rate, average deal amount. Dig in to all the other dashboards from there, which take advantage of Sales Cloud Collaborative Forecasts to show quotas and help you pinpoint forecasting.

- Execs get their own overview highlighting pipeline and sales and service business performance as well as top deals. They can easily jump to dashboards with all the details.

- Managers can start with a leaderboard showing team key performance indicators (KPIs) and dive right in to specifics about quota, changes in the pipeline, team trending, and average sales cycle time. They can also focus on the performance of members of their team and take action to accelerate business closings.

- Reps can view their own quota attainment, bookings, and pipe activities for a given period. They can also quickly find opportunities for new business to help hit their quotas.

- Operations staff can sort through performance by geography, source, and customer to spot trends to help speed new deals.

App creation is easy. Sales Analytics runs a compatibility check against your org to be sure it includes all the data to create the app's datasets and dashboards. If your org doesn't have the required data, error messages tell you what you need to add before creating the app.

Once the compatibility check determines that your org is ready, you have two options:

- Quickly create the app using standard, default settings as determined by Sales Analytics during the compatibility check. Default settings include collaborative forecasting data to show quotas in app dashboards if your org uses the Sales Cloud Collaborative Forecasting feature.

- Use the built-in configuration wizard to choose your own, custom settings to reflect the way you and others on your team want to view data.

Once you've created the app, use its prebuilt dashboards to explore Sales Cloud data from any device that supports Tableau CRM.

You get actionable insights fast from your sales data using the intuitive Tableau CRM interface. And you can drill deeper into key aspects of your business by customizing Sales Analytics around your business needs.

Sales Analytics Prebuilt Dashboards, Lenses, and Datasets

Sales Analytics includes prebuilt dashboards, lenses, and datasets to accelerate your data exploration.

The prebuilt dashboards in Sales Analytics contain key performance indicators (KPIs) that help you get value from your Salesforce data—fast. The dashboards let you manage pipeline, visualize quotas data, and fine-tune forecasts. They also help you understand key business performance drivers, visualize trends, and assign actions. You get fast answers to questions you have about business results.

The app also includes lenses—more narrowly focused visualizations that augment dashboards with greater detail about your sales KPIs. Additionally, when you create Sales Analytics, the app automatically creates datasets based on your existing Salesforce data in the Sales and Service clouds. Datasets include both standard fields from Salesforce objects as well as custom fields computed from your Salesforce data. The dashboards and lenses use data from those datasets to visualize your KPIs to give you the insights you need to run your business.

You can create your own lenses and dashboards using the data in Sales Analytics datasets. To learn more about exploring data in Tableau CRM, see Explore and Visualize Your Data in Tableau CRM.

> **Note:** The dashboards and datasets included in Sales Analytics differ depending on how you answer configuration wizard questions when you create the app.

Sales Analytics Dashboard Guide
Sales Analytics includes prebuilt dashboards to meet the needs of everyone on the team. Sales executives, managers, and representatives as well as the operations staff all have instant access to the KPIs they need to manage the sales business.

Sales Analytics Lenses
Sales Analytics lenses augment app dashboards with targeted visualizations of your org's sales data.

Sales Analytics Datasets
When you create Sales Analytics , Tableau CRM imports your Salesforce data into app datasets based on your wizard selections. The datasets drive the app's dashboards and lenses.

Sales Analytics Calculated Fields
Sales Analytics adds custom fields computed from your Salesforce data to app datasets. The table lists the fields in alphabetical order and shows how they are computed and the datasets that include them.

Sales Analytics Dashboard Guide

Sales Analytics includes prebuilt dashboards to meet the needs of everyone on the team. Sales executives, managers, and representatives as well as the operations staff all have instant access to the KPIs they need to manage the sales business.

✏️ **Note:** The dashboards included in Sales Analytics differ depending on how you answer configuration wizard questions when you create the app.

General Purpose and Embedded Dashboards

- Sales Analytics Home
- Account (Embedded)
- Opportunity (Embedded)
- Sales Overview Home (Embedded) on page 1466

Dashboards for Sales Managers

- Forecast
- Leaderboard
- Team Activities
- Team Benchmark
- Team Trending
- Team Whitespace

Dashboards for Sales Representatives

- Sales Overview Home (Embedded) on page 1466
- Accounts on page 1482
- Sales Rep Overview
- Trending
- Whitespace (individual)

Dashboards for Sales Executives

- Executive Overview – Pipeline Performance
- Executive Overview – Sales Performance
- Executive Overview – Service Performance

Dashboards for Sales Operations

- Company Overview
- Company Trending
- Lead Analysis
- Sales Performance (Operations)
- Sales Stage Analysis

1. Sales Analytics Home Dashboard

   Start exploring Sales Analytics with an overview of top-level year-to-date key performance indicators (KPIs).

2. Sales Analytics Account Dashboard (Embedded)

   Get a snapshot of an account, including its average sales cycle, opportunity amount, and win rate. The dashboard is intended for use when embedded in a Salesforce account or opportunity page.

3. Sales Analytics Opportunity Dashboard (Embedded)

   Get a summary of an opportunity. You embed the dashboard in a Salesforce opportunity page.

4. Sales Analytics Sales Overview Home (Embedded)

   Designed to be embedded in the Lightning Home page, the dashboard lets sales reps see their KPIs the second they log into Salesforce. They can also quickly access the Sales Analytics dashboards that matter most to them.

5. Sales Analytics Forecast Dashboard

   View quota attainment and forecast for each member of the team, and fine-tune the forecast by reviewing opportunities by stage in a selected period.

6. Sales Analytics Leaderboard Dashboard

   Gives sales leaders an overview of team and individual rep performance, including quota attainment, pipe coverage, bookings, pipe generation, closed won business, average sales cycle time, and sales activities.

7. Sales Analytics Team Activities Dashboard

   See what your team's been doing to keep on top of opportunities. Look at their total, completed, and overdue tasks and review each member's calls, events, emails, and tasks.

8. Sales Analytics Team Benchmark Dashboard

   Shows a ranking of sales rep results across your team's KPIs, such as closed won deals, best case, and average deal size. Also shows each rep's performance against team average.

9. Sales Analytics Trending Dashboards

   View company, team, and pipeline changes for a given period, including beginning and end values of the pipe as well as what's moved in and out.

10. Sales Analytics Team Whitespace Dashboard

    Use this dashboard to uncover resell and upsell opportunities by viewing whitespace. *Whitespace* means accounts that you can revisit to create new business.

11. Sales Analytics Sales Rep Accounts Dashboard

    A snapshot of opportunity, case, and activity trends to help sales reps stay on top of all their accounts.

12. Sales Analytics Sales Rep Overview Dashboard

    Sales reps: start here to see the status of opportunities you own and get ideas for the quickest ways to make quota. Review quota attainment, bookings, and pipeline for a given time period. Also uncover key opportunities for accelerating deal closing and review the number of activities you've completed in the selected period.

13. Sales Analytics Executive Overview – Pipeline Performance Dashboard

    Sales executives get instant pipeline status. You can view the current amount in the pipe, comparison to the previous period, projected closing, and top deals by lead source, product, account, and region. Also get top-level views of sales and service performance.

14. Sales Analytics Executive Overview – Sales Performance Dashboard

    Sales executives get instant insight into closed business and quickly recognize trends. See bookings for the period, changes from previous period, as well as win rate, lead conversion rate, and top deals. Also see top-level views of pipeline and service business.

15. Sales Analytics Executive Service Performance

    Sales executives get instant insight into customer service performance. See average case duration for a selected period compared with previous periods and case duration by account, service channel, and geography.

16. Sales Analytics Company Overview Dashboard

    Intended to give sales leaders and operations staff high in the corporate hierarchy an overview of the sales business and operations.

17. Sales Analytics Lead Analysis Dashboard

    Boost operational efficiency by taking a close look at your team's lead conversions. Quickly identify leads that convert the most quickly and visualize the team's conversion rate. And get a view into top leads by source and region.

18. Sales Analytics Sales Performance Dashboard (Operations)

    Sales leaders: Drill into your closed business by geography, account, product, source, or other criteria to get a 360-degree view of the team's operational efficiency.

19. Sales Analytics Sales Stage Analysis Dashboard

    Shows how deals move through stages of the sales process and if deals are moving smoothly. Also reveals bottlenecks and exposes at-risk opportunities.

Sales Analytics Home Dashboard

Start exploring Sales Analytics with an overview of top-level year-to-date key performance indicators (KPIs).

USER PERMISSIONS

To use Tableau CRM apps:
- Use Analytics Templated Apps

To use Sales Analytics:
- Access Sales Cloud Analytics Templates and Apps

To create and manage Tableau CRM apps:
- Manage Analytics Templated Apps

- Edit Analytics Dataflows
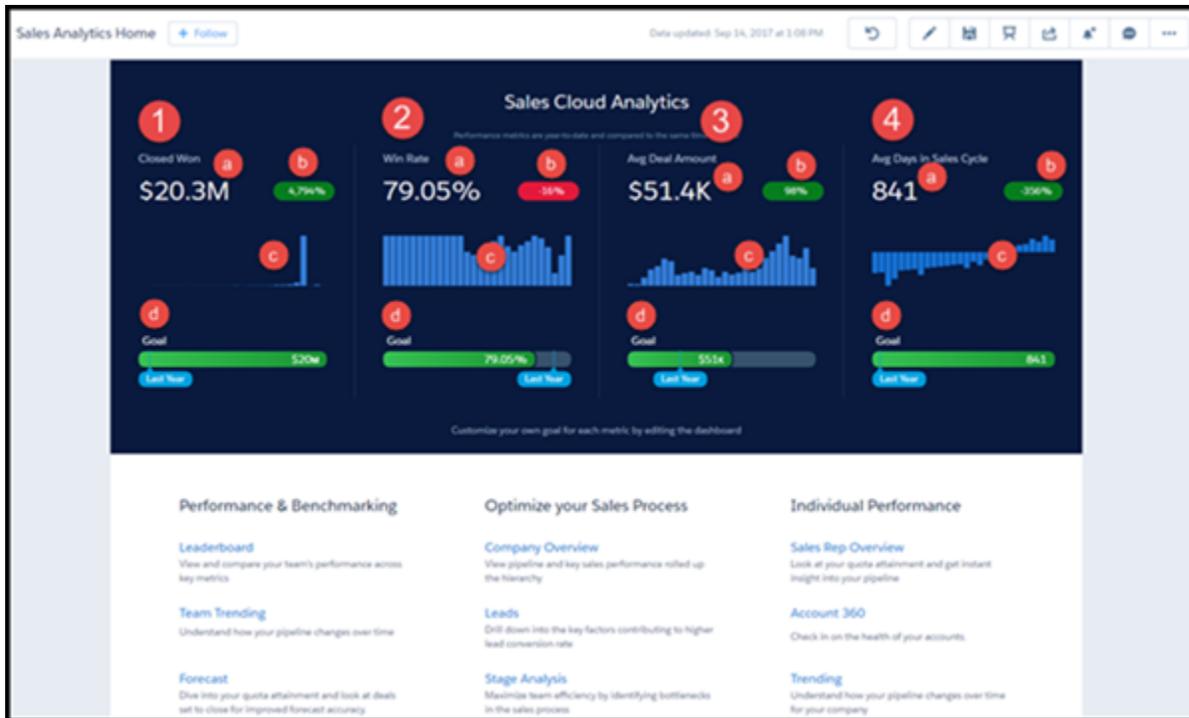
Default Behavior and Recommended Options

Get an overview of your sales business for your current fiscal year, including comparisons to the previous year and the team's progress against goals. Includes closed won, win rate, average deal amount, and average days in the sales cycle. Also, follow links to see all the app's other dashboards and KPIs. All KPIs are based on row visibility settings defined in the Opportunities dataset.

Customize the goal for each metric by editing the dashboard:

1.
   With the dashboard open in Tableau CRM Studio, click the edit icon [pencil icon] to open the designer.

2. Click the Goal widget.

3. In the properties panel on the right, expand **Breakpoints**. Then change **Max** to the goal you'd like to use.

4. Save the dashboard.

Wizard and Other Setup Options

Closed Won (1) and Avg Deal Amount (3) KPIs use amounts from the field selected for opportunity amount, wizard Page 5, Question 1. All other KPIs based on standard fields that can't be changed.

Supports Opportunity Splits if the feature is enabled in your org. See Opportunity Splits.

Datasets Used

* Opportunities.

KPI Calculations

- **Closed Won (1)**

  - Total (a). Total closed won year-to-date (YTD) based on Opportunity.IsWon = true.

  - Comparison (b). (Total closed won YTD - Total closed won same time last year) / Total amount closed won YTD.

  - Over time (c). Total closed won by month for past 3 years.

  - Goal (d). Total closed won this year vs. last year.

- **Win Rate (2)**

  - Total (a). Total won YTD / Total closed won or lost YTD.

  - Comparison (b). (Win rate YTD - Win rate last YTD) / Win rate last YTD.

  - Over time (c). Win rate by month for past 3 years.

  - Goal (d). Total won YTD / Total closed won or lost YTD vs. same for last year.

- **Avg Deal Amount (3)**

  - Total (a). Average amount for won deals YTD.

  - Comparison (b). (Average amount for won deals YTD - Average amount for won deals last YTD) / Average amount for won deals last YTD.

  - Over time (c). Average amount for won deals by month for past 3 years.

  - Goal (d). Average deal size YTD vs. same for last year.

- **Avg Days in Sales Cycle (4)**

  - Total (a). Average won opportunity (Opportunity CloseDate - CreatedDate) for YTD.

  - Comparison (b). (Average cycle YTD - Average cycle last YTD) / Average cycle last YTD.

  - Over time (c). Average days to complete (win) sales cycle each month for past 3 years.

  - Goal (d). Average cycle YTD vs same for last year.

Sales Analytics Account Dashboard (Embedded)

Get a snapshot of an account, including its average sales cycle, opportunity amount, and win rate. The dashboard is intended for use when embedded in a Salesforce account or opportunity page.

Default Behavior and Recommended Options

Embed the dashboard in either Lightning and Aloha page layouts.

Edit the dashboard to set target lines and set the upper bounds for gauges. With the dashboard open in Tableau CRM Studio, click the

edit icon [pencil icon] to open the designer, then click the widget you want to edit. Then look for the value you want to change in the properties panel on the right and change it there. Save the dashboard.

Wizard and Other Setup Options

Average Discount and Breakdown by Product widgets. Only included if you select Products on the wizard page that lets you add objects to the app. Otherwise it includes two activities widgets.

Cases widget. Only included if you select Cases on the wizard page that lets you add objects to the app.

If your org uses account hierarchy, the dashboard can be embedded so that all child account KPIs roll up into KPIs for the parent.

Embedding Syntax

Here's the syntax for embedding the dashboard in either Salesforce Classic or Lightning Experience pages. Note the difference between using a filter for single accounts or account hierarchy. `Opportunity` is the developer name for the opportunity dataset. Use the exact developer name of the dataset when embedding the dashboard.

Salesforce Classic

Embedding the dashboard in the Account page layout.

Here's the syntax to filter by a single account:

```
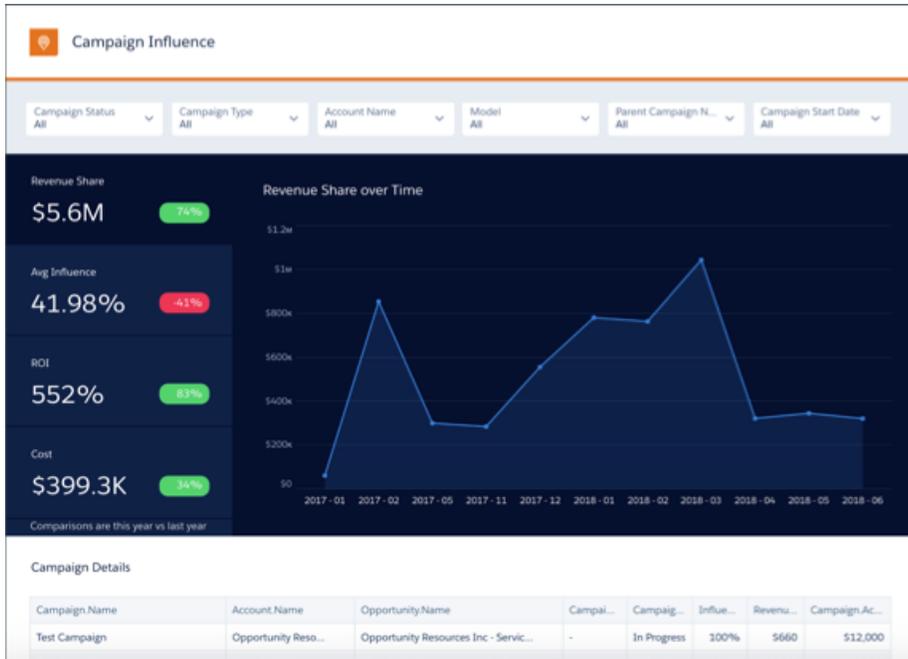{"datasets": { "opportunity": [{"fields": ["AccountId"] ,"filter": { "operator": "in",
"values": ["$Id"] }
          } ]}}
```

Here's syntax to filter using account hierarchy. Roll ups include KPIs for the account and its child accounts:

```
{"datasets": { "opportunity": [{"fields": ["Account.Parent.AccountParents"] ,"filter": {
"operator": "in",
          "values": ["$Id"] } } ]}}
```

Embedding the dashboard in the Opportunity page layout.

Here's the syntax to filter by a single account:

```
{"datasets": { "opportunity": [{"fields": ["AccountId"] ,"filter": { "operator": "in",
"values":
          ["$AccountId"] } } ]}}
```

Here's syntax to filter using account hierarchy. Roll ups include KPIs for the account and its child accounts:

```
{"datasets": { "opportunity": [{"fields": ["Account.Parent.AccountParents"] ,"filter": {
"operator": "in",
          "values": ["$AccountId"] } } ]}}
```

Lightning Experience

📝 Note: Syntax for Lightning Experience uses single quotes instead of double quotes.

Embedding the dashboard in the Account page layout.

Here's the syntax to filter by a single account:

```
{'datasets': { 'opportunity': [{'fields': ['AccountId'] ,'filter': { 'operator': 'in',
'values': ['$Id'] } }
      ]}}
```

Here's syntax to filter using account hierarchy. Roll ups include KPIs for the account and its child accounts:

```
{'datasets': { 'opportunity': [{'fields': ['Account.Parent.AccountParents'] ,'filter': {
'operator': 'in', 'values':
          ['$Id'] } } ]}}
```

Embedding the dashboard in the Opportunity page layout.

Here's the syntax to filter by a single account:

```
{'datasets': { 'opportunity': [{'fields': ['AccountId'] ,'filter': { 'operator': 'in',
'values': ['$AccountId'] } }
        ]}}
```

Here's syntax to filter using account hierarchy. Roll ups include KPIs for the account and its child accounts:

```
{'datasets': { 'opportunity': [{'fields': ['Account.Parent.AccountParents'] ,'filter': {
'operator': 'in', 'values':
          ['$AccountId'] } } ]}}
```

KPI Calculations

Default Widgets

- Average Sales Cycle. Average of opportunity close date - Created date for won opportunities for life of account.
- Win Rate. Total won/ Total closed (won or lost) for life of account.
  - Total Won. Total won for life of account.

- – Total Lost. Total lost for life of account.

- Open Opportunities. Open opportunities with any close date for the account.

  - – Count. Total count of open opportunities with any close date for the account.

  - – Push. Total number of open opportunities for the account that have been pushed at least once.

Optional Widgets

Analytics creates these widgets depending on selections you make on the wizard page that lets you add objects to the app.

- Average Discount. 1 - (Product total price for all products sold to account / Product list price). Only appears if you add Products.

- Product Breakdown. Sum of product total price grouped by product. Only appears if you add Products.

- Number of Open Activities. (Not shown.) Count of open tasks and events related to account. Only appears if you add Products.

- Number of Overdue Activities. (Not shown.) Count of overdue tasks related to account. Only appears if you do not add Products.

- Open Cases. Number of open cases related to account. Only appears if you add Cases.

Sales Analytics Opportunity Dashboard (Embedded)

Get a summary of an opportunity. You embed the dashboard in a Salesforce opportunity page.
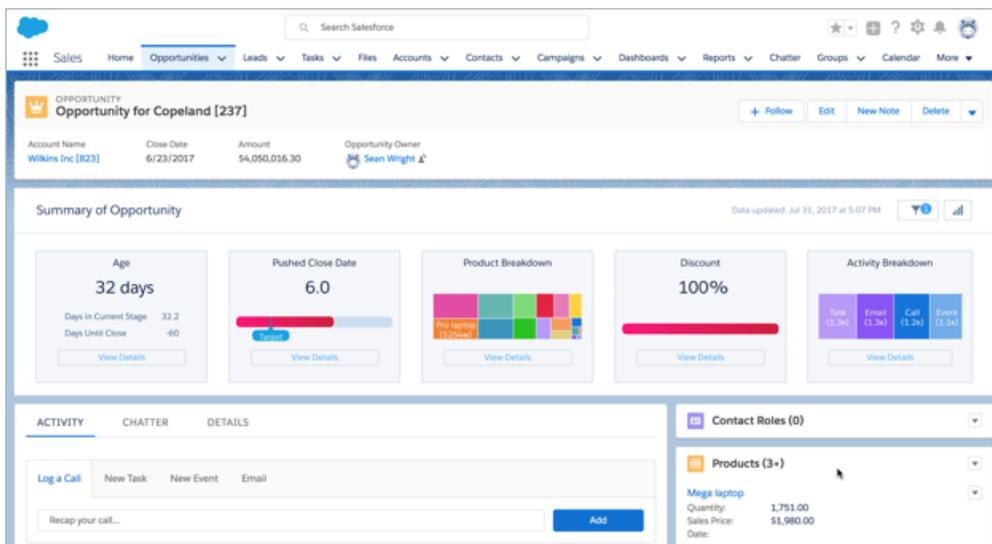
Default Behavior and Recommended Options

📝 Note: This dashboard is only included if Sales Wave is set up to include products. See Wizard and Other Setup Options.

Embed the dashboard in either Lightning and Aloha page layouts.

Edit the dashboard to set target lines and set the upper bounds for gauges. With the dashboard open in Tableau CRM Studio, click the

edit icon [✏️] to open the designer, then click the widget you want to edit. Then look for the value you want to change in the properties panel on the right and change it there. Save the dashboard.

Wizard and Other Setup Options

To add this dashboard to your app, select Products on the wizard page that lets you add objects to the app.

Datasets Used

- Opportunity Products
- Activities

Embedding Syntax

Here's the syntax for embedding the dashboard in either Salesforce Classic or Lightning Experience Opportunity page layout. Note the difference between using a filter for single accounts or account hierarchy. `Opportunity` is the developer name for the opportunity dataset. Use the exact developer name of the dataset when embedding the dashboard.

Salesforce Classic

```
{"datasets": { "opportunity": [{"fields": ["Id"] ,"filter": { "operator":
        "in", "values": ["$Id"] } } ]}}
```

Lightning Experience

📝 Note: Syntax for Lightning Experience uses single quotes instead of double quotes.

```
{'datasets': { 'opportunity': [{'fields': ['Id'] ,'filter': { 'operator': 'in', 'values':
 ['$Id'] } }
        ]}}
```

KPI Calculations

- Age. Number of days opportunity has been open; Today's date - Created date.
- Days in Current Stage. (Not shown.) Number of days the opportunity has spent in the current stage.
- Days until Close. (Not shown.) Expected close date - today's date
- Pushed Close Date. Number of times the close date has been pushed into the future.
- Product Breakdown. Grouping of product names according to product total price.
- Discount. (Product total price for all products attached to the opportunity/ Product list price)
- Activity Breakdown. Number of Activities (open or closed) broken down by activity type.

Sales Analytics Sales Overview Home (Embedded)

Designed to be embedded in the Lightning Home page, the dashboard lets sales reps see their KPIs the second they log into Salesforce. They can also quickly access the Sales Analytics dashboards that matter most to them.
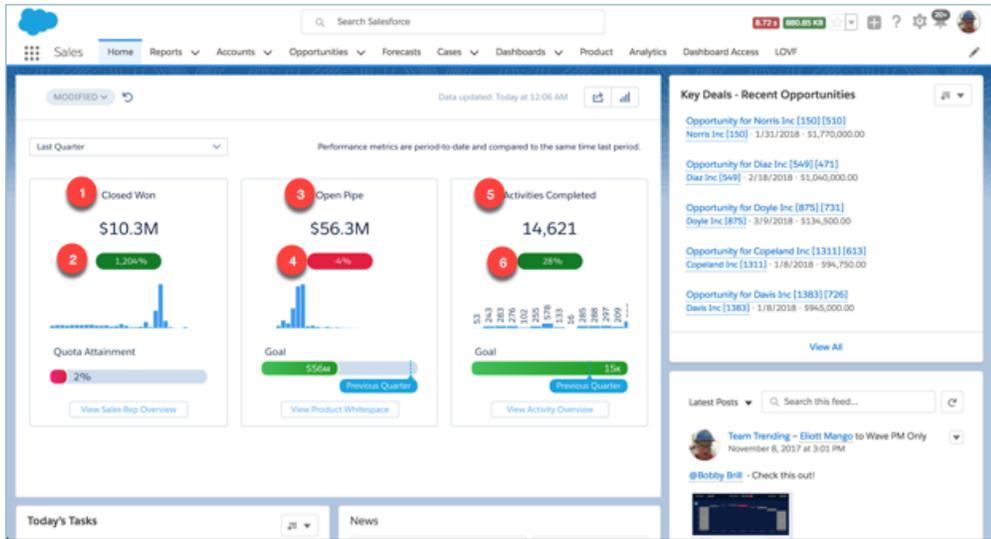
Default Behavior and Recommended Options

Embed the dashboard in the Lightning Home page using the Lightning App Builder without using any filter syntax. By default, data is filtered to show data for the rep viewing the dashboard. You see only opportunities and activities that you own.

Edit the dashboard to set target lines and the upper bounds for gauges. With the dashboard open in Tableau CRM Studio, click the edit

icon to open the designer, then click the widget you want to edit. Look for the value you want to change in the properties panel on the right and change it there. When you're done, save the dashboard.

Click **View Sales Rep Overview**, **View Product Whitespace**, and **View Activity Overview** to open other dashboards.

Wizard and Other Setup Options

The **View Product Whitespace** button appears and links to the Whitespace dashboard only if you select Products on the wizard page that lets you choose to add objects. Otherwise, you see a **View Pipeline Trending** button, which links to the Trending dashboard.

Datasets Used

- Opportunities
- Opportunity Splits (if you add Opportunity Splits to the app on the wizard page that lets you add objects)
- Pipeline Trending
- Activities

KPI Calculations

- Closed Won (1). Total amount closed won opportunities in the selected period.
- Closed Won comparison (2). (Total amount closed won opportunities in the selected period - Total closed won opportunities in the previous period) / Total closed won opportunities in the previous period.
- Open Pipe (3). Total amount of open opportunities with close dates within the selected period.

- Open Pipe comparison (4). (Total open pipe in the selected period - Total open pipe in the previous period) / Total open pipe in the previous period. The amount of open pipe in the previous period is based on data from the same day of the previous period. For example, if the period is *this month*, the amount is for the same day for the previous period minus 1 month.

- Activities Completed (5). Total number of activities completed in the selected period.

- Activities Completed comparison (6). (Total number of activities completed in the selected period - Total number of activities completed in the previous period) / Total number of activities completed in the previous period.

Sales Analytics Forecast Dashboard

View quota attainment and forecast for each member of the team, and fine-tune the forecast by reviewing opportunities by stage in a selected period.
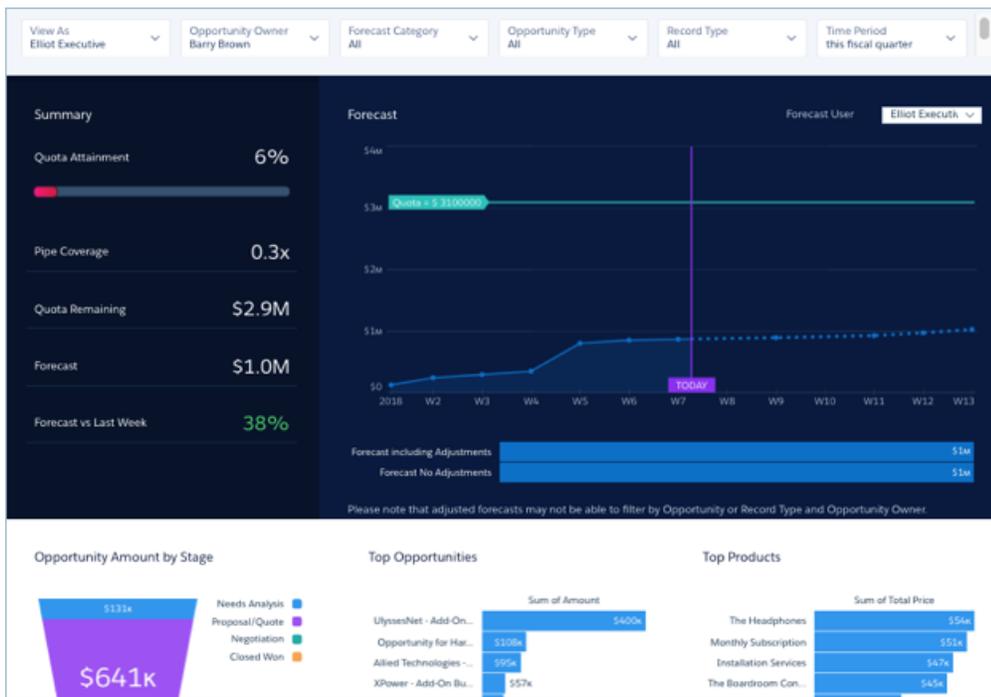
Default Behavior and Recommended Options

If your org uses the Sales Cloud Collaborative Forecasts feature, the dashboard shows opportunity forecast and forecasts with manager adjustments. Historical opportunity data draws from the standard Opportunity Amount field.

Select from the Forecast User filter to view forecast for each team member.

Change the Time Period filter to Next Month or Next Quarter to identify possible opportunities to move into the current period. When you choose other categories from the Forecast Category filter, the Pipe Coverage metric changes accordingly.

Wizard and Other Setup Options

View As filter. If Salesforce is set up to use the Sales Cloud Collaborative Forecasts feature, the filter shows forecast user names. If your org doesn't use Collaborative Forecasts, filter shows role developer names. In either case, pipe rolls up to the selected user. If your org doesn't use Salesforce role hierarchy, this filter does not appear.

Opportunity Type filter. Values shown determined by selection made in wizard page 5, question 3. By default, the filter uses the standard Opportunity Type field.

Quotas. Dashboard automatically includes quota information from Sales Cloud Collaborative Forecasts Quotas if your org uses this feature. If it does not, you can manually upload quotas data into the app from an external tool as a CSV file. See Collaborative Forecasting and Quotas Data in Sales Analytics.

Products. Dashboard includes products if you include products (opportunity line items) when you create the app. Top products are sorted by the product amount selected in the wizard, page 6, question

1.

Opportunity Details chart. Amount column changes to Schedule Amount, Close Date column changes to Schedule Date if you select Product Schedules on page that lets you add data to the app.

Supports Opportunity Splits if the feature is enabled in your org. See Opportunity Splits.

Datasets Used

- Opportunities (or Opportunity Splits)
- Users
- User Allocation
- Forecasting Item (if org uses Collaborative Forecasts)
- Oppty Products (if you set up app with products)
- Pipeline Trending (for Forecast last week comparison)

KPI Calculations

Filters at Top

- View As. Role developer names or forecast managers, depending on how app is set up.
- Opportunity Owner. Any user who has ever owned an opportunity, including inactive users or users who do not own any deals in the selected period.
- Forecast Category. Values from standard opportunity forecast category field.
- Opportunity Type. Values from standard opportunity type field, or another field selected during app setup.
- Record Type. Opportunity record type names defined in your org.
- Time Period. Based on Salesforce fiscal year settings.

Summary (Top Left Panel)

- Quota Attainment. Closed won / Quota amount per opportunity owner based on close date and quota date in selected period. May include future closed opportunities.
- Pipe Coverage. Open pipeline/ Remaining quota per opportunity owner based on selected period. Comparisons are to date. Does not include future closed opportunities.
- Quota Remaining. Quota amount - Closed won based on close date and quota date in selected period.
- Forecast. Closed won + Amount of categories selected in Forecast Category filter. May include opportunities set to close outside of selected period.
- Forecast vs Last Week. Current opportunity forecast amount for categories selected in Forecast Category filter compared with historical opportunity amount for the same categories.

Forecast (Top Right Panel)

- Forecast. Cumulative forecast for categories selected Forecast Category filter. Based on opportunity close date within selected period.
- Forecast No Adjustments/Forecast including adjustments. Total opportunity amount of forecast for categories selected Forecast Category filter compared with adjusted forecast for user selected in Forecast User filter.

Charts at Bottom

- Opportunity Amount by Stage. Opportunities grouped by stage based on close dates within selected period and other values selected in filters.
- Top Opportunities. Total amount open for selected opportunity owner based on close date in selected period.
- Top Products. Top products based on total product amount for all opportunities in forecast.

Sales Analytics Leaderboard Dashboard

Gives sales leaders an overview of team and individual rep performance, including quota attainment, pipe coverage, bookings, pipe generation, closed won business, average sales cycle time, and sales activities.
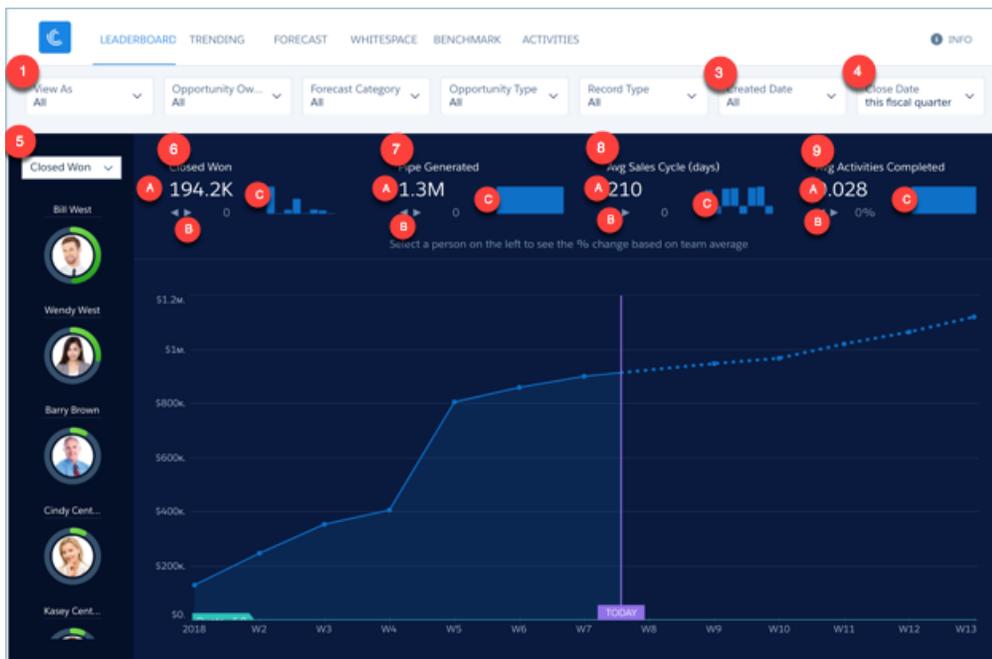
Default Behavior and Recommended Options

The Leaderboard gives a quick overview of team performance for a given time period. View results for the entire team or for any of your subordinates. The team is made up of sales reps that roll up to the user selected in the **View As** filter (1), depending on the configuration of the app. (See below under Wizard and Other Setup Options.)

Select a key performance indicator (KPI) at the top of the left column (2) to view a stacked ranking of your team according to that KPI. Examples of KPIs include the following:

- Quota progress.
- Pipe coverage.
- Open pipe.
- Pipe generated.
- Remaining quota.
- Closed won.
- Completed activities.
- Average sales cycle.

Change the **View As** filter (1) and view KPIs from that user's perspective.

The leaderboard column on the left (5) displays opportunity owners only. It does not roll up metrics up the role hierarchy; the Company Overview dashboard provides that view.

Wizard and Other Setup Options

Quotas. Dashboard automatically includes quota information from Sales Cloud Collaborative Forecasts Quotas if your org uses this feature. If it does not, you can manually upload quotas data into the app from an external tool as a CSV file. See Collaborative Forecasting and Quotas Data in Sales Analytics on page 1526.

Opportunity Type filter. Values shown determined by selection made in wizard page 5, question 3. By default, the filter uses the standard Opportunity Type field.

What field indicates that an opportunity is new business? *
Select the field you use to identify that an opportunity in Salesforce is new business.

Opportunity Type

Close Date filter. Changes to schedule date if you select Product Schedules on page that lets you add data to the app.

View As filter . Contents of this filter depend on your org:

- If your org uses the Sales Cloud Collaborative Forecasts feature, the filter lists forecasting managers as defined by the forecast hierarchy.
- If your org defines a role structure/hierarchy but doesn't use Collaborative Forecasts, the filter lists manager roles as defined by the hierarchy.
- If your org does not define role hierarchy, the filter lists opportunity owners.

Top Deals details chart. Amount column changes to Schedule Amount, Close Date column changes to Schedule Date if you select Product Schedules on page that lets you add data to the app.

Closed won and quota amounts calculated based on the amount field selected in the wizard, Page 5, Question

1.
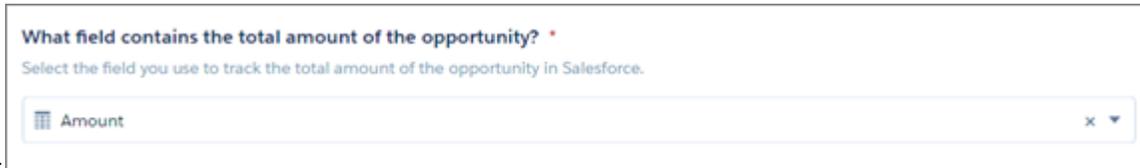
What field contains the total amount of the opportunity? *

Select the field you use to track the total amount of the opportunity in Salesforce.

Amount × ▼

Supports Opportunity Splits if the feature is enabled in your org. See Opportunity Splits.

Datasets Used

- Opportunity
- User Allocation
- Activities
- User
- Opportunity Splits (if you select Opportunity Splits on the wizard page that lets you add objects)

KPI Calculations

- **Top of Chart**
  - Closed Won (6a). Total amount closed won during selected time period. May include future closed opportunities.
  - Closed Won comparison (6b). (Total amount closed based on selected user in **View As** filter - Average total amount closed by team) / Average total amount closed by team. *Team* defined by all opportunity owners.
  - Closed Won over time (6c). Total amount closed won by quarter for past eight quarters
  - Pipe Generated (7a). Based on selections from **Created Date** (3) and **Close Date** (4) filters. For example, to see what pipe is generated this period and closes any time, set **Created Date** (3) to **this fiscal quarter** and **Close Date** (4) to **all time**.
  - Pipe Generated comparison (7b). (Total pipe created by selected user in **View As** filter - Average total pipe created by team) / Average total pipe created by team.
  - Pipe Generated over time (7c). Win rate by month for past three years.
  - Avg Days in Sales Cycle (8a). Average of opportunity close date - created date for won opportunities year to date.
  - Avg Days in Sales Cycle comparison (8b). (Average cycle for selected user in View As filter - Average cycle by team) / Average cycle by team..
  - Avg Days in Sales Cycle over time (8c). Average days to complete (win) sales cycle in the selected period
  - Avg Activities Completed (9a). Average number of activities completed per team member in selected time period. Team made up of opportunity owners included in role selected in **View As** filter (1).
  - Avg Activities comparison (9b). (Average cycle for selected user in View As filter - Average cycle by team) / Average cycle by team.
  - Avg Activities over time (9c). Activities completed per quarter.

- **Upper-Left Filter Options (5)**
  - Closed Won. Total amount closed won as of close date in selected period per opportunity owner.
  - Open Pipe. Total amount open as of close date in selected period per opportunity owner.
  - Quota Attainment. Closed won / Quota amount per opportunity owner as of close date and quota date in selected period.
  - Remaining Quota. Quota amount - Closed won as of close date and quota date in selected period.
  - Pipe Generated. Total amount of opportunities created since created date in selected period per opportunity owner.

- Avg Sales Cycle. Average sales cycle since created date in selected time period per opportunity owner.

- Activities Completed. Number of activities completed in the selected time period per opportunity owner.

- Pipe Coverage. Open pipeline / Remaining quota per opportunity owner in selected period. Values selected in Forecast Category filter impact open pipeline calculation.

- **Top Deals (open opportunities) table (not pictured)**

  - **Pushed**. Number of times an opportunity close date has been pushed to a future date. Only shows opportunities in the table for which the closed date has been pushed into the future at least once.

  - **Neglected**. Opportunities that have not been touched for 60 days. Time period can be changed by editing app dataflow.

Sales Analytics Team Activities Dashboard

See what your team's been doing to keep on top of opportunities. Look at their total, completed, and overdue tasks and review each member's calls, events, emails, and tasks.
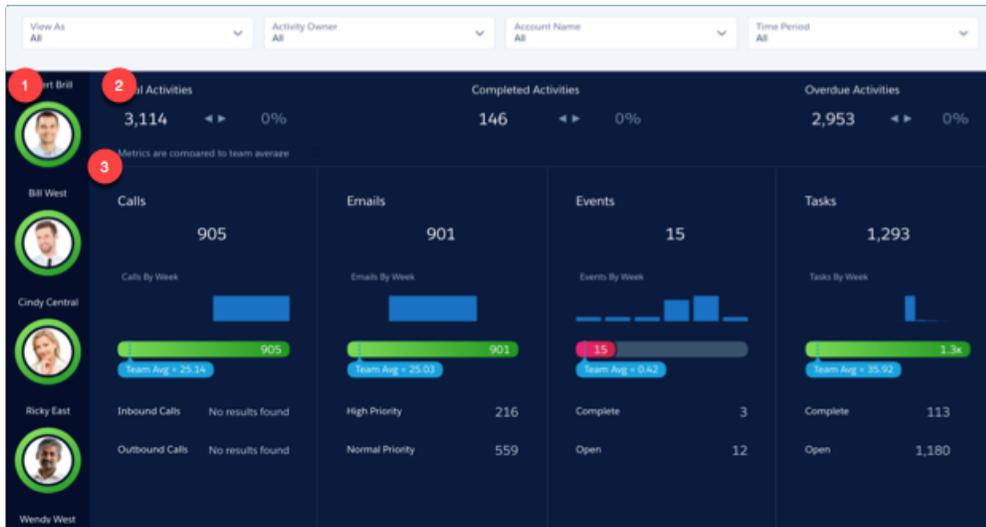
📝 Note: **Multi Org Sales Analytics Version Only**. Use the **Data Source**menu to select the org data to view in the dashboard.
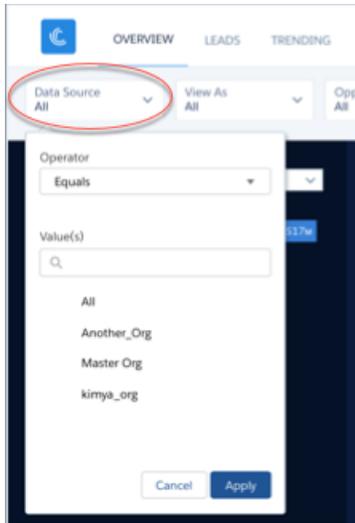
Default Behavior and Recommended Options

Measure team and individual activity based on Salesforce tasks and events data. Start by looking at total, completed, and overdue tasks for your entire team. Select team members in the panel on the left (1) to see their individual activities and review their calls, events, emails, and tasks. Also see how their level of activity compares to the team average.

📝 Note: The dashboard does not yet support Einstein Activity Capture.

Wizard and Other Setup Options

View As filter. If Salesforce is set up to use the Sales Cloud Collaborative Forecasts feature, the filter shows forecast user names. If your org doesn't use Collaborative Forecasts, filter shows role developer names. In either case, pipe rolls up to the selected user. If your org doesn't use Salesforce role hierarchy, this filter does not appear.

Datasets Used

Activities.

KPI Calculations

Top Section (2)

- Total Activities. Total number of activities with due date in the selected period.
- Total Activities comparison. (Total activities for selected user - Total activities) / Total activities.
- Completed Activities. Total number of activities marked as complete based on due date in the selected period. Events are marked as complete if the event date is in the past.
- Completed Activities comparison. Completed activities for the selected user - Completed activities total / Completed activities total.
- Overdue Activities. Total count of any activity which is overdue. Includes tasks not marked as complete with due dates are in the past.

Middle and Lower Sections (3)

- Calls. Number of tasks of type *call* with due date in selected period.

- Calls by Week. Number of calls per week with due date in selected period.

- Calls vs. Team Avg. Number of calls for selected user compared with average number for team. Team is made up of activity owners who roll up to user selected in **View As** filter or user selected in **Activity Owner** filter.

- Inbound Calls. Number of inbound calls based on Call Type field in the standard Tasks object

- Outbound Calls. Number of outbound calls based on Call Type field in the standard Tasks object.

- Emails. Number of tasks of type *email* with due date in selected period.

- Emails by Week. Number of emails per week with due date in selected period.

- Emails vs. Team Avg. Number of emails for selected user compared with average number for team. Team is made up of activity owners who roll up to user selected in **View As** filter or user selected in **Activity Owner** filter.

- High Priority emails. Number of high-priority emails based on standard Tasks object Priority field.

- Normal Priority emails. Number of normal emails based on standard Tasks object Priority field.

- Events. Number of events with due date in selected period.

- Events by Week. Number of events per week with due date in selected period.

- Events vs Team Avg. Number of events for selected user compared with average number for team. Team is made up of activity owners who roll up to user selected in **View As** filter or user selected in **Activity Owner** filter.

- Complete events. Number of events marked closed.

- Open events. Number of future events within the selected period.

- Tasks. Number of open tasks due in the selected period, including overdue tasks.

- Tasks by Week. Number of tasks per week with due date in selected period.

- Tasks vs. Team Avg. Number of tasks for selected user compared with average number for team. Team is made up of activity owners who roll up to user selected in **View As** filter or user selected in **Activity Owner** filter.

- Complete tasks. Number of tasks marked closed.

- Open tasks. Number of future events in the selected period.

Sales Analytics Team Benchmark Dashboard

Shows a ranking of sales rep results across your team's KPIs, such as closed won deals, best case, and average deal size. Also shows each rep's performance against team average.
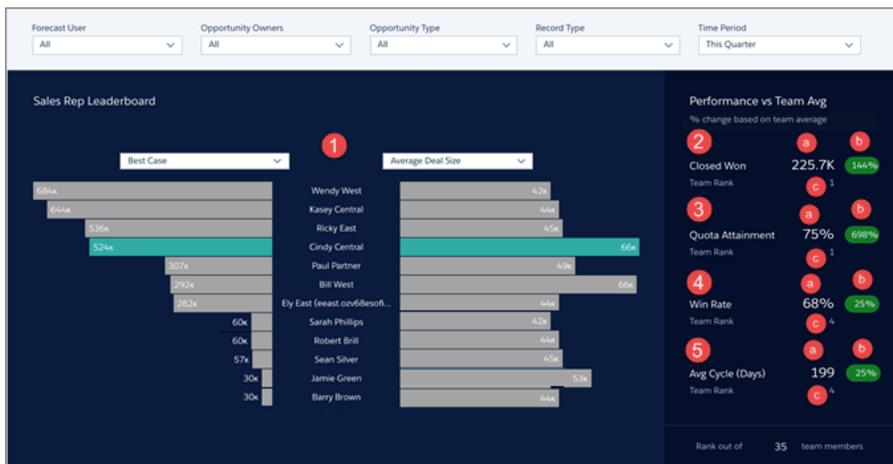
Default Behavior and Recommended Options

Select two KPIs in the main (bar) chart (1) to compare one measure of performance against another. Both selectors contain the same KPIs so you can compare whatever is on the left with whatever is on the right. The rank order of your team sorts according to the measure selected on the left.

Select a specific rep to view their scorecards in the right column, which also shows the rep's team rank. Team is defined by all the reps that show up in the bar chart, depending on what you select in the Forecast User/Manager Role filter (**Forecast User** in image).

Wizard and Other Setup Options

Forecast User/Manager Role filter (*Forecast User* in image).If Salesforce is set up to use the Sales Cloud Collaborative Forecasts feature, the filter shows forecast user names. If your org doesn't use Collaborative Forecasts, filter shows role developer names. In either case, pipe rolls up to the selected user. If your org doesn't use Salesforce role hierarchy, this filter does not appear.

Opportunity Type filter. Values shown determined by selection made in wizard page 5, question 3. By default, the filter uses the standard Opportunity Type field.
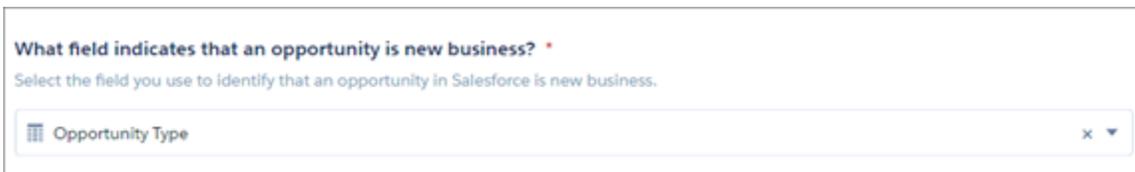
Time Period filter. All time periods based on Salesforce fiscal year setting.

Supports Opportunity Splits if the feature is enabled in your org. See Opportunity Splits.

Datasets Used

- Opportunities (or Opportunity Splits)
- User Manager
- User Allocation
- Activities

KPI Calculations

All KPIs appear in the right-hand column.

- Closed Won (2a). Total amount closed won based on selected time period, which may include future opportunities.
- Closed Won Comparison (2b). (Total amount closed won for selected time period to date [for owner selected in main chart (1)] - Average total amount closed won for team during same period) / Average total amount closed won for all team members during the same period. Comparative value appears only if you select a sales rep in main chart (1), otherwise you see 0.
- Closed Won Team Rank (2c). Rank of rep selected in the main chart (1) based on closed won business in the selected time period. Rank based on opportunity owners who roll up to the manager selected in **Forecast User** filter as well as the owners selected in **Opportunity Owner** filter. Rank appears only if you select a sales rep in main chart (1), otherwise you see a 1.
- Quota Attainment (3a). Closed Won / Quota Amount based on selected time period, which may include future opportunities.
- Quota Attainment Comparison (3b). (Quota attainment for rep selected in main chart (1) - Quota attainment for team) / Quota attainment for team. Comparative value appears only if you select a sales rep in main chart (1), otherwise you see 0.

- Quota Attainment Team Rank (3c). Rank of rep selected in main chart (1) based on quota attainment in the selected time period. Rank based on opportunity owners who roll up to manager selected in **Forecast User** filter as well as the owners selected in **Opportunity Owner** filter. Rank appears only if you select a sales rep in main chart (1), otherwise you see a 1.

- Win Rate (4a). Total amount won / Total amount closed (Won or Lost) based on selected time period, which may include future opportunities.

- Win Rate Comparison (4b). (Win rate for the owner selected in main chart (1) - Win rate for team) / Win rate for team. Comparative value appears only if you select a sales rep in main chart (1), otherwise you see 0.

- Win Rate Team Rank (4c). Rank of rep selected in main chart (1) based on their win rate in the selected time period. Rank based on opportunity owners who roll up to the manager selected in **Forecast User** filter as well as the owners selected in **Opportunity Owner** filter. Rank appears only if you select a sales rep in main chart (1), otherwise you see a 1.

- Avg Sales Cycle (5a). Number of days from CreatedDate in the selected time period, which may include future opportunities.

- Avg Sales Cycle Comparison (5b). (Sales cycle for rep selected in main chart (1) - Average sales cycle for team) / Average sales cycle for team. Comparative value appears only if you select a sales rep in main chart (1), otherwise you see 0.

- Avg Sales Cycle Team Rank (5c). Rank of rep selected in main chart (1) based on sales cycle in the selected time period. Rep with shortest sales cycle receives top ranking. Rank appears only if you select a sales rep in main chart (1), otherwise you see a 1.

Sales Analytics Trending Dashboards

View company, team, and pipeline changes for a given period, including beginning and end values of the pipe as well as what's moved in and out.
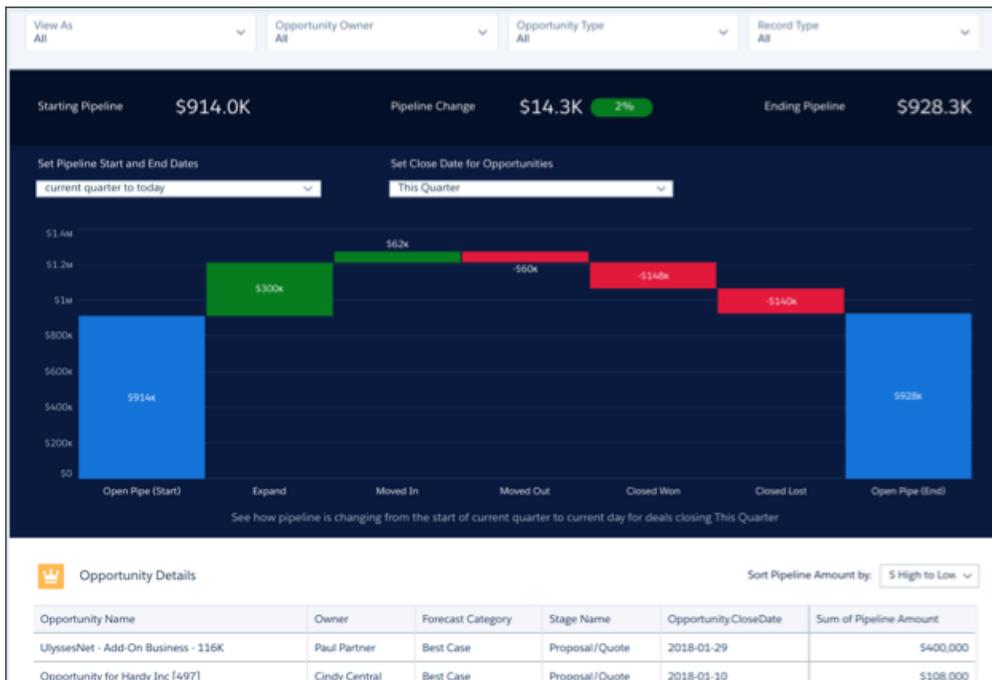
> Note: **Multi Org Sales Analytics Version Only**. Use the **Data Source** menu to select the org data to view in the dashboard.
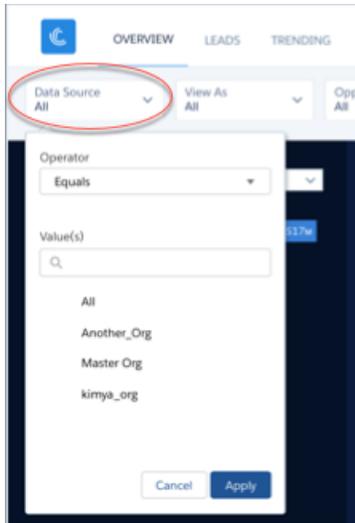
📝 **Note:** The following pertains to Team Trending (shown), Company Trending, and Trending (individual) dashboards, except as noted.

Default Behavior and Recommended Options

Dashboards show the pipeline for all opportunity owners rolling up to the manager using the app. Depending on which dashboard you view, choose another user to change opportunity owners reflected in the chart.

- Team Trending (shown): Change opportunity owner using the **Forecast User/Manager Role/Opportunity Owner** filter (*Forecast User* in image). Filter name changes depending on how your org is set up. See Wizard and Other Setup Options.
- Company Trending: Change opportunity owner using the **View As** filter.
- Trending (individual): Change opportunity owner using the **Opportunity Owner** filter.

The dashboard supports only fields within the opportunity history table. Even if you select a custom amount field in the wizard, the dashboard contains only the standard opportunity amount field. Filters at the top use only current opportunity values; they don't reflect historical values. For example, if the owner of the opportunity has changed, historical data does not reflect the change.

The dashboard does not support opportunity splits.

Wizard and Other Setup Options

View As filter . Contents of this filter depend on your org:

- If your org uses the Sales Cloud Collaborative Forecasts feature, the filter lists forecasting managers as defined by the forecast hierarchy.
- If your org defines a role structure/hierarchy but doesn't use Collaborative Forecasts, the filter lists manager roles as defined by the hierarchy.
- If your org does not define role hierarchy, the filter lists opportunity owners.

📝 **Note:** Pertains only to Team Trending dashboard.

Opportunity Type filter. Values shown determined by selection made in wizard page 5, question 3. By default, the filter uses the standard Opportunity Type field.

What field indicates that an opportunity is new business? *

Select the field you use to identify that an opportunity in Salesforce is new business.

▦ Opportunity Type   ✕   ▾

Datasets Used

Pipeline Trending, including data from Opportunity History.

KPI Calculations

Time period filters

Time periods based on Salesforce fiscal year setting.

- **Set Pipeline Start and End Dates.** Select the start and end dates for when you'd like to view the pipeline. Click **Year**, **Quarter**, **Month**, **Weeks**, or **Days** to indicate unit of time. Then either click edit icon and type in a number or drag slider handles to set the start and end for the period you'd like to view. For example, click **Quarters**, then drag left handle to **-3** and drag right handle to **+1** to view pipeline starting three quarters ago until one quarter from now.

- **Set Close Date for Opportunities.** Select the close date for the opportunities you'd like to include in your view of the pipeline.

Main (Waterfall) Chart

- *Values along top*

    - Starting Pipeline. Total amount open during the selected starting period.

    - Pipeline Change. (Total amount open during selected starting period - total amount open during selected end period/ Total amount open during selected starting period). Does not include future closed opportunities.

    - Ending Pipeline. Total pipeline at the end of the period.

- *Values in vertical bars*

    - Open Pipe (Start). Total pipeline at the start of the time period.

    - New. Opportunities created after the start period.

    - Expand. Existing opportunities in which total amount was increased since start of the period.

    - Moved In. Opportunities with close dates moved into the selected time period. (Where close date was changed after start of the period.)

    - Moved Out. Opportunities with a close date that no longer falls within the selected time period. (Where close date was changed after start of the period.)

    - Reduce. Opportunities in which total amount is less than at the beginning of the period.

    - Closed Won. Opportunities closed won after the beginning of the period.

    - Closed Lost. Opportunities lost after the beginning of the period.

    - Open Pipe (Today). Total pipeline at the end of the period.

Sales Analytics Team Whitespace Dashboard

Use this dashboard to uncover resell and upsell opportunities by viewing whitespace. *Whitespace* means accounts that you can revisit to create new business.
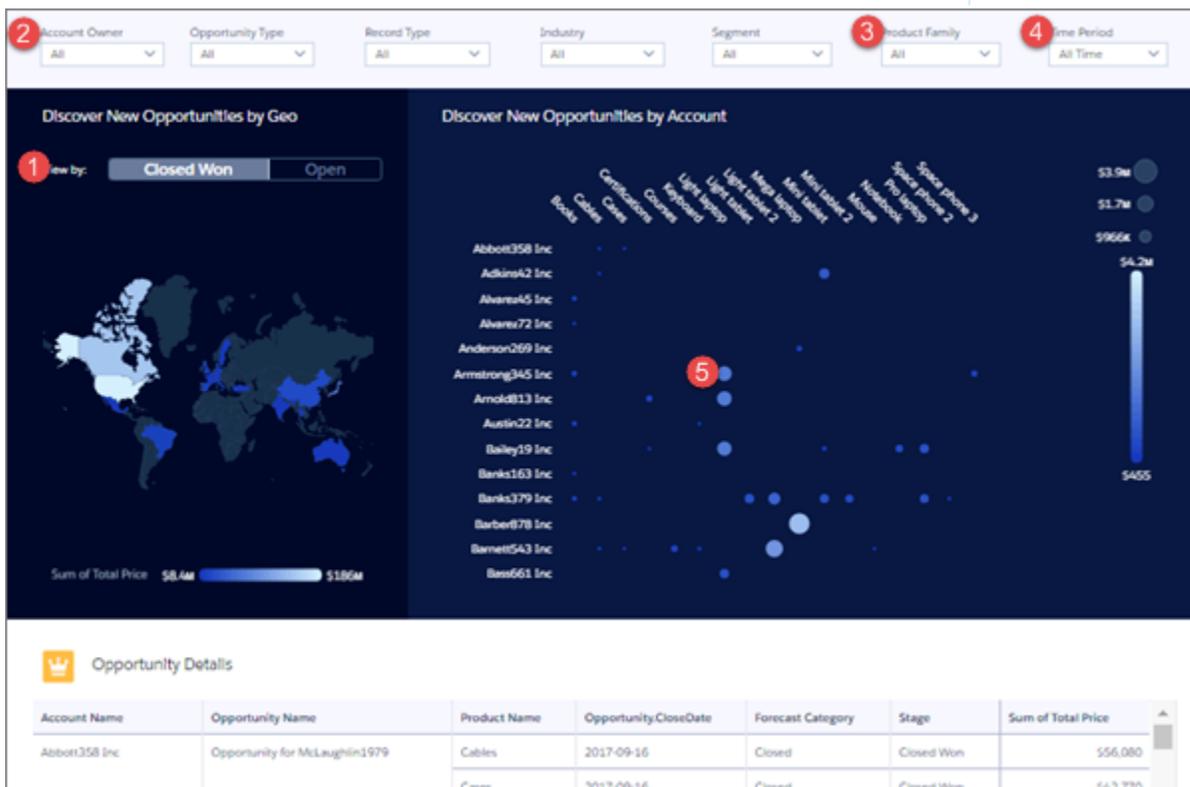
📝 **Note:** The following pertains to Team Whitespace (shown) and Whitespace (individual) dashboards, except as noted.

Default Behavior and Recommended Options

Start by viewing products your team or individual reps have sold by account and what's in the pipeline. You can also see accounts not associated with products and products that have not been sold to any account. Use filters to group similar accounts to spot product whitespace.

Team version shows accounts where you and subordinate members of your team are named owners or other accounts that you can see based on security settings in your org.

Opens to all closed won and open deals for a given time period for the entire team or an individual rep. Click **Closed Won** (shown in image) or **Open** in the **View by** toggle (1) to see just those deals. Select a sales rep's name from the **Account Owner** filter (2) to view deals for that rep. Change the **Product Family** (3) or **Time Period** (4) filters to narrow time period and specific products.

Position cursor over a bubble (5) to see the size of a deal and which products are involved. Bubble sizes increase according to the amount of the deal. Click the bubble to open the details, then take action from the opportunity to recommend steps to pursue a new opportunity.

Wizard and Other Setup Options

Sales Analytics includes this dashboard only if you select Products on the wizard page that lets you add objects to your app.
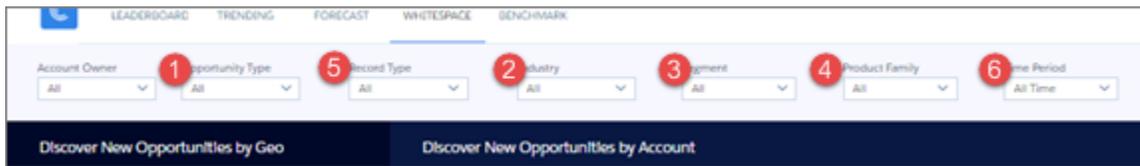
Wizard selections made when you create the app determine the labels and values in the filters numbered in the image. Select basic create or accept default values in custom create to use labels and values from standard Salesforce fields. Select other fields from the following wizard questions to customize filters:

1. **Opportunity Type** in image. Wizard page 5, question 3. Product Family in image. Wizard page 6, question 2.

2. **Industry** in image. Wizard page 4, question 2.

3. **Segment** in image. Wizard page 4, question 1.

4. **Product Family** in image. Wizard page 6, question 2. Only available if you add products to the app on the wizard page that lets you add objects.

Record Type filter (5). Only appears if your org uses record types.

Time Period filter (6). All time periods based on Salesforce fiscal year setting.

Opportunity Details chart. Amount column changes to Schedule Amount, Close Date column changes to Schedule Date if you select Product Schedules on page that lets you add data to the app.

Datasets Used

- Accounts
- Products
- Oppty Products

Combination of these three datasets lets you view accounts with or without products and products with or without opportunities with accounts.

KPI Calculations

Main Chart

Shows product total price by account for the selected period for either closed won, open, or both depending on selected filters. Row visibility settings from Opportunities dataset determine which accounts appear in the chart.

Sales Analytics Sales Rep Accounts Dashboard

A snapshot of opportunity, case, and activity trends to help sales reps stay on top of all their accounts.
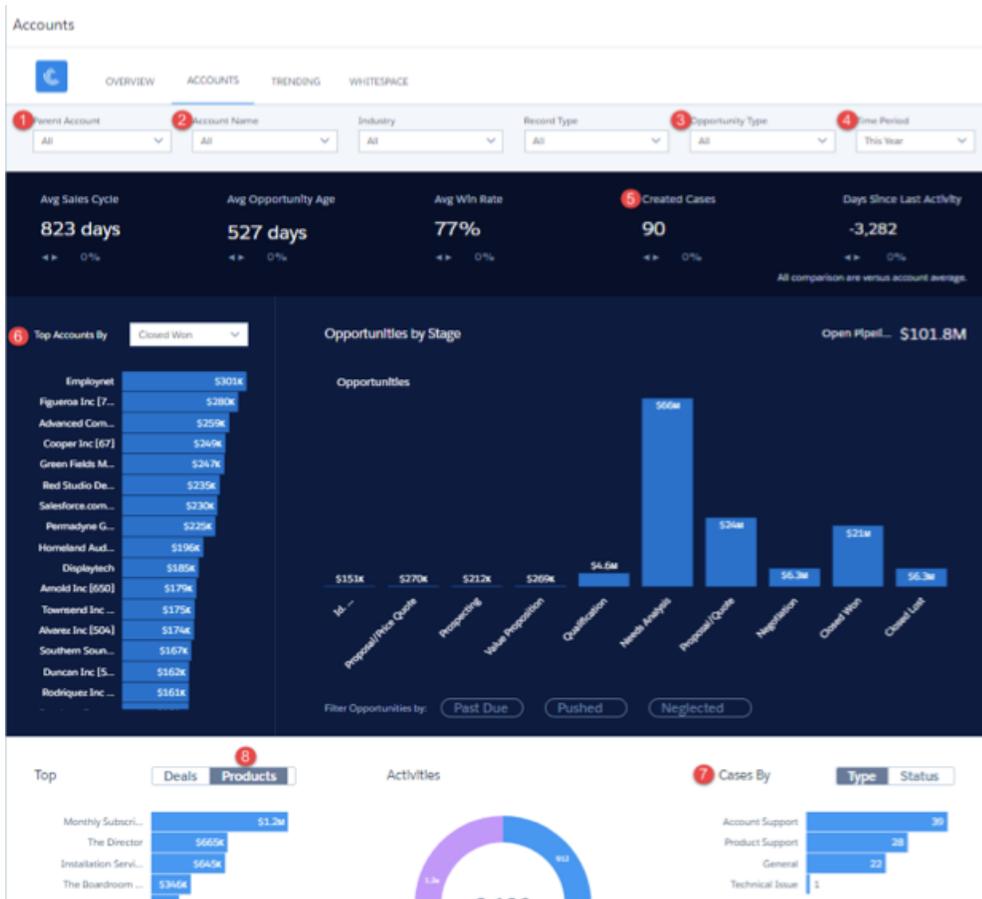
Note: **Multi Org Sales Analytics Version Only**. Use the **Data Source** menu to select the org data to view in the dashboard.
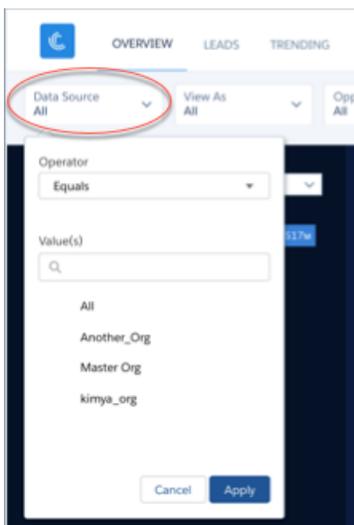
Default Behavior and Recommended Options

View data for a specific account by selecting from the **Parent Account** (1) or **Account Name** (2) filters along the top or the **Top Account** widget (6) on the left. The dashboard shows all accounts and related opportunities visible to the person running the app. Percentages at top of main chart compare metrics for the account selected from **Top Accounts** (6, at left) with totals of all other accounts, based on selections in filters along top.

Wizard and Other Setup Options

Parent Account filter. If you use the standard Parent Account field on the Accounts object, Sales Analytics includes this filter. It's populated with the account at the top of the hierarchy.

Opportunity Type filter. Values shown determined by selection made in wizard page 5, question 3. By default, the filter uses the standard Opportunity Type field.
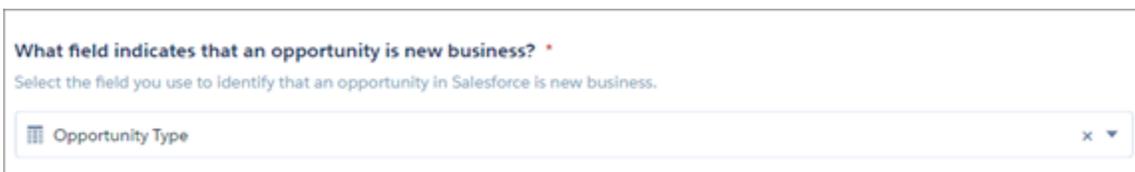
Time Period filter. All time periods based on Salesforce fiscal year setting.

Cases widgets. Only included if you use the Cases object in Salesforce and you select Cases on the wizard page that lets you add objects
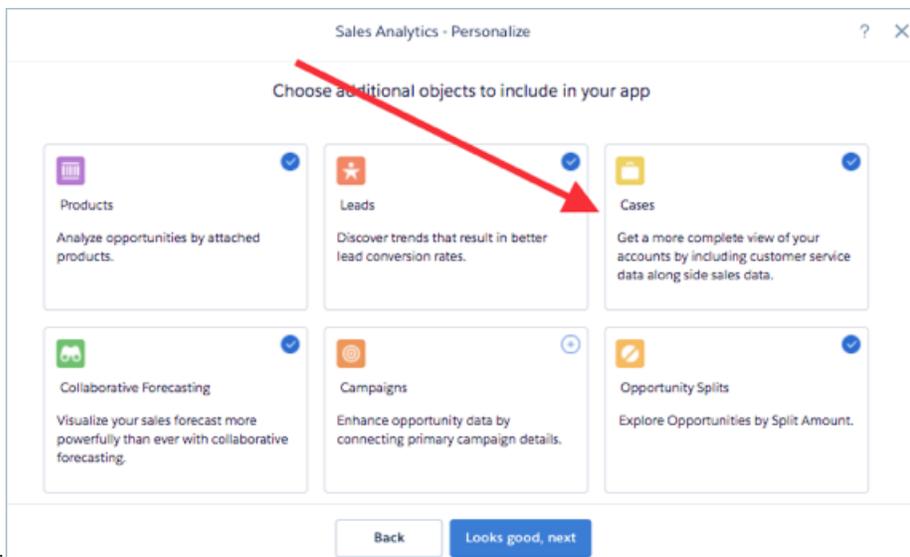
to your app.

Top Products chart. Uses values from fields selected in wizard page 6, questions 2 and
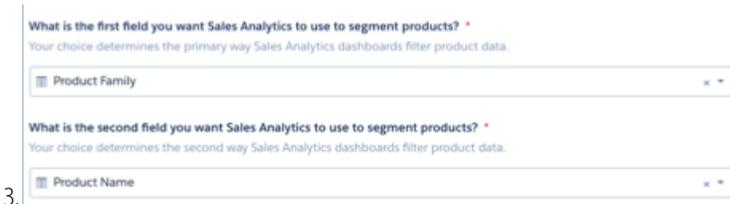
3.

Opportunity Details chart. Amount column changes to Schedule Amount, Close Date column changes to Schedule Date if you select Product Schedules on page that lets you add data to the app.

Datasets Used

- Accounts
- Opportunities
- Oppty Products (if you select Products on the wizard page that lets you choose to add objects)
- Cases (if you select Cases on the wizard page that lets you choose to add objects)
- Activities
- Quota

KPI Calculations

Top of Main Chart

- Avg Sales Cycle (1). Average of (opportunity close date - created date) for won opportunities.
- Avg Sales Cycle comparison (2). (Average cycle for account selected - Average cycle for all accounts depending on filter selections) / Average cycle for all accounts depending on filter selections for closed won opportunities.
- Avg Opportunity Age (3). Average of current date- created date for open opportunities.
- Avg days in sales cycle comparison (4). (Average cycle for selected account - Average cycle for all accounts depending on filter selections) / Average cycle for all accounts depending on filter selections for open opportunities.
- Avg Win Rate (5). Total won/ (Total won + Total lost).
- Avg Win Rate comparison (6). (Win rate for selected account - Win rate for all accounts depending on filter selections) / Win rate for all accounts depending on filter selections.
- Created Cases (7). Total count of created cases in selected period.
- Created Cases comparison (8). (Created cases for selected account - created cases for all accounts depending on filter selections) / Created cases for all accounts depending on filter selections.
- Days Since Last Activity (9). Today - Last update date or Last activity date.
- Days Since Last Activity comparison (10). (Days since last activity for selected account - Days since last activity for any account depending on filter selections) / Days since last activity for any account depending on filter selections.

Main Chart Area

- Top Accounts By…(1). Shows top accounts for selected period. Can filter by Closed Won Amount, Avg Sales Cycle, #Activities Completed, or Open Pipeline.

- Opportunities by Stage (2). Shows amounts for selected period. Further filter by:

  - Past Due (3). Opportunities with close date in the past.

  - Pushed (4). Opportunities with close date pushed to future at least once.

  - Neglected (5). Opportunities with no activity in at least 60 days.

- Top deals or products (6). Products toggle included if you select Products on the wizard page that lets you add objects to your app. Amounts based on selections from top charts.

- Activities (7). Total number of activities by type.

- Cases By (8). Only included if you use the Cases object in Salesforce and you select Cases on the wizard page that lets you add objects to your app. Select either Type or Status.

Sales Analytics Sales Rep Overview Dashboard

Sales reps: start here to see the status of opportunities you own and get ideas for the quickest ways to make quota. Review quota attainment, bookings, and pipeline for a given time period. Also uncover key opportunities for accelerating deal closing and review the number of activities you've completed in the selected period.

Default Behavior and Recommended Options

The top chart (1) shows deal closings over time by all forecast categories. Change the Forecast Category filter (2) to Best Case, Commit, or another category to get ideas for the quickest path to making your quota.

The Areas to Focus on to Close More Deals chart (3) opens to all sales stages. Select one or more sales stages from the filter (4) to view top open deals and neglected opportunities that can help you hit quota.

The dashboard shows accounts where you're the named owner.

Wizard and Other Setup Options

Quotas. Dashboard automatically includes quota information from Sales Cloud Collaborative Forecasts Quotas if your org uses this feature. If it does not, you can manually upload quotas data into the app from an external tool as a CSV file. See Collaborative Forecasting and Quotas Data in Sales Analytics on page 1526.

Opportunity amounts. In wizard page 6, question 1: If you select a custom field that's tracked historically, the dashboard shows

this-week-versus-last-week comparisons.

Neglected opportunities (5). Determined by a flag with the threshold of 60 days in the Sales Analytics dataflow. Choose your own threshold by using the Dataflow Editor on page 876 to change the value.

Account bar chart (6). To see amounts, your org needs to use the standard Accounts object Annual Revenue field.

Datasets Used

- Opportunities
- Opportunity Splits (if you add Opportunity Splits to the app on the wizard page that lets you add objects)
- Quota

KPI Calculations

Main Chart

- Closed Won. Total amount closed won for selected period. May include future closed opportunities.
- Open Pipe. Total open pipeline amount for selected period.
- Activities Completed. Number of activities completed in selected period.

Widgets at Bottom

- Top Open Deals. Open opportunities set to close in the selected period ordered by amount.
- Neglected Opportunities. Open opportunities set to close in selected period with no activities (tasks/events) or record updates in the past 60 days. Can change number of days by editing the Sales Analytics dataflow.
- Accounts with no opportunities. Accounts ordered by annual revenue that have zero opportunities.

Sales Analytics Executive Overview – Pipeline Performance Dashboard

Sales executives get instant pipeline status. You can view the current amount in the pipe, comparison to the previous period, projected closing, and top deals by lead source, product, account, and region. Also get top-level views of sales and service performance.
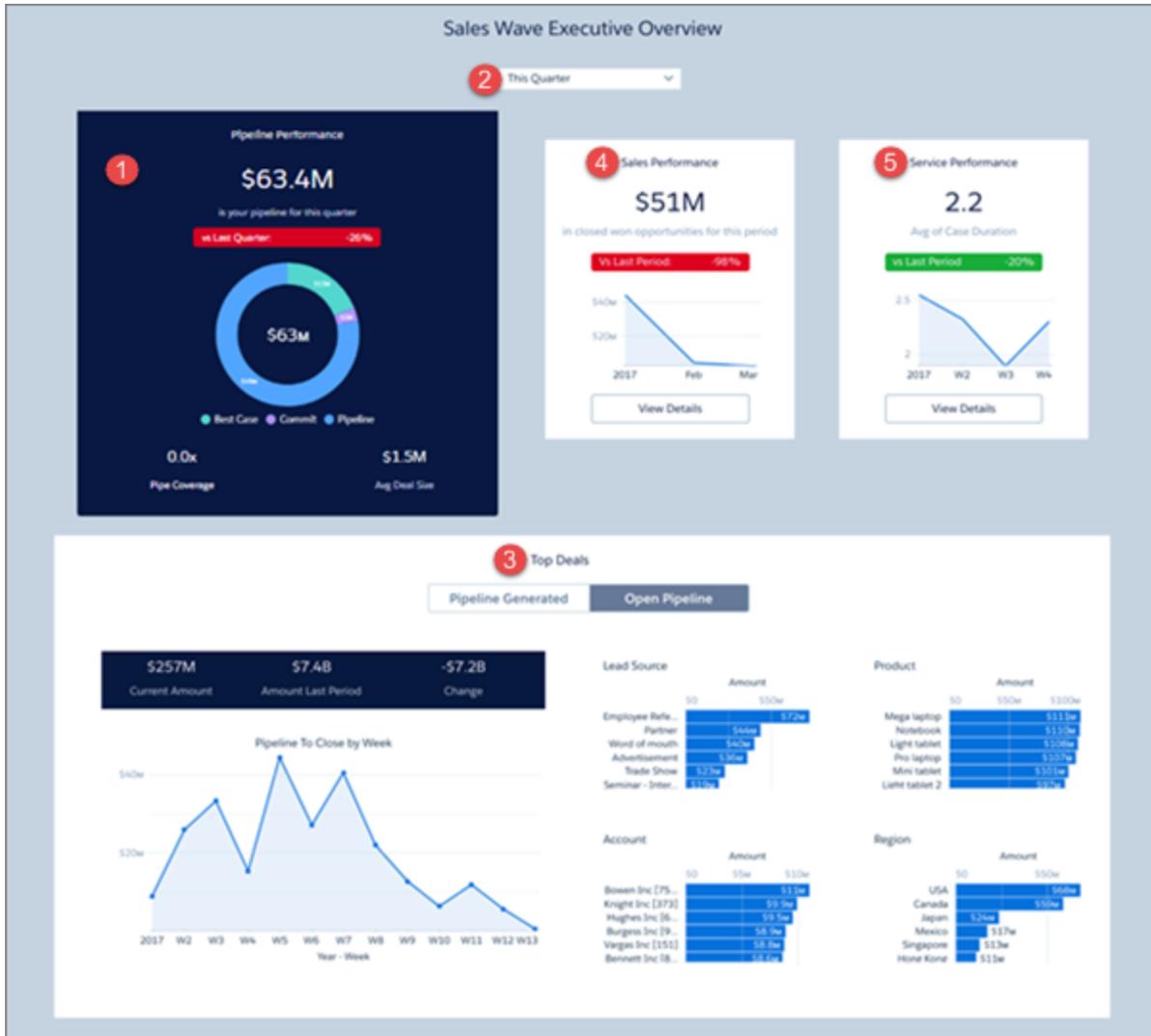
Default Behavior and Recommended Options

Pipeline Performance chart (1) shows results for the current quarter. View results from a different period by changing the selection in the filter at top (2). Also shows pipeline coverage for the period, assuming you track quotas for your team. And shows average deal size for open opportunities set to close that period.

Top deals widget (3) defaults to deals set to close during selected period (**Open Pipeline**). Switch toggle to **Pipeline Generated** to see deals generated during the period.

To view sales and service performance overviews, click **View Details** buttons in Sales Performance (4) and Service Performance (5) charts.

Wizard and Other Setup Options

- Products. Only included if you add Products to your app on the wizard page that lets you add objects.
- Services data. Only included if you add Cases to your app on the wizard page that lets you add objects.

Datasets Used

- Opportunities
- Opportunity Products (if you add Products to the app)
- User Allocation
- Pipeline Trending
- Cases (if you add Cases to the app)

KPI Calculations

Pipeline Performance
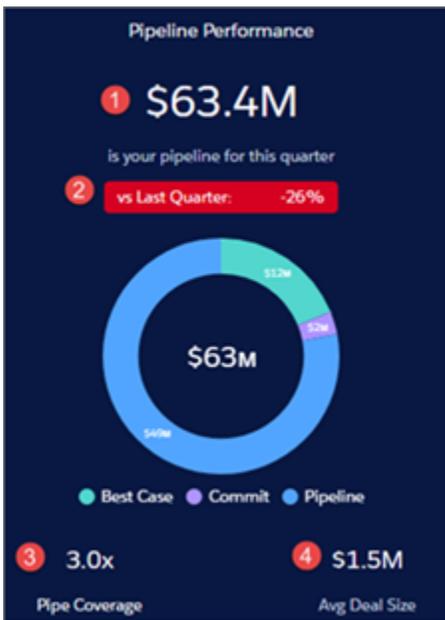
1. Pipeline Performance. Total amount open opportunities for selected period.

2. Comparison to prior period (quarter, year, month, etc. depending on selected period). (Total amount open opportunities to date in selected period - Total amount open opportunities in prior period) / Total amount open opportunities in prior period. Prior period uses historical open pipeline amount.

3. Pipe coverage. Total open quota / Remaining quota for selected period. Remaining quota equals Sum of total sales rep quota - Closed won.

4. Avg Deal Size. Average Amount for open opportunities in selected period.

Top Deals

1. This period (quarter, year, month, etc., depending on selected period). Either total open set to close this quarter or total created this quarter (open or closed) depending on whether you select Open Pipe or Pipe Generated (a).

2. Last period (quarter, year, month, etc., depending on selected period). Either historical amount open at the same time last period or total pipeline generated last period.

3. Change. This Period - Last Period.

Product Name widget appears only if you add Products to your app.

Sales Performance

1. Sales performance. Total closed won in selected period.

2. Comparison to prior period (quarter, year, month, etc. depending on selected period). (Total closed won in current period - total closed won prior period) / Total closed won prior period.

3. Total closed over time. Total closed won by week.

Service Performance

1. Service performance. Average days to close cases during selected period.

2. Comparison to prior period (quarter, year, month, etc. depending on selected period). (Average time to close in selected period - Average time to close in prior time period) / Average time to close in prior time period.

3. Timeline. Average time to close cases by week.

Sales Analytics Executive Overview – Sales Performance Dashboard

Sales executives get instant insight into closed business and quickly recognize trends. See bookings for the period, changes from previous period, as well as win rate, lead conversion rate, and top deals. Also see top-level views of pipeline and service business.

Default Behavior and Recommended Options

Sales Performance chart (1) shows results for the current quarter. View results from a different period by changing the selection in the filter at top (2)

Performance Trends chart (3) defaults to closed won opportunities. View other closed opportunities (such as lost) by changing the filter (4) selection.

Top deals widget (5) defaults to accounts. Toggle between **Parent Account** and **Opportunity**.

To view pipeline and service performance overviews, click **View Details** buttons in Pipeline Performance (6) and Service Performance (7) charts.

Wizard and Other Setup Options

- Lead Conversion Rate (8). Only included if you add Leads to the app with the wizard page that lets you add objects.
- Service Performance (7). Only included if you add Cases to your app on the wizard page that lets you add objects.

Datasets Used

- Opportunities
- User Allocation
- Leads (if you add Leads to the app)
- Cases (if you add Cases to the app)

KPI Calculations

Sales Performance

1. Sales Performance. Total closed won in selected period.

2. Comparison to prior period (quarter, year, month, etc., depending on selected period) (Total closed won to date in current period - Total closed won prior period) / Total closed won prior period.

3. Quota Attainment. Total quota / Total closed. Based on sum of all sales rep quotas.

4. Won Last Period. Total won in prior selected period (quarter, year, month, etc., depending on selected period).

5. Win rate. Total closed won amount / Total closed amount win rate. Total closed won amount / Total closed amount.

6. Avg Sales Cycle. Average time to close for all closed won opportunities.

Performance Trends
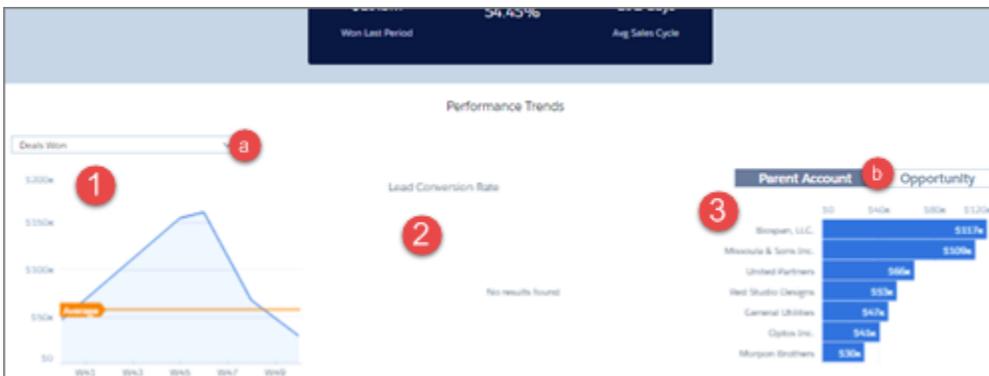
1. Deals over time. Either closed won or closed lost depending on filter (a) selection, week by week for selected period. Average line shows average closed won or lost.

2. Lead Conversion Rate. Leads converted to closed won opportunities / Total leads created in selected period, by week. Only shows if customer adds leads when creating app with wizard. (Not shown in images.)

3. Top accounts or opportunities. Depending on toggle (b) selection, shows top deals according to opportunity name or account name (or parent account name if your org uses account hierarchy, as shown in image). Shows either top closed won or closed lost depending on filter (a) selection.

Product Name widget appears only if you add Products to your app.

Pipeline Performance
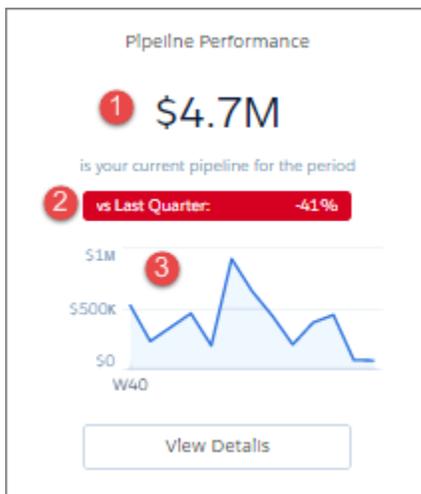
1. Pipeline Performance. Total open opportunity amount for selected period.

2. Comparison to prior period (quarter, year, month, etc., depending on selected period). (Total open opportunity amount to date this period - Total open opportunity amount in prior period) / Total open opportunity amount in prior period. Prior period uses historical open pipeline amount.

3. Open deals over time. Open deals by week of close date.

Service Performance
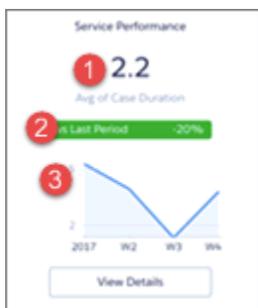
1. Service performance. Average days to close cases during selected period.

2. Comparison to prior period (quarter, year, month, etc. depending on selected period). (Average time to close in selected period - Average time to close in prior time period) / Average time to close in prior time period.

3. Timeline. Average time to close cases by week.

Sales Analytics Executive Service Performance

Sales executives get instant insight into customer service performance. See average case duration for a selected period compared with previous periods and case duration by account, service channel, and geography.

Default Behavior and Recommended Options

> 🔖 **Note:** This dashboard is only included if Sales Wave is set up to include cases. See Wizard and Other Setup Options.

Service Performance chart (1) shows results for the current quarter. View results from a different period by changing the selection in the filter at top (2).

Service Details chart (3) defaults to showing cases by Parent Account. Select Channel from toggle at top to view cases according to the channels where they came from (for example, email, chat, and phone). Select Geography to view cases by locale.

To view pipeline and sales performance overviews, click Details buttons in Pipeline Performance (4) and Sales Performance (5) charts.

Wizard and Other Setup Options

To add this dashboard to your app, select Cases on the wizard page that lets you add objects to the app.
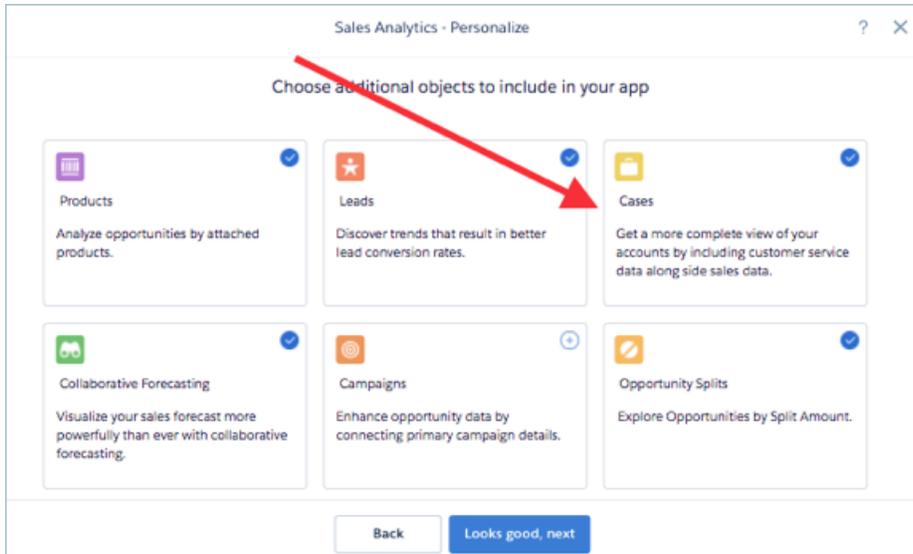
Datasets Used

- Cases
- Accounts

KPI Calculations

Service Performance

- Service performance (1). Average days to close cases during selected period.
- Comparison to prior period (2). (Average time to close to date in selected period - Average time to close in prior period) / Average time to close in prior period.
- Average case duration by quarter (3). Average time to close for cases per quarter, this year and last year.

Service Details (at bottom)

- Account. Average case duration by total amount closed won by account.
- Channel. Average duration and number of closed cases by channel
- Geography. Number of cases created by region. *Region* uses the answer for the primary field from the Accounts object selected in wizard.
- Created Cases. Total number of cases created in selected period.
- Created Cases vs last period. (Number of cases created in current period - Number of cases created in previous period) / Number of cases created in previous period.
- Closed Cases. Total number of cases closed within selected period.

- Cases Closed vs last period. (Number of cases closed in current period - Number of cases closed in previous period) / Number of cases closed in previous period.

- Open cases. Total number of cases currently open. (Not impacted by selected period.)

- Open Cases vs last period. (Number of open cases today - Number of open cases at the start of the previous period) / Number of open cases at the start of the previous period. (Number of open cases at the start of previous period is estimated based on number of cases created and closed since the beginning of the previous period.)

- Bookings. Total amount closed won in selected period.

- Bookings vs last period. (Total amount closed won in current period - Total amount closed won in previous period) / Total amount closed won in previous period).

Pipeline Performance
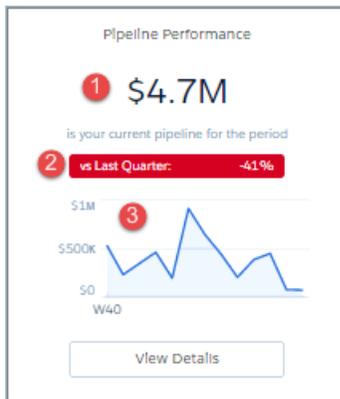
1. Pipeline Performance. Total open opportunity amount for selected period.

2. Comparison to prior period (quarter, year, month, depending on selected period). (Total open opportunity amount to date this period - Total open opportunity amount in prior period) / Total open opportunity amount in prior period. Prior period uses historical open pipeline amount.

3. Open deals over time. Open deals by week of close date.

Sales Performance

1. Sales performance. Total closed won in selected period.

2. Comparison to prior period (quarter, year, month, depending on selected period). (Total closed won in current period - total closed won prior period) / Total closed won prior period.

3. Total closed over time. Total closed won by week.

Sales Analytics Company Overview Dashboard

Intended to give sales leaders and operations staff high in the corporate hierarchy an overview of the sales business and operations.

Note: **Multi Org Sales Analytics Version Only**. Use the **Data Source** menu to select the org data to view in the dashboard.

Default Behavior and Recommended Options

Shows quota progress and pipeline rolled up to the role one level below the manager selected in the **View As** filter. Defaults to showing results for the manager currently running the app. Select a different manager from the **View As** filter to view rollup of pipeline and quota for all team members subsidiary to that manager in the role hierarchy. You can remove any KPIs your organization doesn't care about by removing them from the dashboard layout. To learn how to build and edit Analytics dashboards, see Build Tableau CRM Dashboards.

Wizard and Other Setup Options

- **View As filter.** If Salesforce is set up to use the Sales Cloud Collaborative Forecasts feature, the filter shows forecast user names. If your org doesn't use Collaborative Forecasts, filter shows role developer names. In either case, quota and pipe roll up to the selected user. If your org doesn't use Salesforce role hierarchy, this filter does not appear.
- **Opportunity Type filter**. Values shown determined by selection made in wizard page 5, question 3. By default, the filter uses the standard Opportunity Type field.

- **Time Period filter.** All time periods based on Salesforce fiscal year setting.
- Geography chart at bottom of dashboard uses values from fields selected in wizard page 4, questions 4 and 5.

- Product chart at bottom of dashboard uses values from fields selected in wizard page 6, questions 2 and 3, if you added Products object to your app. If you set up app without Products, the chart shows fields selected from wizard page 4, questions 1 and 2 about how you segment customers.
- **Top Deals details chart**. Amount column changes to Schedule Amount, Close Date column changes to Schedule Date if you select Product Schedules on page that lets you add data to the app.

Supports Opportunity Splits if the feature is enabled in your org. See Opportunity Splits.

Datasets Used

- Opportunity
- Oppty Products (optional)
- User Manager
- Roles (optional)
- User Allocations
- ForecastingItems (optional)

KPI Calculations

Role Hierarchy Filter Options (1)

- Open Pipeline. For selected role, total roll-up opportunity amount for any open opportunity with a close date in selected time period.
- Closed Won. For selected role, total roll-up opportunity amount for any closed won opportunity with a close date in selected time period.
- Avg Deal Size (Closed Won). For selected role, total roll-up average opportunity amount for all closed won opportunities with close dates in selected time period.
- Avg Deal Size (Open). For selected role, total roll-up average opportunity amount for all open opportunities with close dates in selected time period.
- Opportunity Count (Closed Won). For selected role, total roll-up opportunity count for all closed won opportunities with close dates in selected time period.
- Opportunity Count (Open). For selected role, total roll-up opportunity count for all open opportunities with close dates in selected time period.

Top of Main Chart

- Quota Attainment (2). Closed won/quota. Quota is the sum of the sales reps quotas that roll up to the role or user selected in the View As filter

- Pipe Coverage (3). Open pipe/remaining quota. Open Pipe amount based on value selected in Stage and Forecast filters.
- Remaining Quota (4). Quota - closed won

Bottom of Main Chart

- Total Forecast (5). Closed won + opportunity amount for categories selected in **View As** and **Opportunity Type** filters
- Closed Won (6). Total amount of opportunities won in the selected time period.
- Open (7). Based on Forecast Category. Open pipeline based on close date from the period selected in the **Time Period** filter and categories selected in the **Forecast Hierarchy** filter.

Last two amounts shown in horizontal bar in middle of main chart.

Pipeline by Segment Charts

- Owners (8). Total opportunity amount grouped by opportunity or account owner. Based on values selected in filters at top of the dashboard.
- Geographies (9). Total opportunity amount grouped by geography based on close date from the selected time period.
- Products (10). Product total price grouped by product family or product name, depending on how you answered wizard page 6, questions 2 and 3 about products.

Sales Analytics Lead Analysis Dashboard

Boost operational efficiency by taking a close look at your team's lead conversions. Quickly identify leads that convert the most quickly and visualize the team's conversion rate. And get a view into top leads by source and region.

Note: **Multi Org Sales Analytics Version Only**. Use the **Data Source** menu to select the org data to view in the dashboard.
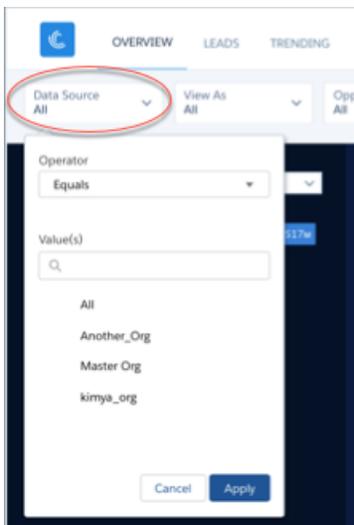
Default Behavior and Recommended Options

Default view shows leads for the entire team. To see leads according to each team member, select a member's name from the **Owner Name** filter (2).

In the Lead Performance Details chart (5), open the action menus from Full Name (a) or Company (b) fields and post to Chatter, create a task or event, or perform another action. Converted leads in Salesforce can't be edited, so you can't perform actions from a converted lead.

Wizard and Other Setup Options

Wizard selections made when you create the app determine the labels and values in the KPIs numbered in the image just above. Select basic create or accept default values in custom create to use labels and values from standard Salesforce fields. To customize KPIs, select other fields from the wizard.

1. **Lead Score** in image, defaults to showing count of leads grouped by their rating. Customize with wizard page 8, question 1.

2. **Lead Source** in image. Defaults to showing count of leads grouped by their source. Customize with wizard page 8, question 2.

3. **Leads by Region** in image. Defaults to showing count of leads grouped by region. Customize with wizard page 8, question 3.

Industry filter (3 in main image at top). Change filter by choosing a field other than Industry in wizard page 8, question 4.

Lead Performance Details chart. Amount column changes to Schedule Amount, Close Date column changes to Schedule Date if you select Product Schedules on page that lets you add data to the app.

Datasets Used

- Leads
- Users

KPI Calculations

Filters Across Top

1. Status. Values from your org's standard Lead Status field.

2. Owner Name. Owner or queue name for owner of lead.

3. Industry. Values from your org's Industry field. Can change filter with configuration wizard. See Wizard and Other Setup Options.

4. Lead Created Date. Based on Salesforce fiscal year settings.

Top Chart
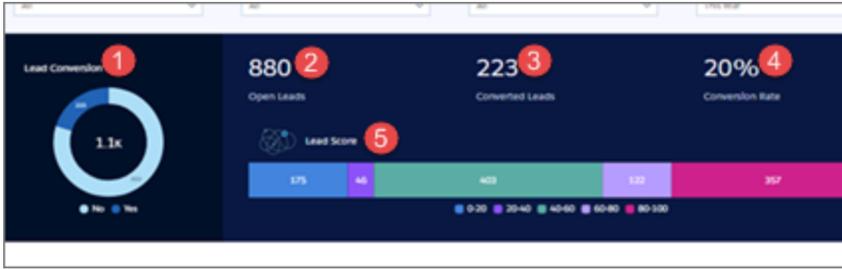
1. Lead Conversion. Counts of closed and open leads. (Leads grouped by IsConverted.)

2. Open Leads. Total number of open leads created during period selected in Lead Created Date filter.

3. Converted Leads. Number of converted leads created during period selected in Lead Created Date filter. Determined by the isConverted flag on Leads. Counts only leads converted to contacts, accounts, or opportunities.

4. Conversion Rate. Leads converted / Leads created during period selected in Lead Created Date filter.

5. Lead Score. Counts of leads grouped by Rating. Can change grouping with configuration wizard. See Wizard and Other Setup Options.

Middle Chart
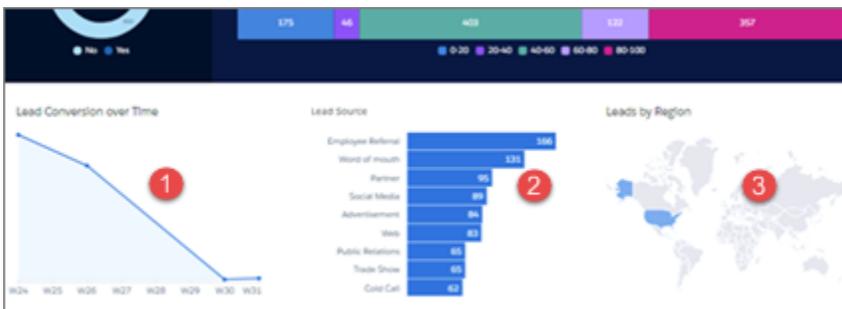
1. Lead Conversion over Time. Weekly count of converted leads.

2. Lead Source. Count of leads grouped by source. Can change grouping with configuration wizard. See Wizard and Other Setup Options.

3. Leads by Region. Counts of leads grouped by region. Can change grouping with configuration wizard. See Wizard and Other Setup Options.

Sales Analytics Sales Performance Dashboard (Operations)

Sales leaders: Drill into your closed business by geography, account, product, source, or other criteria to get a 360-degree view of the team's operational efficiency.
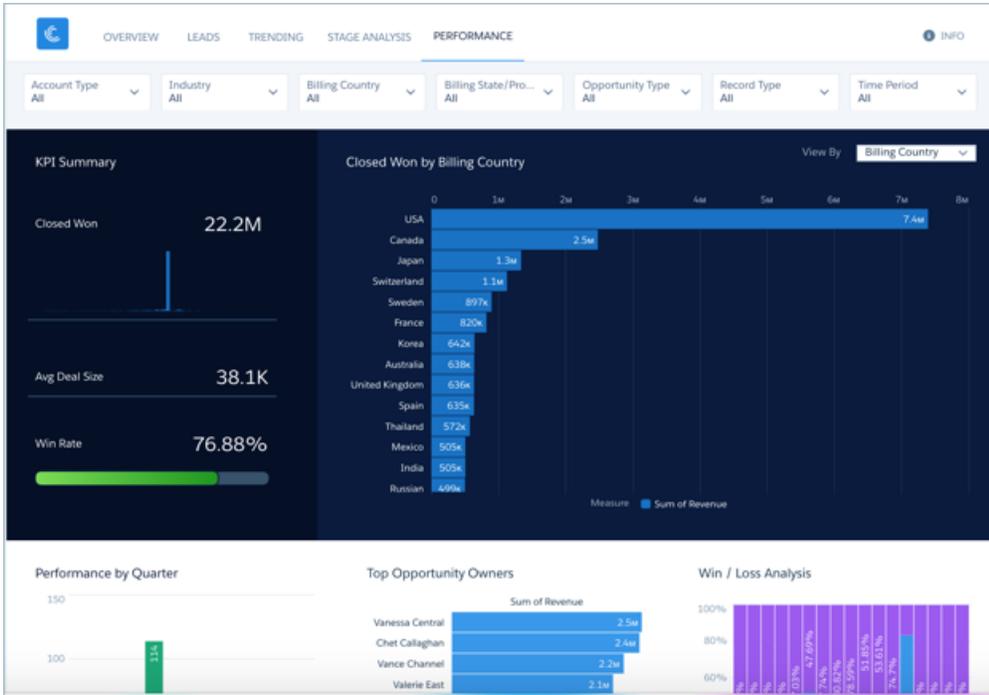
Note: **Multi Org Sales Analytics Version Only**. Use the **Data Source** menu to select the org data to view in the dashboard.
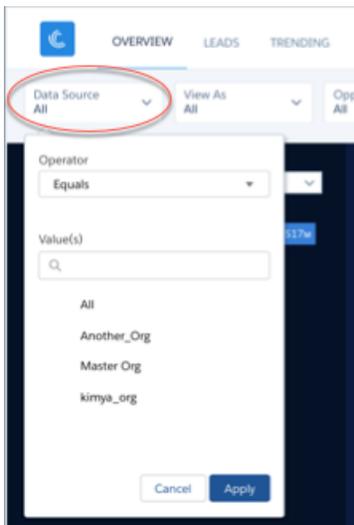
Default Behavior and Recommended Options

View all performance KPIs through a single dashboard filtered by the type of data you'd like to view.

In the main chart, change the **View By** filter (top right) to group data by dimensions available in your org. The title and type of the chart change to reflect the data being shown. The image shows deals grouped by **Billing Country** in a map. Data grouped by product or account type is displayed with the titles *Performance by Product* or *Performance by Account Type* in a bar chart. The chart also shows closed won deals by default. Select Period Over Period in the **Display** filter to get a comparative view of closings over time. The period

being compared depends on the selection in the **Time Period** filter. If you select This Quarter, for example, the chart compares the time to date in this quarter with the same period in the previous quarter.

Wizard and Other Setup Options

Wizard selections made when you create the app determine the labels and values in the following filters (shown at top of image). Select basic create or accept default values in custom create to use labels and values from standard Salesforce fields. Select other fields from the following wizard questions to customize filters:

- **Account Type** in image. Wizard page 4, question 1. Opportunity Type in image. Wizard page 5, question 3.
- **Industry** in image. Wizard page 4, question 2.
- **Billing Country** in image. Wizard page 4, question 4.
- **Billing State/Province** in image. Wizard page 4, question 5.
- **Opportunity Type** in image. Wizard page 4, question 3.

Other filters can't be customized. Selections they contain are determined by the app and/or values in your org.

Opportunity Details chart. Amount column changes to Schedule Amount, Close Date column changes to Schedule Date if you select Product Schedules on page that lets you add data to the app.

Datasets Used

Uses Opportunity Products if you add products to the app. Otherwise uses Opportunities.

KPI Calculations

Header at Top and Main Chart

- **Closed Won** for period selected in **Time Period** filter (year, month, etc.). Closed won total based on close date within the selected period. May include future closed opportunities.
- **Avg Deal Size**. Average size of won opportunities based on filter selections.
- **Win Rate**. Total amount won / Total amount closed based on filter selections.
- **Closed Won by** chart at right shows closed won opportunities based on **View By** filter selection (**Billing Country** in image).

Charts at Bottom

- **Performance by Quarter**. Quarter / Same quarter previous years.
- **Top Ten Opportunity Owner**. Top opportunity owners by amount.
- **Win / Loss Analysis**. Total amount closed grouped by IsWon.

Sales Analytics Sales Stage Analysis Dashboard

Shows how deals move through stages of the sales process and if deals are moving smoothly. Also reveals bottlenecks and exposes at-risk opportunities.

> 📝 **Note:** **Multi Org Sales Analytics Version Only**. Use the **Data Source** menu to select the org data to view in the dashboard.
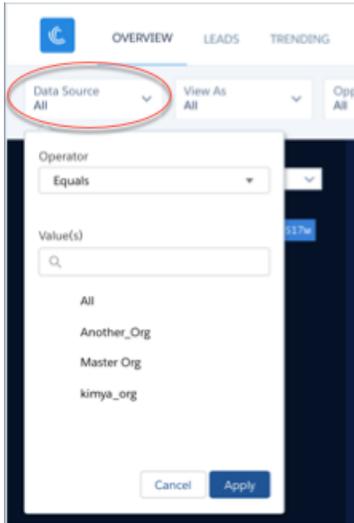
Default Behavior and Recommended Options

Compares how long current opportunities currently take in each stage of your process with the same data for historical deals. Click a stage in the main chart and the chart to the right shows deal movement from that stage to others. The main chart shows historic and current average number of days deals spend in that stage.

By comparing current with historical data, the dashboard determines which deals are at risk and displays them in neglected, stalled, and pushed opportunities charts at bottom of dashboard.

Data comes from the Opportunity History table, which supports only the standard amount and stage name fields.

The Stage Name filter uses stage names in the standard StageName field in the Opportunities object.

Wizard and Other Setup Options

The Sales Analytics dataflow determines neglected opportunities. Change the threshold from the default 60 days to your team's preferred length of time using the Dataflow Editor. See Configure the Dataflow with the Dataflow Editor on page 876.

Datasets Used

- Opportunities
- Pipeline Trending

KPI Calculations

Filters at Top

- **Account**. Account names related to opportunities in the dashboard.

- **Owner Name**. Names of opportunity owners. May include inactive users or users who do not own any deals in the selected period.
- **Owner Role**. Role name of opportunity owners.
- **Stage Name**. Names in the standard StageName field in the Opportunities object.
- **Record Type**. Opportunity record type names defined in your org.
- **Close Date**. Opportunity closed date based on Salesforce fiscal year settings.
- **Historical Time Frame**. Period used to calculate average time in stage.

Main Chart, Top

- **Historical Average**. Number of days each opportunity stay in each stage. Average days in stage for all opportunities in the OpportunityHistory object, based on period set in **Historical Time Frame** filter.
- **Open Opportunities**. Number of days current open opportunities stay in each stage. Average days in stage to date for all open opportunities with close dates within period selected in **Close Date** filter.

Opportunity Days in Each Stage. Shows number of opportunities moved from stage selected in main chart to other stages based on period set in **Historical Time Frame** filter.

Main Chart, Bottom

- Probability to close. Total closed won/Total opportunities based on the selected stage during period set in **Historical Time Frame** filter. If no stage is selected, shows win rate.
- Won Deals. Total amount won during period set in **Historical Time Frame** filter.
- Lost Deals. Total amount won during period set in **Historical Time Frame** filter.

Charts Below Main Chart

- Neglected Opportunities. Top open opportunities by amount that have not been touched in 60 days that are set to close in the selected period.
- Stalled Opportunities. Top open opportunities by amount that have stayed in current stage at least one day longer than historical average for all opportunities. Based on close date in the selected period.
- Pushed Opportunities. Top open opportunities by push count based on close date in the selected period.

Sales Analytics Lenses

Sales Analytics lenses augment app dashboards with targeted visualizations of your org's sales data.

Lens Name	Contents	Edit Through User Interface?
Explore Opportunities	Details about opportunities, starting with pushed opportunities. Narrow the time period to look at pushed opportunities for a year, quarter, month, and so on. Select a new grouping, such as by team or by opportunity owner. Filter by a different stage, status, product, or any other way you break down opportunities to get a more refined view of opportunities in your pipe.	Yes.
Historical Pipeline By Forecast	Augments Team Trending dashboard with detail about the pipeline for the past six months broken out by forecast status.	No. Requires SAQL editing. For advanced users only.
Historical Pipeline By Stage	Augments Team Trending dashboard with detail about the pipeline for the past six months broken out by sales stage.	No. Requires SAQL editing. For advanced users only.
Sales Stage Details	Augments Sales Stage Analysis dashboard, shows where specific deals stand in the sales process. Select a bubble in the chart and click drill-down button at top right to learn specifics about a deal (for example, forecast category, stage, owner, age).	Yes.

Sales Analytics Datasets

When you create Sales Analytics , Tableau CRM imports your Salesforce data into app datasets based on your wizard selections. The datasets drive the app's dashboards and lenses.

Dataset Name	Contents	Special Requirements
Accounts	Data about accounts. Includes accounts without opportunities.	None.
Opportunities	Data about accounts, opportunities, and users.	None.
Oppty Products	Data about products with opportunities and accounts.	None.
Pipeline Trending	Opportunity historical Data. Tracks opportunity amount, close date, and stage changes over the life of an opportunity.	None.
Quota	Quotas data.	Upload a CSV file on page 1526 with quotas data to update this dataset if you store quotas data externally to Salesforce and don't use the Sales Cloud Collaborative Forecasts feature.
User Allocation	Applies role hierarchy information to quotas data during manager roll-up.	None.
Users	Provides details about all users in an org.	None.
Roles	Basic data about user roles.	None.
Cases	Data about cases related to accounts.	Only included if you choose to import cases data to Sales Analytics through the configuration wizard.
Activities	Data about events and tasks.	None.
ForecastingItem	Data about forecasts and forecast adjustments.	Only included if you use the Sales Cloud Collaborative Forecasts feature.
Leads	Salesforce leads data, including converted leads connected to accounts and opportunities.	Only included if you choose to import leads data to Sales Analytics through the configuration wizard.

Dataset Name	Contents	Special Requirements	
Campaigns	Salesforce campaign data, including opportunities connected to campaigns.	Only included if you choose to import campaigns data to Sales Analytics through the configuration wizard.	
Campaign Members	Data about campaign members with campaigns. See who's connected to which campaigns.	Only included if you choose to import campaigns data to Sales Analytics through the configuration wizard.	
Opportunity Splits	Split opportunity data.	Only included if you choose to import opportunity splits to Sales Analytics through the configuration wizard.	
Product Schedules	Data about products sold according to a scheduled payment contract.	Only included if you choose to import product schedules to Sales Analytics through the configuration wizard.	

Sales Analytics Calculated Fields

Sales Analytics adds custom fields computed from your Salesforce data to app datasets. The table lists the fields in alphabetical order and shows how they are computed and the datasets that include them.

* - Indicates that dataset may not be part of your app, depending on your configuration wizard selections.

Field Label	Field API Name	Type	Computation	Datasets
Amount_isUpdated	Amount_isUpdated	Boolean	If the opportunity history record is an amount update: 1. Otherwise: 0	Pipeline Trending
CloseDate_isUpdated	CloseDate_isUpdated	Boolean	If the opportunity history record is a close date: 1. Otherwise: 0	Pipeline Trending
Days Overdue	DaysOverdue	Number	If activity is overdue, the number of days the activity is overdue	Activities
Days Past Due	DaysPastDue	Number	For open opportunities where close date is in the past: Date dataflow last run - Close date	Opportunities Pipeline Trending Activities Oppty Products * Product Schedules * Opportunity Splits *

Field Label	Field API Name	Type	Computation	Datasets
Days Since Last Activity	DaysSinceLastActivity	Number	Either Date that dataflow is run, Date opportunity last modified, or Date dataflow last run (whichever is earlier) - Date of last activity for the opportunity	Opportunities Pipeline Trending Activities Oppty Products * Product Schedules * Opportunity Splits *
Discount_Percent	Discount_Percent	Number	1 - { (Product amount field selected in wizard) / (ListPrice * Quantity) }	Oppty Products
IsLastUpdate	IsLastUpdate	Boolean	If the opportunity history record is the last update for the opportunity: True. Otherwise: False	Pipeline Trending
hasOpportunity	hasOpportunity	Boolean	If the user or account has ever owned an opportunity: True. Otherwise: False	Users Accounts
IsLastStageUpdate	IsLastStageUpdate	Boolean	If this is the last stage change in opportunity history: 1. Otherwise: 0	Pipeline Trending
IsLost	IsLost	Text	If opportunity was lost: Yes. Otherwise: No	Opportunities Pipeline Trending Activities Oppty Products * Product Schedules * Opportunity Splits *
IsOverdue	IsOverdue	Boolean	If activity date is past due: True. Otherwise: False	Activities
IsPulled	IsPulled	Boolean	If close date is changed to an earlier date: 1. Otherwise: 0 Note: Sum of this field on an opportunity results in the number of times a close date was pulled in.	Opportunities Pipeline Trending Activities Oppty Products * Product Schedules * Opportunity Splits *

Field Label	Field API Name	Type	Computation	Datasets
IsPushed or	IsPushed	Boolean	If close date is changed to a date in the future: 1. Otherwise: 0 Note: Sum of this field on an opportunity results in the opportunities push count.	Opportunities Pipeline Trending Activities Oppty Products * Product Schedules * Opportunity Splits *
Lead Age in Days	LeadAge	Number	For nonconverted leads: Date the dataflow runs - Lead created date. For converted leads: Lead conversion date - Lead created date	Leads
Neglected	Neglected	Boolean	If Days Since Last Activity (calculated field above) is > 60: True. Otherwise: False	Opportunities Pipeline Trending Activities Oppty Products * Product Schedules * Opportunity Splits *
OpenClosedWonLost	OpenClosedWonLost	Text	If opportunity is open: Open. If opportunity is won (any won stage): Closed Won. If opportunity is lost (any lost stage): Lost	Opportunities Pipeline Trending Activities Oppty Products * Product Schedules * Opportunity Splits *
Opportunity Age	OpportunityAge	Number	For closed opportunities: Close date - Created date. For open opportunities: Date dataflow last run - Created date	Opportunities Pipeline Trending Activities Oppty Products * Product Schedules * Opportunity Splits *
Opportunity Owner or UniqueUserName	UniqueUserName	Text	For users with same first and last name, duplicates include Salesforce username in parentheses	All datasets

Field Label	Field API Name	Type	Computation	Datasets
Owner or Queue Name	UserOrQueueName	Text	Combines lead owners that are either users or queue	Leads
Owner Type	OwnerType	Text	Designates whether lead is owned by a user (value: User) or a queue (value: Queue)	Leads
Past Due	IsPastDue	Boolean	For open opportunity with close date in the past: True. Otherwise: False	Opportunities Pipeline Trending Activities Oppty Products * Product Schedules * Opportunity Splits *
Pushed	Pushed	Boolean	If opportunity has been pushed at least once: True. Otherwise: False	Opportunities Pipeline Trending Activities Oppty Products * Product Schedules * Opportunity Splits *
Total Opportunity Age	OpportunityAge	Number	For open opportunities converted from leads: Dataflow run date - Lead created date. For closed opportunities converted from leads: Opportunity close date - Lead created date	Leads
Stage_isUpdated	Stage_isUpdated	Boolean	If the opportunity history record is a stage: 1. Otherwise 0	Pipeline Trending
Time in Stage	Duration_Seconds	Number	Duration of opportunity in current stage Note: Pipeline Trending uses this calculation for all stages in history for a single opportunity.	Opportunities Pipeline Trending Activities Oppty Products * Product Schedules * Opportunity Splits *

Field Label	Field API Name	Type	Computation	Datasets
ValidToDate	ValidToDate	Number	Date to which any opportunity history record was valid before changing to the next value	Pipeline Trending
ValidFromDate	ValidFromDate	Date	Date when history record is created	Pipeline Trending

Set Up Permissions for the Sales Analytics

Set up your organization to use the Sales Analytics by enabling Tableau CRM and assigning permission sets to users.

> **Important:** Customers set up to use Tableau CRM Growth licenses have the required user permissions to access Sales Analytics. The following instructions apply only to customers with the Sales Analytics license.

Each Sales Analytics license is a single-user license that provides access to Sales Analytics. It includes a single Sales Analytics app single-user license. The table shows data storage limits for the app. If you require more data, you can purchase Analytics Cloud - Additional Data Rows, which entitles you to an additional 100 million rows.

Table 10: Sales Analytics Data Storage Limits

License	Limit
Sales Analytics Apps	25 million rows when used without Tableau CRM Growth license. Use of Sales Analytics app license does not increase data limit for platform license.
Analytics Cloud - Additional Data Rows	100 million rows.

> **Important:** Sales Analytics license data storage limits are contractual, not technical. Licensee agrees to strictly monitor its total number of data rows.

Your org can use the Sales Analytics with or without the Tableau CRM platform.

> **Important:** The Sales Analytics Apps license does not support use of Tableau CRM platform bulk actions or Apex steps functionality. This limitation is contractual, not technical. Licensee agrees to strictly monitor its use of Tableau CRM platform features.

To give administrators or users in your org access to the Sales Analytics, enable Tableau CRM and assign ready-made permission sets to them. By assigning permission sets, you also assign the Sales Analytics Apps or Sales Cloud Einstein permission set licenses.

> **Warning:** Assign the Sales Analytics Admin permission set sparingly because it lets users administer Sales Analytics, which lets them create, edit, and delete the app. Assign it only to users who administer or manage the app. Users with **Sales Analytics User** and Editor or Manager access to the app can create, edit, and delete app assets.

1. First, enable Analytics. In Salesforce Setup, enter `Analytics` in the **QuickFind** (search) field, then click **Getting Started**.

2. Click **Enable Analytics**.

3. Now, assign permission sets. In Salesforce Setup, enter `Users` in the **QuickFind** field.

4. Click **Permission Sets**.

5. Scroll through the list of permission sets until you see **Sales Analytics Admin** and **Sales Analytics User**.

6. For users who require administrator-level access to Sales Analytics, assign the **Sales Analytics Admin** permission set. This permission set enables all user permissions for the app shown in the table.

   a. Click **Sales Analytics Admin**.

   b. Click **Manage Assignments**, then **Add Assignments**.

   c. Check the boxes next to the names of the users who require administrator-level access to the app.

   d. Click **Assign**, then click **Done**.
      The selected users can now create and manage Sales Analytics.

7. To assign user-level access to Sales Analytics, assign the **Sales Analytics User** permission set.

   a. Click **Sales Analytics User**.

   b. Click **Manage Assignments**, then **Add Assignments**.

   c. Check the boxes next to the names of the users who require access to the app.

   d. Click **Assign**, then click **Done**.
      The selected users can now use Sales Analytics.

After you've completed these steps, create and share Sales Analytics with users in your organization. Users can only explore Sales Analytics dashboards and datasets after you've shared the app with them.

Create and Share Sales Analytics

Follow these steps to create Sales Analytics and start uncovering the value of your Salesforce data.

For rapid app creation, choose Basic creation, which uses default settings to create Sales Analytics. To set up Sales Analytics according to your team's specific Sales Cloud analytics requirements, choose Custom creation.

If you use the Sales Cloud Collaborative Forecasts feature to store quota data, the data is automatically available to Sales Analytics. If you track quotas outside of Salesforce, you need to upload a quotas .CSV to include quotas in Sales Analytics dashboards. See Collaborative Forecasting and Quotas Data in Sales Analytics.

> **USER PERMISSIONS**
>
> To create and manage Tableau CRM apps:
> - Manage Analytics Templated Apps

1. Log in to Salesforce and open Tableau CRM Studio. In Salesforce Classic, select **Analytics Cloud** Studio from the Lightning Platform menu (top right of the Salesforce window). In Lightning Experience, from the App Launcher ( ), find and open the **Analytics Cloud Studio** app.

2. Click **Create**, select **App**, select **Sales Analytics**, and then click **Continue** to open the configuration wizard. *If you've created an app before*: Choose between creating a brand new app or creating an app based on settings from a previously-created app. Click **Continue**. Sales Analytics runs a compatibility check against your org to be sure it includes the data to successfully create the app's datasets and dashboards. If it doesn't, follow the instructions in the error message to add the required data and start the app creation process again.

3. Once the org compatibility check succeeds, click **Looks good, next.**

4. Choose between using preselected standard settings or custom settings to set up your app. Select **Basic** to set up your app quickly based on standard settings determined by the org compatibility check. The standard settings include collaborative forecasting if you use the Sales Cloud Collaborative Forecasts feature. Select **Custom** to open the configuration wizard, which you use to make your own, custom settings to reflect the way you and others on your team want to view data. For details about using the configuration wizard, see Customize Sales Analytics with the Configuration Wizard.

5. If you choose **Basic**: You're almost done. Click **Looks good, next**, and skip to Step 9.

6. If you choose **Custom**: Click **Looks good, next**. The wizard asks you to choose Salesforce objects to add to Sales Analytics. It then takes you through a set of questions about how you prefer to view data. Go to Customize Sales Analytics with the Configuration Wizard for complete instructions on how to use the wizard.

7. Give your app a name that's easily recognizable to others in your company and click **Create**. That starts a dataflow that creates the app and its assets (which include a dataflow definition file, datasets, and dashboards).

8. Before it creates your app, Sales Analytics checks to see that the Analytics Cloud Integration User has access to all fields in Salesforce you'd like your app to use. The check happens in the background, and if it succeeds, the app creates successfully. If it fails, you see an error that says the Integration User does not have access to specific fields. If you receive that error, here's what to do:

    a. In Salesforce Setup, go to Manage Users, then Profiles.

    b. Open the profile for the Analytics Cloud Integration User.

    c. Scroll down to Field Level Security and click **View** next to the objects indicated in the error message.

    d. Check **Read Access** for the fields indicated in the error message.

    e. Refresh your browser cache, and click **Create** again.

9. The app creation process can take a few minutes. You can check the status of the dataflow: Open the Tableau CRM Home page, click the gear menu at the upper right of the page, and select Data Manager. From pulldown menu, select Dataflow View and look for your app.

Now that you've created the app, share it with users in your organization. You can only share it with users who have the Use Analytics Templated Apps and Access Sales Cloud Analytics Templates and Apps permissions enabled. (For more about Sales Analytics permissions, see Set Up Salesforce Permissions for the Sales App).

1. Open your app if it's not already open. If you've navigated away from Tableau CRM Studio, go back to it, select **All Items**, find your app, and click it.

2.
Click the Share icon at upper right.

3. In the next screen, use the search field under **Invite others:** to find other users in your org.

4. Select whether you want to make the selected user a Viewer, Editor, or Manager of the app.

    🛑 **Important:** Users with the "Use Analytics Templated Apps" permission and Editor or Manager access to the app can create, edit, and delete assets in the app.

5. Click **Add**, then click **Save**.

🛑 **Important:** When users are deactivated, they lose share and delete access to all apps they manage. To avoid "stranding" an app, be sure that manager access is assigned to at least one active user BEFORE deactivating the user who's the manager of the app.

SEE ALSO:

    Customize Sales Analytics with the Configuration Wizard
    Upgrade Sales Analytics App

Customize Sales Analytics with the Configuration Wizard

Create Sales Analytics with the configuration wizard so the app reflects how your company prefers to view Salesforce data.

When you start the app creation process, Sales Analytics opens the configuration wizard. The wizard guides you through the following steps.

1. Checks your org to be sure it meets minimum data requirements and to detect features that can be added to your app. The results let you know if you have to add data or change Salesforce settings to create the app. It also lets you know about available features.

2. Asks you to choose between basic and custom create options. Basic is intended for first-time app users, while custom lets experienced administrators fine-tune app setup.

3. Choose the basic option, and Analytics Cloud creates the app quickly with default settings.

4. Choose custom and the wizard guides you through the steps to fine-tune your app.

   a. Add features to your app

   b. Answer a series of questions about how you use data in your org.

The following provide details about each using each part of the wizard. Read them in the order shown to get the best results when you create Sales Analytics. Click the question mark [?] in the top-right corner of each page of the wizard to see help for that page.

1. Sales Analytics Wizard Data and Feature Check

   At the start of app creation, Sales Analytics scans your org. It checks to make sure you can create the app and looks for features and data to add to your app.

2. Choose Sales Analytics Wizard App Creation Options

   After Sales Analytics checks your org's data and features, choose between basics and custom app creation options.

3. Use the Sales Analytics Wizard Basic Create Option

   Select the basic app creation option when you create Sales Analytics for the first time or when you want to create the app quickly.

4. Use the Sales Analytics Wizard Custom Create Option

   Custom app creation gives more advanced Salesforce administrators fine-grained control over Sales Analytics features and data.

5. How to Answer Custom Wizard Questions

   The custom version of the Sales Analytics configuration wizard asks you a series of questions about your data. Follow these general guidelines, and the specifics that follow, to get the best result.

6. Opportunities Questions, Sales Analytics Custom Wizard Step 5 of 9

   Tell Sales Analytics how you prefer to view data from the Opportunities object.

7. Products Questions, Sales Analytics Custom Wizard Step 6 of 9

   Tell Sales Analytics how you prefer to view data from the Opportunities object.

8. Questions To Fine Tune Sales Analytics, Custom Wizard Step 7 of 9

   Answer questions about data visibility as well as account team, opportunity team, and record type data in Sales Analytics.

9. Questions About Leads Sales Analytics, Custom Wizard Step 8 of 9

   Answer questions about how Sales Analytics displays data from the Leads object.

10. Bring Data into Sales Analytics, Custom Wizard Step 9 of 9

   Optional questions that let you add data to Sales Analytics.

SEE ALSO:

   Create and Share Sales Analytics

Sales Analytics Wizard Data and Feature Check

At the start of app creation, Sales Analytics scans your org. It checks to make sure you can create the app and looks for features and data to add to your app.

Sales Analytics checks your org for data and features when you open the configuration wizard. If it finds any issues you need to correct before creating the app, you see messages that tell you what to do. Position your cursor over the tooltip [info] for more information.

Here are details about each phase of the check.

Minimum Requirements

To enable app creation, your org must have the following:

- At least one account, opportunity, task, and event.
- At least one event and one task connected with an opportunity.
- Three opportunity history rows.

If you see an error message, go to Sales Cloud, create the required record or row, and try creating the app again.

Features to Add to Your App

The wizard checks to see if you track data about the following in your org:

- Opportunity Line Items
- Account Teams
- Opportunity Teams
- Cases
- Campaigns
- Leads
- Collaborative Forecasts
- Collaborative Forecasting Quota
- Opportunity Splits

The tooltip [info] tells you which are available in your org. Basic app creation automatically adds those features. Custom app creation lets you choose the ones to add.

📝 **Note:** Selecting both Opportunity Splits and Collaborative Forecasting pulls in quota data for the split type you've selected in your org.

Sales Analytics only supports the standard OpportunityRevenue forecasting type unless you use opportunity splits for tracking revenue. If that's the case, the app takes over the splits forecasting type. (Splits have to add up to 100%). The app does not support use of product schedules in combination with splits or collaborative forecasts.

Field-Level Security

The wizard checks if the Analytics Integration User has access to all fields needed to create the app using the basic create option. If the Integration User can't access all fields, you see a warning and the tooltip lists the fields that lack access. Go to Salesforce Setup and provide access to those fields. See Create and Share Sales Analytics App, Step 8, for details. Until all fields are available, you can't use the basic create option.

Salesforce Settings

The wizard looks for your org's settings for fiscal year start date and currency and uses them for Sales Analytics. It also checks to be sure your org has at least one account with a parent. If it does not, you see a warning.

Additional Data Check

The wizard checks to see if your org uses Manager and Role Hierarchy. It uses that information to customize Sales Analytics dashboards. See Sales Analytics Dashboard Guide for details.

Choose Sales Analytics Wizard App Creation Options

After Sales Analytics checks your org's data and features, choose between basics and custom app creation options.

Select **Basic** if you're using Sales Analytics for the first time. It sets up your app quickly based on standard settings determined by the org compatibility check. You get an immediately useful version of the app so you can see how it works. Experiment with it and share it with your team. Based on what you learn, change the standard settings by recreating the app using the custom create option.

Sales Analytics disables basic app creation if the Analytics Integration User can't access all required fields. If the basic option is disabled, click **Back** to return to the org check and follow the instructions in the field-level security tooltip and error message.

Select **Custom** and follow additional steps in the wizard to make your own, custom settings to reflect the way you and others on your team want to view data. The initial, default settings in the screens that follow are the ones used when you create an app using the basic create option. Custom create lets you vary these settings to meet your team's specific needs. You can choose to add or delete features detected by the org check. You can also make specific choices about the data used in the app's dashboards.

Whether you use basic or custom create, Sales Analytics runs a final scan of your org's field-level security settings. The scan detects if the Analytics Integration User has access to all data fields required to create the app. If the scan fails, you see a message telling you how to fix the issue. For more information about field-level security settings, see See Create and Share the Sales Analytics App, Step 8, for details.

USER PERMISSIONS

To create and manage Tableau CRM apps:
- Manage Analytics Templated Apps

- Edit Analytics Dataflows

Use the Sales Analytics Wizard Basic Create Option

Select the basic app creation option when you create Sales Analytics for the first time or when you want to create the app quickly.

- Select **Basic**.
- Click **Looks good, next**.
- Name your app, and click **Create**.

Basic create settings

The basic create option uses data, features, and default settings detected in Salesforce during the compatibility check that runs when you start app creation. It uses the following when creating your app:

USER PERMISSIONS

To create and manage Tableau CRM apps:
- Manage Analytics Templated Apps

- Edit Analytics Dataflows

- Data from Opportunities, Accounts, Users, Tasks, and Events objects.

- Opportunity Line Items data, if the check detects more than 10 rows in the Opportunity Line Items table.

- Collaborative Forecasting data, if the check detects more than 10 rows in the Forecasting Item table.

- Opportunity Splits data, if the check detects more than 10 rows in the Opportunity Splits table.

- Leads data, if the check detects more than 10 rows in the Leads table.

- Account Hierarchy data, if the check detects at least one account with a parent account.

- Role hierarchy data, if the check detects at least one role with a parent role.

- Product Schedules data, if the check (detects at least 10 rows of data in the Opportunity Line Item Schedule table and you *do not use* Collaborative Forecasting or Opportunity Splits.

Using the basic create enables users to view data owned by them and their subordinates. It also uses only standard fields. To use data not listed here, including custom fields, use the custom create option. See Use the Sales Analytics Wizard Custom Create Option

Consult Create and Share Sales Analytics to learn the process for creating your app.

Use the Sales Analytics Wizard Custom Create Option

Custom app creation gives more advanced Salesforce administrators fine-grained control over Sales Analytics features and data.

Add Data and Features

The first window in the custom creation process lets you add data and features to your app. By default, Sales Analytics includes data from selected fields in the following standard Salesforce objects:

- Accounts

- Users

- Roles

- Opportunities

- Projects (Opportunities line item)

- Tasks

- Events

Custom app creation lets you choose some of the fields used in Sales Analytics dashboards.

The compatibility check that run at the start of app creation looks for other available data and features. The wizard displays the results on the first custom create window, letting you know which features you use in your org. If you use a feature, this window tells you if it's available to your app. If you don't use the feature, Sales Analytics tells you it's not available.

Sales Analytics preselects the features used for basic app creation. You have the option of adding others. Preselected options contain a check mark in the upper right corner. Add other available options to your app by clicking them. Click options with a check mark to deselect them. Here are the options Sales Analytics lets you add, if available, and details about adding each.

- **Products**. Select to add Product objects data to your app. If you select, Sales Analytics adds questions to the wizard that ask how to display Products data in the app.

- **Opportunity Splits**. Select to add data about how members of your team share revenue from an opportunity. If you select, Sales Analytics creates an Opportunity Splits dataset and uses the data in the following dashboards:

  - Company Overview

  - Forecast

  - Home

- – Leaderboard
- – Sales Overview Home (Embedded)
- – Team Benchmark

  If you select, Sales Analytics adds questions to the wizard about how to display data about Opportunity Splits and total amount for products in the app. Do not select if you use Product Schedules.

- **Collaborative Forecasting**. Preselected If your org uses the Sales Cloud Collaborative Forecasts feature, which automatically adds quota information from Sales Cloud Collaborative Forecasts Quotas. If it's not preselected and you want to add quotas data to your app, see Collaborative Forecasting and Quotas Data in Sales Analytics. With Collaborative Forecasting selected, Sales Analytics adds questions to the wizard about how to display data about total amounts for opportunity and products in the app. Do not select if you use Product Schedules.

  > **Ⓘ Important:** Sales Analytics only supports the standard OpportunityRevenue forecasting type unless you use opportunity splits for tracking revenue. If that's the case, the app takes over the splits forecasting type. (Splits have to add up to 100%). The app does not support use of product schedules in combination with splits or collaborative forecasts.

- **Product Schedules**. Select if you sell products with scheduled payment contracts. Can only be selected if you do *not* select Opportunity Splits *or* Collaborative Forecasting. With Product Schedules selected, Sales Analytics adds a Product Schedules dataset and uses the data in all dashboards *except* the following:

  - – Company Trending
  - – Sales Stage
  - – Team Trending
  - – Trending

- **Leads**. Select to add Leads object data to your app. With Leads selected, Sales Analytics adds questions to the wizard about how to display data about leads in the app.

- **Cases**. Select to add Cases object data from the Service Cloud to your app. If you select, Sales Analytics adds service performance visualizations to the Executive Overview dashboard. See Sales Analytics Executive Service Performance. With Cases selected, Sales Analytics adds questions to the wizard about how to display data about cases in the app.

- **Campaigns**. Select to add Campaigns object data.

How to Answer Custom Wizard Questions

The custom version of the Sales Analytics configuration wizard asks you a series of questions about your data. Follow these general guidelines, and the specifics that follow, to get the best result.

- **Most questions provide answers in pick lists showing fields from Salesforce objects.** The lists include standard Salesforce fields and any custom fields you've set up on an object. Answer these questions by selecting from the fields shown. Most questions of this type let you choose only one field, and some let you choose multiple fields.

- **You can choose a field from an object only once.** After you select a field, it's no longer available as an answer to other questions.

- **Other questions are yes/no, provide a set of options, or require you to enter text.**

- **Questions marked with an asterisk (*) require answers.**

- **Using the default answers results in a useful set of dashboards.** If you're not sure what to select, use the answer that's preselected. Some questions do not have preselected answers. In those cases, Tableau CRM reminds you to make a selection. If you're not happy with the result, you can delete the app and create it again.

- **You control only parts of the app with wizard settings, such as the filters used in dashboards.** We built the app to provide immediate value without much work on your part.
- **Only some answers make sense given the question's context.** The answer pick-lists contain many fields, but only some make sense for your app. For example, it's unlikely that you'd filter Accounts data by Photo URL, even though the Accounts object includes a Photo URL field.

🛑 **Important:** Fields that aren't available to the Analytics Integration User aren't visible in the wizard. See Schedule Data Sync and the Dataflow for an Tableau CRM Template to give the Integration User access to all the fields you want your app to use.

Accounts Object Questions, Sales Analytics Custom Wizard Step 4 of 9

📝 **Note:** This page is labeled *Step 4 of 9*, even though it's the first page of questions you see after you select custom create. This is the fourth of nine steps for creating Sales Analytics with the custom option.

- Question 1, mandatory: Sales Analytics dashboards provide two filters for customer data from the Accounts object. Your answer to this question designates the field used in the first filter.
- Question 2, mandatory: Your answer designates the field used in the second customer data filter.
- Question 3, mandatory: Sales Analytics dashboards provide two filters for source data. Your answer designates the from Accounts to use for the first source filter. (The second pertains to the Opportunities object, which is covered in the next wizard page.)
- Question 4, mandatory: Sales Analytics dashboards provide two filters for customer geography data from the Accounts object, and your answer to this question designates the field used in the first filter.
- Question 5, mandatory: Your answer designates the field used in the second geography data filter.
- Question 6, mandatory: Your answer designates whether the app's dashboards include account hierarchy data. The Salesforce account hierarchy lets you view relationships between parent accounts and their subsidiaries.

Opportunities Questions, Sales Analytics Custom Wizard Step 5 of 9

Tell Sales Analytics how you prefer to view data from the Opportunities object.

📝 **Note:** This page is labeled *Step 5 of 9*, even though it's the second page of questions you see after you select custom create. This is the fifth of nine steps for creating Sales Analytics with the custom option.

- The selections made on the page that lets you add data to the app determine the first question asked on this page. All are mandatory. See chart for details.

Selections made on page that lets you add data to app	Question contents
– Add *none of* Collaborative Forecasting, Opportunity Splits, or Product Schedules. – Add *only* Collaborative Forecasting.	The question asks you to select the field used to display data about the total amounts of opportunities. Sales Analytics also checks to see if your org supports Historical Trending and displays results here. If you use a custom amount field, Sales Analytics instructs you to select that field.

USER PERMISSIONS

To create and manage Tableau CRM apps:
- Manage Analytics Templated Apps
- Edit Analytics Dataflows

Selections made on page that lets you add data to app	Question contents
- Add *only* Opportunity Splits. - Add *both* Collaborative Forecasting *and* Opportunity Splits.	The question asks you to select the opportunity type you use. Sales Analytics also checks to see if your org supports Historical Trending and displays results here. If you use a custom amount field, Sales Analytics instructs you to select that field. 📝 Note: Selecting both Opportunity Splits and Collaborative Forecasting pulls in quota data for the split type you've selected in your org.
Add *only* Product Schedules.	You see only questions 2 and 3 (below).

📝 Note: To enable Historical Trending, go to Salesforce Setup, type `Historical Trending` in the Quick Find box. Select the Opportunity object and check the field(s) you want to enable. See Set Up Historical Trend Reporting

- Question 2, mandatory: Your answer designates the field used in the second source data filter, which comes from your opportunities. (The first pertains to the Accounts object, which is covered in the previous wizard page.)

- Question 3, required. Your answer determines the way dashboards identify that an opportunity in Salesforce is new business.

Products Questions, Sales Analytics Custom Wizard Step 6 of 9

Tell Sales Analytics how you prefer to view data from the Opportunities object.

📝 Note: This page appears only if you select Products on the page that lets you add objects to Sales Analytics. It's labeled *Step 6 of 9*, even though it's the third page of questions you see after you select custom create. This is the sixth of nine steps for creating Sales Analytics with the custom option.

To create and manage Tableau CRM apps:
- Manage Analytics Templated Apps
- Edit Analytics Dataflows

- Question 1, required. Your answer determines the field dashboards use to display data showing the total amount of a product on an opportunity. You see only questions 2 and 3 (below) if you select Product Schedules on the page that lets you add data to the app.

- Question 2, mandatory: Sales Analytics dashboards provide two ways of segmenting products data, for example by Product Family or Product Name. Your answer to this question designates the field used for the first method.

- Question 3, required. Your answer designates the the second method for segmenting products data.

- ⚠ Warning: If **Product Code** is available in the list of answers, do not select it. Selecting **Product Code** causes app creation to fail.

Questions To Fine Tune Sales Analytics, Custom Wizard Step 7 of 9

Answer questions about data visibility as well as account team, opportunity team, and record type data in Sales Analytics.

> 📝 **Note:** This page is labeled *Step 7 of 9*, even though it's the fourth page of questions you see after you select custom create. This is the seventh of nine steps for creating Sales Analytics with the custom option.

- Question 1: Lets you control user access to data in your app. Here are the available options:
  - Option 1 enforces Salesforce role hierarchy, which means that users can only see data in Opportunities and Accounts owned by them and their subordinates.
  - Option 2 enables team benchmarking. It lets users see data in Opportunities and Accounts owned by them and their subordinates. It also lets users see data from Opportunities and Accounts owned by others at the same level in the role hierarchy.
  - Option 3 lets all users see all Sales Cloud data regardless of role.

- Question 2, mandatory: Select how your app handles Account Team information. Defaults to Option 3, which excludes Account Team data. Select Option 1 to add Account Team member data to datasets or Option 2 to add Account Team data and grant record visibility to members of the team.

- Question 3, mandatory: Select how your app handles Opportunity Team information. Defaults to Option 3, which excludes Opportunity Team data. Select Option 1 to add Opportunity Team member data to datasets or Option 2 to add Opportunity Team data and grant record visibility to members of the team.

- Question 4, optional. Only appears if your app detects that you segment opportunities by record types. Choose record types and Tableau CRM excludes *unselected* record types from datasets. Leave this question blank to include *all* record types.

USER PERMISSIONS

To create and manage Tableau CRM apps:
- Manage Analytics Templated Apps
- Edit Analytics Dataflows

Questions About Leads Sales Analytics, Custom Wizard Step 8 of 9

Answer questions about how Sales Analytics displays data from the Leads object.

> 📝 **Note:** This page appears only if you select Leads on the page that lets you add objects to Sales Analytics. It's labeled *Step 8 of 9*, even though it's the fifth page of questions you see after you select custom create. This is the eighth of nine steps for creating Sales Analytics with the custom option.

- Question 1, mandatory: Choose the field you use to track lead quality. Your answer determines the primary way Sales Analytics groups leads.

- Question 2, mandatory: Choose the field you use to segment leads by source. Your answer determines how Sales Analytics groups leads by source.

- Question 3, mandatory: Choose the field you use to segment leads by geography. Your answer determines how Sales Analytics groups leads by geography.

- Question 4, mandatory: Choose the field you use to segment leads by customer type. Your answer determines how Sales Analytics groups leads by customer type.

USER PERMISSIONS

To create and manage Tableau CRM apps:
- Manage Analytics Templated Apps
- Edit Analytics Dataflows

Bring Data into Sales Analytics, Custom Wizard Step 9 of 9

Optional questions that let you add data to Sales Analytics.

> ✏️ **Note:** This page is labeled *Step 9 of 9*, even though it's the sixth page of questions you see after you select custom create. This is the ninth of nine steps for creating Sales Analytics with the custom option.

- Question 1, optional: Add fields from the Accounts object to Sales Analytics datasets.
- Question 2, optional: Add fields from the Opportunities object to Sales Analytics datasets.
- Question 3, optional: Add fields from the User object to the Sales Analytics Users dataset.
- Question 4, optional: Add custom fields from the Event object to the Sales Analytics Activities dataset.
- Question 5, optional: Add fields from the Product object to the Sales Analytics Oppty Products dataset. This question appears only if you select Products on the page that lets you add data to the app.
- Question 6, optional: Add fields from the Leads object to the Sales Analytics Leads dataset. This question appears only if you select Leads on the page that lets you add data to the app.
- Question 7, optional: Add fields from the Cases object to the Sales Analytics Cases dataset. This question appears only if you select Cases on the page that lets you add data to the app.

Delete Sales Analytics

Delete apps to start app creation all over or to get rid of apps you no longer use.

> 🛑 **Important:** If you delete your app, Tableau CRM doesn't retain the wizard answers you selected for that version of the app. You have to start the creation process over. If you wish to reuse settings from the app, keep it until you've recreated the app and then delete it.

1. Navigate to Tableau CRM Studio and open the app. Make sure you're viewing the app's landing page and that you can see the app name at upper left, with a list of dashboards in the center panel.
2. Click the triangle in the upper left corner and select **Delete**.

3. In the next screen, Sales Analytics asks you to confirm that you want to delete your app. If you are, click the **Delete XX Asset(s)** button.
4. Tableau CRM deletes your app.

Upgrade Sales Analytics App

Take advantage of the latest Sales Analytics features by upgrading your app when we release a new version.

The banner at the top of your app home page tells you that we've released a new version. It also provides a link to more information about the release. There's also a link inviting you to start the upgrade process in the left-hand column of the home page, just below the app name.

Here's how to upgrade your app.

1. Click either the "What's new" link in the banner or the "New version available" link in the left-hand column of your app's home page.

2. You can read information about the new version on top of the page that opens. At the bottom of the page, find the buttons: **Upgrade current app** and **Create new app**.

3. See descriptions of what the buttons do by hovering over them. If you click **Upgrade current app**, you overwrite your current app and all its assets, replacing it with an app based on the new version. This also deletes any customizations, including any new fields or object you've added to the dataflow or changes you've made to security settings and dashboard labels and colors. If you click **Create new app**, you create a second copy of your current app based on the new version. Decide which option suits your circumstances, and click the appropriate button.

4. ⚠ Warning: If you've customized your app, click **Create new app** instead of **Upgrade current app** to make an app copy based on the new version. This preserves the current version and any customizations, which you can then manually copy into the new version of the app.

   If you click **Upgrade current app**, you see a screen warning you that the upgrade option overwrites the current app and gets rid of any customizations you've made to it. If you're OK with that, check the box and click **Continue**. If you're not, click **Back** to return to the previous screen. Clicking **Continue** takes you to the configuration wizard. Skip ahead to step 6.

5. If you click **Create new app**, you're taken to the configuration wizard. You don't see a warning, since you're not overwriting your current app, which is preserved with any customizations you've made to it.

6. The configuration wizard is preloaded with the settings you chose last time you used the wizard. You can either keep those settings or change them. Go through each page of the wizard.

7. Once you complete the wizard, Tableau CRM shows a screen that indicates which assets are impacted by the upgrade. Review the screen to see how many datasets, dashboards, and lenses will be changed, deleted, or added by the upgrade. It also shows how changed assets will be modified—whether the change is to data or appearance. On the same screen, you have the option to download a file with code for changes made to the app. If you've customized the app, download and save the file so you can copy and paste customization code into the upgraded version.

8. *Upgrade option only*: If you're upgrading your app, click **Upgrade current app**. Remember, this overwrites any customizations you've made. You can also click **Back** to go back into the configuration wizard and change your selections or click the **X** in the upper-right corner to cancel.

9. *Create option only*: If you're creating a new app, name your app something different from the current version and click **Create new app**. This option saves your current app and all its customizations. You can also click **Back** to go back into the configuration wizard and change your selections or click the **X** in the upper-right corner to cancel.

If the link below the app name says "Reset app," you're using the latest version and don't need to upgrade.

Reconfigure Sales Analytics

To restore deleted or altered dashboards or change wizard settings, reconfigure an existing app.

⊘ **Important:** Reconfiguring your app gets rid of dashboard customizations, including fields or objects added to the dataflow. It also deletes actions you enable, or changes to security settings and dashboard colors and labels. If you've made any customizations, save copies of your dataflow definition file or dashboards. Then copy them into your reconfigured app.

1. Navigate to Tableau CRM Studio and open the app. Make sure you're viewing the app's landing page and that you can see the app name at upper left, with a list of dashboards below.

2. Click the **Reconfigure app** link below the app name. If you see **Upgrade** instead of **Reconfigure**, you need to upgrade to the new version of the app before you can use the reconfigure feature. See Upgrade Sales Analytics on page 1525.

3. Review the next screen carefully. It warns you that reconfiguring overwrites app customizations. If you're comfortable overwriting customizations, check the box and click **Continue**. If not, click **Back** or the **X** in the upper right corner.

4. Complete the app creation process described in Create and Share Sales Analytics on page 1514.

5. At the end of the process, you're asked to confirm that you want to reconfigure your app. This gives you one more chance to make sure you're comfortable overwriting customizations. If you are, click **OK**. Tableau CRM creates a new version of your app.

⊘ **Important:** If you delete your app, Tableau CRM doesn't retain the answers you selected when you created that version of the app, and you have to start the creation process all over again. If you wish to reuse settings from the app, keep it until you've completed recreating the app and then delete it.

Collaborative Forecasting and Quotas Data in Sales Analytics

Sales Analytics gives you a choice for how to include your team's quotas depending on whether you use the Sales Cloud Collaborative Forecasts feature to store quotas data.

📝 **Note:** Sales Cloud Einstein customers who use Sales Analytics must use Collaborative Forecasts to see quota data. They can't edit the quota data set.

You have the following options for including quotas data in Sales Analytics.

Table 11: Sales Analytics Quotas Data Options

Option	Sales Cloud Settings	Configuration Wizard Settings	Additional Setup Requirements
1	• Sales Cloud Collaborative Forecasts enabled • Collaborative Forecasts includes forecast quota data.	Select Collaborative Forecasting in first wizard screen that lets you add objects.	None. Sales Analytics automatically adds quota data from the Forecasts object.
2	• Sales Cloud Collaborative Forecasts enabled	Select Collaborative Forecasting in first wizard screen that lets you add objects.	Create .CSV file with quotas data, upload to Tableau CRM, rerun dataflow. See instructions below.

Option	Sales Cloud Settings	Configuration Wizard Settings	Additional Setup Requirements
	• Collaborative Forecasts *does not* include forecast quota data.		
3	Sales Cloud Collaborative Forecasts *not* enabled	No setting available. Wizard does not give you the option of adding Collaborative Forecasts data.	Create .CSV file with quotas data, upload to Tableau CRM, rerun dataflow. See instructions below.

✏️ **Note:** Selecting both Opportunity Splits and Collaborative Forecasting pulls in quota data for the split type you've selected in your org.

Wizard settings are optional, depending on whether you want to include Sales Cloud Collaborative Forecasts data in your app.

With Option 1, Sales Analytics automatically adds quotas data to your app and its dashboards. With Options 2 and 3, add quotas data to your app by following these instructions.

✏️ **Note:** When you work with .CSV files you want to import to Sales Analytics, create and open them using only a UTF-8-compliant text editor. Opening them in Microsoft Excel or other spreadsheet software reformats .CSV files and makes them unusable in Sales Analytics

🛑 **Important:** When you create the .CSV file, be sure it contains the following fields, in this order, with exactly these names. Field names are case-sensitive:

1. StartDate (in yyyy-mm-dd format)

2. QuotaAmount

3. OwnerName

4. Username

1. Create a .CSV file to include the fields just described, that is StartDate (in yyyy-mm-dd format), QuotaAmount, OwnerName, Username. For an example, see Sales Analytics Example .CSV File.

2. Save the file to a location that you can easily remember.

3. In Salesforce, go to the Analytics Cloud home page and find the Quota dataset.

4. Click the arrow at the upper right corner of the dataset panel and select **Edit**.

5. Salesforce displays the dataset editing screen for the Quota dataset. Look for **Replace Data** in the upper right corner and click it.

6. In the dialog box that opens, navigate to the .CSV file you created in Step 1, and double-click it.

7. Click **Next** to open the Replace Dataset Data page.

8. If your fiscal period is different than calendar period, that is if it starts on a date other than January 1, update the Quota Metadata file. If your fiscal period starts on January 1, skip to the next step.

    a. Copy the JSON from Sales Analytics Quota Dataset JSON File and paste it into a text editor of your choice.

    b. Change the value of `"fiscalMonthOffset"` from 4 to a number that represents the month your fiscal period starts. In Sales Analytics metadata, the numeral "0" stands for January, "1" stands for February, and so on up to "11," which stands for December. Save the file to your desktop.

    c. In Tableau CRM Studio, go to the Replace Dataset Data page and locate the Data Schema File area of the page. Click the arrow next to Quota .JSON file, select Replace File, find the file you saved and upload it to Tableau CRM.

9. On the Replace Dataset Data page, click **Next** to open the Edit Field Attributes page. The first column —QuotaAmount— should be selected. If not, select it. In the **Field Attributes** panel on the right, make sure **Field Type** is set to **Measure**.

10. Click **Upload**. You're asked to confirm that you want to replace the file. Click **Replace** to upload the file.

11. After uploading your quota data, rerun the dataflow to update the dashboards.

    a. Click the Gear menu at the upper right of the Sales Analytics screen and select Data Manager.

    b. Select Dataflow view from the menu at the top left of the Data Manager screen.

c. Find your app; you may have to scroll down the page. Open the menu on the far right of the screen next to the app icon and name, and click

it. The dataflow assures that Sales Analytics has your company's latest sales data. You can learn more about dataflows from Schedule the Sales App Daily Dataflow.

Edit the Forecast Dashboard Data Source Connection Make Sure Quotas Data Is Accurate

If Sales Analyticsquotas numbers look higher than expected in the Forecast dashboard, it may be because the app counts some quotas twice. To make sure that the dashboard accurately reflects your team's quotas, specify the dataset field the dashboard uses as its data source

Sales Analytics Example .CSV File

Here's an example of the .CSV file you create to update the Sales Analytics Quota (Target) dataset.

Sales Analytics Quota Dataset JSON File

Use this file to update the Sales Analytics Quota dataset with your fiscal year start date if it's *not* January 1.

Edit the Forecast Dashboard Data Source Connection Make Sure Quotas Data Is Accurate

If Sales Analyticsquotas numbers look higher than expected in the Forecast dashboard, it may be because the app counts some quotas twice. To make sure that the dashboard accurately reflects your team's quotas, specify the dataset field the dashboard uses as its data source

The Forecast dashboard automatically rolls up quota amounts from a manager's direct reports. If the manager also has a quota assigned, that amount is rolled up with the team's quota amount. That can cause the manager's quota to be counted twice. To avoid double counting, assign different quota amounts to managers than the sum of their reps' quotas. Here's how to do that.

1.
Open the Forecast dashboard, and click [✏️] to edit it.

2.
Click [•••] in the upper right corner to open more options.

3.
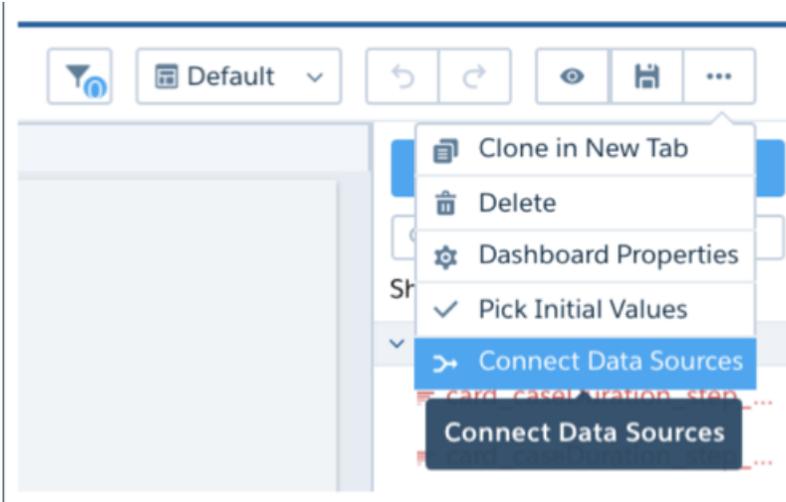
Select **Connect Data Sources**.

4.

In the **Data Source Connections** window, scroll to find the **RoleNames** connection and click [pencil icon] to edit
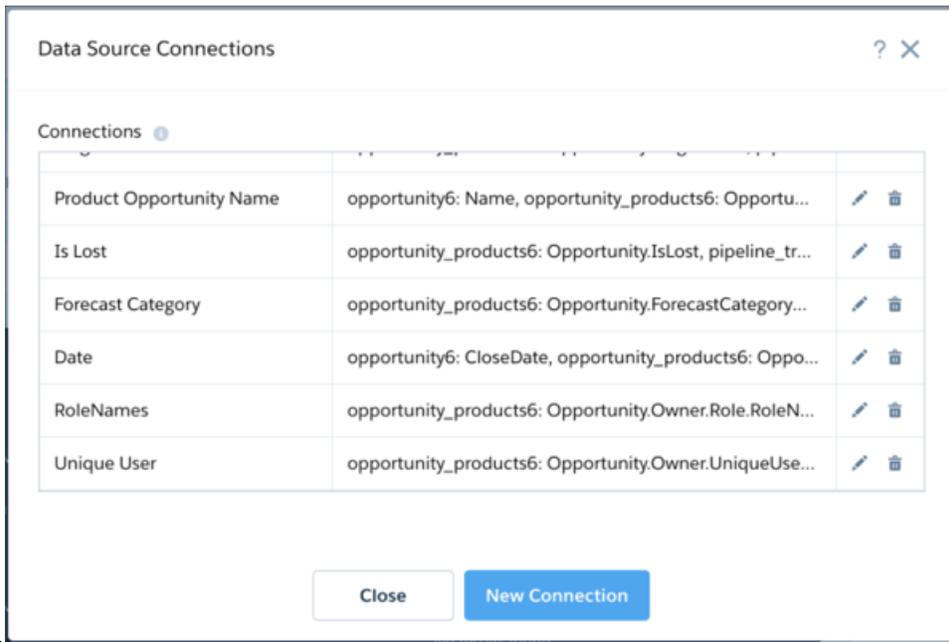
it.

5. In the **New Connection** dialog box, look for the **Data Source** field containing the word 'quota'. That's the data source for the Quota
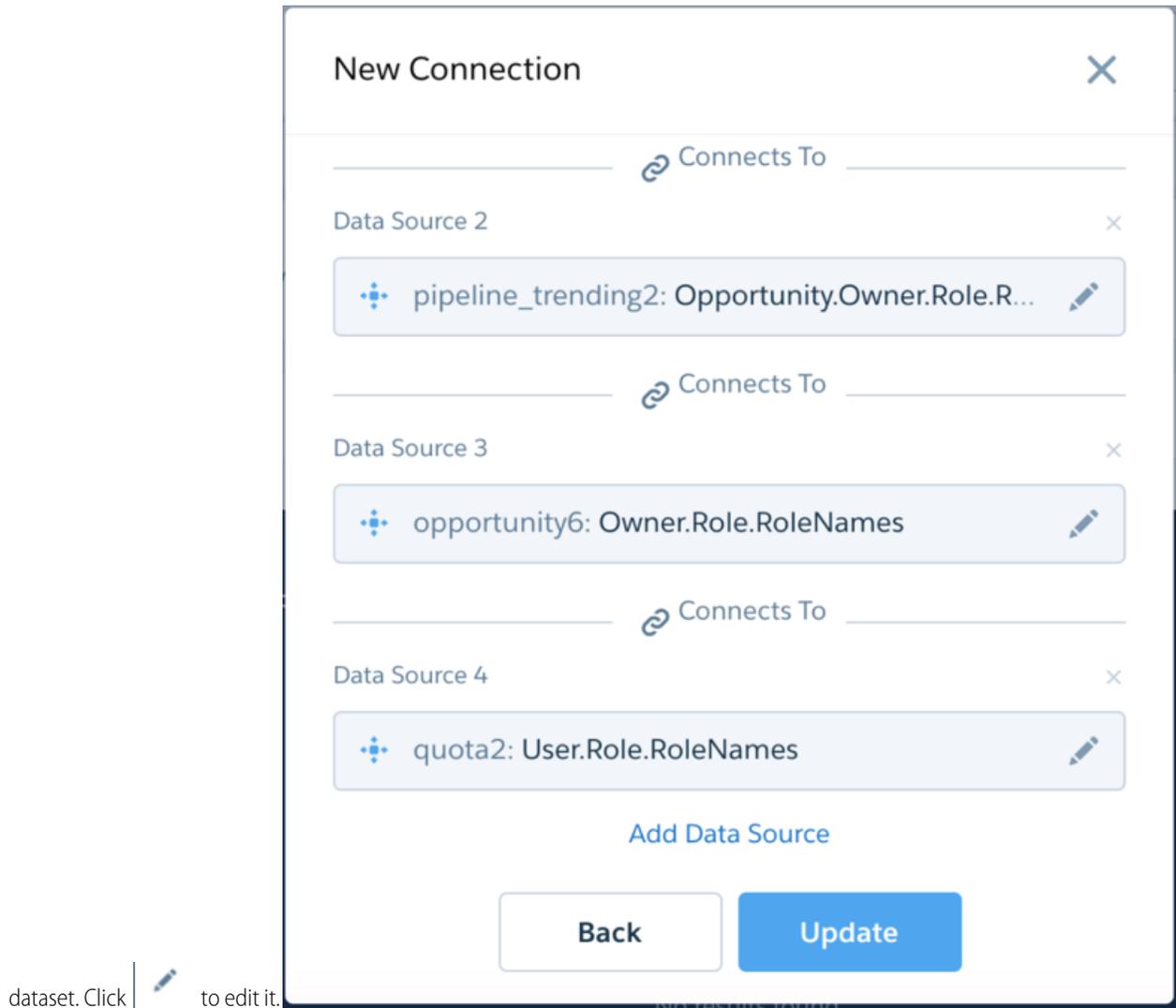
dataset. Click ✎ to edit it.

6. Select the **Quota** dataset. In the list of fields, scroll to find the **User.UniqueUserName** field and select it.

7. In the **New Connection** dialog, click **Update**. Then click **Close** in the **Data Source Connections** window.

8. Click the disk icon in the upper right corner to save the dashboard.

Sales Analytics Example .CSV File

Here's an example of the .CSV file you create to update the Sales Analytics Quota (Target) dataset.

📝 Note: This file is for example purposes only. Create a unique .CSV file with quota data for members of your team including the following fields:

- QuotaAmount
- StartDate
- OwnerName
- Username

Save the .CSV file in UTF-8 format. Field names are case-sensitive and must appear in your file exactly as shown here.

> ⊘ **Important:** Do not open the .CSV file with Microsoft Excel or another spreadsheet application, which can corrupt the file format.

See Collaborative Forecasting and Quotas Data in Sales Analytics on page 1526.

👁 Example:

```
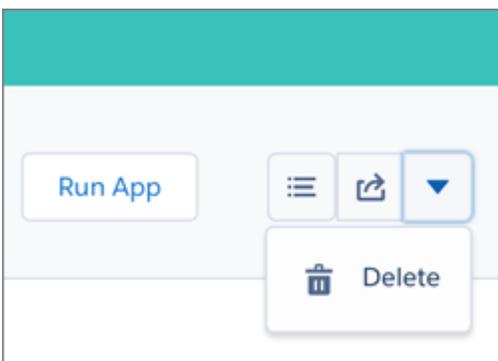QuotaAmount,StartDate,OwnerName,Username
150000,2016-01-01,Chris Riley,trailhead9.ub20k5i9t8ou@example.com
150000,2016-02-01,Chris Riley,trailhead9.ub20k5i9t8ou@example.com
150000,2016-03-01,Chris Riley,trailhead9.ub20k5i9t8ou@example.com
150000,2016-01-01,Harold Campbell,trailhead14.jibpbwvuy67t@example.com
150000,2016-02-01,Harold Campbell,trailhead14.jibpbwvuy67t@example.com
150000,2016-03-01,Harold Campbell,trailhead14.jibpbwvuy67t@example.com
150000,2016-01-01,Jessica Nichols,trailhead19.d1fxj2goytkp@example.com
150000,2016-02-01,Jessica Nichols,trailhead19.d1fxj2goytkp@example.com
150000,2016-03-01,Jessica Nichols,trailhead19.d1fxj2goytkp@example.com
150000,2016-01-01,Catherine Brown,trailhead16.kojyepokybge@example.com
150000,2016-02-01,Catherine Brown,trailhead16.kojyepokybge@example.com
150000,2016-03-01,Catherine Brown,trailhead16.kojyepokybge@example.com
150000,2016-01-01,Kelly Frazier,trailhead7.zdcsy4ax10mr@example.com
150000,2016-02-01,Kelly Frazier,trailhead7.zdcsy4ax10mr@example.com
150000,2016-03-01,Kelly Frazier,trailhead7.zdcsy4ax10mr@example.com
150000,2016-01-01,Dennis Howard,trailhead4.wfokpckfroxp@example.com
150000,2016-02-01,Dennis Howard,trailhead4.wfokpckfroxp@example.com
150000,2016-03-01,Dennis Howard,trailhead4.wfokpckfroxp@example.com
```

Sales Analytics Quota Dataset JSON File

Use this file to update the Sales Analytics Quota dataset with your fiscal year start date if it's *not* January 1.

Copy the contents of this file into an editor of your choice and change the `"fiscalMonthOffset"` value (shown in **bold**) to the month your fiscal period begins. In metadata, the numeral "0" stands for January, "1" stands for February, and so on up to "11," which stands for December. In the code below, the number is set to "4", which stands for May. Use the number that represents the month your fiscal period begins. Then save the file and upload it to Sales Analytics following the instructions in Collaborative Forecasting and Quotas Data in Sales Analytics.

```
{
    "objects": [
        {
            "connector": "CSV",
            "fullyQualifiedName": "Quota_csv",
            "label": "Quota.csv",
            "name": "Quota_csv",
            "fields": [
                {
                    "fullyQualifiedName": "QuotaAmount",
                    "name": "QuotaAmount",
                    "type": "Numeric",
                    "label": "QuotaAmount",
                    "precision": 18,
                    "defaultValue": "0",
                    "scale": 0
                },
```

```
                {
                        "fullyQualifiedName": "StartDate",
                        "name": "StartDate",
                        "type": "Date",
                        "label": "StartDate",
                        "format": "yyyy-MM-dd",
                "fiscalMonthOffset": 4,
                "isYearEndFiscalYear": true

                },
                {
                        "fullyQualifiedName": "OwnerName",
                        "name": "OwnerName",
                        "type": "Text",
                        "label": "OwnerName"
                },
                {
                        "fullyQualifiedName": "Username",
                        "name": "Username",
                        "type": "Text",
                        "label": "Username"
                }
            ]
        }
    ]
}
```

Schedule the Sales Analytics Data Sync and Dataflow

Schedule a data sync and dataflow to rerun every day to assure that Sales Analytics uses up-to-date data.

When you create Sales Analytics, the creation process includes a dataflow that imports the latest Sales Cloud data to Tableau CRM. You can schedule a data sync and dataflow to rerun every day to assure that your app uses up-to-date Salesforce data. Schedule the sync and dataflow to take place sometime outside normal business hours so the processes don't interrupt your use of the app.

📝 Note: The Sales Analytics dataflow runs only once when you create the app. Schedule it to run daily so the app uses the latest sales data.

You can also watch the video ▶ Build Interactive Tableau CRM Dashboards (English Only), which covers the steps described here.

1. In Tableau CRM Studio, click the wheel icon at upper right and select **Data Manager**. Or, click the **Data Manager** link in the left-hand column.

2. First, schedule the sync. Select the **Connect** tab on the left.

   📝 Note: If you can't see the Connect tab, you need to enable data sync in your org. See Enable Data Sync and Connections on page 693.

3. Click the arrow ▼ to the far right of **SFDC_LOCAL**, which is the name of the connection your app uses. From the menu that appears,

select **Schedule**.

4. Set a time for running the data sync. It's best to select a time outside normal working hours so the sync and dataflow don't interrupt business activities. Then click **Save**.

5. Next, schedule the dataflow. Select the **Dataflows & Recipes** tab on the left.

6. Look for the dataflow that contains the name of your app, and click the triangle ▼ to the far right.

7. Select **Schedule**, then check the box next to **Event-based**. You see a message telling you that the dataflow runs after the data sync—exactly what you want.

8. Click **Save**.

The sync and dataflow for your app now runs ever day at the time you set.

Integrate Sales Analytics With Salesforce

USER PERMISSIONS		EDITIONS
To use Tableau CRM templated apps:	Use Analytics Templated Apps	Available in Salesforce Classic and Lightning Experience.
To use Sales Analytics:	Access Sales Cloud Analytics Templates and Apps	
To create and manage Tableau CRM apps:	Manage Analytics Templated Apps	Available with Tableau CRM, which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.
	Edit Analytics Dataflows	
To edit a dataset's extended metadata (XMD) file:	Edit Analytics Dataflows	

Make Sales Analytics more usable by performing a variety of optional integrations and customizations.

Overriding role names in datasets, embedding the "Sales Analytics - Account Overview" dashboard in your Salesforce Accounts page, and expose Sales Analytics as a tab in Salesforce Classic can make it easier for your team to use the app.

Add Actions to Sales Analytics Dashboards

Sales Analytics dashboards use actions that are enabled on the Account and Opportunity page layouts. To add actions, you edit the page layout for those objects.

In Salesforce, go to the Accounts or Opportunities tab, open the page for an account or opportunity, and click the **Edit Layout link** at the top right of the page.

In the Quick Actions in the Publisher section, click the **override the global publisher layout link**. If you or someone else in your org has already done this, you won't see the link. Note that this section may be called something slightly different in the org you use, for example "Quick Actions in the Salesforce Classic Publisher."

In the Account Layout or Opportunity Layout panel (depending on which object you're adding actions to) at the top of the page, click **Quick Actions** on the left. Drag the action(s) that begin with the word "New" (for example, New Task, New Note) to the Quick Actions in the Publisher section.
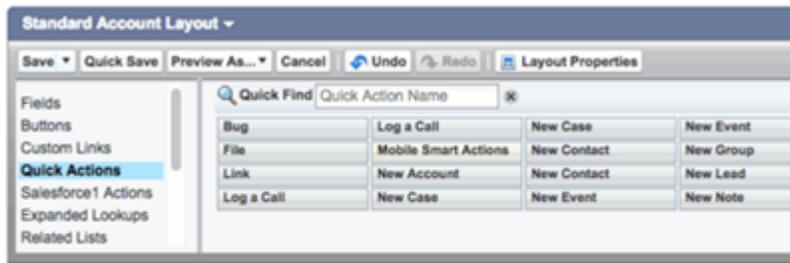
Click **Save** to save your changes. The actions you added to the layout are now available from charts and tables in Sales Analytics dashboards.

Override Role Name in Datasets

Sales Analytics uses Salesforce Role Hierarchy to help you to understand your company's sales performance. When you first create the app, Sales Analytics dashboards show the number associated with role name in the hierarchy. That's not the most useful way to view the role. To see the name of the person responsible for that role, you can update the role value by editing the corresponding dataset's Extended Metadata (XMD) file.

Let's say your role hierarchy is the following:

```
CEO (Manager: Paul)
                Sales_WW (Manager: Yves)
                        Sales_WEST
                        Sales_EAST
```

To have the manager's name appear in Sales Analytics dashboards, update the Opportunity dataset XMD file with the following:

```
{"labels": {
        "keys": {
                "RoleName": {
                "CEO": "Paul",
                "Sales_WW": "Yves"
                 },
            }
}}
```

For more information, see Analytics Cloud Extended Metadata (XMD) Reference.

Embed the Account Overview Dashboard in Account Page Layout

We've optimized the Sales Analytics Account Summary and Opportunity Summary dashboards to be embedded in the Salesforce Account page layout. You can do that by following the instructions in Embed Your Dashboard in a Salesforce Page.

To be able to filter the dashboard based on the account that you're viewing, use this string as filter:

```
{
```

```
        "opportunity1": {
                "AccountId": ["$Id"]
        }
    }
```

`opportunity1` corresponds to the system name of the Opportunity dataset. Its name can be different in your organization.

Make Sales Analytics Available in a Salesforce Classic Tab

You can access Sales Analytics by going to Analytics in the Lightning Platform menu and selecting the app from the Tableau CRM home page. To make it easier for your team to reach the app, you can expose it as a tab in Salesforce Classic. To do so, create a new tab that points to a Visualforce page that includes the "Sales Analytics - Overview" dashboard. Complete instructions are available in Add a Tableau CRM dashboard to a Visualforce Page.

Understand Sales Analytics Limitations

Sales Analytics requires that Sales Cloud include specific data and supports a limited set of Salesforce objects.

Sales Analytics Limitations

These limitations apply to Sales Analytics.

Sales Analytics Data Requirements

Sales Cloud data must meet the following requirements for Sales Analytics dashboards to function correctly:

- Use standard sales objects.
- Have at least one event and one task connected with an opportunity.
- Enable history tracking for Amount, Stage, and CloseDate fields on the Opportunities object.
- Cases must be connected to Accounts if you choose to import Cases data to Sales Analytics using the configuration wizard.
- There are two requirements if you choose to import Leads data to Sales Analytics using the configuration wizard.
    1. At least one lead must be converted to an opportunity.
    2. The lead must be connected to an account.
- There are two requirements if you choose to import Campaigns data to Sales Analytics using the configuration wizard.
    1. At least one opportunity must be connected to a campaign.
    2. At least one campaign member must be connected to a campaign.
- There are two requirements if you choose to import opportunity record types to Sales Analytics using the configuration wizard.
    1. At least one opportunity record type must be defined.
    2. The opportunity record type must be connected to at least one opportunity.

Sales Analytics Support for Salesforce Objects and Fields

Sales Analytics supports all Salesforce standard and custom objects and data. To add custom objects or additional fields not included when you first create the app, you need to update the Sales Analytics dataflow. For details see Design Datasets with Dataflows and the Dataset Builder on page 871.

When you first create the app it includes only a predefined set of objects and fields. Sales Analytics creates a dataflow that exposes selected fields from the following standard Salesforce objects:

- Accounts
- Users

- Roles
- Opportunities
- Products (Opportunities line item)
- Tasks
- Events

The fields provided are determined by the answers you provide when you create the app using the built-in configuration wizard.

Sales Analytics supports standard and custom fields on **standard** **Sales Cloud objects**. The fields provided are determined by the answers you provide when you create the app using the built-in configuration wizard.

Sales Analytics Support for Non-Salesforce Data

Sales Analytics does not support external data sources except a CSV file that contains quota data at the user level. Importing other external data requires an extra license. See your Salesforce representative for details.

To include data about quotas in Sales Analytics, you must upload a CSV file with the details. Sales Cloud Einstein customers who use Sales Analytics must use Collaborative Forecasts to see quota data. They can't edit the quota data set. For more information, see Collaborative Forecasting and Quotas Data in Sales Analytics.

Other Contractual Sales Analytics Limitations

The Sales Analytics Apps license does not support use of Tableau CRM platform bulk actions or Apex steps functionality. This limitation is contractual, not technical. Licensee agrees to strictly monitor its use of Tableau CRM platform features.

Sales Analytics Limitations for Sales Cloud Einstein and Tableau CRM Growth or Plus (Platform) Customers

Salesforce makes Sales Analytics available through three stock-keeping units (SKUs). Consult this chart to see limitations for each.

	Standalone	Sales Cloud Einstein	Tableau CRM Growth or Plus
Data sources	Salesforce data and CSV file for quotas data (see Sales Analytics Support for Non-Salesforce Data, above)	Salesforce data	Salesforce and external data
Object support	Standard and custom objects	Standard objects	Standard and custom objects
Data volume	25 million rows	25 million rows	- Tableau CRM Plus: 10 billion rows - Tableau CRM Growth: 100 million rows
Can customize existing dashboards?	Yes	Yes	Yes
Can create dashboards?	Yes	No	Yes
Can customize existing datasets?	Yes	No	Yes
Can create datasets?	Yes (using standard Salesforce objects and up to 10 custom objects)	No	Yes
Can create custom Tableau CRM apps?	No	No	Yes

	Standalone	Sales Cloud Einstein	Tableau CRM Growth or Plus
Supports Einstein Discovery and Experience Cloud integration?	No	No	Yes
Supports bulk actions and APEX steps?	No	No	Yes
Supports Sales Cloud Einstein artificial intelligence?	No	Yes	No
Supports Salesforce Inbox?	No	Yes	No

Get to Know Sales Analytics Data Terminology

To make the best use of Sales Analytics, it's helpful to understand the metrics and terms used in the app.

Table 12: Sales Analytics Dashboards

Metric/Term	Formula	Description/Notes
Closed Won		Amount of opportunities that have been closed and won.
Closed Lost		Amount of opportunities that have been closed and lost.
Closed Total	Closed Won + Closed Lost	Total amount of opportunities that have been closed.
Quota		Amount that was planned/committed/targeted for a certain time period.
Quota Attainment	Closed Won / Quota	How much of quota has been achieved.
Expected to Close	Many	Can use Forecast Category or Stage Name to calculate.
Forecast	Closed Won + Expected to Close	Amount that is expected/outlooked to end at for a time period.
Forecast to Quota	Forecast / Quota	Ratio of forecast to quota.
Need to Close	Quota - Closed Won	Amount needed to hit quota.
Need to Find	Quota - Forecast	Amount needed to hit quota, after accounting for opportunities expected to close in the future.
Open Pipe		Total amount of open opportunities in the pipeline, where Forecast Category does not equal Closed or Omitted.

Metric/Term	Formula	Description/Notes
Open Pipe Coverage	Open Pipe / Need to Close	How much in pipe to cover what is needed to close.
Bookings		Amount of Closed Won opportunities for new business.
Diff	This Year - Last Year	Difference between two items.
Y / Y %	(This Year / Last Year) - 1	Year-over-year growth rate.
Avg Selling Price	Total Amount ($) / Total Count (#)	Average selling price for Opportunities, Products, and so on.
Avg Win Rate by $	Closed Won / Closed Total	Success rate on closed opportunities, based on amount ($).
Avg Win Rate by #	Closed Won / Closed Total	Success rate on closed opportunities, based on count (#).
Avg Sale Cycle	Closed Date - Created Date	How long it took to close the deal.
Avg Discount %	abs ((Selling Price / List Price)-1)	How much was reduced from what was listed.

Einstein Discovery for Sales Analytics

Use the Einstein Discovery for Sales Analytics template to apply the power of Einstein Discovery predictive analytics to your Sales Cloud data.

USER PERMISSIONS

To create and manage the Einstein Discovery for Sales Analytics app:
- Tableau CRM Plus Admin permission set

To use the Einstein Discovery for Sales Analytics app:
- Tableau CRM Plus User permission set
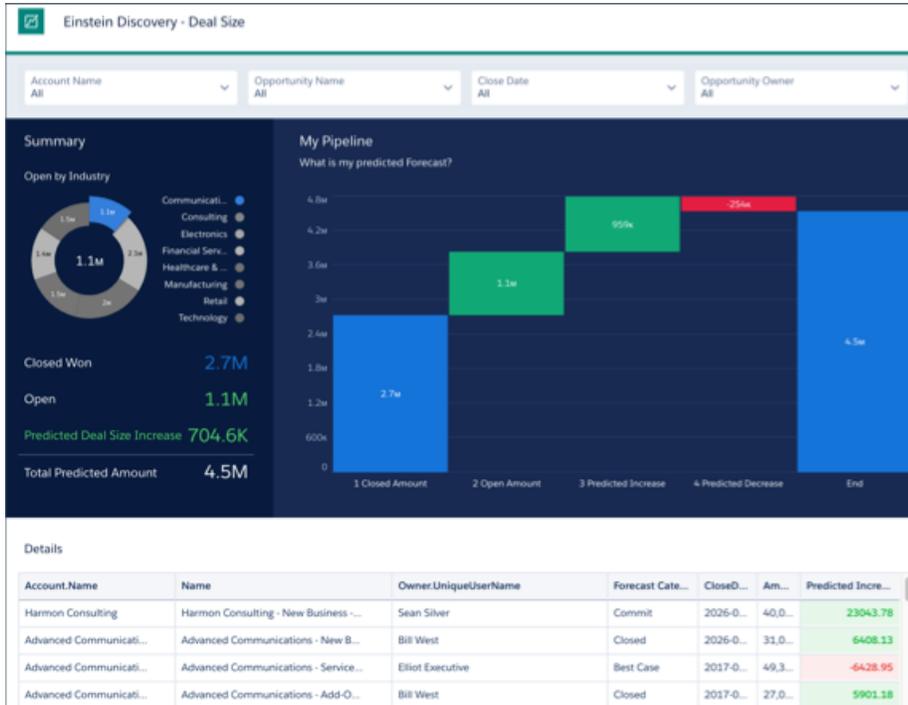
The template creates an Einstein Discovery story and a dataset and dashboard to help you focus on your most promising accounts. Identify your largest potential opportunities or the ones you have the greatest chances of winning or closing quickly. Then act on those accounts.

> ⛔ **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Einstein Discovery for Sales Analytics app.

Org Requirements

Before you create the Einstein Discovery for Sales Analytics, make sure you've created the Sales Analytics on page 1454 and scheduled its daily dataflow.

Also be sure you and all app users have the Tableau CRM Plus license.

Create the Einstein Discovery for Sales Analytics App

Preliminary Steps

1. In Tableau CRM Studio, click **Create** in the upper right corner

2. Select **App**, then **Create App from Template** to open the template picker.

3. Locate the **Einstein Discovery for Sales Analytics** tile, select it, and click **Continue**.

4. Review the app preview page, and click **Continue** to open the configuration wizard.

5. If you're offered a choice between basing your app on an existing app or creating a new one, select **Create a brand new app** and click **Continue**.

Create the Story and App

1. The wizard opens to the first page where you must make two selections.

a. **Choose an opportunities dataset.** The template uses its data to create an Einstein Discovery story. The first dataset in the pick list should be `Opportunities` from the Sales Analytics. We recommend using this dataset the first time you create the app.

b. **What would you like your prediction to help you do?** The pick list includes business results you'd like to improve with predictions from the story. For example, **Minimize Time to Close** predicts opportunities with a chance of closing before or after the dates initially determined by the sales team. Choose the result you'd like to improve.

2. Click **Looks good, next**.

3. That opens the **Add data to your predictive story** page, which lets you add data to improve your story's predictions. The most common fields are preselected. Select fields you want to add, or deselect fields to take away. Avoid fields with many entries, since they can skew your results. For more information about adding data to an Einstein Discovery story, see Create a Story. When you're done, click **Looks good, next**.

4. Name your app, and click **Create**.

Give Tableau CRM a few minutes to create your app. You can track its progress on the page that appears. When you see the **Application Complete!** message, refresh the page. You see your app page with a story, a dataset, and a dashboard showing the predictions. Click it to have a look.

You can add the dashboard to your Sales Analytics app by doing the following:

1. In the upper-right corner, click the three dots icon. Then select, **Clone in New Tab**.
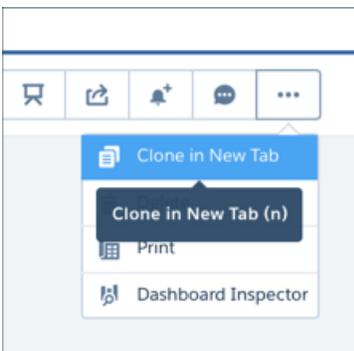
2. In the upper-right corner, click the disk icon to open the Save dialog box. In the **App** field, select the Sales Analytics app you want to include the predictive dashboard. You might have given it a unique name.

3. Click **Save**. The new copy of the dashboard now appears with the other dashboards in your Sales Analytics app.

📝 Note: The Einstein Discovery story can't be updated with the **Reconfigure App** action. To change your story, you must create a new Einstein Discovery for Sales Analytics.

Service Analytics

The Service Analytics template gets you started fast with Analytics and provides a clear path through your Service Cloud data on any device. Whether you're a service manager or agent, you get everything you need in one place to uncover key data insights to help you grow your business.

💡 Tip: Follow the steps in the order shown to get started with Service Analytics. If you haven't used Analytics before, learn more about it from the Analytics Documentation.

1. About Service Analytics

   Salesforce created the Service Analytics template to make it easy for service managers and agents to use data to drive the success of your service business. Learn the app's benefits before you create and use the app to explore your Service Cloud data.

2. Service Analytics Prebuilt Dashboards and Datasets

   The Service Analytics app includes prebuilt dashboards and datasets to accelerate data exploration for both service managers and agents.

3. Set Up Permissions for the Service Analytics

   Set up your organization to use the Service Analytics by enabling Tableau CRM and creating and assigning permission sets.

4. Create Service Analytics

   Follow these steps to create Service Analytics and start uncovering the value of your Salesforce data—fast.

5. Customize Service Analytics with the Configuration Wizard

   Create Service Analytics with the configuration wizard so the app reflects how your company prefers to view Service Cloud data.

6. Share Service Analytics

   Members of your team can only use Service Analytics if you share it with them.

7. Reconfigure Service Analytics

   To restore deleted or altered dashboards or change wizard settings, reconfigure an existing version of Service Analytics.

8. Delete Service Analytics

   Delete apps to start app creation all over or to get rid of apps you no longer use.

9. Upgrade Service Analytics

   Take advantage of the latest Service Analytics features by upgrading your app every time we release a new version.

10. Schedule the Service Analytics Data Sync and Dataflow

    Schedule a data sync and dataflow to rerun every day to assure that Service Analytics uses up-to-date data.

11. Embed Service Analytics Sidebar Dashboards in a Salesforce Page

    Add Service Analytics dashboards to Salesforce pages so your service team can get a complete view of their cases and take appropriate action.

12. Understand Service Analytics Limitations

    Service Analytics requires that Service Cloud include specific data, initially supports a limited set of Salesforce objects, and has other miscellaneous limitations.

About Service Analytics

Salesforce created the Service Analytics template to make it easy for service managers and agents to use data to drive the success of your service business. Learn the app's benefits before you create and use the app to explore your Service Cloud data.

Service Analytics dashboards give you best-practice key performance indicators (KPIs) about your Salesforce service data in a single place. We've based the app's dashboards on learnings from many years of helping businesses manage customer relationships. The goal is to provide the right amount of information at the right time to help both managers and agents make the right decision.

Service managers get a complete view of service customer data that includes trending as well as historical and peer benchmarks. Agents can quickly view a snapshot of each case and customer to help them make quick decisions about their next customer interactions.

We've done a lot of the hard work: Our team has built complex queries, formulas, and ratios that draw from your service and sales data and assembled them into easy-to-read visualizations. You just create the app: using a built-in configurator, answer a few questions about the data and fields you'd like to see, and Tableau CRM takes care of the rest. Once you've created the app, use its prebuilt datasets and dashboards to explore Service Cloud data from any device that supports Tableau CRM.

You get actionable insights fast from your data using the intuitive Tableau CRM interface. And you can drill deeper into key aspects of your business by customizing Service Analytics around your needs.

Note: Your organization can use Service Analytics without Tableau CRM platform by purchasing a Service Analytics license. A Service Analytics license is included with the Tableau CRM platform license.

Service Analytics Prebuilt Dashboards and Datasets

The Service Analytics app includes prebuilt dashboards and datasets to accelerate data exploration for both service managers and agents.

📝 **Note:** The dashboards and datasets included in your instance of Service Analytics can differ. The ones you see depend on how you answer configuration wizard questions when you create the app.

Service Analytics Dashboards

Service Analytics prebuilt dashboards contain best practices that help you get value from your Salesforce data—fast. The dashboards let you manage service cases and forecast and understand key business performance drivers, visualize trends, assign actions. They also help you get fast answers to questions you have about your service business's results.
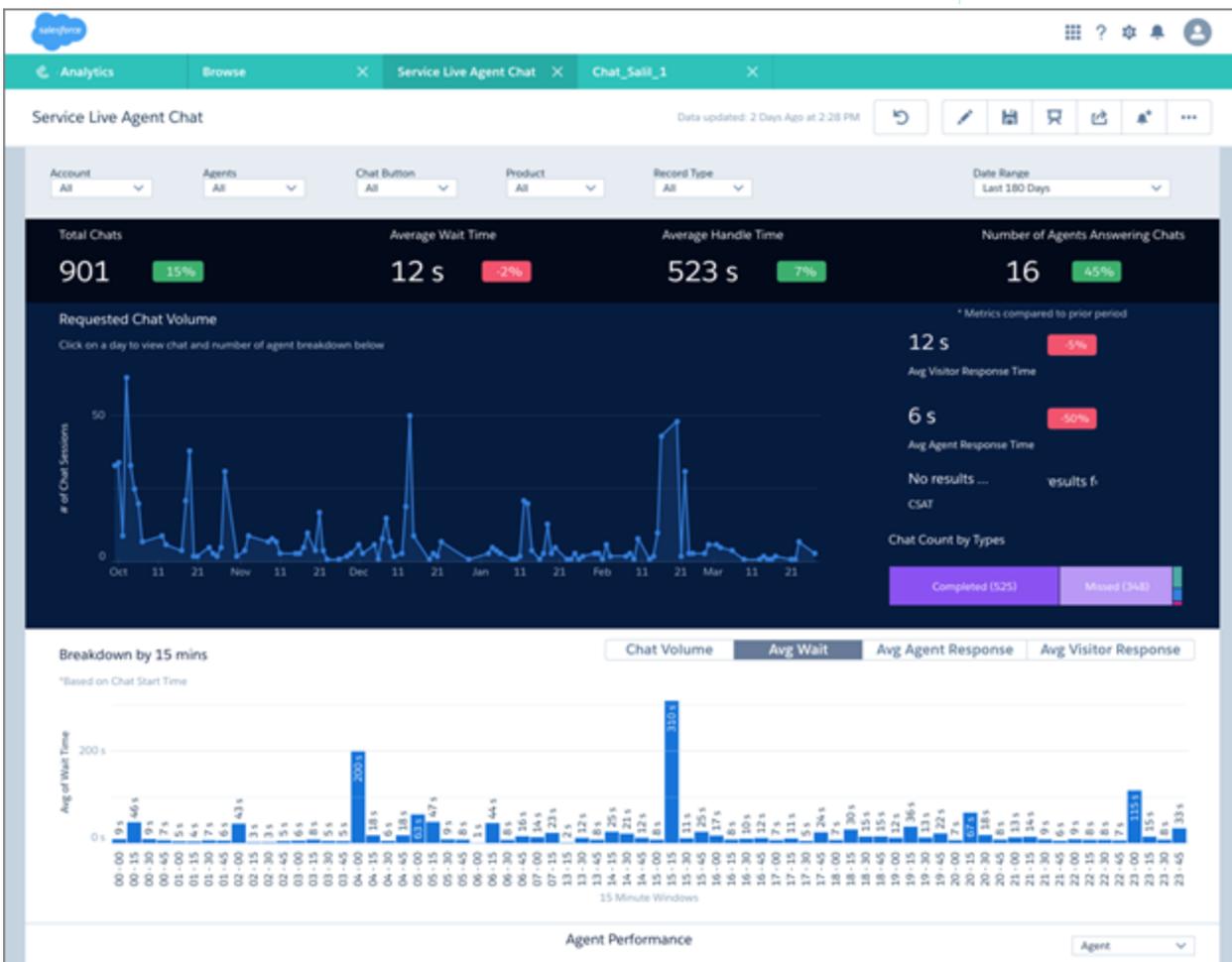
Dashboards available from the app provide a complete view of service customer data that includes all your key service performance indicators in one place. Managers can quickly view average case closing times, customer satisfaction, trending, and historical and peer

benchmarks. They can also get insight into the teams' use of knowledge to resolve cases and other data so they can quickly take appropriate action.

Another set of dashboards -- called sidebars -- are specifically for service agents. Embed sidebars in a Salesforce page, such as the service console, to make them more accessible to agents during their day-to-day work. For a given case, an agent can view customer history, number of cases, and CSAT. Agents can also see previous customer actions and their result so they can find out what worked. The goal is to provide agents the right amount of information at the right time to help them take the right action.

The following table guides you through Service Analytics dashboards. You can also explore further on your own at any point. To learn more about exploring data in Tableau CRM, see

Table 13: Service Analytics Dashboards

Dashboard Name	Contents	Target User
Service Analytics Overview (Service Manager Overview)	Start here. Summarizes key performance indicators (KPIs) from your Service Cloud, including open cases in the backlog, agent average time to close, and first contact resolution rate. Helps you gauge health of the business and surface issues requiring further investigation. Also, provides springboard to all other dashboards.	Service manager
Account Profile	Provides a full breakdown of a selected customer's service history and current backlog and includes customer satisfaction (CSAT) data and trends. If you track Opportunity data, you can also see pending and closed deals.	Service manager
Agent Activity	Helps determine how well agents use their time. Shows how much work agents do and how their work aligns with case-closing numbers and duration as well as CSAT.	Service manager
Agent Performance	Shows agent and team performance against key activity and customer satisfaction metrics. Performance trends and benchmarks on agent performance help you provide direction and drive agent success.	Service manager
Channel Review	Shows CSAT and activities—duration and volume—on cases by channel to help you monitor success of each channel.	Service manager
Backlog	Gives complete view of your backlog to show how efficiently your team resolves cases and how quickly your backlog is growing. Helps you prioritize and acquire/assign resources accordingly.	Service manager

Dashboard Name	Contents	Target User
Customer Satisfaction	Highlights CSAT measures so you can identify specific areas to improve the service experience. Indicates most/least satisfied customers and associates agents, products, and channels with CSAT.	Service manager
Knowledge Impact	Provides a view into how knowledge articles attached to cases impact CSAT and resolution time. See which agents used the most articles to determine who needs training/coaching in the use of knowledge to help resolve service issues	Service manager
Knowledge Use	Helps you understand how agents use knowledge articles to help you drive article creation. See which articles are attached the most and the least and which have the most views or votes (high ratings) over time.	Service manager
Chat	Shows how your team uses chat to help solve support cases. Includes number of chats, days of the week with most activity, missed or abandoned chats, chat length, and average handle time.	Service manager
Omni	Gives managers insight into agent utilization based on Omni-Channel work record tracking. Includes incoming agent work volume, average speed to answer, average handle time, average active time, and other agent utilization metrics.	Service manager
Open Cases	Shows current open case workload to help you prioritize, investigate problematic cases, and view escalations and SLA compliance. Let's you prioritize cases by accounts and pending deals so you can take appropriate action.	Service manager
Telephony	Helps you understand the impact of telephone contact with customers during service case resolution. Filter call volume and duration by inbound or outbound call, the result of the call, and other factors. Also relates call volume to agent performance.	Service manager
Sidebar – By Customer	Provides a snapshot of customers, including the service products they're using, case reason, and priority. Also shows which	Service agent

Dashboard Name	Contents	Target User
	agents have helped customers and provides a CSAT overview.	
Sidebar – By Similarity	Shows a snapshot of historical data to help guide further customer interactions. Shows if there are other cases similar to the ones they're resolving. Helps agents identify potential product issues or candidates for creating knowledge and find agents who have worked similar cases.	Service agent
Sidebar – By Case History	Gives an overview of case history or lifecycle so agents can be more informed during customer conversations. Includes a snapshot of case interactions and duration of case against the average.	Service agent
My Performance Summary	Gives agents visibility to their own productivity so they're always aware of their results towards targets as tracked in Salesforce. Includes ranking within the support center, trends, and benchmarking so they're always prepared for meetings with managers and reviews. Typically results in higher agent satisfaction and performance and lower turnover rates.	Service agent

Service Analytics Datasets

The table lists all the possible standard datasets that become part of Service Analytics when you create the app.

Table 14: Service Analytics Datasets

Dataset Name	Contents
Service Activity	Replaces Service Event and Service Task datasets used in Classic version of app. Data about events and tasks associated with cases, including last modified date and case duration. (Your Service Cloud data must include at least one event and one task for Service Analytics dashboards to function correctly.)
Service Case	Data about cases, including duration, last modified date, and customer satisfaction measure.
Service Case History	Data about case history, including owner, type, and creation date.
Service Chat Transcript	One of two Service Analytics datasets with data about your team's use of Chat usage. Includes data from Chat transcripts, such as response times, numbers of messages per chat, and chat duration.

Dataset Name	Contents
Service Chat Transcript Event	One of two Service Analytics datasets with data about your team's Chat usage. Includes data about Chat events, such as if chats are accepted, queued, declined, or transferred.
Service Knowledge	Data about use of knowledge articles in cases, such as number of views, ratings, last referenced dates, and case associations.
Service Knowledge Attached	Data about attachment of knowledge articles to cases including whether articles are shared by email, article version number, and case type.
Service Omni Agent Work	One of two Service Analytics datasets with data from Omni-Channel Agent Work object. Includes data that helps you track information about the assignments your agents work on, including the queue, channel, status, and speed to answer. (Your org must use Omni-Channel to create and route work items for Service Analytics to create this dataset. You must also answer Yes to the configuration wizard question about Omni-Channel.)
Service Omni User Presence	One of two Service Analytics datasets with data from Omni-Channel User Presence object. User presence records contain fields that help you track information about agent availability. (Your org must use Omni-Channel to create and route work items for Service Analytics to create this dataset. You must also answer Yes to the configuration wizard question about Omni-Channel.)
Service Opportunities	Data about opportunities so you can filter cases by opportunity for prioritization and appropriate action.
Service Event (*Only in Classic version of app.*)	Data about events associated with cases, including last modified date and case duration. (Your Service Cloud data must include at least one event for Service Analytics dashboards to function correctly.)
Service Task (*Only in Classic version of app.*)	Data about tasks associated with cases, including last modified date and case duration. (Your Service Cloud data must include at least one task for Service Analytics dashboards to function correctly.)

Set Up Permissions for the Service Analytics

Set up your organization to use the Service Analytics by enabling Tableau CRM and creating and assigning permission sets.

⛔ **Important:** Customers set up to use Tableau CRM Growth licenses have the required user permissions to access Service Analytics. The following instructions apply only to customers with the Service Analytics license.

USER PERMISSIONS

To set up Tableau CRM users:
- Manage Analytics

Each Service Analytics license is a single-user license that provides access to Service Analytics. It includes a single Service Analytics app single-user license. The table shows data storage limits for the app. If you require more data, you can purchase Analytics Cloud - Additional Data Rows, which entitles you to an additional 100 million rows.

Table 15: Service Analytics Data Storage Limits

License	Limit
Analytics Cloud - Service Analytics	25 million rows when used without Tableau CRM Growth license. Use of Service Analytics app license does not increase data limit for platform license.
Analytics Cloud - Additional Data Rows	100 million rows.

⊙ **Important:** Service Analytics license data storage limits are contractual, not technical. Licensee agrees to strictly monitor its total number of data rows.

Your org can use the Service Analytics with or without the Tableau CRM platform.

To give administrators or users in your org access to the Service Analytics, enable Tableau CRM and assign ready-made permission sets to them. By assigning permission sets, you also assign the Service Analytics permission set license.

⚠ **Warning:** Assign the Service Analytics Admin permission set sparingly because it lets users administer Service Analytics, which lets them create, edit, and delete the app. Assign it only to users who administer or manage the app. Users with **Service Analytics User** and Editor or Manager access to the app can create, edit, and delete app assets.

1. First, enable Analytics. In Salesforce Setup, enter `Analytics` in the **QuickFind** (search) field, then click **Getting Started**.

2. Click **Enable Analytics**.

3. Now, assign permission sets. In Salesforce Setup, enter `Users` in the **QuickFind** field.

4. Click **Permission Sets**.

5. Scroll through the list of permission sets until you see **Service Analytics Admin** and **Service Analytics User**.

6. For users who require administrator-level access to Service Analytics, assign the **Service Analytics Admin** permission set. This permission set enables all user permissions for the app shown in the table.

    a. Click **Service Analytics Admin**.

    b. Click **Manage Assignments**, then **Add Assignments**.

    c. Check the boxes next to the names of the users who require administrator-level access to the app.

    d. Click **Assign**, then click **Done**.
        The selected users can now create and manage Service Analytics.

7. To assign user-level access to Service Analytics, assign the **Service Analytics User** permission set.

    a. Click **Service Analytics User**.

    b. Click **Manage Assignments**, then **Add Assignments**.

    c. Check the boxes next to the names of the users who require access to the app.

    d. Click **Assign**, then click **Done**.
        The selected users can now use Service Analytics.

After you've completed these steps, create and share Service Analytics with the users in your organization. Users can only explore Service Analytics dashboards and datasets after you've shared the app with them.

Create Service Analytics

Follow these steps to create Service Analytics and start uncovering the value of your Salesforce data—fast.

For rapid app creation, choose the basic creation option, which uses default settings to create Service Analytics. To set up the app according to your team's specific Service Cloud analytics requirements, choose custom creation.

1. Log in to Salesforce.

> (!) **Important:** Service Analytics requires that your org has at least one closed case and one task. It also requires that your org has at least one contact ID associated with a case.

2. Set Salesforce field-level security to enable the Tableau CRM Integration User to see all fields you'd like your app to use. Integration users run the dataflow, and if they don't have proper field-level security permissions, the dataflow can fail.

   a. Go to Setup and enter the name of an object—for example, Accounts or Cases—in the Quick Find box and hit Enter.

   b. Click the name of the object.

   c. The next window shows all the fields for the object. Go to the one(s) where you need to edit field-level security.

   d. Look for the Analytics Cloud Integration User, check the box(es) for the required fields under Visible, and click Save.

   e. Repeat the Steps a through d for all objects with fields you want to use.

   f. Refresh your browser cache.

3. From the App Launcher (⠿), find and open **Analytics Cloud Studio**.

4. *If you've created an app before:* Choose between creating a brand new app or creating an app based on settings from a previously-created app. Click **Continue**. Service Analytics scans your org's data and features. For details about the scan, see Checking Data and Features with the Service Analytics Configuration Wizard.

   > (💡) **Tip:** If you want to create your app with different settings, repeat the previous steps and select the previously-created app here. Complete app creation choosing different wizard settings and give your app a new name.

5. If the scan detects your org is missing required data, Service Analytics displays error messages telling you what to do. Follow the instructions and start app creation again. Position your cursor over tooltips [ⓘ] to see more detail. If there are no errors, click **Looks good, next**.

6. Choose between basic and custom creation options to set up your app. Select **Basic** to set up your app quickly based on standard settings determined by the org data check. Select **Custom** to open the configuration wizard. The wizard helps fine-tune app setup to reflect the way your team wants to view data. See Customize Service Analytics with the Configuration Wizard.

7. If you choose **Basic**: Click **Looks good, next**. Answer the question about the field you use to categorize cases. Click **Looks good, next**, and skip to Step 10. See Service Analytics Wizard Basic Create Option.

8. If you choose **Custom**: The wizard asks you to choose Salesforce objects to add to Service Analytics. It then takes you through a set of questions about how you prefer to view data. See Service Analytics Wizard Custom Create Option on page 1556.

9. Give your app a name that's easily recognizable to others in your company and click **Create**. That starts a dataflow,which creates the app and its assets (a dataflow definition file, datasets, and dashboards). The app creation process can take a few minutes. During that process, Service Analytics runs a final scan to be sure the Analytics Integration User has access to all fields you select in the wizard. If the scan fails, you see an error message with the fields that can't be accessed. Return to Step 2 and give the Integration User access to those fields.

10. You can check the status of the dataflow while you wait. In Tableau CRM Studio, click the gear menu at the upper right of the page, and select **Data Manager**. From the pulldown menu, select **Dataflow View** and look for your app.

Taking a few extra steps can make Service Analytics better serve the needs of your organization:

- Embed the sidebar dashboards on page 1564 in a Salesforce page of your choice, such as the service console. This step provides a seamless analytics experience for service agents. Embedding sidebars in pages used regularly by agents gives them access to key metrics as part of their regular work to help the take the most effective customer action.

- Add datasets to the app based on Salesforce data using the Tableau CRM dataset builder on page 872. Service Analytics includes a default set of datasets based on your Salesforce data. Creating additional datasets lets your service managers and view other Salesforce data through the app.

  📝 **Note:** Service Analytics does not support data sources external to Salesforce or registering new datasets from the app-generated dataflow.

- Add objects, fields, and filters to Service Analytics. Edit the app's dataflow on page 963 to include standard and custom objects and fields from Salesforce.

  🛑 **Important:** When users are deactivated, they lose share and delete access to all apps they manage. To avoid "stranding" an app, be sure that manager access is assigned to at least one active user BEFORE deactivating the user who's the manager of the app.

  📝 **Note:** To create new datasets with just the Service Analytics app license, use the Tableau CRM dataset builder. Tableau CRM associates new datasets created using the builder with the Tableau CRM default dataflow. If you edit the dataflow for the Service Analytics to register new datasets, you can't access the resulting datasets.

Customize Service Analytics with the Configuration Wizard

Create Service Analytics with the configuration wizard so the app reflects how your company prefers to view Service Cloud data.

When you start the app creation process, Service Analytics opens the configuration wizard. The wizard guides you through the following steps.

1. Checks your org to be sure it meets minimum data requirements and to detect features that can be added to your app. The results let you know if you have to add data or change Salesforce settings to create the app. It also lets you know about available features.

2. Asks you to choose between basic and custom create options. Basic is intended for first-time app users, while custom lets experienced administrators fine-tune app setup.

3. Choose the basic option, and Analytics Cloud creates the app quickly with default settings.

4. Choose custom and the wizard guides you through the steps to fine-tune your app.

   a. Add features to your app

   b. Answer a series of questions about how you use data in your org.

The following provide details about each using each part of the wizard. Read them in the order shown to get the best results when you create Service Analytics. Click the question mark [?] in the top-right corner of each page of the wizard to see help for that page.

1. Service Analytics Wizard Data and Feature Check

   At the start of app creation, Service Analytics scans your org. It checks to make sure you can create the app and looks for features and data to add to your app.

2. Choose Service Analytics App Creation Options

   After Service Analytics checks your org's data and features, choose between basics and custom app creation options.

USER PERMISSIONS

To create and manage Tableau CRM apps:
- Manage Analytics Templated Apps
- Edit Analytics Dataflows

3. Use the Service Analytics Basic Create Option

   Select the basic app creation option when you create Service Analytics for the first time or when you want to create the app quickly.

4. Use the Service Analytics Custom Create Option

   Custom app creation gives you fine-grained control over Service Analytics features and data.

5. How to Answer Service Analytics Custom Wizard Questions

   After you add features to Service Analytics, the custom version of the configuration wizard asks you a series of questions about your data. Follow these general guidelines, and the specifics that follow, to get the best result.

6. Data Drill Down Questions, Service Analytics Custom Wizard Step 5 of 6

   Tell Service Analytics how you prefer to drill into data about cases, including category, type, severity, owner, and reason, and if they're resolved at first contact.

7. Questions About Features Added to Service Analytics; Custom Wizard Step 6 of 6

   Provide more specifics about how Service Analytics uses data from the features you add to the app.

8. Set Up Service Analytics CSAT Metrics: Example

   This scenario provides detail to help you answer questions about CSAT data in Step 6 of the Service Analytics configuration wizard.

Service Analytics Wizard Data and Feature Check

At the start of app creation, Service Analytics scans your org. It checks to make sure you can create the app and looks for features and data to add to your app.

Service Analytics checks your org for data and features when you open the configuration wizard. If it finds any issues you need to correct before creating the app, you see messages that tell you what to do. Position your cursor over the tooltip [i] for more information.

Here are details about each phase of the check.

Minimum Requirements

Your org must have at least one closed case and one closed task to enable app creation. If you see an error message, go to Service Cloud, close a case and a task, and try creating the app again.

Features to Add to Your App

The wizard checks to see if you track data about the following in your org:

- Customer satisfaction (CSAT)
- Knowledge
- Case history
- Opportunity
- Queues
- Events
- Business hours
- Case record types
- Opportunity record types
- Telephony
- Chat

- Omni-Channel

The tooltip tells you which are available in your org. Basic app creation automatically adds those features. Custom app creation lets you choose the ones to add.

 Note: CSAT and knowledge data can only be added using custom app creation.

Field-Level Security

The wizard checks if the Analytics Integration User has access to all fields needed to create the app using the basic create option. If the Integration User can't access all fields, you see a warning and the tooltip lists the fields that lack access. Go to Salesforce Setup and provide access to those fields. See Create Service Analytics on page 1552, Step 2, for details. Until all fields are available, you can't use the basic create option.

Fiscal Calendar Setting

The wizard looks for your org's fiscal year start date and uses it for Service Analytics. To choose a different fiscal start date, use the app's Custom create option.

Choose Service Analytics App Creation Options

After Service Analytics checks your org's data and features, choose between basics and custom app creation options.

Select **Basic** if you're using Service Analytics for the first time. It sets up your app quickly based on standard settings determined by the org compatibility check. You get an immediately useful version of the app so you can see how it works. Experiment with it and share it with your team. Based on what you learn, change the standard settings by recreating the app using the custom create option.

Service Analytics disables basic app creation if the Analytics Integration User can't access all required fields. If the basic option is disabled, click **Back** to return to the org check and follow the instructions in the field-level security tooltip and error message.

USER PERMISSIONS
To create and manage Tableau CRM apps:
• Manage Analytics Templated Apps
• Edit Analytics Dataflows

Select **Custom** and follow additional steps in the wizard to make your own, custom settings to reflect the way you and others on your team want to view data. The initial, default settings in the screens that follow are the ones used when you create an app using the basic create option. Custom create lets you vary these settings to meet your team's specific needs. You can choose to add or delete features detected by the org check. You can also make specific choices about the data used in the app's dashboards.

Whether you use basic or custom create, Service Analytics runs a final scan of your org's field-level security settings. The scan detects if the Analytics Integration User has access to all data fields required to create the app. If the scan fails, you see a message telling you how to fix the issue. For more information about field-level security settings, see See Create Service Analytics, Step 2.

Use the Service Analytics Basic Create Option

Select the basic app creation option when you create Service Analytics for the first time or when you want to create the app quickly.

The basic create option uses data, features, and default settings detected during the compatibility check that runs when you start app creation.

The wizard asks you a single, required question about the field you use to categorize cases. Choose the name of the field your business uses most to identify and organize cases, such as service product, type or level. Then click **Looks good, next**.

Name your app, and click **Create**.

USER PERMISSIONS
To create and manage Tableau CRM apps:
• Manage Analytics Templated Apps
• Edit Analytics Dataflows

Basic create can't use customer satisfaction and knowledge data. Service Analytics requires more information about how they're used in your org before creating an app. To add them, use the custom create option, and the wizard asks for the required details.

Basic create uses the following settings:

- To identify cases by geography, the Billing State/Province field from the Accounts object.
- Fifty (50) seconds as the maximum time agents can keep customers waiting during phone calls.

To vary these settings based on how you use Service Cloud, select the custom create option.

Consult Create Service Analytics on page 1552 for details about creating your app.

Use the Service Analytics Custom Create Option

Custom app creation gives you fine-grained control over Service Analytics features and data.

Add Data and Features

The first window in the custom creation process lets you add data and features to your app. By default, Service Analytics includes data from the following standard Salesforce objects:

- Account
- Cases
- Contact
- User
- UserRole
- Task

The compatibility check that run at the start of app creation looks for other available data and features. The wizard displays the results on the first custom create window, letting you know which features you use in your org. If you use a feature, this window tells you it's available to your app. If you don't use the feature, Service Analytics tells you it's not available.

Service Analytics preselects the features used for basic app creation. You have the option of adding others. Preselected options contain a check mark in the upper right corner. Add other available options to your app by clicking them. Click options with a check mark to deselect them. Here are the options Service Analytics lets you add, if available, and details about adding each.

- **Customer Satisfaction (CSAT) Score**. Select to add CSAT data to your app. If you select, Service Analytics adds questions to the wizard that ask how you track CSAT data. See Questions About Features Added to Service Analytics; Custom Wizard Step 6 of 6.
- **Knowledge**. Select to add Salesforce knowledge data to your app. If you select, Service Analytics adds questions to the wizard that ask how you track knowledge data. See Questions About Features Added to Service Analytics; Custom Wizard Step 6 of 6.
- **Business Hours**. Select to add a business hours case duration calculation to datasets. This uses data from the Business Hours field in the Cases object. Be sure that the Analytics Integration User has access to this field by editing field-level security for the Cases object. See Create Service Analytics, Step 2.
- **Case History**. Select to include historical tracking for both case owner and for the field selected in wizard Page 5, Question 4. See Data Drill Down Questions, Service Analytics Custom Wizard Step 5 of 6.
- **Case Record Types**. Select to add record type for the Cases object. If you select, Service Analytics lets you filter cases by record type.
- **Queues**. Select to add queues data from the Group object, which your org uses to assign case ownership. If you select, Service Analytics lets you filter cases by user and queues.
- **Opportunities**. Select to add data from the Opportunities object, which is connected to cases via accounts in your org. If you select, Service Analytics adds sales data to dashboards to give you a complete view of accounts.
- **Opportunity Record Types**. Select to add record type from the Opportunities object. If you select, Service Analytics lets you filter opportunities by record type.

- **Events**. Select to add data from the Events objects, which your org uses to track case activity. If you select, Service Analytics includes events data and combines it with tasks data so you can drill into cases by activities (that is, tasks and events).
- **Telephony**. Select to add telephony data from standard call fields in the Tasks object. Standard fields include call duration, call object identifier, call result, and call type. If you select, Service Analytics creates the Telephony dashboard showing data from those fields.
- **Chat**. Select to add Chat data. If you select, Service Analytics creates the Chat dashboard and dataset based on Chat data.
- **Omni-Channel**. Select to add data from Omni-Channel, which your org uses to create and route work items. If you select, Service Analytics creates the Omni-Channel dashboard and dataset.

How to Answer Service Analytics Custom Wizard Questions

After you add features to Service Analytics, the custom version of the configuration wizard asks you a series of questions about your data. Follow these general guidelines, and the specifics that follow, to get the best result.

- **Most questions provide answers in pick lists of fields from Salesforce objects.** The lists include standard Salesforce fields and custom fields you've set up on an object. Answer these questions by selecting from the fields shown. Most questions of this type let you choose only one field, and some let you choose multiple fields.
- **You can choose a field from an object only once.** After you select a field, it's no longer available as an answer to other questions.
- **Other questions are yes/no (Boolean), provide a set of options, or require you to enter text.**
- **Questions marked with an asterisk (*) require answers.**
- **Default answers result in a useful set of dashboards.** If you're not sure what to select, use the answer that's preselected. Some questions do not have preselected answers. In those cases, Service Analytics reminds you to make a selection.
- **You can only control parts of the app with wizard settings, such as filters and widgets.** We built the app to provide immediate value without much work on your part.
- **Only some answers make sense given the question's context.** The questions often contain many fields, but only some make sense for your app. For example, it's unlikely that you'd filter Accounts data by Photo URL, even though the Accounts object includes a Photo URL field.

Case Metrics Questions, Service Analytics Custom Wizard Step 4 of 6

> 📝 **Note:** This page is labeled *Step 4 of 6*, even though it's the first page of questions you see after you select custom create. This is the fourth of six steps for creating Service Analytics with the custom option.

- Question 1, optional: Asks you to indicate the primary field you use in Salesforce to track case duration, if any. If you don't track case duration, Service Analytics uses its own formula to calculate case duration based on the date a case was opened and closed. If the case is still open, it uses today's date as part of the calculation. Allows only a single selection. If you add business hours to the app, Service Analytics includes a business hours case duration calculation. See Use the Service Analytics Wizard Custom Create Option.
- Question 2, optional: Asks if you use other (secondary) fields related to case duration. You can select multiple fields, for example, **Duration with customer** and **Duration with agent**.
- Question 3, optional: Asks you to select the field you use to track SLA compliance. Typically, you would track SLA compliance with a custom formula field on the Cases object or the standard Milestone Status field. (Milestone Status field values include compliant, open violation, and closed Violation). If you track SLA compliance using a Boolean custom formula field, you can't select that field from the wizard. In that case, do the following: Select any field and then manually edit the dataflow to include the Boolean field. (See Configure the Dataflow Through the Definition File.) Then, find and replace the field name in Service Analytics dashboard JSON (see Analytics Cloud Dashboard JSON Reference).

- Question 4, optional: Asks you to select the field you use to record that a case is resolved on first contact. You would typically track first contact resolution using a Boolean custom formula field on the Cases object or the standard Closed when Created (IsClosedOnCreate) field.
- Question 5, optional: Asks if you'd like to include other Cases object metrics in your app. Choose a numeric field.

Data Drill Down Questions, Service Analytics Custom Wizard Step 5 of 6

Tell Service Analytics how you prefer to drill into data about cases, including category, type, severity, owner, and reason, and if they're resolved at first contact.

📝 **Note:** This page is labeled *Step 5 of 6*, even though it's the second page of questions you see after you select custom create. This is the fifth of six steps for creating Service Analytics with the custom option.

- Question 1, required. Asks about the field you use to categorize cases. Choose the name of the field your business uses most to identify and organize cases, such as service product, type or level.

  📝 **Note:** This is the only question you see if you use basic app creation.

- Question 2, optional. Asks about the field you use to record customers' type of support. Examples include gold, silver, or bronze; and premium and basic.
- Question 3, required. Asks which field you use to categorize case severity—usually the field you use to prioritize cases.
- Question 4, required. Asks which field you use to track case status of case. You filter and analyze cases in dashboards based on your selection.
- Question 5, required. Asks about the field you use to identify topics, reasons, and closure codes of cases. Your selection helps you understand which kinds of cases your team solves most efficiently.
- Question 6, required. Asks you to choose the field you use to track types of cases so you can filter and analyze cases according to support type.
- Question 7, required. Asks you to select the field used to track channels customers use to open cases, such as phone, email, mobile, or web.
- Question 8, required. Asks you to select the object used to track cases according to geography. Options are **Accounts**, **Contacts**, and **Cases**. Defaults to **Accounts**, which is used in basic create.
- Question 9, required. Asks you to indicate the field in the object selected in Question 8 to track cases by geography. Defaults to **Billing State/Province**, which is used in basic create.
- Question 10, required. Asks you to indicate your fiscal year start month. Defaults to January; select another month if your fiscal year is different from calendar year. Service Analytics supports only standard fiscal periods.
- Question 11, optional: Select dimension fields from the Cases object you haven't selected in previous questions. Adding dimensions lets you drill into dimensions that are important to your organization. Dimensions are qualitative values, such as date, region, and product name.

Questions About Features Added to Service Analytics; Custom Wizard Step 6 of 6

Provide more specifics about how Service Analytics uses data from the features you add to the app.

📝 **Note:** This page is labeled *Step 6 of 6*, even though it's the third page of questions you see after you select custom create. This is the last of six steps for creating Service Analytics with the custom option.

Most of the questions on this page appear only if you add CSAT and knowledge features to your app. See Use the Service Analytics Wizard Custom Create Option. Follow the instructions carefully to assure success.

- Question 1, required. You see this question only if you add CSAT to your app. Asks which object you use to track CSAT. Service Analytics defaults to **Cases**. If you use a different object to track CSAT, select it from the pick list and answer the additional questions that appear.

- Question 2, required. You see this question only if you add CSAT to your app. Asks you to select the field from the object selected in Question 1 you use to track CSAT.

  - Question 2a, required. You see this question only if you select an object other than Cases in Question 1. Service Analytics has to create a join between that object and Cases for you to see CSAT data in dashboards. (For an example, see Set Up Service Analytics CSAT Metrics: Example.) Enter the API name the field where you store case identification information. Usually, that's the **Id** field, in which case you enter `Id`.

    If it's a field other than **Id**, find the API name for the field. Go to **Setup—>Cases—>Fields**. In the top chart showing standard fields, field name column shows the API name. For example **IsSelfServiceClosed** is the API name for the **Closed by Self-Service User** field. The chart further down for custom fields includes an API Name column.

  - Question 2b, required. You see this question only if you select an object other than Cases in Question 1. Determine which field from the object selected in Question 1 to use for the other side of the join with the Cases object. Use the field that contains case identification. Enter the API name for the field here.

- Question 3, required. You see this question only if you added knowledge to your app. Asks about the article type you want to see data about in Service Analytics dashboards. Choose only a single article type. To see the names of article types in Service Cloud, go to Salesforce Setup and enter `knowledge` in the Quick Find box.

- Question 4, required. You always see this question in the wizard. Gives you the option of having Analytics Cloud check to see if data in your org supports the answers you select. We recommend that you select **Yes** and run the check, because app creation fails if your org doesn't contain the right data. Analytics Cloud displays an error message that tells you what to fix. Make any required changes in Salesforce, refresh your browser cache, and come back to the wizard and finish creating your app.

SEE ALSO:

Set Up Service Analytics CSAT Metrics: Example

Set Up Service Analytics CSAT Metrics: Example

This scenario provides detail to help you answer questions about CSAT data in Step 6 of the Service Analytics configuration wizard.

If you track customer satisfaction (CSAT) data, such as the CSAT score for a service case, in an object other than Cases, take particular care in how you answer the questions on this page.

For example, if you store the CSAT score in a custom object, for example "CSAT Survey", select this object in question 1. In Question 2, specify the field that has the CSAT metric you would like to report, for example "CSAT Score." Service Analytics needs to relate the custom object specified in Question 1 to the Cases object so dashboards can include data about cases, accounts, and agents by CSAT. This relationship is created through a join.

To specify the relationship between the Cases object and your custom CSAT object, designate join keys from both objects and enter their API names in Questions 3 and 4. On the Cases object, this will typically be the "Id" field. On the example custom object "CSAT Survey," this is the custom field that stores the case ID, which in this scenario is the field with the API name `CaseId__c`.

The following diagram shows the join between the two objects.

Share Service Analytics

Members of your team can only use Service Analytics if you share it with them.

You need to create an app before you can share it. See Create the Service Analytics App You can only share it with users who have the following permissions enabled:

- Access Service Cloud Analytics Templates and Apps
- Use Analytics Templated Apps
- Access Sales Cloud Analytics Templates and Apps

1. Open your app if it's not already open. If you've navigated away from Tableau CRM Studio, go back to it, select **All Items**, find your app, and click it.

2. Click the Share icon ⬆ at upper right.

3. In the next screen, use the search field under **Invite others:** to find other users in your org.

4. Select whether you want to make the selected user a Viewer, Editor, or Manager of the app.

   ⛔ Important: Users with the "Use Analytics Templated Apps" permission and Editor or Manager access to the app can create, edit, and delete assets in the app.

5. Click **Add**, then click **Save**.

Reconfigure Service Analytics

To restore deleted or altered dashboards or change wizard settings, reconfigure an existing version of Service Analytics.

> ⛔ **Important:** Important. Reconfiguring your app gets rid of dashboard customizations, including fields or objects added to the dataflow. It also deletes actions you enable, or changes to security settings and dashboard colors and labels. If you've made any customizations, save copies of your dataflow definition file or dashboards. Then copy them into your reconfigured app.

1. Open the app landing page. If you see **Upgrade** instead of **Reconfigure**, you need to upgrade to the new version of the app before you can use the reconfigure feature. See Upgrade the Service Analytics App.

2. Click the **Reconfigure app** link.
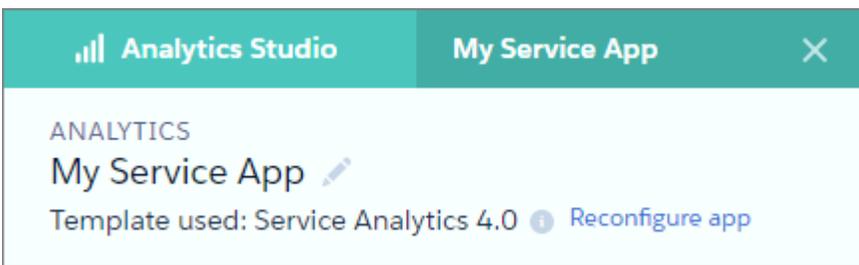
3. Review the next screen carefully. It warns you that reconfiguring overwrites customizations, including changes or additions you've made to your app. If you're comfortable overwriting customizations, check the box and click **Continue**. If not, click **Back** or the **X** in the upper right corner.

4. Complete the app creation process described in Create the Service Analytics App.

5. At the end of the process, you're asked if it's OK to reconfigure your app. This gives you one more chance to make sure you want to overwrite customizations. If you are, click **OK**. Service Analytics creates a new version of your app.

Delete Service Analytics

Delete apps to start app creation all over or to get rid of apps you no longer use.

> ⛔ **Important:** If you delete your app, Tableau CRM doesn't retain the wizard answers you selected for that version of the app. You have to start the creation process over. If you wish to reuse settings from the app, keep it until you've recreated the app and then delete it.

1. Navigate to Tableau CRM Studio and open the app. Make sure you're viewing the app's landing page and that you can see the app name at upper left, with a list of dashboards in the center panel.

2. Click the triangle in the upper left corner and select **Delete**.

3. In the next screen, Service Analytics asks you to confirm that you want to delete your app. If you are, click the **Delete XX Asset(s)** button.

4. Tableau CRM deletes your app.

Upgrade Service Analytics

Take advantage of the latest Service Analytics features by upgrading your app every time we release a new version.

The banner at the top of your app home page tells you that we've released a new version. It also provides a link to more information about the release. There's also a link inviting you to start the upgrade process in the left-hand column of the home page, just below the app name.

Here's how to upgrade your app.

1. Click either the "What's new" link in the banner or the "New version available" link in the left-hand column of your app's home page.

2. You can read information about the new version on top of the page that opens. At the bottom of the page, find the buttons: **Upgrade current app** and **Create new app**.

3. See descriptions of what the buttons do by hovering over them. If you click **Upgrade current app**, you overwrite your current app and all its assets, replacing it with an app based on the new version. Upgrading also deletes any customizations, including any new fields or object that you've added to the dataflow or changes you've made to security settings and dashboard labels and colors. If you click **Create new app**, you create a second copy of your current app based on the new version. Decide which option suits your circumstances, and click the appropriate button.

4. ⚠ Warning: If you've customized your app, click **Create new app** instead of **Upgrade current app** to make an app copy based on the new version. Creating a new app preserves the current version and any customizations, which you can then manually copy into the new version of the app.

   If you click **Upgrade current app**, you see a screen warning you that the upgrade option overwrites the current app and gets rid of any customizations you've made. If you're OK with overwriting customizations, check the box and click **Continue**. If you're not, click **Back** to return to the previous screen. Clicking **Continue** takes you to the configuration wizard. Skip ahead to step 6.

5. If you click **Create new app**, you're taken to the configuration wizard. You don't see a warning, since you're not overwriting your current app, which is preserved with any customizations you've made to it.

6. The configuration wizard is preloaded with the settings you chose last time you used the wizard. You can either keep those settings or change them. Go through each page of the wizard.

7. Once you complete the wizard, Tableau CRM shows a screen that indicates which assets are impacted by the upgrade. Review the screen to see how many datasets, dashboards, and lenses will be changed, deleted, or added by the upgrade. It also shows how

changed assets will be modified—whether the change is to data or appearance. On the same screen, you have the option to download a file with code for changes made to the app. If you've customized the app, download and save the file so you can copy and paste customization code into the upgraded version.

8. *Upgrade option only*: If you're upgrading your app, click **Upgrade current app**. Remember, this overwrites any customizations you've made. You can also click **Back** to go back into the configuration wizard and change your selections or click the **X** in the upper-right corner to cancel.

9. *Create option only*: If you're creating a new app, name your app something different from the current version and click **Create new app**. This option saves your current app and all its customizations. You can also click **Back** to go back into the configuration wizard and change your selections or click the **X** in the upper-right corner to cancel.

If the link below the app name says "Reset app," you're using the latest version and don't need to upgrade.

Schedule the Service Analytics Data Sync and Dataflow

Schedule a data sync and dataflow to rerun every day to assure that Service Analytics uses up-to-date data.

When you create Service Analytics, the creation process includes a dataflow that imports the latest Services Cloud data to Tableau CRM. You can schedule a data sync and dataflow to rerun every day to assure that your app uses up-to-date Salesforce data. Schedule the sync and dataflow to take place sometime outside normal business hours so the processes don't interrupt your use of the app.

> **Note:** The Service Analytics dataflow runs only once when you create the app. Schedule it to run daily so the app uses the latest sales data.

1. In Tableau CRM Studio, click the wheel icon at upper right and select **Data Manager**. Or, click the **Data Manager** link in the left-hand column.

2. First, schedule the sync. Select the **Connect** tab on the left.

   > **Note:** If you can't see the Connect tab, you need to enable data sync in your org. See Enable Data Sync and Connections on page 693.

3. Click the arrow ▼ to the far right of **SFDC_LOCAL**, which is the name of the connection your app uses. From the menu that appears,

   select **Schedule**.

4. Set a time for running the data sync. It's best to select a time outside normal working hours so the sync and dataflow don't interrupt business activities. Then click **Save**.

5. Next, schedule the dataflow. Select the **Dataflows & Recipes** tab on the left.

6. Look for the dataflow that contains the name of your app, and click the triangle ▼ to the far right.

7. Select **Schedule**, then check the box next to **Event-based**. You see a message telling you that the dataflow runs after the data sync—exactly what you want.

8. Click **Save**.

Embed Service Analytics Sidebar Dashboards in a Salesforce Page

Add Service Analytics dashboards to Salesforce pages so your service team can get a complete view of their cases and take appropriate action.

Service Analytics includes three specialized dashboards—called sidebars—to give service agents instant access to visualizations of data about their cases.

- The By Customer sidebar gives agents a snapshot of their customers, including products, case reasons and priority, and CSAT.
- The By Similarity sidebar shows data about previous cases so agents can guide customer interactions based on what they did in similar cases.
- The By Case History sidebar provides a quick view of the lifecycle of a case so agents can be informed during customer conversations.

Administrators can embed them in a Salesforce page so agents can access Service Analytics seamlessly while they carry out their normal case work. For example, if your Salesforce makes the Service Cloud available through a service console, that's an ideal place to embed the sidebar dashboards.

To embed a sidebar in Lightning Experience, use Wave Dashboard Components. Read the process in Embed Tableau CRM Dashboards in Lightning Pages, then see Steps 3, 4, and 5 below to create the filter.

To embed a sidebar in Salesforce Classic, follow the steps here.

1. Create a Visualforce page for each sidebar. Go to Setup and enter Visualforce Pages in the Quick Find box, then select Visualforce Pages, and click New. Enter a label with the name of the sidebar and, optionally, a description.

2. Replace the text that appears when you open the editor with the following: `<apex:page standardController="case"> <wave:dashboard dashboardId="Your_dashboard_id" height="1000px" showTitle="false" filter="Filter_condition" /></apex:page>`

    a. Replace `Your_dashboard_ID` with the dashboard ID for the sidebar. You can find the dashboard ID by opening the sidebar in Service Analytics; the ID is the last 15 characters of the URL for the sidebar.

   Depending on which sidebar you're embedding, go to one of the following steps:

   - **By Customer sidebar:** Go to Step 3.

- **By Similarity sidebar:** Go to Step 4.
- **By Case History sidebar:** Go to Step 5.

3. If you're creating a page for the By Customer sidebar, replace `filter="Filter_condition"` with one of the following, then skip to Step 6.

   a. If you're embedding a sidebar dashboard from the Service Analytics (Classic) app built with the old dashboard designer in a Salesforce Classic page, use this syntax: `filter="{'ServiceCase1': {'AccountId': ['{!case.AccountId}']}}"`.

   b. If you're using the latest version of Service Analytics based on the new dashboard designer, use this syntax to embed the dashboard in a Salesforce Classic page: `filter="{'datasets' : {'ServiceCase1': [{'fields':['AccountId'], 'filter': { 'operator':'in', 'values': '{!case.AccountId}'} }]}}"`.

   c. If you're embedding a dashboard from the latest version of Service Analytics in a Lightning Experience page, use this syntax: `filter="{'datasets' : {'ServiceCase1': [{'fields':['AccountId'], 'filter': { 'operator':'in', 'values': ['$AccountId']} }]}}"`.

   In all cases, replace `ServiceCase1` with the name of the Service Case dataset. To find the name, open a dashboard, hit **Command+e** to view the JSON for the dashboard, and search for `ServiceCase`. The name of the dataset is appended with a numeral that's incremented each time you create the app, such as `ServiceCase2`. Enter the entire dataset name including the numeral.

4. If you're creating a page for the By Similarity sidebar, replace `filter="Filter_condition"` with one of the following, then skip to Step 6.

   a. If you're using Service Analytics (Classic) with previous dashboards built with the old dashboard designer, use this syntax to embed the dashboard in a Salesforce Classic page: `filter="{'ServiceCase': {'Reason': ['{!case.Reason}'],'Product__c': ['{!case.Product__c}']}}"`.

   b. If you use the latest version of Service Analytics based on the new dashboard designer, use this syntax to embed the dashboard in a Salesforce Classic page: `filter="{'datasets' : {'ServiceCase1': [ {'fields':['Reason'], 'filter': { 'operator':'in', 'values': '{!case.Reason}'}}, {'fields':['Product__c'], 'filter': { 'operator':'in', 'values': '{!case.Product__c}' }} ]}}"`.

   c. If you're embedding a dashboard from the latest version of Service Analytics in a Lightning Experience page, use this syntax: `filter="{'datasets' : {'ServiceCase1': [{'fields':['Reason'], 'filter': { 'operator':'in', 'values': ['$Reason']}}, {'fields':['Product__c'], 'filter': { 'operator':'in', 'values': ['$Product__c'] }} ]}}"`.

   In all cases, do the following.

   - Replace `ServiceCase1` with the name of the Service Case dataset. See Step 3 to learn how to find the name.

   - Replace `Product_c` with the API name of the custom field you use to track products. This field is the answer you selected in the configuration wizard, Page 2, Question 4. Find the API name of this field by going to Setup and typing `Cases` in the Quick Find box. Then click Fields and review your custom field names to find the field you use for products. Look for the API name of the field, and use that in place of `Product_c` in both places in the query.

   📝 Note: If you use a custom field to track case reasons, replace `Reason` with the API name of your custom field. Find the API name using the same technique you use for finding the custom field you use to track products.

5. If you're creating a page for the By Case History sidebar, replace `filter="Filter_condition"` with one of the following, then skip to Step 6.

   a. If you're using Service Analytics (Classic) with previous dashboards built with the old dashboard designer, use this syntax to embed the dashboard in a Salesforce Classic page: `filter="{'ServiceCase1': {'Id': ['{!case.Id}'],`

```
'OwnerId' : ['{!case.OwnerId}'], 'Product__c' : ['{!case.Product__c}'], 'Reason'
: ['{!case.Reason}']}}".
```

Also do the following:

- Replace `'Product__c'` with the name of the custom field you use to track product family. See Step 4 to learn how to find the name of that field.

- Replace `'Reason'` with the API name of the field you use to track the case reason. This field is the answer you selected in the configuration wizard, Page 2, Question 5. Find the API name of this field by going to Setup and typing `Cases` in the Quick Find box. Then click Fields and review your custom field names to find the field you use for case reason. Look for the API name of the field, and use that in place of `'Reason'` in the query

b. If you're using the latest version of Service Analytics based on the new dashboard designer, use this syntax to embed the dashboard in a Salesforce Classic page: `filter="{'datasets' : {'ServiceCase1': [ {'fields':['Id'], 'filter': { 'operator':'in', 'values': '{!case.Id}'} }]}}"`.

c. If you're embedding a dashboard from the latest version of Service Analytics in a Lightning Experience page, use this syntax: `filter="{'datasets' : {'ServiceCase1': [ {'fields':['Id'], 'filter': { 'operator':'in', 'values': ['$Id']} }]}}"`.

In all cases, replace `'ServiceCase1'` with the name of the Service Case dataset. See Step 3 to learn how to find the name.

6. Save the page, and open it in preview mode to be sure it shows the dashboard.

7. Add the page to the console or whatever page you prefer in Salesforce. Through Setup, find the specific page layout you want to include the sidebar. For example, some customers use a page layout for information from the Cases object. In that case, enter `Cases` in the QuickFind box in Setup, then open Page Layouts under Cases, and look for the page layout you want to edit.

8. Click Edit Layout and select the **Custom Console Components** tab near the top. Go the Right Sidebar section and select one of the pages you just created. First, select Visual Force page from **Type**, then select the page you just created from the **Component** drop-down. Enter a label.

9. Set the width to 450px and save the layout. The sidebar should appear embedded in the page you just edited.

10. Repeat this step with the other two sidebars, as needed.

Understand Service Analytics Limitations

Service Analytics requires that Service Cloud include specific data, initially supports a limited set of Salesforce objects, and has other miscellaneous limitations.

Service Analytics Limitations

The following limitations apply to Service Analytics.

Service Analytics Data Requirements

Service Cloud data must include at least one each of the following for Service Analytics dashboards to function correctly: Events, tasks, closed cases, or published articles attached to a case or an opportunity. It also requires that your org have at least one contact ID associated with a case.

Custom formula fields on the Cases object must exist or be created for the SLA missed (text) and FCR (Boolean) fields.

Service Analytics Support for Salesforce Objects and Fields

Service Analytics supports all Salesforce standard and custom objects and data. When you first create the app by default it includes only a predefined set of objects and fields. The app creates a dataflow that exposes selected fields from the following Salesforce objects:

- Account

- Cases

- Contact

- User

- UserRole

- Task

Service Analytics also uses fields from the following options depending on the answers you select in the configuration wizard:

- Customer satisfaction (CSAT)

- Knowledge

- Business hours

- Case history

- Case record types

- Queues

- Opportunities

- Opportunity record types

- Events

- Telephony

- Chat

- Omni-Channel

Service Analytics supports standard and custom fields on standard Salesforce objects. The fields provided are determined by the answers you provide when you create the app using the built-in configuration wizard.

Service Analytics limits custom objects support to no more than 10 custom objects for each org that implements Service Analytics. This limitation is contractual, not technical.

To add custom objects or extra fields not included when you first create the app, you must update the Service Analytics dataflow. For details, see Design Datasets with Dataflows and the Dataset Builder on page 871.

More Service Analytics Limitations

Service Analytics does not support external data sources or registering new datasets from the app-generated dataflow.

Including CSAT score in dashboards/dataflow is optional. The app configuration wizard lets you associate CSAT from a custom case field or Custom Object associated to the Cases object.

You must manually add any security predicates or other dataset filters to the app dataflow.

The app does not support multi-hierarchy (team/manager) when you create it. You must manually add that functionality.

The Knowledge Usage dashboard currently reports on a single Article Type, which you select when you use the configuration wizard.

Salesforce Analytics for Veeva Template

For Veeva CRM customers only: Salesforce Analytics for Veeva assembles key metrics in one place to give you insight into pharmaceutical reps processes and effectiveness.

Create an app from the Salesforce Analytics for Veeva template to get instant insight into how your reps conduct sales calls and manage accounts. You get data about reps' use of key messages, products, and samples to manage their effectiveness. And use Einstein Discovery to understand the impact of sale call activities on account orders.

Use the app to see key messages used during sales calls, which products were discussed, and if samples have been left behind. Also, see order and sales call activities by account, and take action right from the dashboard to drive your sales team's activities. And stay on top of all your accounts. You can see who's ordered what products, the number of calls for each account, and the samples left behind during those calls.

🛑 **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Salesforce Analytics for Veeva app.

Org Requirements

🛑 **Important:** The Salesforce Analytics for Veeva is only for Veeva CRM customers.

Your org requires the following before you can create an app from the Salesforce Analytics for Veeva template:

- Make sure you and all app users have the Tableau CRM Growth license.
- Set Salesforce field-level security to enable the Analytics Integration User to see the fields that you want to analyze. During app creation, Analytics checks your org's field-level security and lets you know if you have to edit it.

Use Salesforce Analytics for Veeva

Open the app.

1. From the app picker ⚏ , select **Tableau CRM Studio** to open the Tableau CRM home page.
2. Under **Browse** in the left column, select **All Items**.
3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

Click **Run App**. The app opens to the Account Order Analytics dashboard. Click **Sales Call Analytics** to view that dashboard.

Account Order Tableau CRM Dashboard

View this dashboard for insights into your accounts. Select account to see trends for orders, calls, products, and samples, benchmarked against average for all accounts. View summary data in the top chart, including numbers of calls to accounts, orders placed, products included in orders, and samples left during calls to accounts. Order details appear in the chart at the bottom.

Sales Call Analytics

Gives you insights into the effectiveness of your team's sales calls. Left side of the top chart shows number of sales calls, average duration, key messages used, samples left behind, and products offered. Chart at top right shows weekly call frequency for period selected in Date filter.

In the charts below, view the key messages your team uses most frequently, the samples left behind, and products offered during calls.

Use the list of call attendees at the bottom to view their participation in calls in addition to their status. Take action for an attendee by rolling the cursor over the account name. Then select a Salesforce action to use, such as posting to Chatter, sending a message, or kicking off a task or event.

Snapshot Analytics Template

Create an app using the Snapshot Analytics template to trend data from any existing Tableau CRM dataset or Salesforce object.

Snapshot Analytics simplifies trending your Salesforce data. The template's wizard lets you snapshot either an existing Analytics Cloud dataset or a Salesforce object. If you target a Salesforce object, Tableau CRM imports its data into a dataset. With either option, Analytics Cloud automatically creates a dashboard with a snapshot of your data.

🛑 **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Snapshot Analytics app.

Org Requirements

Your org requires the following before you can create an app from the Snapshot Analytics template:

- Make sure you and all app users have the Tableau CRM Growth license.

- Make sure you and others who create the app have the 'Use Any API Client' permission. Without that permission, you may not see the datasets you want to use for the app in the wizard. To assign that permission, request API Access Control from Salesforce Customer Support. See Restrict Access to APIs with Whitelisted Connected Apps.

- Set Salesforce field-level security to enable the Analytics Integration User to see all fields used in the app. See Set Field Level Security to Enable Creation of an Tableau CRM Template

The Snapshot Analytics Configuration Wizard

Snapshot Analytics includes a configuration wizard. Follow these steps to use it.

1. **Choose existing dataset or Salesforce object page.** Choose whether you want to snapshot an existing dataset in Analytics Cloud or create a new dataset using a Salesforce object. Click **Looks good, next**.

   🛑 **Important:** If you don't see the dataset you want to use in the wizard, be sure you have the 'Use Any API Client' permission. See *Org Requirements*, above.

2. Complete one of the following two procedures, depending on your choice.

   a. **Start with a Tableau CRM dataset option.** Select the dataset you'd like to snapshot, and click **Looks good, next**

   b. **Start with a Salesforce object option.** Lets you choose to create dataset from a Salesforce object parent and up to two levels of child objects (a child and a grandchild). Once you're done, click **Looks good,next**.

      - **1. Choose an object.** The page opens to a single question asking you to choose the parent object.

      - **2. Choose a child object (optional).** After you choose an object, you have the option of selecting a child object. For example, say that you choose Account for the parent object and Contact for the child. Snapshot Analytics adds only accounts related to contacts to the dataset. If you leave this field blank, Snapshot Analytics adds all data from the parent object.

      - **3. Choose a second child object (optional).** You have the option of selecting a secondary child object, which becomes a grandchild of the parent object. For example, say that you choose Account for the parent object, Contact for the first child,

and Event for the second child. Snapshot Analytics adds only accounts related to both contacts and events to the dataset. If you leave this field blank, Snapshot Analytics uses only the first child.

c. **Add fields from Salesforce objects to the Snapshot Analytics dataset.** If you choose the option to start with a Salesforce object, you can add fields from any of the three selected objects to the dataset created by Snapshot Analytics. Choose as many as you like, then click **Looks good, next**

3. **Set time to preserve data.** To save data storage space, you can set a time to have snapshot data automatically deleted from Tableau CRM. Select Never and Tableau CRM preserves your data until you manually delete it.

Use Snapshot Analytics

You can set the snapshot frequency by rescheduling the Snapshot Analytics dataflow. You can record only one snapshot per day. See Schedule Data Sync and the Dataflow for an Tableau CRM Template.

Edit the new dataset created from a Salesforce object to add actions, measure formatting, and dimension labels to the app. See Edit a Dataset and Enable Actions for Tableau CRM Lenses and Dashboards.

Revenue Operations Analytics

Use the Revenue Operations Analytics template to help sales teams gain performance insights to build stronger pipelines, improve forecast accuracy, and generate more revenue.
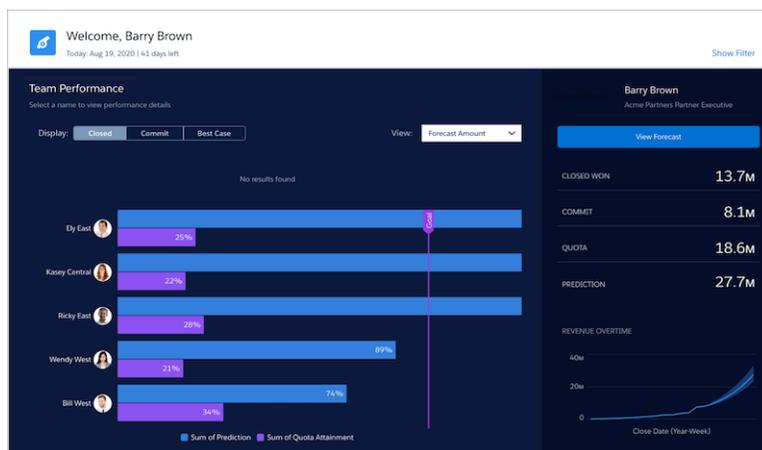
> **USER PERMISSIONS**
>
> To create and manage the Revenue Operations Analytics app:
> - Tableau CRM Plus Admin permission set
>
> To use the Revenue Operations Analytics app:
> - Tableau CRM Plus User permission set

The template creates datasets, dashboards, and an Einstein Discovery story to help your sales managers and teams gauge their team performance, revenue forecasting, forecast tracking, activity intelligence, and deal overview. The Revenue Operations Analytics app delivers forecasting analytics with better pipeline visibility and simpler deal inspection.

🛑 **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Revenue Operations Analytics app.

Salesforce Org Requirements

Note these requirements to create an app from the Revenue Operations Analytics template:

- Make sure that you and all app users have the Tableau CRM Plus license.
- Your org must have at least one of each of the following record types: Opportunity, Account, Forecasting Item, Forecasting Type, User, and UserRole.

- Set Salesforce field-level security to enable the Analytics Integration User to see all fields used in the app. See Set Field Level Security to Enable Creation of an Tableau CRM Template.

Create the Revenue Operations Analytics App

1. In Tableau CRM Studio, click **Create** in the upper right corner

2. Select **App**, then **Create App from Template** to open the template picker.

3. Locate the **Revenue Operations Analytics** tile, select it, and click **Continue**.

4. Review the app preview page, and click **Continue** to open the configuration wizard.

5. If you're offered a choice between basing your app on an existing app or creating a one, select **Create a brand new app** and click **Continue**. Revenue Operations Analytics runs a compatibility check against your org to be sure it includes the data to successfully create the app's datasets and dashboards. If it doesn't, follow the instructions in the error message to add the required data and start the app creation process again.

6. When the org compatibility check succeeds, click **Looks good, next**.

7. Name your app, and click **Create**.

Give Tableau CRM a few minutes to create your app. You can track its progress on the page that appears. When you see the **Application Complete!** message, refresh the page. You see your app page with a story, the datasets, and dashboards showing team performance and activity intelligence. Click it to have a look.

Schedule the Revenue Operations Analytics Dataflows

The Revenue Operations Analytics app creates two dataflows, a main dataflow to process the revenue data and a secondary dataflow for opportunity data snapshots. To create snapshots of your opportunity data for period over period comparison of your revenue forecast, you must schedule both of the dataflows for your app.

1. In Tableau CRM Studio, click the wheel icon at upper right and select **Data Manager**. Or, click the **Data Manager** link in the left-hand column.

2. First, schedule the sync. Select the **Connect** tab on the left.

   📝 Note: If you can't see the Connect tab, you must enable data sync in your org. See Enable Data Sync and Connections.

3. Click the arrow 🔽 to the far right of **SFDC_LOCAL**, which is the name of the connection your app uses. From the menu that appears,
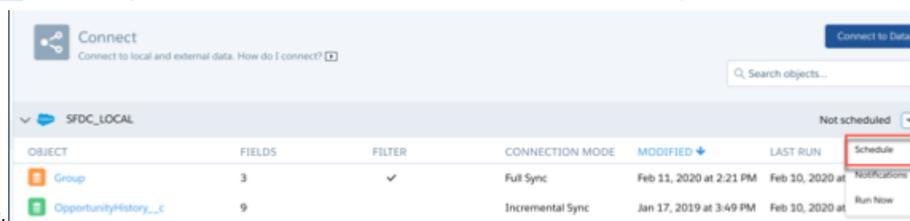
   select **Schedule**.

4. Set a time for running the data sync. It's best to select a time outside normal working hours so the sync and dataflow don't interrupt business activities. Then click **Save**.

5. Next, schedule the dataflow. Select the **Dataflows & Recipes** tab on the left.

6. Schedule the main dataflow:

   a. Look for the dataflow that contains the name of your app appended with `_RevenueOperations`, and click the triangle 🔽 to the far right.

   b. Select **Schedule**, then check the box next to **Time-based**.

    c. Select the time to run the dataflow. You can schedule the data to run by minute, hour, week, or month. See Schedule a Dataflow to Run Automatically for more details.

    d. Click **Save**.

7. Schedule the snapshot dataflow:

    a. Look for the dataflow that contains the name of your app appended with `_RevOpsSnapshot`, and click the triangle ▼ to the far right.

    b. Select **Schedule**, then check the box next to **Time-based**. Schedule the snapshot dataflow to run after the main dataflow finishes.

    c. Click **Save**.

Both dataflows now sync and run every day at the time you set.

Revenue Operations Analytics Datasets

Dataset Name	Contents	Special Requirements
Account	Data about accounts.	None.
Opportunity	Data about opportunities. Includes records joined on opportunity split and opportunity amount.	None.
Opportunity Splits	Data about opportunities splits. The cumulative split amount is used for the opportunity amount.	Only included if your org has opportunity splits enabled.
Forecast	Data about forecasts, including forecast item, forecast quota, and forecast type data.	Only included if your org has forecast enabled.
User Roles	Data about users, including role, opportunity, and role hierarchy data.	Only included if your org has user roles enabled.
Forecasting Quota	Data about forecasting quotas.	None.

Understand the Einstein Discovery Likelihood to Win Story

The Revenue Operations Analytics app generates an Einstein Discovery story to predict ways to maximize your sales team's win rates. This story contains insights on the top revenue drivers—the factors that have top positive and negative impacts on the forecast. The AI or machine learning model is natively generated at app creation. It incorporates the best practices to create a predictive model to gauge the win rate for every opportunity. To view these insights, open the Likelihood to Win story from your app, and you can then easily enhance the model with custom fields to further enrich predictive insights.

Upload Quota Dataset

If your org has Sales Cloud Collaborative Forecasts enabled, Revenue Operations Analytics automatically adds quotas data to your app and its dashboards and the following steps aren't necessary. If your org doesn't have Sales Cloud Collaborative Forecasts enabled, you must add quotas data to your app by following these instructions. Without quotas data present in the app, the Team Performance charts can't give an accurate visualization of performance metrics.

> 📝 **Note:** When you work with .CSV files you want to import to Revenue Operations Analytics, create and open them using only a UTF-8-compliant text editor. Opening them in Microsoft Excel or other spreadsheet software reformats .CSV files and makes them unusable in Revenue Operations Analytics

> ⊘ **Important:** When you create the .CSV file, be sure it contains the following fields, in this order, with exactly these names. Field names are case-sensitive:
>
> 1. StartDate (in yyyy-mm-dd format)
>
> 2. QuotaAmount
>
> 3. OwnerName
>
> 4. Username

1. Create a .CSV file to include the fields described, that is StartDate (in yyyy-mm-dd format), QuotaAmount, OwnerName, Username. For an example, see Sales Analytics Example .CSV File.

2. Save the file to a location that you can easily remember.

3. In Salesforce, go to the Analytics Cloud home page and find the Quota dataset.

4. Click the arrow at the upper right corner of the dataset panel and select **Edit**.

5. Salesforce displays the dataset editing screen for the Quota dataset. Look for **Replace Data** in the upper right corner and click it.

6. In the dialog box that opens, navigate to the .CSV file you created in Step 1, and double-click it.

7. Click **Next** to open the Replace Dataset Data page.

8. If your fiscal period is different than calendar period, that is if it starts on a date other than January 1, update the Quota Metadata file. If your fiscal period starts on January 1, skip to the next step.

    a. Copy the JSON from Sales Analytics Quota Dataset JSON File and paste it into a text editor of your choice.

    b. Change the value of `"fiscalMonthOffset"` from 4 to a number that represents the month your fiscal period starts. In Revenue Operations Analytics metadata, the numeral "0" stands for January, "1" stands for February, and so on, up to "11," which stands for December. Save the file to your desktop.

    c. In Tableau CRM Studio, go to the Replace Dataset Data page and locate the Data Schema File area of the page. Click the arrow next to Quota .JSON file, select Replace File, find the file you saved and upload it to Tableau CRM.

9. On the Replace Dataset Data page, click **Next** to open the Edit Field Attributes page. The first column —QuotaAmount— is selected. If not, select it. In the **Field Attributes** panel on the right, make sure **Field Type** is set to **Measure**.

10. After uploading your quota data, rerun the dataflow to update the dashboards.

    a. Click the Gear menu at the upper right of the Revenue Operations Analytics screen and select Data Manager.

    b. Select Dataflow view from the menu at the top left of the Data Manager screen.
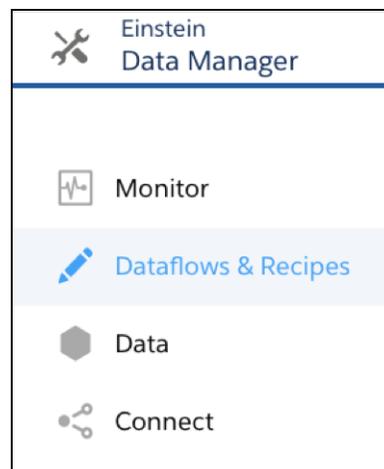
    c. Find your app's dataflow. Open the menu on the far right of the screen next to the app icon and name, and click Start. That's it.

1. Revenue Operations Analytics Revenue Operations Dashboard

   Use the Revenue Operations Analyticsdashboard to help sales teams gain performance insights to build stronger pipelines, improve forecast accuracy, and generate more revenue.

2. Revenue Operations Analytics Activity Intelligence Dashboard

   The Activity Intelligence dashboard shows your team's activities, pipeline, and closed deals. The dashboard helps you understand which activities lead to a greater win rate and which team members need coaching.

3. Sales Manager Dashboard (Embedded)

   Use the Sales Manager embedded dashboard to help sales managers understand their total opportunities, the gap to goal, and the predicted sales.

Revenue Operations Analytics Revenue Operations Dashboard

Use the Revenue Operations Analyticsdashboard to help sales teams gain performance insights to build stronger pipelines, improve forecast accuracy, and generate more revenue.

The dashboard helps your sales managers and teams gauge their team performance, revenue forecasting, forecast tracking, and deals overview.
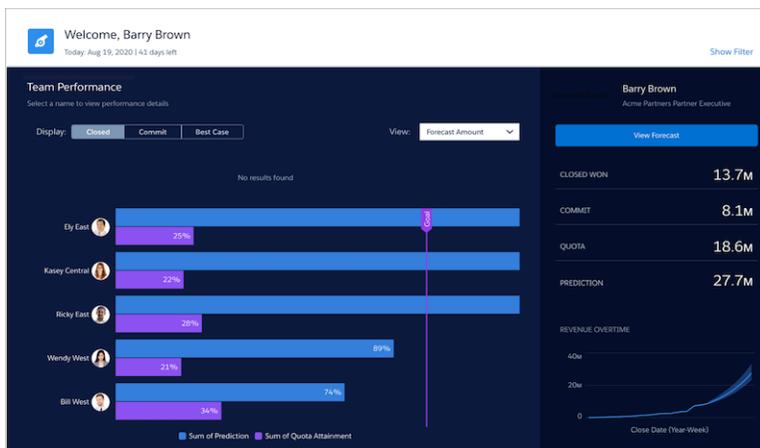
To view details, click **View Forecast**. This page allows you to target sales wins by viewing the top positive and negative drivers that affect sales forecasts in the **Top Drivers** chart powered by the Einstein Discovery Story packaged in the app. You can see the top positive and negative drivers that influence your forecast in the order of their impact. In addition, you can view the top drivers in the same dashboard as your metrics in context without having to navigate away from it
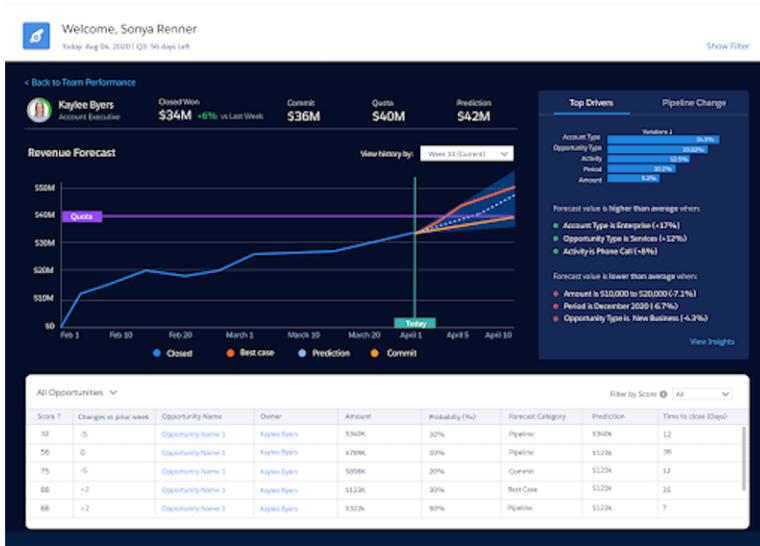
KPI Calculations

- **Quota Attainment:** total qualifying amount from opportunities won that have a close date in the period divided by rep's quota.

- **Win Rate:** total won opportunity amount from all opportunities that were closed in this quarter.

- **Quota:** total quota of the team.

- **Prediction:** closed opportunity amount generated by TimeSeries for this quarter.

- **Coverage:** ratio of open pipe to quota required to close this quarter.

- **Top Drivers:** view the top positive and negative drivers that affect sales forecasts generated by the Likelihood to Win Einstein Discovery story.

Revenue Operations Analytics Activity Intelligence Dashboard

The Activity Intelligence dashboard shows your team's activities, pipeline, and closed deals. The dashboard helps you understand which activities lead to a greater win rate and which team members need coaching.

The Activity Intelligence dashboard enables sales managers to quickly see the team members who need coaching and determine activities to help them achieve the most wins. Team members appear in order of win rate, and sales managers can see the activities for each team member. If you select a team member, you can view the pipeline, closed wins, and activities, such as calls, emails, events, and tasks for that team member. Individual team members can see their own activities, pipeline, and wins.

The first page shows the win details across your team so that you can easily see who is winning deals and who is lagging. This page can help you quickly identify the team members who can benefit from coaching.
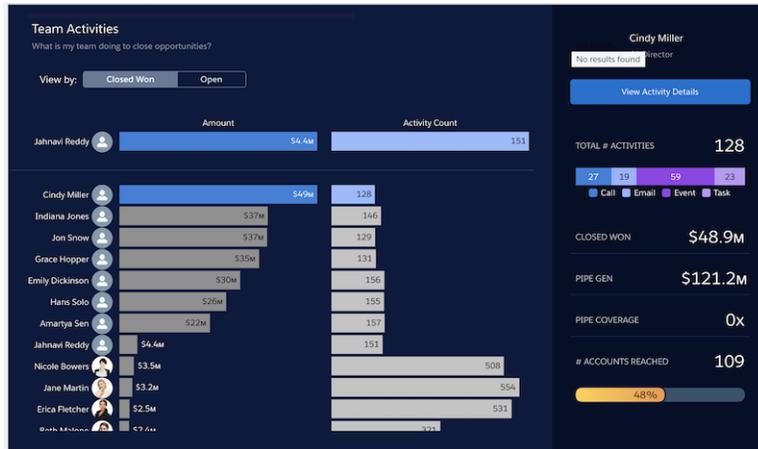
To drill down into the opportunities for a team member, in the right panel, select **View Activity Details**. You can view accounts and opportunities for a team member in the Activity Highlight page. In the right panel, the dashboard summarizes the highlights for a team member:the open pipeline, the quota attainment, and the number of activities this team member has completed as compared to the top three performers on the team.
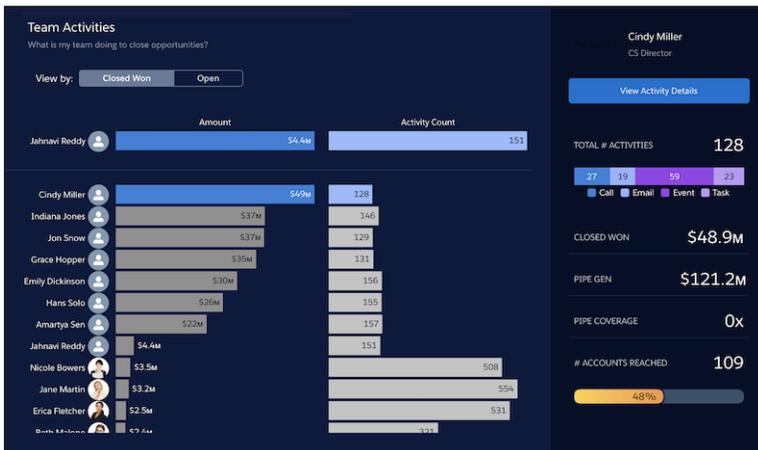
KPI Calculations

- **Total Activities:** total activities associated with opportunities in the org
- **Closed Won:** total closed won amount on all the opportunities
- **Pipeline Generated:** opportunity amount on all the new deals generated this quarter
- **Account Reached:** accounts that have at least one activity logged
- **Quota Attainment:** closed won amount or quota amount

Sales Manager Dashboard (Embedded)

Use the Sales Manager embedded dashboard to help sales managers understand their total opportunities, the gap to goal, and the predicted sales.

Default Behavior and Recommended Options

Embed the dashboard in either Lightning and Aloha page layouts.

Edit the dashboard to set target lines and set the upper bounds for gauges. With the dashboard open in Tableau CRM Studio, click the

edit icon to open the designer, then click the widget you want to edit. Then look for the value you want to change in the properties panel on the right and change it there. Save the dashboard.

Use the Dashboard

In the left column, you can see metrics for Quota Attainment, Commit, Prediction to Close, and Gap to Goal. For details, hover over the metric to open a dropdown menu. The dropdown menu can take you to details about the metric, as well as allowing you to explore the lens associated with the metric, add the metric to a watchlist should the value increase or decrease, send an email notification for changes to the metric, annotate the entry, or share the dashboard to a user or groups of users.

In the right pane, you can view the number of activities associated with an open opportunity, the predicted win rate for the opportunity, and the amount changes for the opportunity. To do this, select the toggle button next to the options.
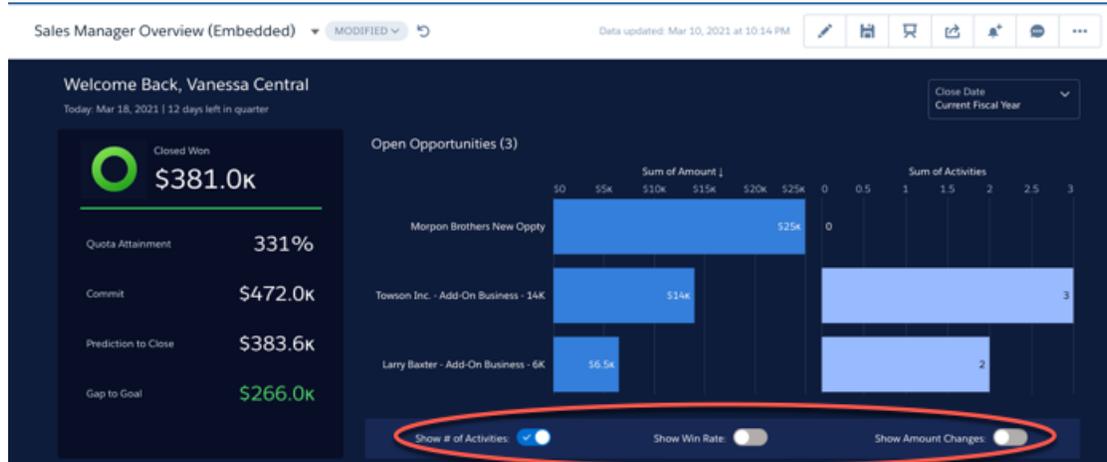
KPI Calculations

- **Quota Attainment:** total qualifying amount from opportunities won that have a close date in the period divided by rep's quota.

- **Closed Won:** total closed won amount on all the opportunities.

- **Commit:** the total amount of closed won opportunities plus the opportunities with a Forecast category of commit with a close date in the selected time range.

- **Prediction:** closed opportunity amount generated by TimeSeries for this quarter.

- **Gap to Goal:** the amount needed to reach the goal (quota). This value represents the total closed or forecasted amount minus the quota amount.

- **Total Activities:** total activities associated with opportunities in the org.

Social Case Analytics Template

The Social Case Analytics template creates an app that provides ready-made insights into team performance on each social channel.

The Social Case Analytics template gives service managers a fast way to get started analyzing Service Cloud social customer service data. The app's ready-made dashboard keeps you on top of how your agents use social channels to deliver customer service. Its visualizations help you:

- Plan capacity with visualizations of historic volume and handle time data.

- Improve productivity and lower cost per case by tracking per-case post volume.

- Identify agents who need coaching by analyzing your team's time-to-close and response times.

- Measure social post volume and trends by language, geography, and handle.

> 🛑 **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Social Case Analytics app.

Org Requirements

Your org requires the following before you can create an app from the Social Case Analytics template:

- Make sure you and all app users have the Tableau CRM Growth license.

- Social Case Analytics requires that your org:

  - Includes at least one social post linked to a case.

  - Uses case record types.

- Set Salesforce field-level security to enable the Analytics Integration User to see all fields used in the app. See Set Field Level Security to Enable Creation of an Tableau CRM Template

Use Social Case Analytics

Open the app.

1.
    From the app picker ⣿ , select **Tableau CRM Studio** to open the Tableau CRM home page.
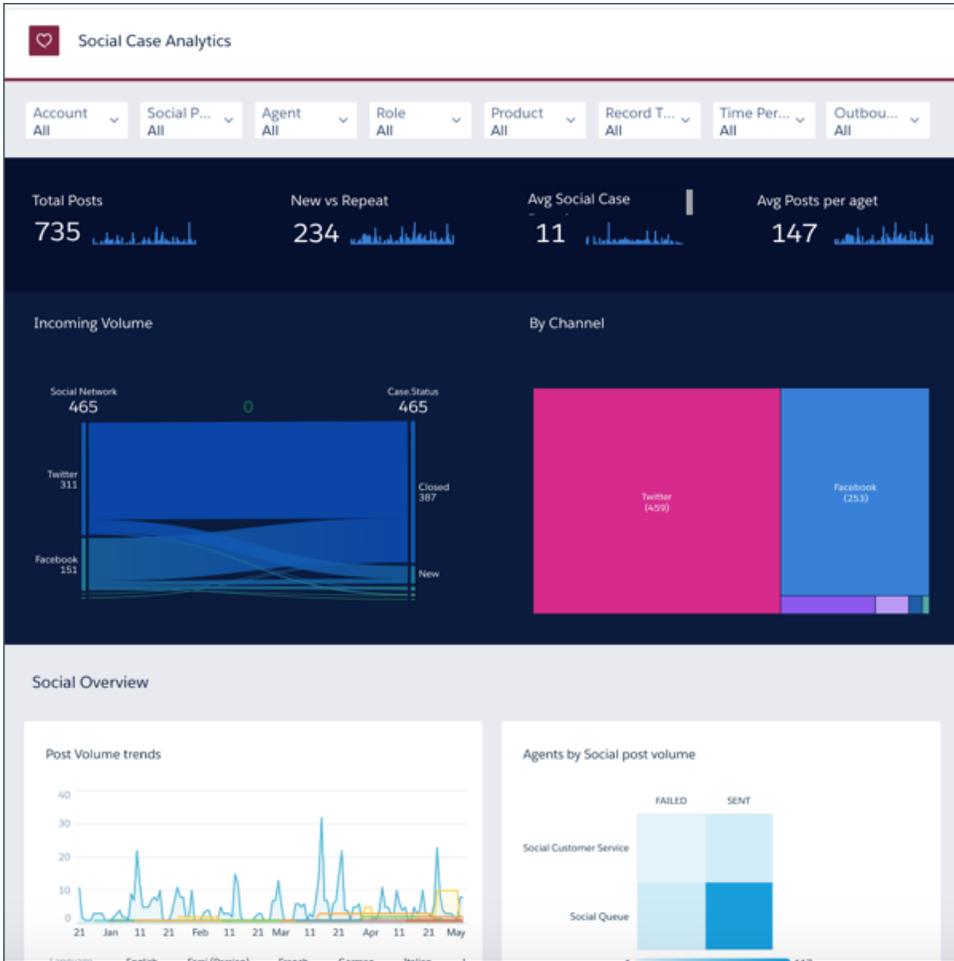
2. Under **Browse** in the left column, select **All Items**.

3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.

The Social Case Analytics Social Posts dashboard gives you a wealth of data visualizations about your service agents social channel interactions. Summary metrics include the following:

- Total posts to social channels

- Unique social personas

- Average social case duration

- Average first response time

- Average outbound posts per agent

- Incoming volume by day and channel

Filter summary metrics by the following:

- Social persona

- Agent
- Role
- Product
- Record Type
- Time period

You can also review post volume trends and each agent's social post volume in separate charts.

The app includes SocialPost and SocialPersona datasets, which you can use to build your own custom explorations.

Subscription Analytics Template

Salesforce CPQ customers: Create an app from the Subscription Analytics template to track your renewal business.

Subscription Analytics shows the value of your subscription-based products, the status of accounts up for renewal, and a list of all your subscription accounts in a single dashboard.

> **Important:** See Create Apps from Tableau CRM Templates: Start Here for general app creation procedures for all Tableau CRM templated apps. The information here provides specifics about creating and using the Subscription Analytics app.

Org Requirements

> **Important:** Subscription Analytics can only be used by Salesforce CPQ customers.

Your org requires the following before you can create an app from the Subscription Analytics template:

- Make sure you and all app users, including the Analytics Integration User, have the Tableau CRM Growth and Salesforce CPQ licenses.
- Your org needs to use at least one Salesforce CPQ Subscription record.
- Set Salesforce field-level security to enable the Analytics Integration User to see all fields used in the app. See Set Field Level Security to Enable Creation of an Tableau CRM Template.

Use Subscription Analytics

Open the app.

1. From the app picker ⋮⋮⋮ , select **Tableau CRM Studio** to open the Tableau CRM home page.
2. Under **Browse** in the left column, select **All Items**.
3. Select the **Apps** tab, then click your app to open it. If you can't immediately find it, consult your Salesforce administrator to find out the name they gave it when creating the app.
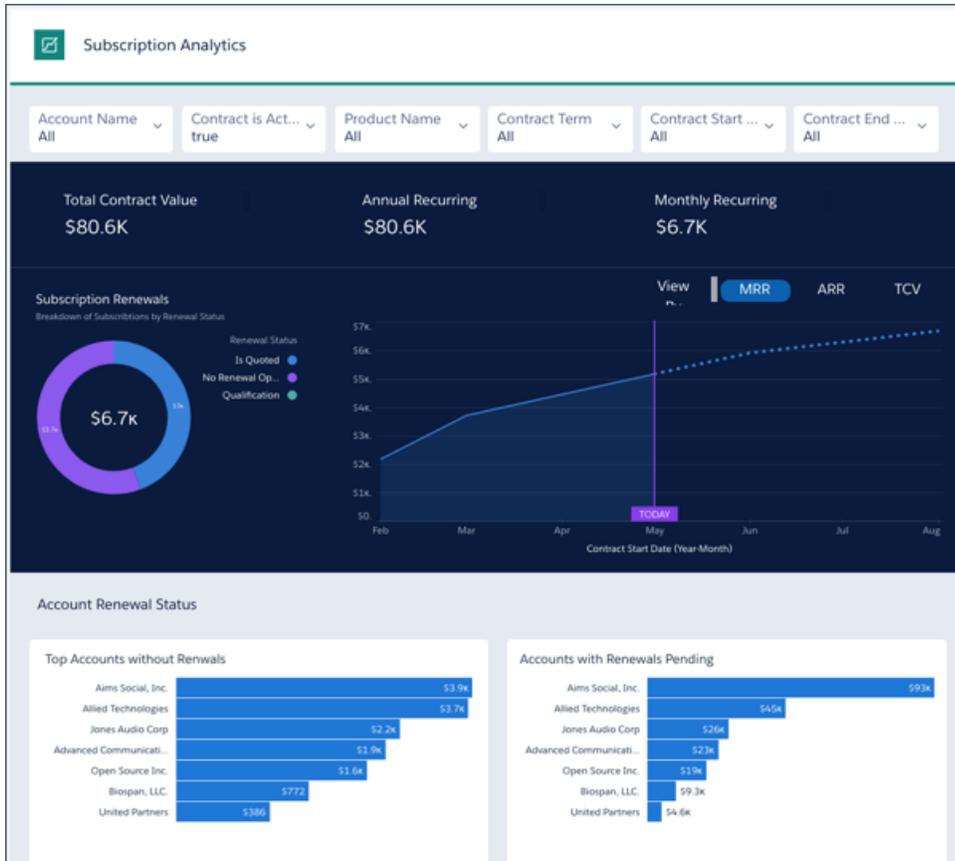
Click **Run App** to open its dashboard. Use the chart at the top to get an overview of your subscription business. You can see the following:

- Total value of your contract business.

- Recurring revenue from contracts by month and year.

- Value of contracts by stage in the renewal process.

Charts in the middle provide detail about renewals, including renewal warning and pending renewal metrics.

Scroll down to see a list of all accounts with subscription-based contract. Take action on an account by hovering over its ID and clicking the disclosure triangle to the right. That opens the Actions Menu from the record. From there, post to Chatter, create a task or an event, or perform another action to move the process along.

Predict Client Churn Risk for Wealth Management Analytics

Predict Client Churn Risk for Wealth Management Analytics allows financial advisors to intelligently predict customer churn. Using the power of Einstein Discovery, the app prescribes corrective actions to help minimize occurrences of churn.

📝 Note: Predict Client Churn Risk for Wealth Management Analytics is included with Tableau CRM for Financial Services, which is available at an extra charge for customers with Financial Services Cloud Basic or Standard licenses and the FSCAnalyticsPlus (Tableau CRM for Financial Services) license. See Deploy Tableau CRM for Financial Services for complete deployment instructions.

Predict Likelihood to Add Assets for Wealth Management Analytics

Predict Likelihood to Add Assets for Wealth Management Analytics allows financial advisors to intelligently forecast the likelihood of adding assets to accounts. Using the power of Einstein Discovery, the app prescribes actions on how to increase the chances to grow your account assets.

> **Note:** Predict Likelihood to Add Assets for Wealth Management Analytics is included with Tableau CRM for Financial Services, which is available at an extra charge for customers with Financial Services Cloud Basic or Standard licenses and the FSCAnalyticsPlus (Tableau CRM for Financial Services) license. See Deploy Tableau CRM for Financial Services for complete deployment instructions.

Analytics for Wealth Management Template

Analytics for Wealth Management a comprehensive analytics solution that's part of a suite of Tableau CRM apps for Financial Services Cloud customers. Its extensive sets of dashboards apply the power of Tableau CRM to all significant data and KPIs from the Financial Services Cloud. .

> **Note:** Analytics for Wealth Management is included with Tableau CRM for Financial Services, which is available at an extra charge for customers with Financial Services Cloud Basic or Standard licenses and the FSCAnalyticsPlus (Tableau CRM for Financial Services) license. See Deploy Tableau CRM for Financial Services for complete deployment instructions.

Wealth Starter Analytics Template

Creating an app from the Wealth Starter Analytics template gives Financial Services Cloud customers a quick way to apply the power of Tableau CRM to data about their book of business.

> **Note:** Wealth Starter Analytics is included with Tableau CRM for Financial Services, which is available at an extra charge for customers with Financial Services Cloud Basic or Standard licenses and the FSCAnalyticsPlus (Tableau CRM for Financial Services) license. See Deploy Tableau CRM for Financial Services for complete deployment instructions.

Develop on the Tableau CRM Platform

The Tableau CRM platform meets a wide range of development needs. Learn about the solutions you can deliver with Tableau CRM platform technologies.

Your choice of development tools.

Choose between developing with Microsoft Visual Studio (VS) Code, the Salesforce Command Line Interface (CLI), and Workbench. See Tableau CRM Development Tools on page 1585.

Use your investment in Tableau CRM apps development with Tableau CRM Templates.

Create an app and all its assets in your development or scratch org, then turn it into a template to migrate it to other orgs. Templates let you distribute your app to customers, partners, and other teams within your company. They can use the template to create a custom version of the app using their own data. You can do all your template development with Microsoft VS Code using Salesforce CLI commands. Run commands to convert your Tableau CRM app into a template, and then write the code to customize your template from within VS Code. VS Code displays all your template assets in an organized directory structure that makes team development and merges into Github a breeze.

- Learn how organizations use Tableau CRM apps.
- Get an overview of Tableau CRM Templates.
- Develop templates based on your Tableau CRM apps.
- Use Visual Studio Code and Salesforce CLI commands for Tableau CRM template development.

Migrate Tableau CRM assets between orgs.

Use change sets, bundle assets together in managed packages, and distribute and track packages through AppExchange and the License Management App. Also use the metadata API to manage customizations for your org.

- Learn about Tableau CRM migration, packaging, and distribution on page 1588.

Make Tableau CRM and Einstein Discovery Analytics and Discovery functionality available through Salesforce apps, external apps, and web pages.

Use Tableau CRM SDKs and the Tableau CRM Dashboard Component to facilitate access to insights created inside Tableau CRM. Your users can view and interact with those insights from wherever they normally do their work without having to open Tableau CRM.

And the Einstein Prediction Service API lets you include access to Discovery predictions in applications and websites.

- Learn about Tableau CRM SDKs on page 1588.
- Learn about the Tableau CRM Dashboard Component on page 1589.
- Use the Einstein Discovery Prediction Service on page 1754.

Create Tableau CRM applications programmatically.

The Tableau CRM REST API helps you speed up development of user functionality. Instead of undertaking repetitive steps using the declarative interface, construct API calls and commands that automate creating and enhancing apps, dashboards, lenses, datasets, dataflows, and recipes. Call API endpoints through the Salesforce CLI. You can also use Microsoft VS Code to access CLI commands and simplify development on the platform. Workbench also provides access to Tableau CRM API endpoints.

- Learn about the Tableau CRM REST API.
- Simplify Tableau CRM Development with Visual Studio Code and the Analytics Plugin for the Salesforce CLI on page 1586.
- Set up Visual Studio Code.
- Salesforce Analytics CLI Plugin Command Reference
- Use Workbench for testing and debugging.

Build custom data queries.

Access data in Tableau CRM and construct advanced queries using Salesforce Analytics Query Language (SAQL)—the language Tableau CRM uses to get data for visualizations. You can embed SAQL queries that return actionable insights wherever your users do their everyday work.

- Learn about SAQL.
- Create SAQL queries on page 1269.

And consult the following resources.

Tableau CRM Development Tools
Salesforce offers various developer tools for Tableau CRM administrators and code-level developers.

Distribute Your Tableau CRM Apps with Tableau CRM Templates
Tableau CRM Templates gives you a great way to leverage your Tableau CRM app development efforts. Turn your app into a template, which other organizations can use to create a version of the app using their own data.

Tableau CRM Migration, Packaging, and Distribution
Migrate Tableau CRM assets using change sets, bundle them together in managed packages, distribute and track packages through AppExchange and the License Management App, and use the metadata API to manage customizations for your org.

The Analytics Cloud SDK

Use the Tableau CRM SDK to embed Tableau CRM functionality directly where your users work everyday, without having to transition between Lightning Experience and Tableau CRM Studio. The Tableau CRM SDK lets you communicate and interact with Tableau CRM assets from Lightning Apps, Apex, Visualforce, and more. You can create one cohesive experience powered by Tableau CRM features directly in Salesforce pages and apps.

The Analytics Cloud Dashboard Component

The Tableau CRM dashboard component is an Aura component used to embed Tableau CRM dashboards in Visualforce and Lightning pages. The component can render a live Tableau CRM dashboard or it can be interactive with the page using events and methods to update the dashboard state.

The Tableau CRM REST API

Access Analytics Cloud datasets and lenses programmatically using the Tableau CRM REST API.

Salesforce Analytics Query Language (SAQL)

Use SAQL to access data in Analytics Cloud datasets. Analytics Cloud uses SAQL behind the scenes in lenses and dashboards to gather data for visualizations.

SEE ALSO:

Use Visual Studio (VS) Code and the Salesforce Extensions for Tableau CRM Template Development

Tableau CRM Development Tools

Salesforce offers various developer tools for Tableau CRM administrators and code-level developers.

Build Tableau CRM Assets with Tableau CRM Studio

Tableau CRM Studio provides the foundational tool for building Tableau CRM assets—the datasets, dataflows, recipes, lenses, and dashboards that make up Tableau CRM apps.

Develop with the Tableau CRM Plugin for the Salesforce CLI

The Salesforce CLI is a powerful command-line interface that simplifies development and build automation when working with a Salesforce org. The CLI plugin extends that experience to Tableau CRM. Plugin commands work from either the CLI or Microsoft Visual Studio (VS) Code.

Install the Analytics CLI Plugin

Follow these steps to set up the Analytics plugin for Salesforce command-line interface (CLI) for your Tableau CRM development project.

Use Visual Studio (VS) Code and the Salesforce Extensions for Tableau CRM Template Development

Microsoft Visual Studio (VS) Code provides the fastest, most efficient way to work with Tableau CRM templates. Here's how to install and set up the tool.

Call Tableau CRM REST API Endpoints from Workbench

Workbench provides an unsupported way to access all Tableau CRM REST API endpoints. *Advanced programming skills required*.

Build Tableau CRM Assets with Tableau CRM Studio

Tableau CRM Studio provides the foundational tool for building Tableau CRM assets—the datasets, dataflows, recipes, lenses, and dashboards that make up Tableau CRM apps.

Use Tableau CRM Studio to import data, transform the data with dataflows and recipes, and create datasets you use for explorations. Then explore your data and build lenses and dashboards with data visualizations that give you the insights to help you run your business. Store everything in apps— the containers of assets that tell a story about an aspect of your business through data visualizations.

Tableau CRM Studio lets you do most of your work declaratively through its intuitive graphical user interface. You can also access the JSON, XMD, SaQL, and other files that control how Tableau CRM consumes and displays data in dashboards and apps. Open those files with your editor of choice to make fine-grained changes to Tableau CRM business user functionality.

To open Tableau CRM Studio, navigate to the Salesforce app picker ⠿ , and select **Tableau CRM Studio**. Sometimes you must scroll or click **View All** to find it. From there, click **Create**, and start building. The Analytics Learning Map provides pointers to help get you going and master Tableau CRM.

Develop with the Tableau CRM Plugin for the Salesforce CLI

The Salesforce CLI is a powerful command-line interface that simplifies development and build automation when working with a Salesforce org. The CLI plugin extends that experience to Tableau CRM. Plugin commands work from either the CLI or Microsoft Visual Studio (VS) Code.

Use Salesforce CLI and Salesforce Developer Experience (SFDX) to create scratch orgs with Tableau CRM Studio, which you can use to develop and test source code. CLI and SFDX work with any Salesforce org.

The CLI plugin gives you access to commands that call a subset of Salesforce Tableau CRM REST API endpoints to manage Tableau CRM assets programmatically. Use plugin commands to convert an app created in Tableau CRM Studio to a template and manage the complete template lifecycle. Download the template JSON files for refinement, using an editor of your choice. For example, adding a configuration wizard to enable app customization and updating the template. And update the template files from the master app when you change it in Tableau CRM Studio or edit the JSON files in VS Code.

You can also share the results of your work with other developers using a version control system. They can push the template JSON files to their own scratch orgs for further testing and development work, including updating and deleting apps and templates.

Use other CLI plugin commands to update and delete apps, dashboards, lenses, and dataflows. You can also restore previous versions of Tableau CRM assets programmatically. You can also use the Tableau CRM Studio user interface to restore previous versions of dashboards and dataflows.

Install the Analytics CLI Plugin

Follow these steps to set up the Analytics plugin for Salesforce command-line interface (CLI) for your Tableau CRM development project.

1. Enable Dev Hub in your org so you can create and manage scratch orgs from the command line. See Enable Dev Hub in Your Org.

2. Install the Salesforce CLI. See Install the Salesforce CLI.

3. Create a Salesforce Developer Experience (SFDX) project. See Create a Salesforce DX Project.

4. Create a scratch org. See Create Scratch Orgs.

5. Install the CLI Analytics plugin by running the command `sfdx plugins:install @salesforce/analytics`.

6. Verify the installation by listing the available Tableau CRM commands. Run the command `sfdx analytics --help`.

For an overview and examples of commonly used CLI commands for managing Tableau CRM assets, refer to the Salesforce Analytics CLI Plugin Command Reference.

Use Visual Studio (VS) Code and the Salesforce Extensions for Tableau CRM Template Development

Microsoft Visual Studio (VS) Code provides the fastest, most efficient way to work with Tableau CRM templates. Here's how to install and set up the tool.

The VS Code Salesforce Analytics Extension Pack lets you work iteratively with templates using Salesforce DX (SFDX) and Salesforce CLI Analytics commands. Manage the lifecycle of a templated app from VS Code. Create a template from a Tableau CRM app. Edit the template JSON files to make the app customizable with a configuration wizard. Deploy the app to a scratch org to test it. And store the files in your source code control system of choice to enable collaborative template development.

Install VS Code and Salesforce Extensions

Prerequisites: Install Salesforce CLI and the Analytics CLI plugin, and create a development org before you install Visual Studio (VS) Code. See Install the Analytics CLI plugin on page 1586.

Install VS Code, and get it set up for template development with the Salesforce Extension Pack and Salesforce Analytics Extension Pack.

1. Go to https://code.visualstudio.com/, and follow the instructions for downloading and installing VS Code.

2. Open VS Code.

3. Click the **Extensions** button, or open the **View** menu, and select **Extensions**.

4. In the **Search Extensions...** field, enter `Salesforce Extension Pack`.

5. Select **Salesforce Extension Pack**, and click **Install**.

6. Search for the **Salesforce Analytics Extension Pack**, and install it.

Create a VS Code Project

1. Open VS Code, open the **View** menu, and select **Command Palette**.

2. Type `SFDX: Create Project with Manifest` in the palette, and select **SFDX: Create Project with Manifest** from the options that appear.

3. Select **Analytics**, and enter a project name that's easy for you to remember.

4. Choose a directory on your computer to store the files for the project, then click **Create Project**. The VS Code refreshes and your project appears in the **Explorer** panel on the left.

Connect VS Code to Your Development Org

1. In VS Code, open the command palette again, type `Auth`, and select **SFDX: Authorize an Org**.

2. Select **Project Default**, then enter an alias for your org. Again, enter an alias you can easily remember. Press the **Enter** key.

3. VS Code opens a browser to the Salesforce login page. Enter the credentials you used when you created your development org.

That opens the Home screen for your org. Any time you want to go right to your development org, enter the `Auth` command in the VS Code palette.

Issue Salesforce CLI commands for creating and managing Analytics templates from the VS Code Command Palette. See Install the Analytics CLI Plugin on page 1586 and the Salesforce Analytics CLI Plugin Command Reference.

Call Tableau CRM REST API Endpoints from Workbench

Workbench provides an unsupported way to access all Tableau CRM REST API endpoints. *Advanced programming skills required*.

Construct API calls in Workbench to manage Tableau CRM assets in a Salesforce org. View, load, and perform CRUD operations on your data. Run Metadata API functions. And test, deploy, and troubleshoot your apps.

You can also use Workbench to develop and test Tableau CRM Templates. Construct REST API calls to Tableau CRM endpoints to manage templates. After creating the template files with a POST call, you retrieve the files by exporting them from the Metadata API. Edit the files with a development tool of your choice.

For a complete reference to calling Tableau CRM API endpoints, see Tableau CRM REST API Developer Guide.

Distribute Your Tableau CRM Apps with Tableau CRM Templates

Tableau CRM Templates gives you a great way to leverage your Tableau CRM app development efforts. Turn your app into a template, which other organizations can use to create a version of the app using their own data.

To learn more and create Tableau CRM Templates, see the *Tableau CRM Templates Developer Guide.*.

Tableau CRM Migration, Packaging, and Distribution

Migrate Tableau CRM assets using change sets, bundle them together in managed packages, distribute and track packages through AppExchange and the License Management App, and use the metadata API to manage customizations for your org.

Migrate Tableau CRM Assets with Change Sets

Use change sets to move customized Tableau CRM assets between orgs that have a deployment connection. For example, create a Tableau CRM app containing dashboards, lenses, datasets, dataflows in your Sandbox org, then migrate the app assets to your production org once testing is complete.

Tips for Migrating Tableau CRM Assets with an Ant Script

If you use the Salesforce Ant Migration Tool to migrate Tableau CRM assets between orgs, consider the following tips.

Package Tableau CRM Assets in Managed Packages

You can create managed packages of Tableau CRM assets, including Tableau CRM apps, dashboards, lenses, datasets, recipes, dataflows, and user XMD. Use packages to distribute those assets to other users or organizations, including those outside your company.

Distribute Tableau CRM Assets Through AppExchange

Once your managed package is certified through our security review, you can make it available to your customers by uploading it to AppExchange. You can specify your package release type, control major and minor version numbers, and specify licensing options through the Licensing Management Application.

Use the Metadata API for Tableau CRM Assets and User XMD

Tableau CRM provides full support for the Metadata API, which can be used to retrieve, deploy, create, update, or delete customizations for your organization.

The Analytics Cloud SDK

Use the Tableau CRM SDK to embed Tableau CRM functionality directly where your users work everyday, without having to transition between Lightning Experience and Tableau CRM Studio. The Tableau CRM SDK lets you communicate and interact with Tableau CRM assets from Lightning Apps, Apex, Visualforce, and more. You can create one cohesive experience powered by Tableau CRM features directly in Salesforce pages and apps.

Tableau CRM Assets SDK

Are you ready to use the power of Tableau CRM directly inside Lightning Experience and create your own custom Tableau CRM functionality? By using the Tableau CRM Assets SDK, you can access the power of REST APIs to retrieve collections of Tableau CRM assets, such as dashboards, lenses, and datasets and describe the details of individual assets. Then, customize the display of the results via a Lightning Component controller. You can also create dynamic SAQL queries against your Tableau CRM data to display runtime results.

Tableau CRM Templates SDK

You have great apps, and you're creating app templates to copy or distribute those apps. And you might even be using the REST APIs to work with templates and folders.

By using the Tableau CRM Template SDK, your application can do many of the same things from a Lightning Component controller.

Tableau CRM Web SDK Events

Would you like your application to communicate with your Tableau CRM dashboards, whether your application is built with the Lightning SDK, Visualforce, or mobile? How about from an application outside of Salesforce? Wouldn't it be great if your application could apply filters or know about dashboard selections and filters made by a user?

Your application could take actions specific for your business if values fall outside of defined ranges. Or you could have an application that is a viewpoint for dashboards made available by different parts of your business application ecosystem. Imagine that: a single information source to present to your executive staff!

The Tableau CRM Web SDK events we're making available are the foundation for a new way of thinking about Tableau CRM applications. Coupled with the Lightning Locker, you can even code your application outside of Salesforce—you can interact with Tableau CRM from any JavaScript application.

Tableau CRM Apex SDK

Is your company one of the gazillion using custom code in Apex, the server-side programming language for Lightning Platform? Would you like it to be easier to query data in Tableau CRM directly from your Apex code? Say hello to the Tableau CRM Apex SDK and send well-formed SAQL queries to Tableau CRM.

This first phase of the Tableau CRM Apex SDK lets developers build SAQL queries and execute them in the security context of the logged-in user, ensuring that security settings are honored. API versioning is supported to avoid breaking applications as the SDK evolves. The SDK also offers `Wave.InvalidParameterException` to help catch bad values supplied to the class methods.

For complete information about the SDK, see the *Analytics Cloud SDK Developer Guide*.

The Analytics Cloud Dashboard Component

The Tableau CRM dashboard component is an Aura component used to embed Tableau CRM dashboards in Visualforce and Lightning pages. The component can render a live Tableau CRM dashboard or it can be interactive with the page using events and methods to update the dashboard state.

Tableau CRM Dashboard Component Attributes
The Tableau CRM dashboard component has attributes that can be set programmatically, allowing you to create, render, and interact with a Tableau CRM dashboard on a page at runtime.

Tableau CRM Dashboard Component Methods
The Tableau CRM dashboard component has methods that allow you to change the state of a Tableau CRM dashboard at runtime.

Tableau CRM Dashboard Component Use Cases
For the best result, use the dashboard component interactively with other components. This section has basic use cases to get you started on how to use the component methods, attributes, and Web SDK events.

For complete information about the Analytics Cloud dashboard component, see the *Analytics Cloud Dashboard Component Developer Guide*.

The Tableau CRM REST API

Access Analytics Cloud datasets and lenses programmatically using the Tableau CRM REST API.

For complete information about the API, see the *Tableau CRM REST API Developer Guide*.

SEE ALSO:

> Enable the Tableau CRM REST API

Salesforce Analytics Query Language (SAQL)

Use SAQL to access data in Analytics Cloud datasets. Analytics Cloud uses SAQL behind the scenes in lenses and dashboards to gather data for visualizations.

Developers can write SAQL to directly access Analytics Cloud data via:

- Tableau CRM REST API

  Build your own app to access and analyze Analytics Cloud data or integrate data with existing apps.

- Dashboard JSON

  Create advanced dashboards. A dashboard is a curated set of charts, metrics, and tables.

For complete information about SAQL, see the *Analytics Cloud SAQL Reference*.

SEE ALSO:

> View the Query Behind Your Lens

Explain, Predict, and Take Action with Einstein Discovery

Einstein Discovery augments your business intelligence with statistical modeling and supervised machine learning to identify, surface, and visualize insights into your business data. It uses predictive and prescriptive analysis to predict future outcomes, as well as suggests ways in which you can improve predicted outcomes. Einstein Discovery requires either the Tableau CRM Plus license or the Einstein Predictions license, both of which are available for an extra cost.

Main Topic	Help and Other Resources
Get Started	
Set Up Einstein Discovery on page 1593	• Learn About Einstein Discovery Permissions and Permission Sets on page 1593 • Assign Permission Sets to Users on page 1594
Get Started Using Einstein Discovery on page 1596	• Introduction to Einstein Discovery on page 1596 • Jumpstart Solutions with Story Templates on page 1598 • Tableau CRM Learning Map • Trailhead: Einstein Discovery Basics • Einstein Discovery Glossary on page 1607

Analyze Your Data and Investigate Results	
	• Trailhead: Understand What Could Happen Insights • Trailhead: Understand What Is The Difference Insights
Analyze Report Data	• Analyze Reports with Einstein Data Insights on page 265

Predict Outcomes and Take Actions	
Build, Deploy, and Manage Models	• About Models on page 1680 • Explore Model Metrics on page 1682 • Implement Recommended Updates on page 1705 • Deploy Models on page 1706 • Compare Models on page 1717 • Manage Prediction Definitions and Models on page 1719
Predict Outcomes	• Display Einstein Predictions Using Automated Prediction Fields on page 1751 • Add Einstein Predictions to a Lightning Page on page 1752 • Einstein Prediction Service on page 1754 • Trailhead: Einstein Prediction Service • Get Predictions in Apex on page 1776 • Get Predictions in Process Automation Formulas on page 1781 • Display Einstein Predictions Using Custom Fields (Deprecated) on page 1784
Get Predictions in Tableau	• Introduction to Einstein Discovery in Tableau on page 1793 • Get Predictions in Tableau Dashboards on page 1794 • Get Predictions in Tableau Calculated Fields on page 1798 • Embed Einstein Predictions in Tableau Flows on page 1799

Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

SEE ALSO:

Explore Data and Take Action with Tableau CRM

Analyze Reports with Einstein Data Insights

Set Up Einstein Discovery

Set up your org to use Einstein Discovery by creating and assigning permission sets and enabling Einstein Discovery.

> 📝 **Note:** Einstein Discovery requires either the Tableau CRM Plus license or the Einstein Predictions license.

Complete the following tasks to set up Einstein Discovery in your org.

1. Learn About Einstein Discovery Permissions and Permission Sets

   Learn about the user permissions and standard Einstein Discovery permission sets associated with either your Tableau CRM Plus or Einstein Predictions license.

2. Assign Permission Sets to Users

   After you have defined permission sets for Einstein Discovery users, you can assign the appropriate permission sets to individual users or groups of users. You can assign multiple permission sets to an individual user.

3. Monitor Usage Statistics for Einstein Discovery

   Einstein Discovery monitors usage statistics in real time. You can view the usage statistics in your org.

Learn About Einstein Discovery Permissions and Permission Sets

Learn about the user permissions and standard Einstein Discovery permission sets associated with either your Tableau CRM Plus or Einstein Predictions license.

Both the Tableau CRM Plus license and Einstein Predictions license provide two identical permission sets.

- **Tableau CRM Plus User** for restricted access to use a story or model
- **Tableau CRM Plus Admin** for admin-level access to all Einstein Discovery features, including the ability to create stories and deploy models

These permission sets are automatically created when the Tableau CRM Plus or Einstein Predictions license is provisioned in your org.

Tableau CRM Plus User Permission Set

The Tableau CRM Plus User permission set provides single-user access to read-only features enabled by the Tableau CRM Plus or Einstein Predictions license.

User Permission	What It Enables
Use Einstein Discovery	Use Einstein Discovery and view the stories that you have permission to view.
View Einstein Discovery Recommendations	View Einstein Discovery recommendations that are connected to a Salesforce object.

Tableau CRM Plus Admin Permission Set

The Tableau CRM Plus Admin permission set provides admin-level access to all features enabled by the Tableau CRM Plus or Einstein Predictions license.

User Permission	What It Enables
(Deprecated) Ignore predicate when creating story from dataset	(Deprecated) Previously required to create an Einstein Discovery story from a dataset with a predicate.
Connect Einstein Discovery Model	Connect an Einstein Discovery model to a Salesforce object.
Create and Update Einstein Discovery Stories	Create stories to discover relationships between a business relevant metric and the explanatory variables that are potential influencers of that metric.
Manage Einstein Discovery	Access all Einstein Discovery features and see all datasets and stories created by users.
Share Einstein Discovery Stories	Export Einstein Discovery stories to Quip.
Use Einstein Discovery	Use Einstein Discovery and view the datasets and stories that you have permission to view.
View Einstein Discovery Recommendations	View Einstein Discovery recommendations that are connected to a Salesforce object.

Custom Permission Sets for Einstein Discovery Users

In addition to the standard permission sets associated with your Tableau CRM Plus or Einstein Predictions license, you have the option to create custom permission sets that give users access to specific combinations of analytics functionality, including Einstein Discovery features. If you create a custom permission set:

- For **Licenses**, select **Tableau CRM Plus**.
- For **System Permissions**, select the Einstein Discovery permissions you want to grant in this permission set.

Assign Permission Sets to Users

After you have defined permission sets for Einstein Discovery users, you can assign the appropriate permission sets to individual users or groups of users. You can assign multiple permission sets to an individual user.

1. From Setup, in the Quick Find box, enter `Permission Sets`, and select **Permission Sets**.

2. Click the permission set you want to assign to users.

   > 📝 Note: Assign the Manage Einstein Discovery permission set only to users who administer Einstein Discovery.

3. Click **Manage Assignments**, and click **Add Assignments**.

4. Select one or more users, and click **Assign**.

5. Click **Done**.

Any selected users now have access to Einstein Discovery.

USER PERMISSIONS

To assign a permission set to users:
- Assign Permission Sets

Monitor Usage Statistics for Einstein Discovery

Einstein Discovery monitors usage statistics in real time. You can view the usage statistics in your org.

To view usage statistics:

1. From Setup, in the Quick Find box, enter `Discovery`.

2. Under **Einstein Discovery and Einstein Data Insights**, select **Usage**.

> **Note:** Times and dates are based on your org's default time zone.

Statistic	Description
Number of predictions run today	Total number of predictions run in your org since 12:00 am today. This metric applies to models that were deployed with automated prediction fields.
Number of story versions created today	Total number of all stories created in your org since 12:00 am today.
Number of concurrent stories which can be analyzed	Total number of story versions that are currently being created in your org.
Number of prediction API calls run today	Total number of predictions run in your org since 12:00 am today. This includes prediction requests from REST API calls, Apex calls, bulk scoring jobs, and predictions embedded in Lightning and Experience pages.
Number of story versions created this month	Total of all stories created in your org since 12:00 am on the first day of the current month.

SEE ALSO:

Einstein Discovery Limits

Get Started Using Einstein Discovery

Start here if you are new to Einstein Discovery.

Introduction to Einstein Discovery

Become familiar with Einstein Discovery.

Jumpstart Solutions with Story Templates

Story templates provide out-of-the box solution starter kits that you can customize to your solution requirements. Story templates provide end-to-end implementations of common business use cases. Designed to integrate seamlessly with your Salesforce data, story templates are built with industry best practices in mind. Einstein Discovery handles the heavy lifting of building the initial training dataset, creating a data pipeline with feature engineering, generating the first Einstein Discovery story and model, deploying a model into Salesforce, and getting predictions on your data and writing them back to the dataset.

Einstein Discovery Limits

Einstein Discovery has limits for Tableau CRM datasets, stories, and predictions.

Einstein Discovery Glossary

Familiarize yourself with terminology that is commonly associated with Einstein Discovery.

Learn More About Einstein Discovery

In addition to product documentation and training, you can supplement your understanding of Einstein Discovery with informal blogs and white papers. Consider these resources to help you on your learning journey.

Introduction to Einstein Discovery

Become familiar with Einstein Discovery.

Using Einstein Discovery to Understand and Improve Outcomes

Einstein Discovery augments your business intelligence with statistical modeling and machine learning to identify, surface, and visualize insights into your business data. Einstein Discovery uses:

- **descriptive analytics** that tell you what happened in your data
- **predictive analytics** that reveal why it happened (diagnostic insights), what could happen (predicted future outcomes based on statistical probabilities), and what is the difference between variables (comparative insights)
- **prescriptive analytics** that suggest ways in which to improve your predicted outcomes (improvements)

Einstein Discovery is integrated into your Salesforce environment so that you can quickly operationalize data analysis, predictions, and improvements with clicks, not code. In addition, developers can use the Einstein Prediction Service to programmatically retrieve predictions and write predictions to custom fields. Data specialists can predict outcomes within dataflows.

To use Einstein Discovery, your org must have either the Tableau CRM Plus or Einstein Predictions license, and Einstein Discovery must be enabled. You must also have the right permissions assigned to your user account. For instructions, see Set Up Einstein Discovery on page 1593.

Implementing Einstein Discovery Solutions

To start using Einstein Discovery, consider the following series of tasks.

#	Theme	Description	More Information
1	Use Case	Define an outcome that you want to improve. This is most likely a key performance indicator (KPI) for your business. Most organizations have many outcomes that are candidates for improvement. Start with one outcome that is particularly significant. Over time, you can develop and apply Einstein Discovery to many different outcomes concurrently.	Define Your Use Case on page 1614
2	Data	Design and populate the Tableau CRM dataset to contain the data you want to investigate. A dataset contains tabular data that you normalize for analysis. You can use Salesforce data as well as data that is external to Salesforce.	Integrate Your Data in Tableau CRM on page 597
3	Story	Based on that dataset, create and configure a story that tells Einstein how to analyze your data. Stories provide the settings and preferences that Einstein Discovery uses to generate insights and build models.	Create and Manage Stories on page 1621
4	Insight	Investigate the insights that Einstein generated during its analysis. Learn what patterns and statistical insights Einstein discovered in your data.	Explore Story Insights on page 1655
5	Model	(Predictive analytics only) Refine and deploy the predictive model associated with your story into Salesforce.	Build, Deploy, and Manage Models on page 1680
6	Prediction	(Predictive analytics only) Predict outcomes using the model you deployed.	Predict Outcomes on page 1750

Iterative Refinement and Continuous Improvement

Einstein Discovery is designed for rapid exploration, experimentation, and implementation. You learn as you go. Every step of the way, you use built-in feedback to check your results, review your assumptions, ask new questions, make adjustments, and try again. Add a column to your dataset. Change story settings. Tweak the model threshold. As you fine tune your approach, each improvement can help lead you to better operational outcomes.

Jumpstart Solutions with Story Templates

Story templates provide out-of-the box solution starter kits that you can customize to your solution requirements. Story templates provide end-to-end implementations of common business use cases. Designed to integrate seamlessly with your Salesforce data, story templates are built with industry best practices in mind. Einstein Discovery handles the heavy lifting of building the initial training dataset, creating a data pipeline with feature engineering, generating the first Einstein Discovery story and model, deploying a model into Salesforce, and getting predictions on your data and writing them back to the dataset.

Maximize Customer Revenue Story Template

Analyze which types of accounts are leading to higher sales. Use a regression model to predict which accounts might bring additional revenue.

Maximize Win Rate Story Template

Explore the key drivers for winning deals without the hassle of prepping your data.

Minimize Time to Close Story Template

Understand the key factors that drive shorter deal cycles without having to worry about where to begin.

SEE ALSO:

Create a Story from a Template

Maximize Customer Revenue Story Template

Analyze which types of accounts are leading to higher sales. Use a regression model to predict which accounts might bring additional revenue.

This article describes the app and associated assets that Einstein Discovery generates when you create a story using the Maximize Customer Revenue story template.

Use Case

Numeric

Prerequisites

Einstein Discovery checks your org to determine whether it meets the requirements to generate an app from this template.

Check for	What's Required
Minimum requirements	At least 400 distinct accounts with closed won opportunities.
Optional objects	• Org uses Opportunity Line Items • Org does not use Leads
Field access	The logged in user has access to all the fields needed to build an app from the template.
Salesforce configuration	Org uses Record Types.
Data	Account sales history up to three years.

Generated Assets

Einstein Discovery generates an app containing the following assets.

Asset Type	Default Name	Description
Story	Maximize Customer Revenue	Explains the driving factors for maximizing the customer revenue for an account, as well as recommendations on how to improve the outcome. The initial story focuses on fields that should provide good predictive signals and avoids obvious data leakage.
Tableau CRM Dataset	Account Training Dataset	Dataset used to train the machine learning model.
Tableau CRM Dataflow	`appName eltDataflow`	Tableau CRM dataflow that pulls relevant account information from Salesforce to generate a dataset that is used to train a machine learning model. Design is based on best practice feature engineering and data filters.
Dashboard	About This Story	- Story description - Dataset description - Number of Accounts Analyzed - Average Customer Value - Fields Created for the Training Dataset
Prediction Definition, Model	Predicted Customer Revenue, Predicted Revenue	Deployed automatically.

Salesforce Objects

- Account
- Opportunity
- Product
- Leads

Story Outcome Variable and Derived Variables

The generated dataset includes the following data. For a complete list, refer to the dataset or story.

Variable Name	Data Type	Description
`Total Revenue` (outcome)	numeric	Sum of all closed won opportunities for each account within the past three years.
`# Leads Converted`	numeric	Number of leads converted for each Account in the past number of days.
`# Won Deals`	numeric	Total number of deals won per Account based on the time frame used to calculate total revenue.
`Aggregates.FirstDealDate`	date	Close date for the first closed won Opportunity for each account based on the time frame used to calculate total revenue.

Variable Name	Data Type	Description
AvgerageDealAge	numeric	Average time to close a deal per Account based on the time frame used to calculate total revenue.
AvgerageDealSize	numeric	Average deal size per Account based on the time frame used to calculate total revenue.
Top Product Family	text	Top product family associated with each Opportunity based on the Opportunity Line Item with the largest total price.
Top Product Name	text	Top product associated with each Opportunity based on the Opportunity Line Item with the largest total price.

SEE ALSO:

Create a Story from a Template

Create a Story from a Template

Jumpstart Solutions with Story Templates

Maximize Win Rate Story Template

Explore the key drivers for winning deals without the hassle of prepping your data.

This article describes the app and associated assets that Einstein Discovery generates when you create a story using the Maximize Win Rate story template.

Use Case

Classification

Prerequisites

Einstein Discovery checks your org to determine whether it meets the requirements to generate an app from this template.

Check for	What's Required
Minimum requirements	At least 400 closed opportunities.
Optional objects	• Opportunity Line Items • Activities
Field access	The logged in user has access to all the fields needed to build an app from the template.
Salesforce configuration	Org uses Record Types.
Data	400 closed opportunities within the past three years (up to 1095 days).

Generated Assets

Einstein Discovery generates an app containing the following assets.

Asset Type	Default Name	Description
Story	Maximize Win Rate	Explains the driving factors for maximizing the likelihood for an opportunity to win as well as recommendations on how to improve the outcome. The initial story focuses on fields that should provide good predictive signals and avoids obvious data leakage.
Tableau CRM Dataset	Opportunity Win Rate Training Dataset	Dataset used to train the machine learning model.
Tableau CRM Dataflow	`appName eltDataflow`	Tableau CRM dataflow that pulls relevant account information from Salesforce to generate a dataset that is used to train a machine learning model. Design is based on best practice feature engineering and data filters.
Dashboard	About This Story	- Story description - Dataset description - Rows Analyzed - Win Rate - Fields Created for the Training Dataset
Prediction Definition, Model	Predicted Likelihood to Win, IsWon	Deployed automatically.

Salesforce Objects

- Account
- Opportunity
- Opportunity History
- Product
- Price Book
- Task
- Event
- User
- UserRole

Story Outcome Variable and Derived Variables

The generated dataset includes the following data. For a complete list, refer to the dataset or story.

Variable Name	Data Type	Description
`Won` (outcome)	text	Story goal is to maximize Won (true).
`# Amount Decreased`	numeric	Number of times the Opportunity amount was decreased.
`# Amount Increased`	numeric	Number of times the Opportunity Amount was increased before it was closed.

Variable Name	Data Type	Description
# Closed Date Pushed	numeric	Number of times the closed date was pushed out on the Opportunity before it closed.
# Completed Activities	numeric	Number of completed activities on the Opportunity.
# Lost Opportunities on Account	numeric	Number of closed lost opportunities associated with an Account.
# Won Opportunities on Account	number	Number of closed won opportunities associated with an Account.
Account Tenure	numeric	How long the Account record has lived in Salesforce. Calculation: (Today - Account Created Date).
Top Product Family	text	Top product family associated with each opportunity based on the Opportunity Line Item with the largest total price.
Top Product Name	text	Top product associated with each opportunity based on the Opportunity Line Item with the largest total price.

SEE ALSO:

Minimize Time to Close Story Template

Understand the key factors that drive shorter deal cycles without having to worry about where to begin.

This article describes the app and associated assets that Einstein Discovery generates when you create a story using the Minimize Time to Close story template.

Use Case

Numeric

Prerequisites

Einstein Discovery checks your org to determine whether it meets the requirements to generate an app from this template.

Check for	What's Required
Minimum requirements	At least 400 closed won opportunities.
Optional objects	- Opportunity Line Items - Activities

Check for	What's Required
Field access	The logged in user has access to all the fields needed to build an app from the template.
Salesforce configuration	Org uses Record Types.
Data	400 won opportunities within the past three years (up to 1095 days).

Generated Assets

Einstein Discovery generates an app containing the following assets.

Asset Type	Default Name	Description
Story	Minimize Time to Close	Explains the driving factors for minimizing the time to close for an opportunity, as well as suggested actions to improve the predicted outcome. The initial story focuses on fields that should provide good predictive signals and avoids obvious data leakage.
Tableau CRM Dataset	Opportunity Training Dataset	Dataset used to train the machine learning model.
Tableau CRM Dataflow	`appName eltDataflow`	Tableau CRM dataflow that pulls relevant account information from Salesforce to generate a dataset that is used to train a machine learning model. Design is based on best practice feature engineering and data filters.
Dashboard	About This Story	- Story description - Dataset description - Rows Analyzed - Average Opportunity Age - Average predicted time to close - Mean Absolute Error - Number of Open Deals with Predictions - Top Open Deal Most Likely to Push - Fields Created for the Training Dataset
Prediction Definition, Model	Minimize Time to Close, Age	Deployed automatically.

Salesforce Objects

- Account
- Opportunity
- Opportunity History
- Product
- Price Book
- Task

- Event
- User
- UserRole

Story Outcome Variable and Derived Variables

The generated dataset includes the following data. For a complete list, refer to the dataset or story.

Variable Name	Data Type	Description
Age (Outcome)	numeric	Age of the Opportunity based on the number of days between the Opportunity Created Date and Close Date.
# Amount Decreased	numeric	Number of times the Opportunity Amount was decreased before it was closed.
# Amount Increased	numeric	Number of times the Opportunity Amount was increased before it was closed.
# Closed Date Pushed	numeric	Number of times the closed date was pushed out on the Opportunity before it was closed.
# Completed Activities	numeric	Number of completed activities on the Opportunity.
# Lost Opportunities on Account	numeric	Number of past closed lost opportunities associated with an Account when each Opportunity is closed.
# Won Opportunities on Account	numeric	Number of past closed won opportunities associated with an Account when each Opportunity is closed.
Account Tenure	numeric	Number of days the account has been a customer at the time each Opportunity was created. Calculation: (Opportunity Created Date - First Won Opportunity Created Date).
Owner Tenure	text	How long the Opportunity owner has been a Salesforce user. Calculation: (Today - User Created Date).
Top Product Family	text	Top product family associated with each Opportunity based on the Opportunity Line Item with the largest total price.
Top Product Name	text	Top product associated with each Opportunity based on the Opportunity Line Item with the largest total price.

SEE ALSO:

Create a Story from a Template

Create a Story from a Template

Jumpstart Solutions with Story Templates

Einstein Discovery Limits

Einstein Discovery has limits for Tableau CRM datasets, stories, and predictions.

Tableau CRM Dataset Limits

Tableau CRM datasets have the following limits:

Limit	Minimum	Maximum
Number of data rows	• Descriptive Insights: 50 • Predictive Insights: 400	20,000,000
Number of columns	3 (1 outcome variable plus 2 dataset columns)	50

Story Limits

Einstein Discovery stories have the following limits:

Limit	Number
Number of story creations per org per day. The Usage Statistics chart displays the cumulative total for this metric.	100
Number of story creations per org per month. The Usage Statistics chart displays the cumulative total for this metric. 📝 **Note:** Customers can purchase more stories (in blocks of 1,000) with Einstein Analytics Plus - Additional Stories (1,000).	500
Number of concurrent story creations per org. The Usage Statistics chart displays the cumulative total for this metric.	2
Number of concurrent queries per user.	10 (shared with Einstein Data Insights)
Number of queries per user per day.	10,000 (shared with Einstein Data Insights)

Prediction Limits

Limits for Einstein predictions are based on how predictions are configured when the model associated with the prediction is deployed. For instructions, see

Automated Prediction Field

The following limits apply to predictions associated with models deployed using the following settings:

• **Create a new prediction field from label**

• **Use an existing prediction field**

This kind of Einstein Discovery prediction has the following limits:

Limit	Number
Number of unique Salesforce objects to which models can be deployed using automated prediction fields. The Usage Statistics chart displays the cumulative total for this metric.	5
Number of active models deployed to a single prediction definition using automated prediction fields. 📝 **Note:** To deploy multiple models to the same prediction definition, select **Use an existing prediction field** during deployment. To change the model order in Model Manager, see Change the Model Evaluation Order in a Prediction Definition on page 1747.	10
Maximum number of unique predictions on a single entity with automated prediction fields	3
Number of predictions requested per org per day using automated prediction fields.	500,000

No Prediction Field

The following limits apply to predictions associated with models deployed using the following setting: **No prediction field (automatic predictions will not be enabled)**. This kind of Einstein Discovery prediction is run programmatically via the Einstein Prediction Service on page 1754 using REST API calls or implementations based on the Einstein Discovery Managed Package.

Limit	Number
Number of active models deployed to a single prediction definition using manually configured predictions fields.	10
Number of Einstein Prediction Service API calls per org per day. The Usage Statistics chart displays the cumulative total for this metric.	50,000
Number of predictions returned from a single Einstein Prediction Service API request. Used for Einstein Prediction Service on page 1754 bulk scoring API calls. For more information, see Score Records in Bulk on page 1789.	200
Number of Einstein Prediction Service API requests per user per hour.	2,000
Number of concurrent Einstein Prediction Service API requests within an org.	5

Other Limitations

- Multi-value fields, which are fields that contain multiple values (such as a list or array), are not supported in Einstein Discovery. To learn more, see this KB article: Limitations of MultiValue Fields in Tableau CRM.
- Accessibility features aren't incorporated into Einstein Discovery.

SEE ALSO:

Monitor Usage Statistics for Einstein Discovery

Einstein Discovery Glossary

Familiarize yourself with terminology that is commonly associated with Einstein Discovery.

Actionable Variable

An *actionable variable* is an explanatory variable that people can control, such as deciding which marketing campaign to use for a particular customer. Contrast these variables with explanatory variables that cannot be controlled, such as a customer's street address or a person's age. If a variable is designated as actionable, the model uses prescriptive analytics to suggest actions (improvements) the user can take to improve the *predicted* outcome.

Actual Outcome

An *actual outcome* is the real-world value of an observation's outcome variable after the outcome has occurred. Einstein Discovery calculates model performance by comparing how closely predicted outcomes come to actual outcomes. An actual outcome is sometimes called an *observed outcome*.

Algorithm

See *modeling algorithm*.

Average

In Einstein Discovery, the *average* represents the statistical mean for a variable.

Bias

If Einstein Discovery detects *bias* in your data, it means that variables are being treated unequally in your model. Removing bias from your story can produce more ethical and accountable models and, therefore, predictions. See *Disparate Impact*.

Binary Outcome

A *binary outcome* is an outcome variable that has only two text values, such as win-lose, pass-fail, or retain-churn. A story that represents a classification use case has a binary outcome variable.

Cardinality

Cardinality is the number of distinct values in a category. Variables with high cardinality (too many distinct values) can result in complex visualizations that are difficult to read and interpret. Einstein Discovery supports up to 100 categories per variable. You can optionally consolidate the remaining categories (categories with fewer than 25 observations) into a category called **Other**. Null values are put into a category called **Unspecified**.

Categorical Variable

A *categorical variable* is a type of variable that represents qualitative values (categories). A story that represents a classification use case has a categorical outcome variable. See *category*.

Category

A *category* is a qualitative value that usually contains categorical (text) data, such as Product Category, Lead Status, and Case Subject. Categories are handy for grouping and filtering your data. Unlike measures, you can't perform math on categories. In Salesforce Help for Tableau CRM datasets, categories are referred to as *dimensions*.

Causation

Causation describes a cause-and-effect relationship between things. In Einstein Discovery, causality refers to the degree to which variables influence each other (or not), such as between explanatory variables and an outcome variable. Some variables can have an obvious, direct effect on each other (for example, how price and discount affect the sales margin). Other variables can have a weaker, less obvious effect (for example, how weather can affect on-time delivery). Many variables have no effect on each other: they are independent and mutually exclusive (for example, win-loss records of soccer teams and currency exchange rates). It's important to remember that you cannot presume a causal relationship between variables based simply on a statistical correlation between them. In fact, correlation provides you with a *hint* that indicates further investigation into the association between those variables. Only with more exploration can you determine whether a causal link between them really exists and, if so, how significant that effect is.

Classification Use Case

In Einstein Discovery, the *classification use case* applies to story outcome variables that are binary outcomes: categorical (text) fields that contain only two values. Examples include variables that are either public or private, churned or not churned, and so on. These fields separate your data into two distinct groups. For analysis purposes, Einstein Discovery converts the two values into Boolean true and false. Predicting a categorical field is a binary classification problem with its own set of metrics to measure model quality. Einstein Discovery uses logistic regression to analyze binary outcomes.

Coefficient

A *coefficient* is a numeric value that represents the impact that an explanatory variable (or a pair of explanatory variables) has on the outcome variable. The coefficient quantifies the change in the mean of the outcome variable when there is a one-unit shift in the explanatory variable, assuming all other variables in the model remain constant.

Correlation

A *correlation* is simply the *association*—or "co-relationship"—between two or more things. In Einstein Discovery, correlation describes the statistical association between variables, typically between explanatory variables and an outcome variable. The strength of the correlation is quantified as a percentage. The higher the percentage, the stronger the correlation. However, keep in mind that correlation is *not* causation. Correlation merely describes the strength of association between variables, not whether they causally affect each other.

Count

A *count* is the number of observations (rows) associated with an analysis. The count can represent all observations in the dataset, or the subset of observations that meet associated filter criteria.

Date Variable

A *date variable* is a type of variable that contains time (temporal) data.

Deployment Wizard

The *Deployment Wizard* is the Einstein Discovery tool used to deploy models into your Salesforce org.

Descriptive Insights

Descriptive insights are insights derived from historical data using descriptive analytics. What Happened insights in Einstein Discovery are descriptive insights. Einstein Data Insights produces descriptive insights for reports.

Disparate Impact

If Einstein Discovery detects *disparate impact* in your data, it means that the data reflects discriminatory practices toward a particular demographic. For example, your data can reveal gender disparities in starting salaries. Removing disparate impact from your story can produce more accountable and ethical models and, therefore, predictions.

Dominant Values

If Einstein Discovery detects *dominant values* in a variable, it means that the data is unbalanced. Most values are in the same category, which limits the value of the analysis.

Drift

Over time, a deployed model's performance can *drift*, becoming less accurate in predicting outcomes. Drift can occur due to changing factors in the data or in your business environment. Drift also results from now-obsolete assumptions built into the story on which the model is based. To remedy a model that has drifted, you can refresh it by adjusting story settings, retraining it on newer data, and redeploying it.

Duplicates

If Einstein Discovery detects a *duplicate* condition in your data, it means that two or more explanatory variables are highly correlated (for example, City and Postal Code). These variables have a *duplicate impact* on the outcome. Einstein Discovery recommends choosing just one variable to improve results. Consider keeping the most descriptive field (for example, City) to make insights more easily interpretable. This condition is also known as *multicollinearity*.

Tableau CRM Dataset

A *Tableau CRM dataset* is a collection of related data that is stored in a denormalized, yet highly compressed, form. The data is optimized for analysis and interactive exploration.

Ethical Use

Ethical use reflects the application of AI and machine learning for fair and unbiased purposes. With Einstein Discovery, it's the practice of producing ethical and accountable stories, insights, and predictions. For an overview, take the Responsible Creation of Artifical Intelligence Trailhead module.

Explanatory Variable

An *explanatory variable* is a variable that you explore to determine whether, and to what degree, it can influence the outcome variable for your story. Einstein Discovery calculates statistical associations between explanatory variables and the outcome variable. Based on the strength of the association, you can investigate further whether and how that explanatory variable affects the outcome variable. An explanatory variable is sometimes called an *input variable*, a *feature*, a *predictor variable*, or an *independent variable*.

Feature Selection

Feature selection involves choosing the optimum set of explanatory variables in a story. Ideally, a story contains the number of explanatory variables that best explain variations in the outcome variable. A story with too few explanatory variables can be too vague to detect underlying patterns in the data, result in an underfitting model. A story with too many explanatory variables can be overly specific and complex to filter out noise in the data, resulting in an overfitting model. Successful feature selection includes the most influential explanatory variables with no significant lurking variables (important explanatory variables that are missing from the story).

First-Order Analysis

In an insight, a *first-order analysis* examines how one explanatory variable explains variation in the outcome variable. First-order analysis is sometimes called *bivariate analysis*.

Generalized Linear Model (GLM)

General Linear Model is a regression-based modeling algorithm that Einstein Discovery uses to build a model from a story.

Goal

A *goal* specifies the desired outcome for your story. A story's goal includes its outcome variable plus your preferred direction (minimize or maximize) you want the outcome to take. For example, your goal could be to maximize margin or to minimize costs. Einstein Discovery uses the story goal to orient its analysis and explain the insights it uncovered from the analysis.

Gradient Boosting

Gradient Boosting is a decision tree-based ensemble machine learning algorithm that Einstein Discovery uses to build a model from a story. Also called *Gradient Boosting Machine* (or *GBM*).

Identical Values

If Einstein Discovery detects *identical values* in your data, it means that all values for a variable belong to the same category. Having identical values increases complexity—but no benefit—to the analysis of your data.

Improvement

An *improvement* is a suggested action, based on prescriptive analytics, that a user can take to improve the likelihood of a desired outcome. Improvements are associated with *actionable variables*, which are explanatory variables that people can control. Taking a suggested action can improve the *predicted* outcome. An improvement is analogous to a *prescription* in prescriptive analytics.

Insight

An *insight* is a statistically significant finding in your data. When you create a story, Einstein Discovery analyzes the data in your dataset and generates insights based on its analysis. Insights provide a starting point for you to investigate the relationships among your story's explanatory variables and its goal.

k-fold Cross-Validation

Model validation process in which Einstein Discovery randomly divides all the observations in the Tableau CRM dataset into four separate partitions of equal size. Next, it completes four test passes (folds) in which three of the partitions serve as the training set

and one partition serves as the test set. For each fold, Einstein Discovery compiles model metrics, then averages the metrics for all four folds.

Leakage

Leakage occurs when the data used to train your model includes one or more variables that contain the information that you are trying to predict. This can result in models that are extremely accurate when, in actuality, they are problematic. To remedy data leakage, remove any variables from your model that are causing the leakage.

Linear Regression

In Einstein Discovery, *linear regression* is an analytical technique used for the numeric use case.

Logistic Regression

In Einstein Discovery, *logistic regression* is an analytical technique used for the classification use case.

Lurking Variable

A *lurking variable* is an explanatory variable that is missing from your story but which significantly explains variations in the outcome variable.

Mean

A *mean* is the statistical average: the sum of all items divided by the number of items.

Measure

A numeric value that quantifies something. See *Numeric Variable*.

Model

A *model* is the sophisticated, custom algorithm that Einstein Discovery generates automatically upon story creation. Models are based on a comprehensive, statistical understanding of past outcomes and are used to predict future outcomes. A model accepts the values of one or more predictor variables as input and produces a predicted outcome as output, along with top factors and improvements (if requested).

Modeling Algorithm

A *modeling algorithm* is what Einstein Discovery uses to create a model for a story. Einstein Discovery uses several algorithms: generalized linear model (GLM) is a regression-based algorithm, while gradient boosting machine (GBM) and XGBoost are decision tree-based machine learning algorithms.

Model Manager

The *Model Manager* is the Einstein Discovery tool used to manage predictions and models you have deployed.

Model Metrics

Model metrics describe the performance of the predictive model associated with your story. It provides metrics (quality indicators, which are sometimes called *fit statistics*) to show how well the model's predictions fit the training data in the dataset. For definitions of quality indicators shown in the Model Metrics tabs, see

Numerical Variable

A *numerical variable* is a type of variable that represents quantitative values (numbers), such as revenue or price. You can do math on numeric variables, such as calculating the total revenue or the average price. A numeric value always has an associated unit of measure, such as currency, volume, or weight. A story that represents a numeric use case has a numeric outcome variable. In the Tableau CRM dataset documentation, a numeric column is referred to as a *measure*.

Numerical Use Case

In Einstein Discovery, the *numerical use case* applies to story outcome variables that are numeric. Predicting a number field is a regression problem with its own set of metrics to measure model quality. Einstein Discovery uses linear regression to analyze numeric outcomes.

Observation

An *observation* represents an instance of the data you want to analyze. An observation is analogous to a row of data in a Tableau CRM dataset, or to a record in a Salesforce object. For example, if your story's goal is to maximize opportunity wins, then each observation represents a single opportunity.

Outcome

An *outcome* is the business result you are trying to analyze or predict. An outcome is typically a key performance indicator (KPI), such as sales margin or opportunity wins.

Outcome Variable

In a story, the *outcome variable* is the column selected as the single, primary focus for analysis and predictions. The goal of a story is to maximize or minimize its outcome variable. An outcome variable is sometimes referred to as the *response*, the *target variable*, or the *dependent variable*.

Outlier

If Einstein Discovery detects *outliers* in your data, it means that a variable contains data points that are unusually distant from the average value. Uncommonly large or small numbers, potentially from data entry errors or rare events, can produce misleading charts or predictions. Depending on your situation, excluding outliers from a story can yield better insights and predictions.

Overfitting

In predictive analytics, *overfitting* occurs when a model performs well in predicting outcomes on the training data in the dataset, but less well when predicting outcomes for other data, such as production data. Using too many explanatory variables can result in an overly complex predictive model that captures the noise in your data. To mitigate overfitting, Einstein Discovery uses ridge regression and regularization. See also *underfitting*.

Predicted Outcome

A prediction. Einstein Discovery calculates model performance by comparing how closely predicted outcomes come to actual outcomes.

Prediction

In Einstein Discovery, a prediction is a derived value, produced by a model, that represents a possible future outcome. You can think of a prediction as the output of a predictive model that is based on the inputs of predictor variables that the model accepts.

Prediction Definition

In Einstein Prediction Service, a parent resource that contains one or more models. If a prediction definition contains multiple models, then each model produces predictions for a different segment of the data.

Prediction Field

A *prediction field* is a field where Einstein stores prediction scores for a Salesforce object. During deployment, Einstein can create this field automatically (called an *automated prediction field*), or a custom prediction field can be created later if needed.

Predictive Analytics

Predictive analytics is the practice of analyzing historical and current data, based on AI, machine learning, predictive modeling, and statistical techniques. Einstein Discovery uses predictive analytics to identify patterns and predict probabilistic future outcomes.

Predictive Model

See *model*.

Predictor or Predictor Variable

A variable that a model expects as input. A prediction request passes values for each predictor variable that the model requires. Based on the provided input values, the model's algorithm produces a prediction as output. Predictors are also known as *independent variables*.

Prescription

See *improvement*.

Prescriptive Analytics

Prescriptive analytics is the practice of suggesting actions to improve predicted outcomes.

Proxy Variable

A *proxy variable* is an explanatory variable that is highly correlated to another explanatory variable in relation to the outcome variable. When a proxy variable, such as a loan applicant's street address, is highly correlated to a protected characteristic, such as ethnicity, it can reflect discriminatory practices that compromise your analysis and predictions with unwanted bias. Einstein Discovery helps

1611

you identify proxy variables so that you can remove them, and the bias they reflect, from consideration in your stories, insights, and predictions.

R^2

R^2 measures a regression's model's ability to explain variation in the outcome. It represents the proportion of the variance in the outcome variable that is predictable from one or more explanatory variables. In general, the higher the R^2, the better the model predicts outcomes. R^2 is a commonly used metric for numeric use cases.

Recommended Updates

When analyzing your data, Einstein Discovery looks for issues, such as outliers or duplicates, that can decrease the value of the analysis. If detected, Einstein Discovery presents you *recommended updates* to fix these data issues in your story.

Response

See *outcome variable*.

Ridge Regression

Ridge regression is a regularization approach that Einstein Discovery uses to mitigate model overfitting by preventing coefficients from getting too large.

Score

(noun) A prediction associated with an observation. (verb) The process of predicting outcomes for a set of observations.

Second-Order Analysis

In an insight, a *second-order analysis* examines how the combination of two explanatory variables explains variation in the outcome variable. In second-order analysis, the combined impact of both variables together on the outcome is sometimes called the *interaction effect*. Second-order analysis is sometimes called *multivariate analysis*.

Segment

A *segment* is a subset of observations (rows) that meet the criteria specified in the segment filter. See *segmentation*.

Segmentation

Segmentation involves filtering your data to focus your prediction on a particular group, such as a single customer type or gender.

Story

A *story* defines the data and analytical settings that Einstein Discovery uses to generate insights and build models. Story settings include the outcome variable, whether to maximize or minimize the outcome variable, the data to analyze in a Tableau CRM dataset, and other preferences. Story settings tell Einstein Discovery how to conduct the analysis and communicate its results.

Story Setup Wizard

The *story setup wizard* is the Einstein Discovery tool used to define your story settings, such as the story goal, data selections, and other preferences. Einstein Discovery uses these story settings to analyze the data, produce insights, and (for predictive stories) generate a model.

Strongest Predictor

If Einstein Discovery detects a *strongest predictor* in your data, it means that a variable explains the most variation in the data. Remove the variable if there is an obvious mathematical relationship between it and the outcome (for example Cost and Price). Similarly, remove the variable if it is known only after the outcome is known (for example, Reason for Churn in a customer churn analysis). Excluding strongest predictor variables can expose more subtle patterns in your data.

Terminal State

Data that is finalized and not expected to change. An example of finalized data is the date on which an order shipped. A record that has reached its terminal state represents an *actual* outcome (also called *observed outcome*). Define the conditions under which your story's outcome variable has attained its terminal state.

Text Variable

See *Categorical Variable, Binary Outcome*.

Threshold

In a classification model, the *threshold* value tells your model how to classify a binary outcome. If the calculated probability is above the threshold value, Einstein classifies the outcome one way (such as True or Positive). If the calculated probability is below the threshold value, Einstein classifies the outcome the other way (such as False or Negative). The default threshold is 0.5, but you can tune this value up or down to accommodate your use case. The threshold is sometimes called the *Classification Threshold* or *Decision Threshold*.

Top Predictors

Top predictors are the conditions that most significantly drive the predicted outcome, in decreasing order of magnitude. A condition is a data value associated with a column. In Einstein Discovery, a predictor consists of one or two conditions. See *predictor variables*.

Training Set

In predictive analytics, the *training set* is the portion of the data in your dataset that Einstein Discovery uses to train your model to make predictions. See also *validation set*.

Underfitting

In predictive analytics, *underfitting* occurs when a model performs poorly in predicting outcomes on the training data in the dataset. Underfitting is often a result of an excessively simple model in which there aren't enough variables for a statistical algorithm to capture the underlying patterns in the data. See also *overfitting*.

Validation Set

In predictive analytics, the *validation set* is the portion of the data in your dataset that Einstein Discovery uses to validate the predictions generated by your trained model. See also *training set*.

Variable

A *variable* represents a characteristic of the data you are analyzing. A variable is analogous to a column in a dataset or a field in a Salesforce object. For example, an opportunity has variables—such as the opportunity type, lead source, fiscal year, lead source, expected amount—that describe properties associated with each opportunity. Each variable has one data type (number, text, or date). Einstein Discovery analyzes relationships among two types of variables: outcome variables and explanatory variables.

XGBoost

XGBoost is a decision tree-based, ensemble machine learning algorithm that Einstein Discovery uses to build a model from a story.

Learn More About Einstein Discovery

In addition to product documentation and training, you can supplement your understanding of Einstein Discovery with informal blogs and white papers. Consider these resources to help you on your learning journey.

Overview

- White Paper: Understanding the Differentiating Capabilities and Unique Features of Salesforce Einstein Discovery within the Machine Learning Space
- Videos: Tableau CRM Training (scroll down to Einstein Discovery videos)
- Blog: Staying focused – Using a methodology to organize your thoughts and project activities

Solution Workflow

To start using Einstein Discovery, consider the following series of tasks.

#	Phase	More Information	
1	Defining Your Use Case	• Blog: What kind of questions can Tableau CRM Stories answer?	

#	Phase	More Information
2	Prepping Data	- Blog: Preparing Your Data for Einstein Discovery - Blog: Create Intelligent Applications with the Predict node in Recipes
3	Managing Stories	- Blog: What are Tableau CRM Stories and when would you use them? - Blog: Ethical AI Can't Wait: 4 Ways To Drive Greater Equality in Your AI - Blog: Accelerating Prescriptive Analytics Using Einstein Discovery Templates
4	Exploring Insights	Blog: Take your Einstein Discovery model from Good to Great
5	Managing Models	- Blog: The complete guide to Einstein Discovery model deployments - Blog: Einstein Discovery: How good is my model really? - Blog: Managing Einstein Discovery Models in the Wild - Blog: The Delicacy of Accuracy: A Deep Dive on Classification Performance
6	Getting Predictions and Improvements	Blog: Einstein Discovery Prediction Service on Postman — 3 Easy Steps

Other Resources

- Blog: Tree-Based Learning Algorithms in Einstein Discovery
- Tableau CRM Learning Days

Define Your Use Case

Begin by defining the focus and purpose of your use case, and determining what data is needed to achieve that purpose.

Preliminary Considerations

To begin exploring data with Einstein Discovery, consider some preliminary questions:

- What outcome (such as a KPI) do you want to explore? Einstein Discovery detects patterns related to continuous (measure) or binary (two-value text) outcomes.
- What explanatory variables do you want to include in analysis? These are factors that can influence your outcome.
- Where can you find this information? What are possible data sources? A Salesforce object? An existing dataset? Data that is external to Salesforce?
- Is there enough data for Einstein to analyze? For details, see Einstein Discovery Limits on page 1605.
- In addition to analyzing historical data (descriptive insights), do you want to predict outcomes and get recommended actions?

Identify the Outcome Variable

Decide which outcome variable you want to explore, and at what granularity. The outcome variable could be a KPI value (such as revenue, discount, cost measure, or duration) or other quantifiable outcome. You can also use categories (text fields) with two values (binary) as an outcome variable, such as Win/Loss. In general, binary outcomes are less accurate to predict than continuous value outcomes. Occasionally, a new metric is created, such as customer revenue by month. It's possible to create outcome variable metrics.

Einstein Discovery algorithms assume that each record is independent and is not related to other records. If relationships exist between records, create a variable within the row of data to capture that behavior. For example, if the same Opportunity has multiple competitors, don't prepare multiple rows of data with the same Opportunity ID. Instead, create more fields on one Opportunity ID and indicate whether each of the top-10 competitors were present in the deal.

In addition, clarify your goal. Einstein Discovery orients its analysis based on maximizing or minimizing the outcome variable. For example, your goal can be to maximize net margin or minimize customer churn.

Identify Explanatory Variables

Think about which variables can possibly describe or influence the outcome. For example, to investigate sales, potential influencer variables can include Discount, Days between Lead Received and Last Contacted, Lead Source, Region, Vertical, Competitor, and Promotion. When selecting predictor variables, you want to gather a maximum amount of information from a minimum number of variables. Einstein Discovery helps this process by eliminating variables that do not have good explanatory power from the story it generates.

Dates are rolled up to a duration when used as input in Einstein Discovery. If a business process has multiple key dates, use the Data Manager to create multiple variables in which to store numeric durations (for example, Days between Lead to Last Contact and Days between Demo to Trial). Common date variable roll-ups include the earliest date and the most recent date. Time durations can also be represented in either absolute or relative form.

> 📝 Note: For proof-of-concept projects, keep your input data from 10 through 25 variables. It's faster to learn and improve your data preparation skills with a less complex model.

Select any fields (predictor variables) that directly affect the outcome. Ensure that the variable data is clean and consistent. The order and meaning of input predictor variables must remain the same from record to record. Inconsistent data formats, "dirty data," and outliers can undermine the quality of analytical findings.

Then you shape the data into analytical fields with derived variables that describe or influence the outcome variable. Shaping data involves subject-matter expertise to creatively select, create, and transform variables for maximum influence.

Identify Data Sources and Fields

Based on your outcome variable, determine which data sources can best represent the fields for business processes associated with the outcome variable. Potential sources include Salesforce objects (including custom objects) and data that is external to Salesforce.

Prepare Data for Analysis

Create and populate a Tableau CRM dataset with the data you want Einstein Discovery to analyze. Effective data preparation is key to getting great results with your story insights, predictions, and improvements.

Engage the data integration power of Tableau CRM to load and transform data from one or more data sources into a dataset. You can pull data from Salesforce and external sources. You can also use third-party tools and utilities to further expedite data cleansing and wrangling tasks. For instructions on using Data Prep and other data integration options, see Integrate Data into Tableau CRM Datasets on page 597.

Determine Data Requirements

Data preparation is a process of iterative refinement. As you dig deeper into your data, new clues emerge. Discoveries can cause you to reassess previous assumptions and adjust your data prep implementation accordingly.

Importance of Preparing Data for Analysis

Data scientists typically invest time and effort to plan and prepare their data. They know how much the quality of the output depends on the design and quality of the input. Even if you aren't a data scientist, you can definitely improve your results by applying basic principles to help you implement your solution. Data preparation involves aggregating and optimizing data associated with the outcome variable you're investigating, along with potential explanatory variables that can influence the outcome.

Explore Data Integration Options

Tableau CRM provides many ways in which to populate your datasets, including Data Prep, as described in Integrate Data into Tableau CRM Datasets on page 597. You can pull data stored in a dimensional data warehouse or in a transactional database format. If so, use record identifiers or primary keys to join fields from multiple tables to create a single, unified, flattened view. Your view contains an outcome variable, along with input predictor variables collected at a level of analytical granularity on which you can make actionable decisions.

For many outcome variables, data is captured at various business process steps in multiple data sources. For example, a sales process can have data in a CRM, an email marketing program, and Excel spreadsheet, and an accounting system. If that is the case, identify the fields in those systems that can link the different data sources together.

Consider Prior Record States

If you want to capture changes in data over a time period, determine whether your data source keeps only the current state values of a record. Transactional application data sources (Salesforce, for example), contain only the most recent values for a record. Other data sources capture transactional data in a chronological log. Each new version of the record is appended to the log, and previous versions of the record are retained in earlier log entries. Getting a prior value requires storing a snapshot of the historical data, or keeping the prior value data in custom fields in the current record.

Determine the Appropriate Level of Granularity for the Insights You Want

What level of insights are of interest to achieve the objective? For example, customer-level insights are of interest when looking at customer revenue. In the Data Manager, use grouping to adjust the granularity of the data. Choose a granularity that is actionable, understandable, and useful so that you can incorporate the results into your business process or application.

A common mistake is to overly aggregate data. Keep the desired outcome in mind, and use data collected in rows at that level of granularity. Data analyzed in Excel can be at a different level from what you want for Einstein Discovery. For example, to understand the

effects of day of the week, provide data at the day level. You cannot predict a day-level outcome from an aggregated, monthly level dataset.

Determine Relevant Time Frames

Statistical analysis datasets can summarize a lifetime of values in just one single row, with many columns that describe different points in time. Don't collect a lifetime of fields if a specific window of time more accurately reflects the outcome variable you want to analyze and predict. Usually events closest to the outcome are stronger predictors than events that happened a long time ago. Consider a reasonable cut-off time to ensure that your data is sufficiently recent to be relevant.

To use Einstein Discovery for predictions, your variables must be at the point in time on which the prediction is based. For example, suppose that your objective is to decrease defaults on loans by not pre-approving loans that are likely to default. In this case, you capture variables, such as a credit score at the time of loan application and prior. If the person was late on two payments after loan origination, it would not be used in the pre-approval analysis because they have already been approved.

Decide How Much Data to Get

To build reliable predictive models, provide Einstein Discovery with as much data as possible to resemble real-world distributions of variables . The actual number of records is not always easy to determine because it depends on patterns found in your data. If you have more noise in your data, you need more data to overcome it. Noise in this context means unobserved relationships in the data that the input predictor variables do not capture. In general, more rows of data are better for analysis accuracy. Columns with more possible values result in finer segmentation of the data, but it can require more rows of data for a statistically sound analysis. For example, 10,000 rows with a binary outcome of gender (either male or female) results in potentially 5,000 observations per gender. But 10,000 rows with a variable indicating 50 states results in potentially 200 observations per state.

Consider the Time Series

Data that changes over time must be reflected in your model and also in your associated dataset. When time sequences (Lead Received > Quote Provided > Deal Closed) are important in predictions, proportionally collect data from those different time periods. The key principle is to provide data that reflects what actually happens in the real world at the right level of outcome metric granularity.

Think Proportionally

When collecting data, think about the balance of values for variables in your raw data. For example, how many vertical industry records are there by time period? When extracting a subset of data, include approximately the same proportion of variables in your input dataset. If you provide more records of one variable (vertical, in our example), you can unintentionally introduce bias into your analysis. If you have datasets with millions of rows, it is less likely to encounter accidental bias.

Provide Known Outcomes in Your Data

Einstein Discovery orients its analysis around a particular outcome, typically a key performance indicator (KPI), such as sales margin or opportunity win. Providing data with known outcome values gives Einstein Discovery something to work with. For example, if you're targeting deal win rates, then your data should reflect deals that are definitively won or lost. If the deal is not complete—it is neither won nor lost—then Einstein omits the deal from analysis because the outcome value is missing.

Consider Bias and Fairness in Your Data

Does the data that you want to use reflect business practices that are possibly biased or unfair? To help you produce ethical and accountable insights and models, Einstein Discovery detects proxy variables and disparate impact in your dataset. You can also flag and filter sensitive variables (such as gender or age) to see where they show up in your insights. If Einstein Discovery exposes bias in your

data, you can simply exclude the biased data from your story. To learn more, see Detect and Remove Bias from a Story on page 1646. In addition, consider excluding biased data during data prep. For an overview of ethical and accountable AI, take the Responsible Creation of Artificial Intelligence Trailhead module.

Analyze Without Overfitting or Underfitting

Einstein Discovery figures out which variables and combinations of variables best explain the behavior of your chosen metric without overfitting or underfitting:

Issue	Approach
Overfitting	Occurs when using too many variable fields in a predictive model. Overfitting captures the noise in your data with an overly complex, unreliable way so that the model memorizes unnecessary details. When new data comes in, the model fails. To avoid overfitting, exclude variables that are too detailed.
Underfitting	Often the result of an excessively simple model. The statistical algorithm cannot capture the underlying patterns in the data.

Thus, there is a delicate balance between being too specific with too many variables and too vague with not enough selected variables.

Data Preparation and Iterative Improvement

Depending on your project, data preparation can be a one-time activity or a periodic one. Analyzing your data is an iterative process that can continue after your solution has been deployed. As new insights are revealed, it is common to experiment by adding or changing aspects of the input data. You can schedule analysis to continually add new data to your model incrementally. It is also common to periodically update your model variables and fields with new information or better focused business questions.

To get started, in your story setup, select a few columns of data, including one outcome variable (for example, revenue, units, or days). Also select several independent predictor variables (for example, product category, region, date of sale, customer type) to find what insights are immediately identified. Then you can continue to experiment with more columns of data or by organizing your dataset in a machine-learning-friendly format.

📝 **Note:** It's best to start with just a few columns and then add more columns as needed with the goal of optimizing your results.

Create Calculated Columns in Your Dataset

Create calculated columns in your dataset to extract more useful information, such as a ratio or aggregation. A calculated column uses a formula to derive its value from other data (such as fields, expressions, and values).

Value of Calculated Columns

Calculated columns can provide a succinct, single representation of meaningful but more complex data relationships. For example:

- Fields that precisely describe the outcome you're analyzing or predicting can improve pattern detection and enable more actionable insights to be found.
- Calculated columns usually results in better analysis and higher model accuracy than any single-variable transformation.

For your use case, consider ways in which you can use calculated columns to boost your analysis and models.

Types of Calculated Columns

Type	Description
Aggregations	Examples of commonly computed aggregated fields include: mean (average), most recent, minimum, maximum, sum, multiplying two variables together, and ratios made by dividing one variable by another.
Ratios	Ratios can communicate more complex concepts, such as a price-to-earnings ratio, in which price or earnings alone can deliver this insight.
Transformations	Transformation refers to the replacement of a variable by a function. For instance, replacing a variable by its square or cube root or logarithm is a transformation. You transform variables when you want to change the scale of a variable or standardize the values of a variable for better understanding. Variable transformation can also be done using categories or bins to create variables: for example, binning continuous Lead Age into Lead Age Groups or Price into Price Categories, such as Discount, Retail, and OEM.

> **Note:** Multi-value fields, which are fields that contain multiple values (such as a list or array), are not supported in Einstein Discovery. To learn more, see this KB article: Limitations of MultiValue Fields in Tableau CRM.

Ways to Calculate Column Values

Tableau CRM provides several approaches for preparing data:

Approach	To Learn More
Data Prep and transformations	Clean, Transform, and Load Data with Data Prep on page 702
dataflows and transformations	Design Datasets with Dataflows and the Dataset Builder on page 871
Data Prep Classic (recipes) and calculated fields	Clean, Transform, and Load Data with Data Prep Classic on page 812

To determine the best way to calculate values for your use case, see Should I Use a Recipe or Dataflow? on page 698

Cleanse and Prepare Data

Tableau CRM can suggest ways in which to clean and prepare your data. It can look at combinations of variables to detect complex predictors and to identify data issues that you can fix.

Assess Your Source Data and Correct Issues at the Source

Assess the condition of your source data. As you collect the data into variables, profile the values. Look for data problems, such as extremes, outliers, missing values, incorrect values, skew, and high cardinality. Common data preparation issues are identified during the data-loading process.

We recommend that you address data quality issues as early as possible. You can repair them in Tableau CRM, in the source system, or in your data preparation process. If you are seeing errors from source applications, a best practice is to resolve the issue at the source system instead of during data preparation.

Address Common Data Issues

The following table describes some approaches for handling common data issues when preparing data for analysis.

> 📝 **Note:** Einstein Discovery detects some of these issues for you during analysis. If found, Einstein prompts you to fix an issue in the story so that fixing it in the dataset is not required. For more information, see Select Recommended Updates to a Story on page 1648.

Issue	Approach
Extreme Values and Outliers	Einstein Discovery algorithms are sensitive to outliers because those values affect averages (means) and standard deviations in statistical significance calculations. If you find unusual values or outliers, confirm whether these data points are relevant and real. Often unusual values are errors. If the extreme data points are accurate, predictable, and reoccurring, do not remove them unless those points are unimportant. You can reduce outlier influence by using transformations or converting the numeric variable to a categorical value with binning.
Missing Values	The most common repair for missing values is imputing a likely or expected value using a mean or computed value from a distribution. If you use a mean value, you could reduce your standard deviation. Thus, the distribution imputation approach is more reliable. Another approach is to remove records with missing values. Don't get too ambitious with filtering out missing values. Sometimes the pattern is in the missing data. Also, if you delete too many records, you undermine the real-world aspects in your analysis. 📝 **Note:** Einstein Discovery orients its analysis around a particular outcome. If the value of the targeted outcome is missing from a particular row, then Einstein Discovery excludes that row from analysis.
Incorrect Values	Predictive algorithms assume that the input information is correct. If only a few rows have incorrect values, decide what to do. Remove those rows from the analysis, or replace the incorrect values with more correct or average values. If there are numerous inaccurate values, determine why the inaccuracies happened and whether it's possible to repair them. Sometimes it's better to remove a highly error-prone variable than include it in the analysis.
Standardize Categorical Values	For category values, ensure consistent category names. Remove spelling variations (such as plurals or abbreviations). Fix typos and other errors. Use labels that are meaningful, recognizable, and easy to interpret.
Skewed Data	For continuous variables, review the distributions, central tendency, and spread of the variable. These variables are measured using various statistical visualization methods. Confirm that continuous variables are normally distributed. If not, try to reduce skewness for optimal prediction. For categorical variables, use a frequency table, along with a bar chart, to understand distributions of each category. If variable values are skewed, Einstein Discovery could produce biased models. When a skewed distribution must be corrected, transform the variable using a function, such as the Box-Cox transformation. After applying the fix, a normal distribution for the variable is achieved. The newly prepared, transformed variable performs much better for predictive modeling purposes.
High-Cardinality Fields	High-cardinality fields are categorical attributes that contain many distinct values. Examples include names, ZIP codes, or account numbers. Although these variables can be highly informative, high-cardinality attributes are rarely used in predictive modeling. Including these attributes vastly increases the dimensionality of the dataset, which can make it difficult for most algorithms to build accurate prediction models.

Issue	Approach
Binary Outcomes and Boolean Variables	If a variable has a binary outcome (only two possible values), and those values are represented by numbers (for example, 1 and 0), then convert those numeric values to text values (for example, "TRUE" and "FALSE" or "NOTCHURNED" and "CHURNED"). For solutions that implement the classification use case (binary outcomes), Einstein Discovery requires the outcome values to be represented as text values. For other variables in the dataset, converting these values to text can improve the interpretability of the charts and explanations in the resulting insights.
Ordinal Variables	Ordinal variables are problematic for predictive models. Ordinal data consists of numerical scores on an arbitrary scale that is designed to show ranking in a set of data points. For example, Low, Medium, and High are ordinal. Predictive algorithms assume that the variable is an interval or ratio variable and therefore be misled or confused by the scale. Ordinal variables are treated as categorical. If you have ordinal values, transform them into continuous or categorical values.
Duplicate, Redundant, or Highly Correlated Variables	Minimize duplicate, redundant, or other highly correlated variables that carry the same information. Einstein Discovery algorithms perform better without these kinds of collinear variables. Collinearity occurs when two or more predictor variables are highly correlated. As a result, one can be linearly predicted from the others with a substantial degree of accuracy. To avoid collinearity, do not include multiple variables that are highly correlated or data that is from the same reporting hierarchy. For example, customers who live in the city of Tampa also live in the state of Florida. To identify high correlation between two continuous variables, review scatter plots. The pattern of a scatter plot indicates the relationship between variables. The relationship can be linear or nonlinear. To find the strength of the relationship, compute correlation. Correlation varies between -1 and $+1$.

Maximize Interpretability for Insights

When preparing your data, consider the downstream effects of your decisions. Keep in mind the importance of making charts and explanations easier for users to review and interpret. For example, using consistent

Analyze Your Data with Stories and Insights

Create stories to analyze your data. Explore the insights that Einstein Discovery revealed in your data using AI, machine learning, and statistical analysis. Insights show you what happened in your data, why it happened, and what could happen.

Create and Manage Stories

Stories tell Einstein Discovery what to analyze. Creating a story involves defining the dataset fields, outcome variable, explanatory variables, and other settings to include in the analysis. Einstein Discovery uses this information to generate insights into your data.

Explore Story Insights

You can explore insights for any story to which you have access. An insight is a statistically significant finding in your data. When you create a story version, Einstein Discovery analyzes the data in your dataset and generates insights based on its analysis. Insights provide a starting point for you to investigate the relationships among your story's explanatory variables and its goal.

Create and Manage Stories

Stories tell Einstein Discovery what to analyze. Creating a story involves defining the dataset fields, outcome variable, explanatory variables, and other settings to include in the analysis. Einstein Discovery uses this information to generate insights into your data.

About Stories

Use Einstein Discovery stories to augment your data analysis by exposing relevant facts, themes, and statistical correlations in your data.

Explaining and Predicting Outcomes

A *story* helps you explore relationships between a business-relevant outcome (called the *outcome variable*) and the possible factors (called *explanatory variables*) that can potentially influence the outcome. For example, a story can focus on margin as the outcome variable. For explanatory variables, the story can consider business data (such as customer type or campaign) as potential contributing factors to the margin. Based on its understanding of the relationships in your data, Einstein can predict future outcomes and suggests ways in which to improve the probability of even better outcomes.

Analytical Settings

A story defines the data and analytical settings that Einstein Discovery uses to generate insights and build predictive models. Story settings include the outcome variable, whether to maximize or minimize the outcome variable, the data to analyze in a Tableau CRM dataset, and other preferences. Story settings tell Einstein Discovery how to conduct the analysis and communicate its results.

Insights

A story contains answers, explanations, predictions, and suggested actions that arranged into an organized presentation with logical flow and related sections. The story is filled with insights about your data as they relate to the outcome you're interested in. Einstein Discovery walks you through what has happened and why, what has changed, what is likely to happen, and what you can do about it. Now anyone can tap into the power of data science and conduct statistical analtics and predictive analytics.

Shared Stories

Depending on your user permissions, you can create stories or use stories that others have created and given you access to.

Create a Story

Create a story from a Tableau CRM dataset or a template. Einstein Discovery thoroughly analyzes your data and generates revealing and meaningful insights for you to explore. If you create a story that includes predictions, Einstein Discovery also generates a predictive model.

Create a Story from a Dataset

Create a story to uncover the meaningful relationships among variables in a Tableau CRM dataset. Einstein Discovery augments your analysis by quickly sifting through huge amounts of data to uncover statistically significant correlations for you to investigate further.

Create a Story from a Template

Quickly create a story using Einstein Discovery story templates. Each story template provides a starter implementation for a common Einstein Discovery use case. A template gives you what you need to build the app, prepare and load your data, create an initial story, deploy a model, and save predictions into your Tableau CRM dataset.

Cancel Story Creation

You can cancel story creation before Einstein finishes creating it. Canceling a story interrupts Einstein data analysis and abandons any story elements created after you clicked **Create Story**. This feature allows you to change your mind and not wait until Einstein after completes analyzing your data and generating insights.

Create a Story from a Dataset

Create a story to uncover the meaningful relationships among variables in a Tableau CRM dataset. Einstein Discovery augments your analysis by quickly sifting through huge amounts of data to uncover statistically significant correlations for you to investigate further.

Ways to Start Creating a Story from a Dataset

There are three ways in which to launch the **Create Story** command:

- From the **Create** menu. When prompted, choose **Create from Dataset**.

- While viewing a lens.

- From the dataset dropdown, as described in step 1 that follows.

  Note: You cannot create stories from live datasets, which are described in Explore Data Directly in Snowflake on page 1058.

Steps to Create a Story

1. Select a Dataset

    Select the Tableau CRM dataset that contains the data you want to analyze.

2. Configure Story Settings

    Select the goal, title, and location for your story.

3. Select the Story Type

    Choose whether you want Einstein to generate predictions and insights, or insights only (What Happened insights).

4. Select Story Options

    Choose how you want to select the fields in your dataset to analyze. Einstein can select the fields for you automatically, or you can select and configure them manually.

5. Select Data Options

    Manually select and configure the fields in your dataset that you want Einstein to analyze.

6. Analyze Data

    Einstein Discovery prepares your story, analyzes your data, and displays its progress each step of the way.

Select a Dataset

Select the Tableau CRM dataset that contains the data you want to analyze.

Considerations for Dataset Selection

- To get great results with your stories, models, and predictions, start by preparing the data you want to analyze in your dataset. For instructions, see Prepare Data for Analysis on page 1615.

- To create a story, your dataset must contain a minimum number of rows for Einstein to analyze, as described in Einstein Discovery Limits on page 1605. If your dataset contains enough rows to create descriptive insights but not enough rows to create predictive insights, Einstein Discovery generates only What Happened insights.

- For story creation, you can use a dataset with a predicate (row-level security). Your access must be at least 50 rows of data (400 rows for a predictive model). All users who access the story can see the results of the story. They don't need the same row-level access as the story creator.

- A story is based on a snapshot of the data in the Tableau CRM dataset. The initial data snapshot is taken when the story is created. Thereafter, the story is *locked* to that snapshot by default. If data has changed in the source dataset, users with sufficient permissions

can optionally refresh the story based on the most recent data (see Open a Story on page 1634). Otherwise, subsequent changes to the story do not affect the snapshot, and subsequent changes to the dataset are ignored.

Consider selecting datasets that are more static than dynamic. If your dataset is continually updated, Salesforce periodically removes older versions of the dataset, which can include the snapshot upon which your story is based. If the snapshot is no longer available, then you can still view the story but cannot edit it.

To select a dataset:

1. In Tableau CRM Studio, find the dataset you want to use to create the story. To find the datasets you are authorized to use, select the Datasets subtab.

2. Open the dropdown on the Tableau CRM dataset.

3. Click **Create Story**.

> 📝 Note: If you don't see **Create Story** in the dropdown menu, ask your administrator about your user permissions and whether Einstein Discovery is set up for your org.

Configure Story Settings

Select the goal, title, and location for your story.
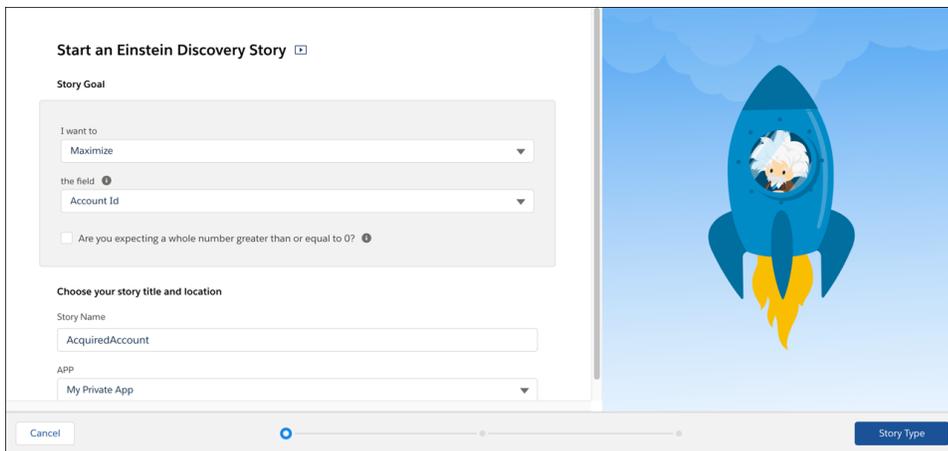
1. Specify the desired outcome (or *goal*) of your story. A story goal includes two parts: the outcome variable in your dataset that you want to analyze (typically a key performance indicator), and the preferred direction (minimize or maximize) that you want the outcome to take. For example, your goal could be to maximize margin or to minimize costs. Einstein Discovery uses the story goal to orient its analysis and explain the insights it uncovered from the analysis.

Field	Description
I want to	Select one of the following options: • **Maximize** tells Einstein that you want to maximize the outcome. • **Minimize** tells Einstein that you want to minimize the outcome.
the field	Select the outcome that you want to use. You can choose two types of fields: • number fields (measures)

Field	Description
	• text fields (categories) with two values (such as public/private, win/lose, approved/rejected, and so on) ✎ Note: For a field to appear in this list, it must contain values in a minimum number of rows in the dataset: • descriptive insights only: 50 rows or more • predictive insights: 400 rows or more For additional information, see Einstein Discovery Limits on page 1605.
Are you expecting a whole number greater than or equal to 0?	Calculations often result in fractions. Select this option if you want your story outcome values to show up as whole numbers (using automatic rounding). Certain values make sense only when expressed as whole numbers, such as the number of orders per month, or the number of customers entering a retail store.

2. Specify the name and location of your story.

Option	Description
Story Name	Accept the default name or enter a new name.
APP	Select the app where you want to save the story. If you want to share this story with others, save it into a shared app.

3. Click **Story Type**.

Select the Story Type

Choose whether you want Einstein to generate predictions and insights, or insights only (What Happened insights).

1. Select one of the following options.

Option	Description
Insights Only	Produces only What Happened insights. Choose this option for faster story creation if you don't need predictions and improvements. After your story is created, you can change your mind and manually add predictive analysis by enabling the **Create Predictive Model** setting. Note: This option omits models, so model-related features described in Build, Deploy, and Manage Models on page 1680 and Predict Outcomes on page 1750 are not available in this type of story.
Insights & Predictions	Produces all insight types. Choose this option if you want this story to include Why It Happened, What Could Happen, and What Is The Difference insights. This option tells Einstein to perform descriptive, predictive, and prescriptive analysis on your data.

2. Click **Setup Options**.

Select Story Options

Choose how you want to select the fields in your dataset to analyze. Einstein can select the fields for you automatically, or you can select and configure them manually.
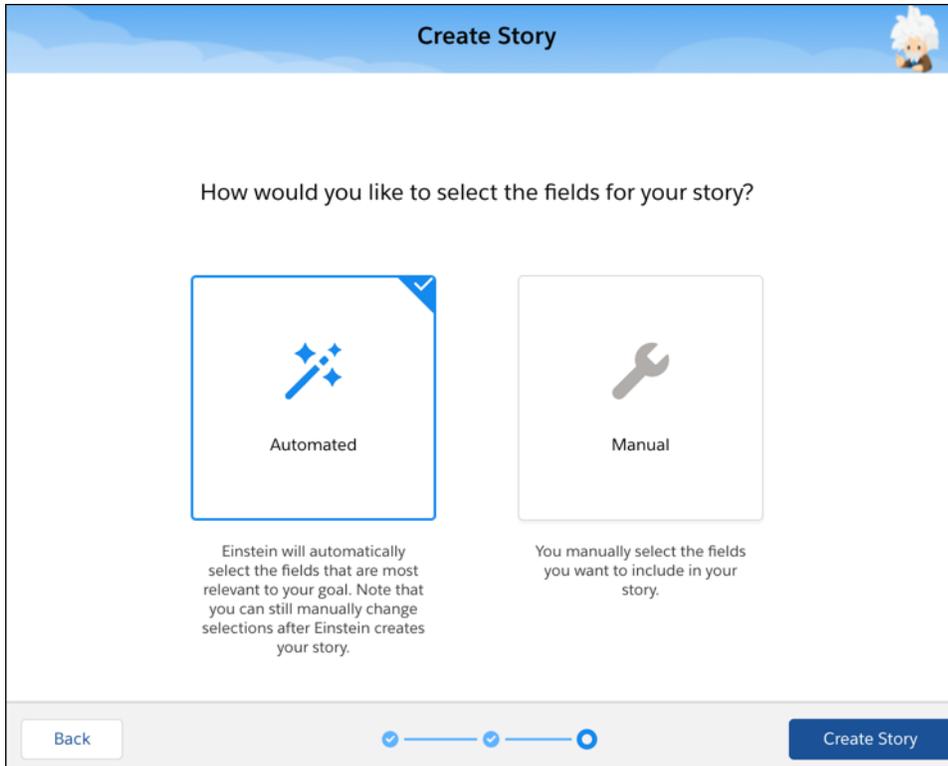
1. Select one of the following options:

Option	Description
Automated	Einstein Discovery searches your dataset and automatically selects the columns that best correlate with your story's outcome. After your story is created, you can manually change column selections (see Edit Story Settings on page 1639).
Manual	Allows you to manually select the fields to use in your story.

2. Do one of the following:

   - If you chose **Automated**, click **Create Story**.
   - If you chose **Manual**, click **Data Options**.

Select Data Options

Manually select and configure the fields in your dataset that you want Einstein to analyze.

📝 Note:

   - The list of fields includes numeric fields and text fields with values represented at least 25 times.
   - Multi-value fields, which are fields that contain multiple values (such as a list or array), are not supported in Einstein Discovery. To learn more, see Einstein Discovery Limits on page 1605.

📝 **Note:** Changing data settings in this screen affects the data in your *story*. It does not affect the data stored in the underlying dataset.

1. Select the other fields to include in your story. Select a minimum of two fields.

   - To help you decide, for each field, Einstein shows you its statistical correlation with the selected outcome. The **Correlation** column shows how much each field contributed to the outcome, in descending order of impact. If a field has little or no correlation to the outcome, consider removing it from the story to improve the model.

   - To filter the list of fields, type one or more characters in the **Search Variables** box.

2. To configure other settings for an individual field, click it and change its settings in the right panel.

For instructions, see:

> 📝 **Note:** The **Filters Applied** column shows any selected filters for a field.

3. Enable **Create Predictive Model** if you want Einstein Discovery to make predictions and recommendations for this story. Disable it if you want only What Happened insights.

4. If **Create Predictive Model** is selected, you can choose among different algorithms for Einstein Discovery to use to create the model associated with this story. Select one from the **Algorithm** list.

Algorithm	Description
GLM	Default. Generalized Linear Model is a regression-based algorithm.
GBM	Gradient Boost Machine is a decision tree-based ensemble machine learning algorithm.
XGBoost	XGBoost is a decision tree-based ensemble machine learning algorithm.
Random Forest	Random Forest is a supervised learning algorithm that uses multiple decision trees, randomization, and other optimization techniques.

Alternatively, select **Model Tournament** to have Einstein Discovery run all four algorithms and then show the results of the algorithm that performed best.

5. When ready, click **Create Story**.

Analyze Data

Einstein Discovery prepares your story, analyzes your data, and displays its progress each step of the way.
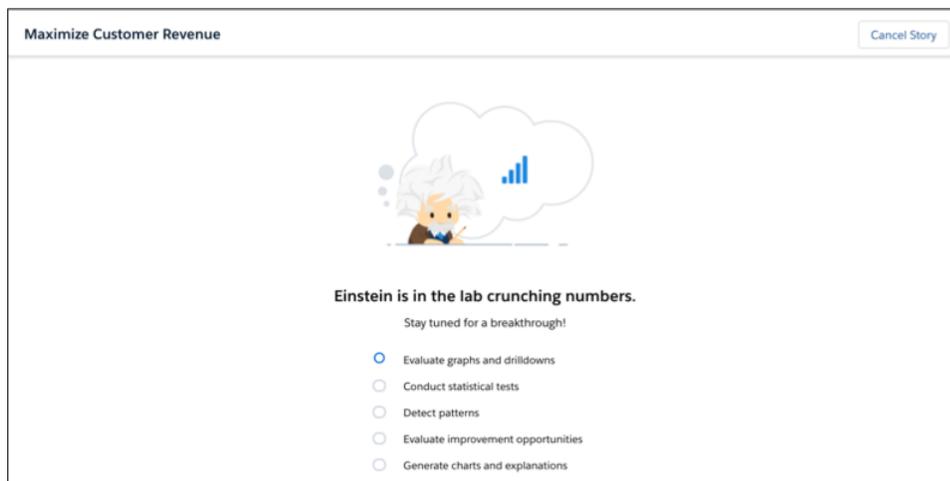
When finished, Einstein opens the new story.
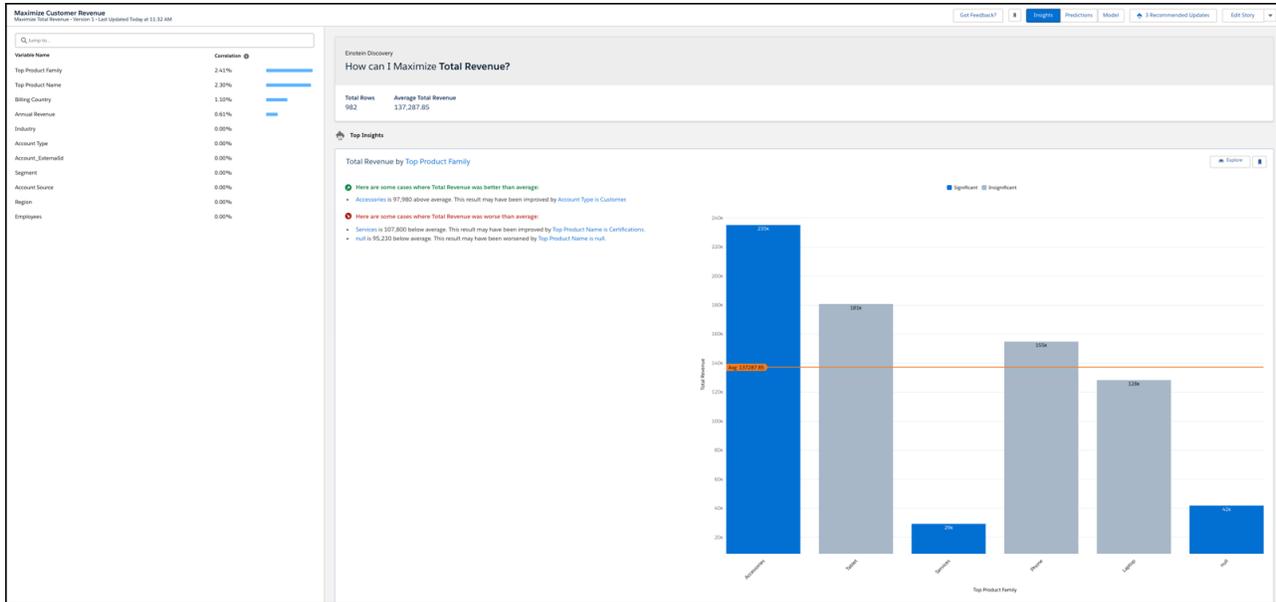
The story provides a list of variables, insights and charts that are ranked according to their impact on the selected outcome variable. In addition, Einstein Discovery determines their level of interest and relevance to the outcome.

Once created, your new story is added to the list of stories to which you have access. If you saved the story to a shared app, then users with access to that app have access to your story as well.

Create a Story from a Template

Quickly create a story using Einstein Discovery story templates. Each story template provides a starter implementation for a common Einstein Discovery use case. A template gives you what you need to build the app, prepare and load your data, create an initial story, deploy a model, and save predictions into your Tableau CRM dataset.

To create a story from a story template:

1. In Tableau CRM Studio, click **Create**, then **Story**.
2. Select one of the available templates, and then click **Continue**.
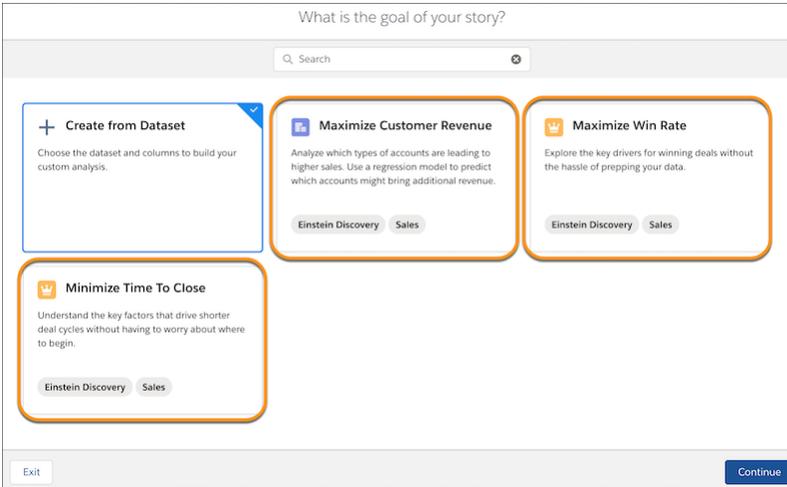
3. Review summary information about the template.

The template overview includes a description (1), a list of supported objects (2), and sample insights (3). Click **Continue**.
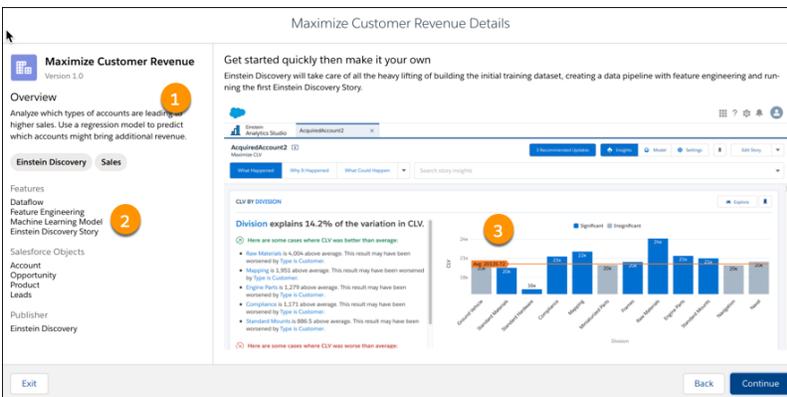
4. Einstein Discovery runs through a series of checks to confirm that your organization's data meets minimum requirements. If your org passes, click **Looks good, next**. Otherwise, review any messages and check the requirements for your template in Jumpstart Solutions with Story Templates on page 1598.

5. Give your app a name, then click **Create**.
   Einstein Discovery proceeds to create the app and associated assets, including a dataset, dataflow, story, model, and documentation (as a dashboard). When finished, your initial story opens.

6. Explore the initial story, insights, dataflow, deployed model, and other assets that Einstein Discovery created for you. As you explore, consider ways in which you can customize your app to better reflect your business requirements and goals.

Even if your organization passed the minimum data requirements for the selected story template, app creation can fail due to other reasons, including:

Issue	Description
Story concurrency limits exceeded	No more than two stories can be created concurrently in your org. App creation can fail if there are already two stories in the org that are currently being analyzed.
Dataflow run limits exceeded	During app creation, Einstein Discovery story templates run a dataflow twice: (1) to create the dataset used to train the predictive model, and (2) to use the predictive model to generate prediction scores and write them back to the Tableau CRM dataset. App creation can fail if you exceed the maximum number of dataflow runs in your org in a 24-hour period.
Data Sync-related limits exceeded	Story templates can add objects to Data Sync. App creation can fail if your org has already created the maximum number of data sync objects. Consider contacting your Salesforce account representative to request an increase in the number of Data Sync objects allowed.
Data Sync-related errors	With Data Sync enabled, dataflows in story templates rely on Data Sync objects to run successfully. If app creation triggers Data Sync-related errors, address them in the Data Manager before trying to create the template again.

SEE ALSO:

Jumpstart Solutions with Story Templates

Maximize Customer Revenue Story Template

Maximize Win Rate Story Template

Minimize Time to Close Story Template

Cancel Story Creation

You can cancel story creation before Einstein finishes creating it. Canceling a story interrupts Einstein data analysis and abandons any story elements created after you clicked **Create Story**. This feature allows you to change your mind and not wait until Einstein after completes analyzing your data and generating insights.

To cancel story creation, click the **Cancel Story** button.
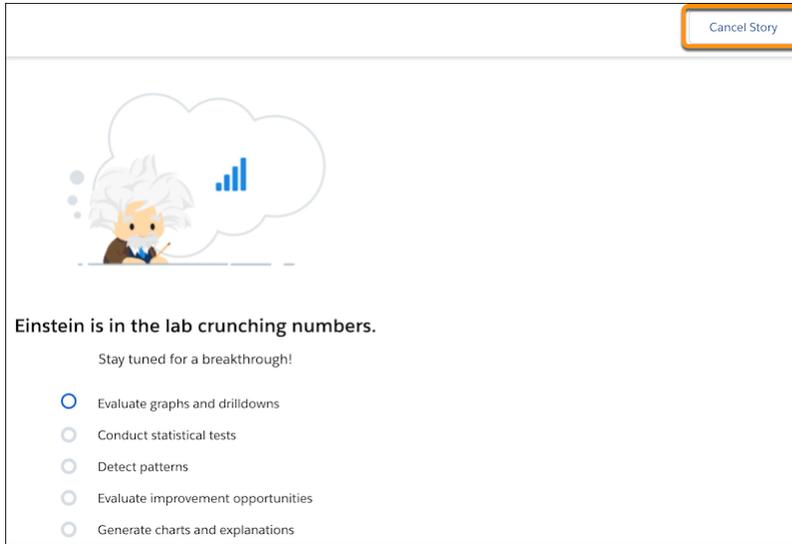
Clicking the button halts the creation of a new story (or a new story version) and returns you to the Home page.

Open a Story

Open a story to explore relationships in your data.

To open a story:

1. In your Salesforce org, from the App Launcher (⋮⋮⋮), find and open **Tableau CRM Studio**.

2. Find the story you want to open. To filter the list, select the **Stories** subtab.

   If you don't see the **Stories** subtab, ask your administrator about your user permissions and whether Einstein Discovery is set up for your org.

3. Click the story you want to open.
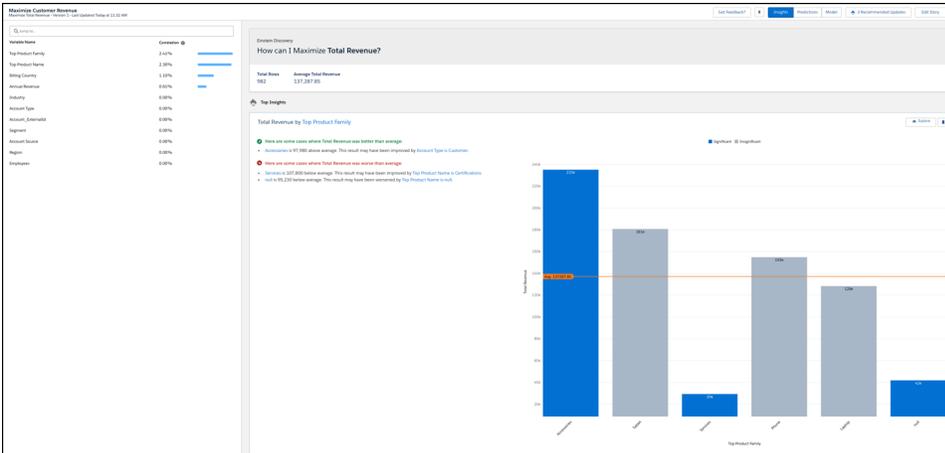
Einstein Discovery displays the story. You can have multiple stories open at the same time.

Navigate a Story

Navigate a story you've opened.

Story Interface

The story interface consists of a variable filter panel, story navigation bar, and insights list.

Story Toolbar

The Story Toolbar provides commands to manage a story.

Story Interface

The story interface consists of a variable filter panel, story navigation bar, and insights list.

Elements of the Story Interface

When a story is open, the interface consists of the following regions.

Area	Name	Description
1	Story Headline	Name of this story, selected goal, most recent version.
2	Story Toolbar on page 1638	Tools to: • view suggested story improvements, story insights, and Model Metrics • view and update story settings • change the selected outcome • rename or delete this story • publish this story to Quip • deploy the model story model to a Salesforce object
3	Variables Panel	Shows you the list of explanatory variables in your story and their correlation to the story outcome.
4	Story Version Summary	Summary of story insights, including version comparison.
5	Insights List	List of insights associated with this story.

Story Headline

The basis of the story. This section consists of the following regions.

Area	Name	Description
1	Story Name	Name of this story
2	Version Update	Date and time the most recent story version was created.
3	Story Goal	Desired story outcome.
4	Story Version	Selected version.

Story Version Summary

Summary of the story. This section consists of the following regions.

Area	Name	Description
1	Goal	Includes outcome variable plus the preferred direction (minimize or maximize) the story outcome should to take.
2	Row Count	The number of observations (rows) associated with an analysis.
3	% Change in Row Count from Previous Version	Row change difference between the current version to the previous one.
4	Outcome Average	Statistical mean.

Area	Name	Description
5	% Change in Outcome Average from Previous Version	Outcome difference between the current version to the previous one.
6	What Changed (Compare to a Different Version)	Compare the current version to a previous one.

Story Toolbar

The Story Toolbar provides commands to manage a story.

Area	Name	Description
1	**Got Feedback?**	Submit product feedback.
2	**Bookmark Filter**	• If selected, shows only bookmarked insights in the Insights list. • If not selected, shows all insights in the Insights list. See Bookmark an Insight in a Story on page 1677.
3	**Insights**	Display insights for this story. See Navigate Story Insights on page 1655.
4	**Predictions**	Display predictions for this story. See Navigate Story Insights on page 1655.
5	**Model**	Display metrics for this model. See Explore Model Metrics on page 1682.
6	**Improvements Found**	Improve a story. See Select Recommended Updates to a Story on page 1648.
7	**Edit Story**	Edit story settings. See Edit Story Settings on page 1639.
8	Click the dropdown menu arrow to see the remaining tools.	
9	**Change Outcome**	Run the story setup wizard and select different story settings (such as the story goal or data options).
10	**Deploy Model**	Deploy a model associated with this story into Salesforce. See Deploy Models on page 1706.
11	**Version History**	Display version history. See Export and Share Insights on page 1678.
12	**Publish to Quip**	Export pinned insights to a sharable Quip document. See Export and Share Insights on page 1678.

Area	Name	Description
13	**Properties**	Save this story (with its current settings) into an app. Once saved, your new story is added to the list of stories to which you have access. If you save the story to a shared app, then users with access to that app have access to this story as well. See Rename or Move a Story on page 1654.
14	**Delete**	Delete this story. See Delete a Story on page 1655.

Edit Story Settings

Working with stories is often an iterative process of further refinement. As you investigate insights in your story, you can decide to improve it by revising your settings and creating a newer version. For example, you can include or exclude a column and rerun the analysis. By interacting with the story, you overlay your intuition and domain knowledge to make the story more insightful and its recommendations more pertinent.

To edit a story:

1. Open the story.

2. From the Story Toolbar, click **Edit Story**.
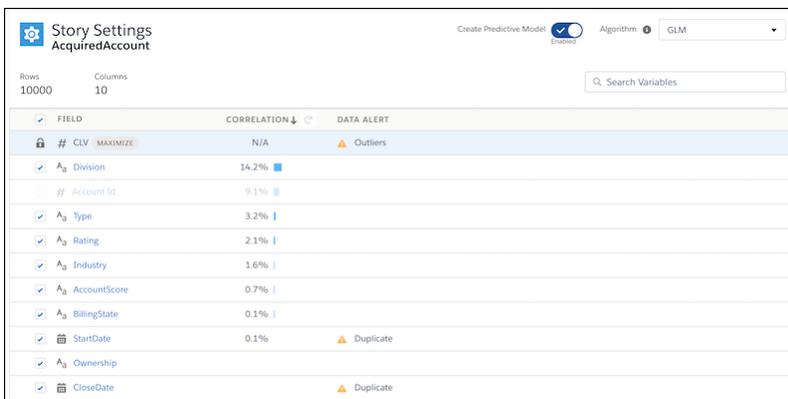
> **Note:** You could see a banner indicating there have been changes to the underlying dataset since the story was created. Click `Review story settings using the latest data`, review the story settings and click `Update Story` to create a new story version based on the latest available data.

3. Change field selections, if you want.

   The **Correlation** column shows how much each field contributed to the outcome, in descending order of impact. If a field has little or no correlation to the outcome, consider removing it from the story to improve the model.

   > **Note:**
   > - The list of fields includes numeric fields and text fields with values represented at least 25 times.
   > - Multi-value fields, which are fields that contain multiple values (such as a list or array), are not supported in Einstein Discovery. To learn more, see Einstein Discovery Limits on page 1605.

4. Review data alerts and make any story improvements you want.

The **Data Alert** column identifies fields that you improve, such as fields with outliers or duplicates, to get better insights for your data. For other story improvements, see Select Recommended Updates to a Story on page 1648.

5. To configure other settings for an individual field, click it and change its settings in the right panel.
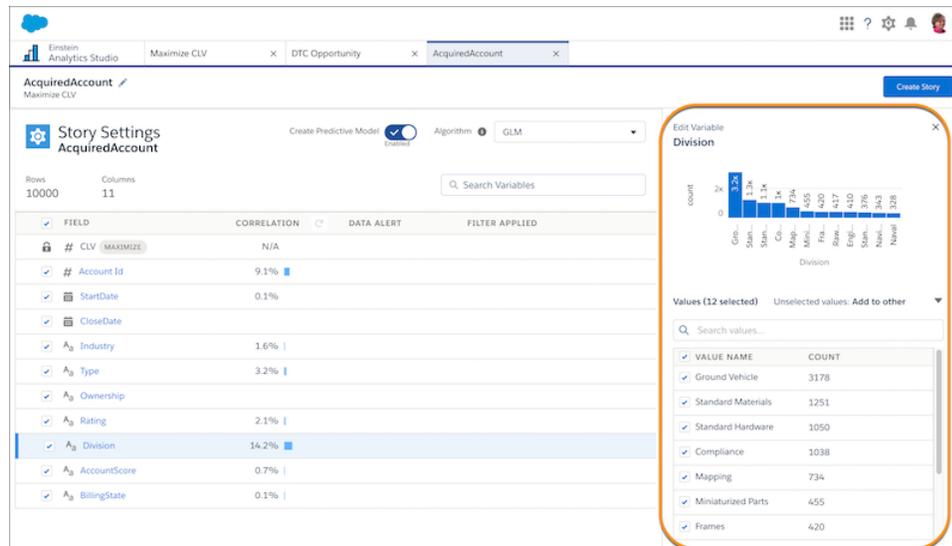
For instructions, see:

- Configure Filters and Settings for Number Fields on page 1642
- Configure Filters and Settings for Text Fields on page 1643
- Configure Filters and Settings for Date Fields on page 1645

📝 Note: The **Filters Applied** column shows any selected filters for a field.

To view the updated field correlation, click the recalculate icon ⟳ .

6. Enable **Create Predictive Model** if you want Einstein Discovery to make predictions and recommendations for this story. Disable it if you want only What Happened insights.

7. If **Create Predictive Model** is selected, you can choose among different algorithms for Einstein Discovery to use to create the model associated with this story. Select one from the **Algorithm** list.

Algorithm	Description
GLM	Default. Generalized Linear Model is a regression-based algorithm.
GBM	Gradient Boost Machine is a decision tree-based ensemble machine learning algorithm.
XGBoost	XGBoost is a decision tree-based ensemble machine learning algorithm.
Random Forest	Random Forest is a supervised learning algorithm that uses multiple decision trees, randomization, and other optimization techniques.

Alternatively, select **Model Tournament** to have Einstein Discovery run all four algorithms and then show the results of the algorithm that performed best.

8. Choose **Update Story**. Add your version notes in the Version Description (Optional) window, then click **Update** to save.
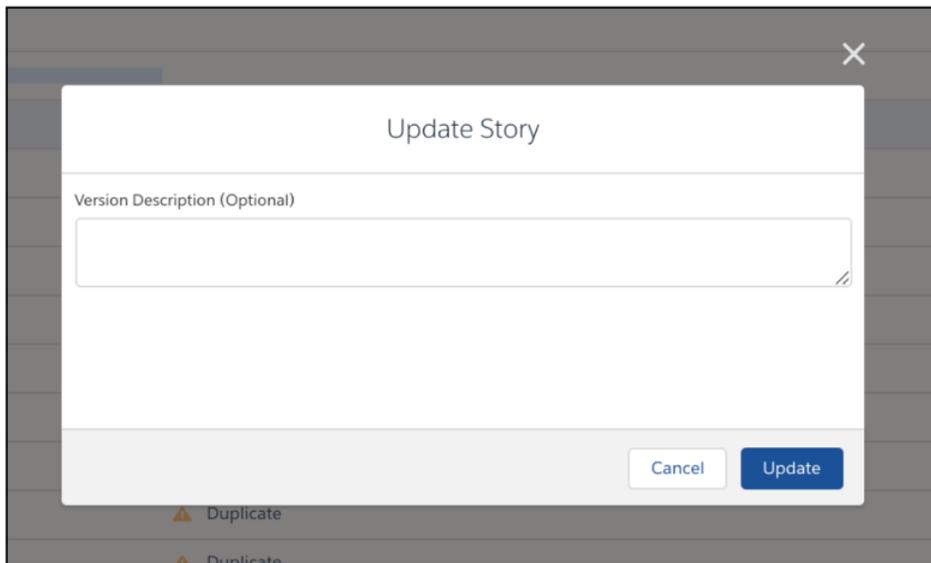
9. To abandon edits, including changes to the dataset, by choosing **Close**.
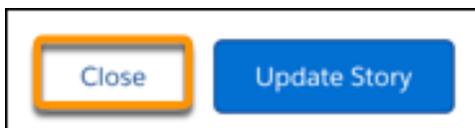
Based on your changed settings, Einstein Discovery begins preparing your story, analyzing your data, and displaying its progress each step of the way. When finished, it creates a story and opens it.

Configure Filters and Settings for Number Fields

Configure settings for individual number fields in your story.

Configure Filters and Settings for Text Fields

Configure settings for individual text fields in your story.

Configure Filters and Settings for Date Fields

Configure settings for individual date fields in your story.

Configure Filters and Settings for Number Fields

Configure settings for individual number fields in your story.

1. On the Story Settings page, click a number field. The Edit Variable panel appears.
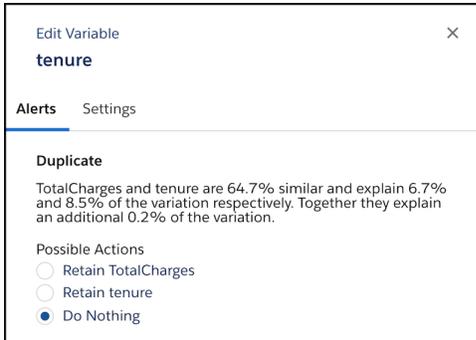
> Edit Variable   ✕
> **tenure**
>
> **Alerts**   Settings
>
> **Duplicate**
>
> TotalCharges and tenure are 64.7% similar and explain 6.7% and 8.5% of the variation respectively. Together they explain an additional 0.2% of the variation.
>
> Possible Actions
> ○ Retain TotalCharges
> ○ Retain tenure
> ● Do Nothing

2. In the **Alerts** tab, respond to any suggestions regarding data issues for this field, such as outliers or duplicates. For more information, see Select Recommended Updates to a Story on page 1648.

3. In the Settings tab, configure the following settings.

> Edit Variable   ✕
> **#**
>
> [Distribution graph showing count vs range with bars: 67 67 67 67 67 67 67 67 67 68]
>
> **Filter Range (Inclusive)**
>
> From   To
> [Min: 1]   [Max: 671]
>
> **Select Bucketing Method**
> ○ Bucket by count
> ○ Bucket by width
> ○ Manual bucketing
>
> Number of categories
> [10]
>
> ☐ This variable contains sensitive data

Setting	Description
Distribution Graph	A graph shows the number of values that occur across the range of number ranges.
Filter Range (Inclusive)	Create a numeric range by adding minimum and maximum values to the From and To fields.

Setting	Description
Select Bucketing Method	Select a method to bucket data: by count, by width, or by manually setting one or more value ranges. 📝 Note: For numeric variables with low cardinality (ten or fewer observations), Einstein Discovery displays numbers along the x-axis rather than ranges (buckets).
Number of Categories	Specify the number of bins to show in charts. Einstein Discovery automatically aggregates numbers into groups based on behavior.
This variable contains sensitive data	Select this option to exclude a variable from the model so that it does not influence predictions and recommendations. If selected, Einstein Discovery shows a shield icon next to the title of the insight to remind you it's a sensitive variable. 🛡 TENURE BY CONTRACT This enables you to evaluate and assess the field's impact in the story. Einstein Discovery still notifies you if it shows a 50% or higher correlation to the story's outcome variable.

Changes take effect after you create the story.

Configure Filters and Settings for Text Fields

Configure settings for individual text fields in your story.

1. On the Story Settings page, click a text field. The Edit Variable panel appears.

> **Edit Variable** ✕
> **InternetService**
>
> **Alerts**   Settings
>
> **Duplicate**
> InternetService and StreamingTV are 71.7% similar and explain 10.4% and 5.3% of the variation respectively. Together they explain no additional variation.
>
> Possible Actions
> ○ Retain InternetService
> ○ Retain StreamingTV
> ● Do Nothing

2. In the **Alerts** tab, respond to any suggestions regarding data issues for this field, such as outliers or duplicates. For more information, see

3. In the **Settings** tab, configure the following settings.

Setting	Description
Distribution Graph	A graph shows the number of values that occur across categories.
Categories	Select the categories you want to include in the story. Depending on the following options, excluded categories are either omitted from analysis or merged into the Other category.
Sort values in insights chart by	Specify the sort order for insights: by alphabetically or count.
This variable contains sensitive data	Select this option to exclude a variable from the model so that it does not influence predictions and recommendations. If selected, Einstein Discovery shows a shield icon next to the title of the insight to remind you it's a sensitive variable. This enables you to evaluate and assess the variable's impact in the story. Einstein Discovery still notifies you if it shows a 50% or higher correlation to the story's outcome variable.

Changes take effect after you create the story.

Configure Filters and Settings for Date Fields

Configure settings for individual date fields in your story.

1. On the Story Settings page, click a date field. The Settings panel appears.
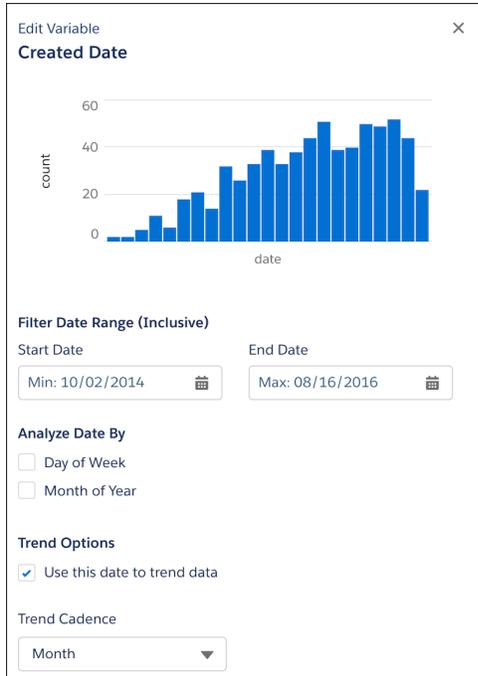
2. In the Settings panel, configure the following settings.

Setting	Description
Data alerts	If Einstein Discovery detected data issues in the selected field, such as fields with outliers or duplicates, respond to the suggestions. For more information, see Select Recommended Updates to a Story on page 1648.
Distribution Graph	A graph shows the number of values that occur across the range of dates.
Filter Date Range (Inclusive)	Optionally, restrict the date range to include in the analysis by changing the start and end dates.
Analyze Date By	Optionally, select how to analyze dates by: • **Day of Week** • **Month of Year** Specify the time period to analyze: by Day, Month, or Year. For trending (line) charts, use time periods to specify how to represent historical data.
Trend Options	Optionally, specify the following trend options: • Select **Use this date to trend data**. • Select a **Trend Cadence**: Day, Week, or Month

Changes take effect after you create the story.

Detect and Remove Bias from a Story

Einstein Discovery helps you practice ethical use of AI by detecting bias in your data so that you can remove its distorting effects on your analysis and predictions. Bias indicates that variables are being treated unequally in your model.

Examples of bias include:

- *Proxy variables*, where a variable is highly correlated to a sensitive variable, such as a loan applicant's street address and ethnicity.

- *Disparate impact* is a type of analysis we do to try to understand how different groups are being treated by the model.

Einstein Discovery lets you flag data that could potentially be associated with unfair treatment—such as race, gender, religion, national origin, sexual orientation, disability, age, and so on—as sensitive variables. Einstein Discovery then displays a shield icon next to stories and insights associated with sensitive variables as a reminder of possible bias while you investigate your data. Removing biased variables from your story can produce more ethical and accountable insights and models. For an overview, take the Responsible Creation of Artificial Intelligence Trailhead module.

To detect and remove bias in your story:

1. In Story Setup, select a variable that you suspect can indicate bias, click the **Settings** tab in the right panel, and select **This variable contains sensitive data**.

2. Click **Update Story** to create a new story version.

Einstein Discovery shows you a disparate impact analysis if there is a significant discrepancy in the way different classes are being treated by the model.
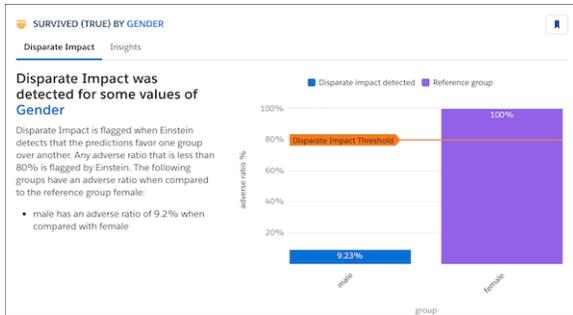
3. Filter the list so that it contains only insights associated with sensitive variables, click the shield button.

The shield icon indicates that this insight is associated with a variable that has been flagged as sensitive.

4. Click **Edit Story** to view story settings. Notice how Einstein displays the shield icon and the **Disparate Impact** data alert for the variable.

5. Investigate your story's insights to help you decide whether to include a sensitive variable (because it serves your analysis), or to exclude it and remove its biased influence from the story. To remove a variable, simply disable it and click **Update Story** to create a new story version without it.

Select Recommended Updates to a Story

Einstein Discovery uncovers issues in your data and suggests fixes that you can implement to improve your story's insights and predictions.

During data validation, Einstein Discovery searches for the following types of issues in your data. If found, Einstein Discovery presents you with the option to fix them in your story.

> **Note:** In the following table, statements that begin with *Methodology* provide supplemental information for data scientists and other advanced users.

Issue	Description
Duplicates	Indicates that two or more variables are highly correlated (for example, City and Postal Code). These variables have a *duplicate impact* on the outcome. Einstein Discovery recommends choosing just one variable to improve results. Consider keeping the most descriptive field (for example, City) to make insights more easily interpretable. This condition is also known as *multicollinearity*. *Methodology*: Einstein Discovery raises this data alert when the Cramér's V algorithm (used to test for multicollinearity) returns 0.5 or higher for two variables.
Outliers	Indicates the presence of uncommonly large or small numbers, potentially from data entry errors, or rare situations. Outliers can affect the accuracy of insights and predictions. Einstein Discovery recommends excluding outliers to improve results. *Methodology*: For a variable, Einstein Discovery calculates its global mean and global standard deviation in the dataset. The data alert identifies outliers as any value that is greater than, or less than, five standard deviations away from the global mean.
Strongest Predictors	Indicates a variable that explains the most variation in the data. Consider removing a variable if, for example: • there is an obvious mathematical relationship between this variable and the outcome (for example Cost and Price), or • if the variable is known only after the outcome is known (for example Reason for Churn in a customer churn analysis) A very high correlation (especially greater than 80%) can indicate possible *data leakage*. Leakage occurs when the data used to train your model includes one or more variables that contain the information that you are trying to predict.

Issue	Description
	If you suspect data leakage, exclude strongest predictor variables from your story. *Methodology*: Einstein Discovery raises this data alert for a variable when its R-squared value with the outcome variable exceeds 0.3.
Identical Values	Indicates that all values in a variable are identical (they belong to the same, single category), which adds no value to the analysis. Einstein Discovery recommends removing the variable from the story.
Dominant Values	Indicates that most values in a variable are in the same category, which limits the value of the analysis. *Methodology*: Einstein Discovery raises this data alert for a variable when one value occurs 90% of the time (the value has a relative frequency of 0.9 or higher).
Recommended Buckets	Einstein Discovery uses a clustering algorithm that analyzes the numerical variables in your story and can suggest different bucketing (grouping of data points based on ranges). Bucketing can help improve the interpretability of your insights.
High Cardinality	Einstein Discovery automatically alerts you when it detects variables containing more than 100 unique values. The options to address high cardinality variables include: • Let Einstein Discovery ignore unique values above 100 and group them into a reserve category. • Add one high cardinality variable, with a maximum of 200 unique values, to a story. Enable **Allow high cardinality for {field name}**. • Combine field values to minimize unique values. Other high cardinality variables are limited to 100 unique values. Enable **Combine remaining values into Other (Recommended).** • Limit the number of unique values by clicking the variable's **Settings** tab and deselecting the values to exclude.

1. Open the story.
2. In the Story toolbar, click the **Recommended Updates** button.

   📝 Note: The **Recommended Updates** button is clickable only if Einstein Discovery found opportunities to improve your story. Otherwise, the button is grayed out.

3. In each tab, accept any improvements you want to implement in your story.

> 📝 Note: Improvements made to a story affect the story settings and results. They do not change any underlying data in the associated Tableau CRM dataset.

4. To create a story with the selected improvements and current settings, click **Create New Story**. To configure other settings before story creation, click **Customize Story**.

When you create a story, Einstein Discovery includes your selected improvements in its data analysis, which can produce better results.

Track Story Versions

Each time you create a story, whether it's brand new or an update of an existing story, Einstein Discovery creates a new version and keeps previous versions so that you can track and manage your progress.

View Story Versions

Revisit and work with previous story versions.

Cancel a Story Version

Ever wish you could include a change before clicking Create Story? Ever wish you could cancel story creation instead of waiting for a story to finish just to delete it? You now have the option of interrupting a story after submitting it.

Compare Story Versions

Updated data or changed story settings can motivate the creation of a new story version. Compare story versions to better understand what's changed between them.

View Story Versions

Revisit and work with previous story versions.

Each time a user creates a story with updated settings, Einstein saves a snapshot of the previous version. You can open and work with previous versions of a story.

> 📝 **Note:** Einstein Discovery advises you if disparate impact occurs when recent version updates affect sensitive variables. Use the available story versions to help you evaluate your findings and decide if bias has occurred.

1. From the Story Toolbar, click the dropdown menu arrow, and then click **Version History**.
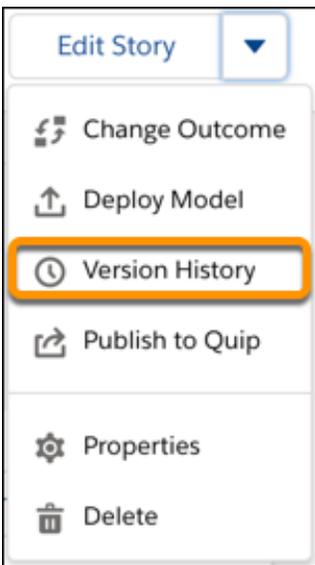
2. Add informative and detailed descriptions to the versions to help you track updates. In the **Version History** list, click the version's edit icon and add a description.
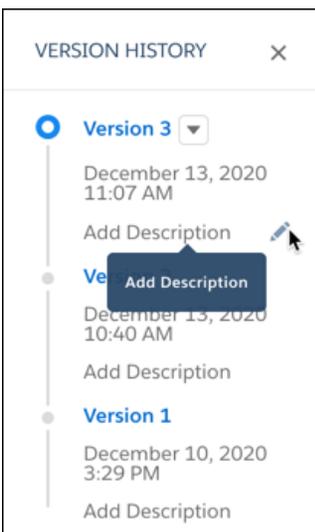
3. Click a version to open it.
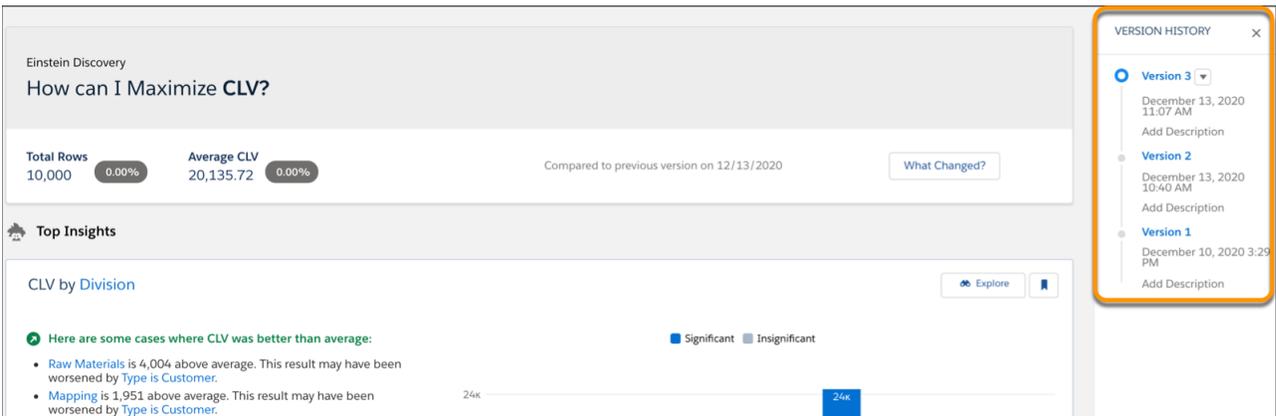
You can work with this story just as you would any other story.

Cancel a Story Version

Ever wish you could include a change before clicking Create Story? Ever wish you could cancel story creation instead of waiting for a story to finish just to delete it? You now have the option of interrupting a story after submitting it.

To cancel a new version:

1. Click the **Cancel Story** button after initiating a new version.
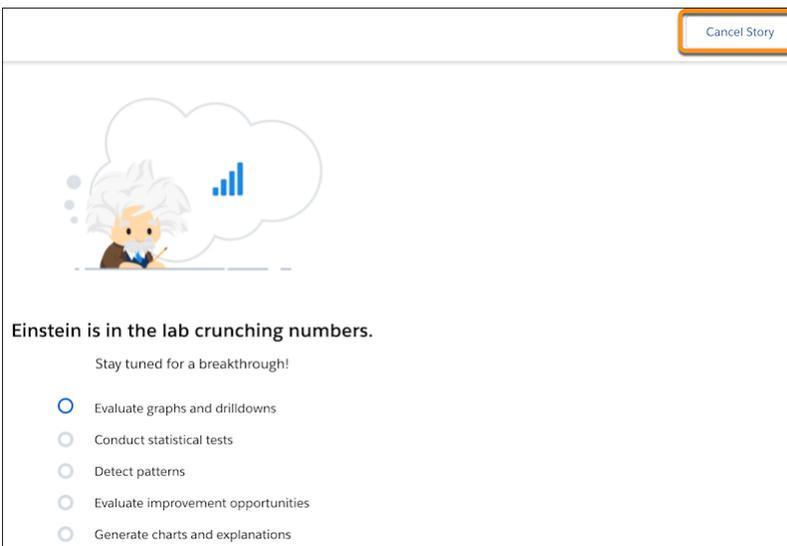
Canceling a revision returns you to the current story version.

Compare Story Versions

Updated data or changed story settings can motivate the creation of a new story version. Compare story versions to better understand what's changed between them.

Compare story versions to examine the effects of changes on your analysis. For example, suppose Einstein notifies you that the data in your dataset has changed, and you created a new story version. You can compare the latest version with a previous version to learn more about how changes in your data affected the outcome. Similarly, suppose you changed story settings—such as adding or removing a column, setting a filter, or making a recommended update. If you created a new story version, then you can compare it with a previous version to determine which settings produce better results.

1. Open a story with multiple versions. The story overview prompts the user with a **What Changed?** button, including a change summary with the net difference between versions.
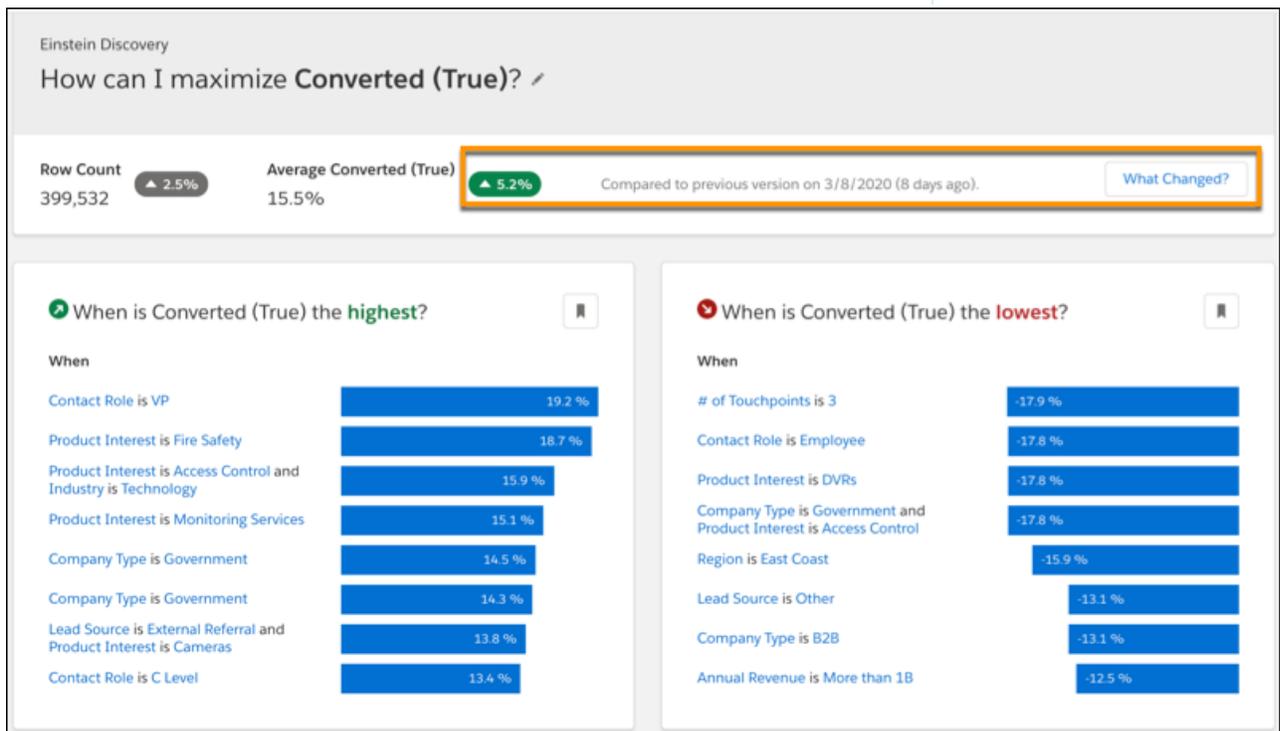
2. Click **What Changed?** to compare the current version to a previous one. Einstein automatically selects the immediately previous version and compares them side by side.
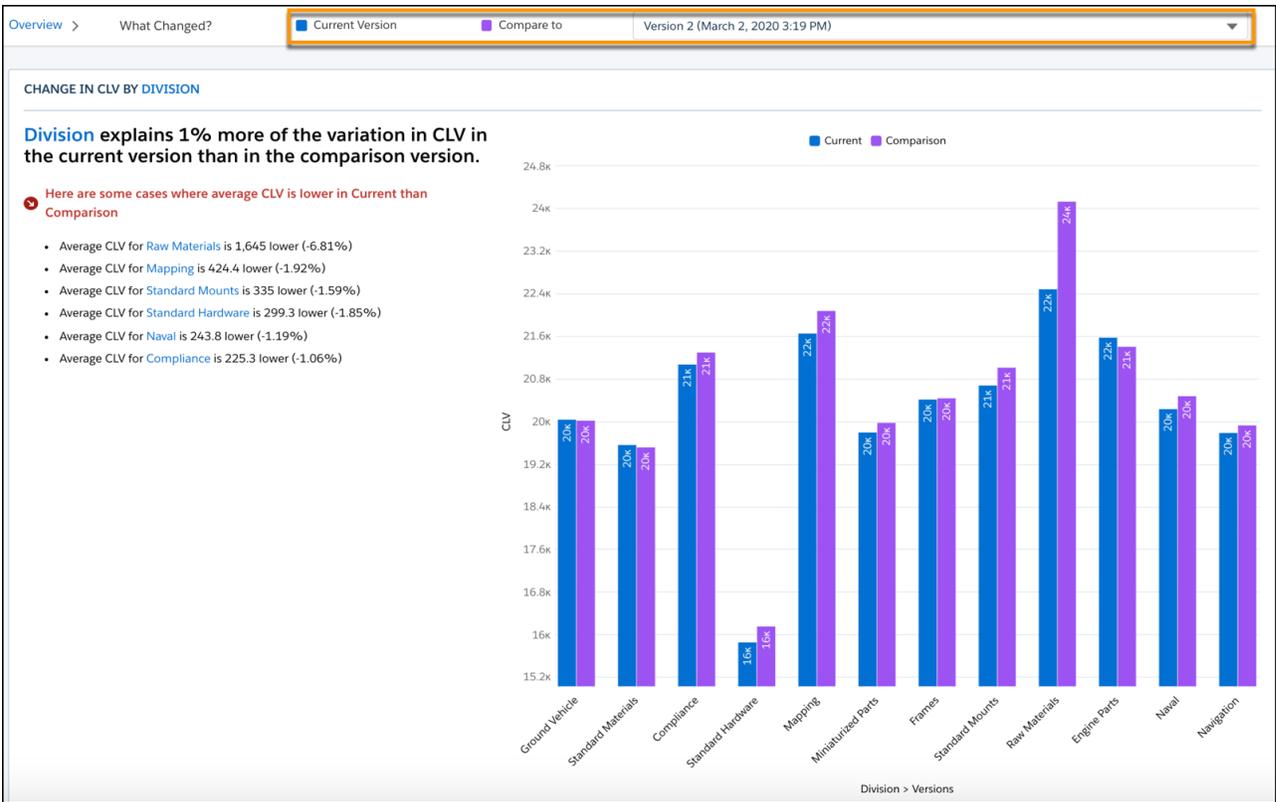
3. To compare with a different version, select it from the dropdown list.

Rename or Move a Story

Change a story's name or move it to a different app.

To save a story:

1. From the Edit Story Toolbar, click the dropdown menu arrow, and then click **Properties**.

2. For **App**, optionally select a different app (such as a shared app).

3. For **Story name**, optionally specify a different name.

4. Click **Save**.

Einstein Discovery saves your story selections.

Delete a Story

Delete a story you no longer need. Once you delete a story, it cannot be recovered.

To delete a story:

1. From the Edit Story menu, click **Delete**.

2. When Einstein Discovery prompts you to confirm deletion, choose **Delete**.

It can take up to five minutes for a deleted story to be removed permanently.

Explore Story Insights

You can explore insights for any story to which you have access. An insight is a statistically significant finding in your data. When you create a story version, Einstein Discovery analyzes the data in your dataset and generates insights based on its analysis. Insights provide a starting point for you to investigate the relationships among your story's explanatory variables and its goal.

Navigate Story Insights

Start here to learn how to navigate insights and become familiar with the user interface.

Explore Insights for a Variable

Explore how an explanatory variable is related to your story's goal.

Compare a Category or Bucket With the Global Average

Explore how a category or bucket relates to a variable's global average. Isolating a value helps you better understand how it relates to the story's goal.

Explore Why a Value Does Better or Worse Than Average

Learn why a category or bucket does better or worse than the global average.

Compare Categories or Buckets

Compare categories or buckets to learn how they relate to your story's goal. For example, you can compare the sales performance between two different territories (such as "North America" and "Europe"). Einstein Discovery displays a waterfall chart to help you visualize the comparison between territories. Comparative insights help you determine whether the patterns you focus on are real and not accidents of the data.

Explore Predictions and Improvements

Explore predicted outcomes and suggested ways to improve those predicted outcomes. Perform interactive, "what if" analyses and change feature selections to see prediction scores, top prediction factors, and top improvements to enhance prediction scores.

Bookmark an Insight in a Story

Bookmarking an insight allows you to export it to a Quip document. It also makes it easier to quickly find it in the Insights List when the Bookmark filter on the toolbar is disabled.

Get a Lens View of an Insight

Show the data in an insight as a lens.

Export and Share Insights

Export story insights to Quip so that you can share them with others.

Navigate Story Insights

Start here to learn how to navigate insights and become familiar with the user interface.

Insights Interface

Insights have similar interface components.

Interact with Insights

Interact with insights and drill into details to gain more understanding about your data.

Insights Interface

Insights have similar interface components.

Overview

Each insight consists of the following areas.
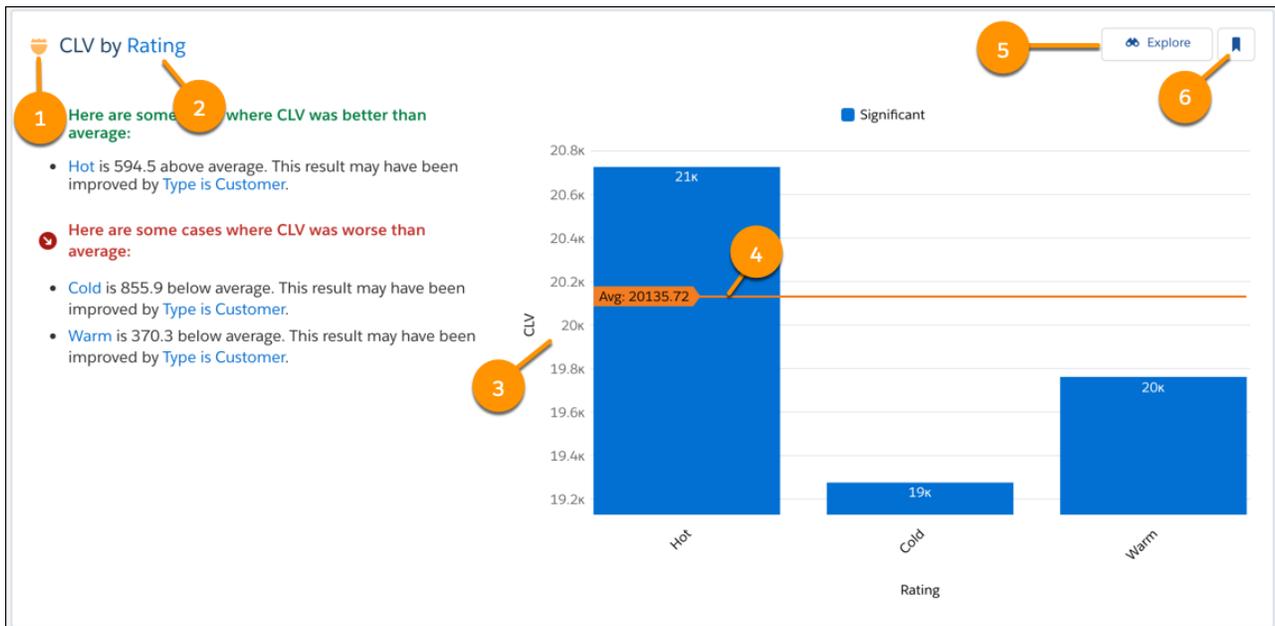
Area	Name	Description
1	Sensitive Field Badge	Shield indicating variable has sensitive data.
2	Insight Title	Selected variable's impact on story outcome.
3	Insight Chart	Insights based on the analysis of a dataset.
4	Version Update	Current version's outcome average.
5	Explore Button (Lens)	View into a dataset used in an exploratory mode.
6	Bookmark Insight	Save the insight for future reference.

If multiple insights are displayed, the insights are typically shown in the following order.

- Single variable that best explains the variability in the outcome (best single predictor).

- Variable that with the previous variable most improves the explanatory power (best two-variable predictor).
- If appropriate, another variable that with the first variable improves the explanatory power.
- Next single variable that best explains the variability on its own (second-best single predictor).

Variables Panel

Variables are conveniently ordered by correlation. This section consists of the following areas.
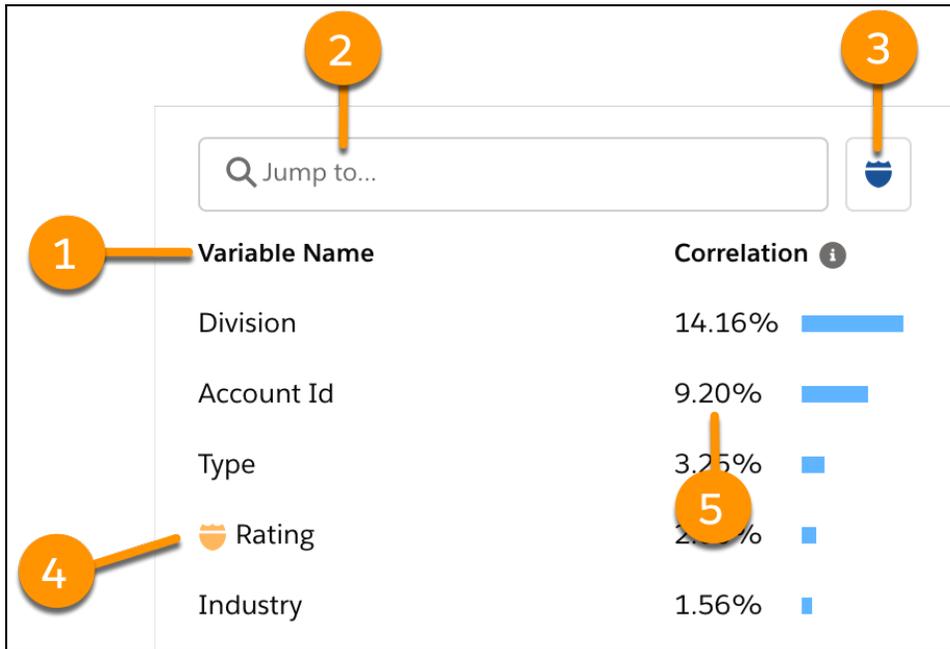
Area	Name	Description
1	Variable List	Listing of fields you are analyzing.
2	Search Box	Input field for searching variables or values.
3	Sensitive Fields Filter	Button that returns variables identified as sensitive.
4	Sensitive Fields Badge	Variables identified as carrying sensitive data.
5	Correlation to Outcome	Measurement of a field's correlation to the outcome variable.

Filter Selector

Surface insights by variable or value. This section consists of the following areas.
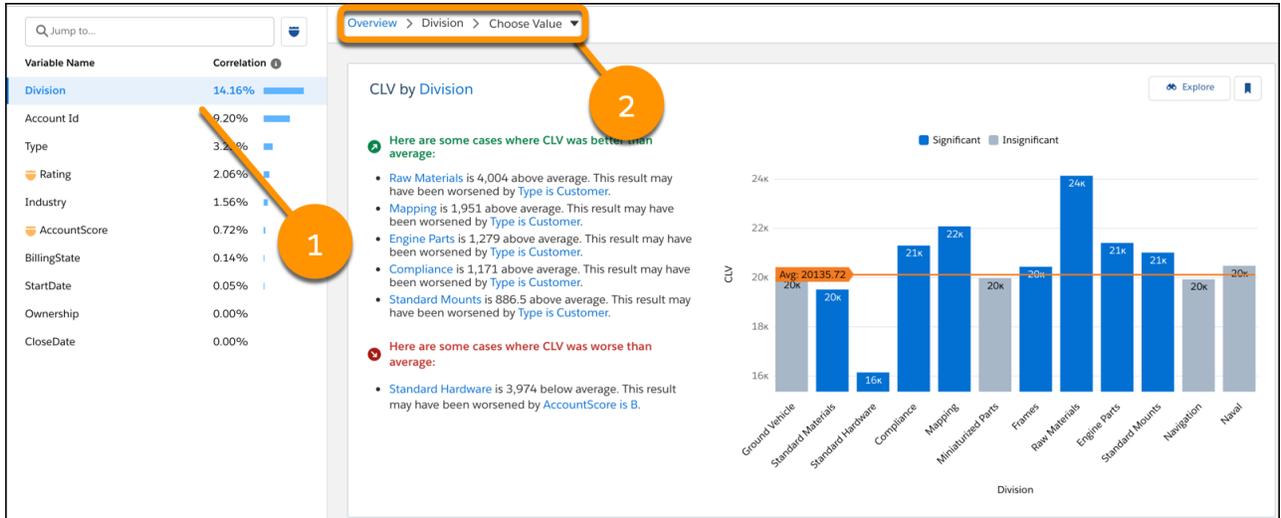
Area	Name	Description
1	Selected Variable	Chosen story variable.
2	Filter Selector	Dropdown menu to select a variable value.

Interact with Insights

Interact with insights and drill into details to gain more understanding about your data.

Explore Details from the Chart

To learn more about an insight, hover over a bar in the graph. A popup shows supporting statistics for the bar.
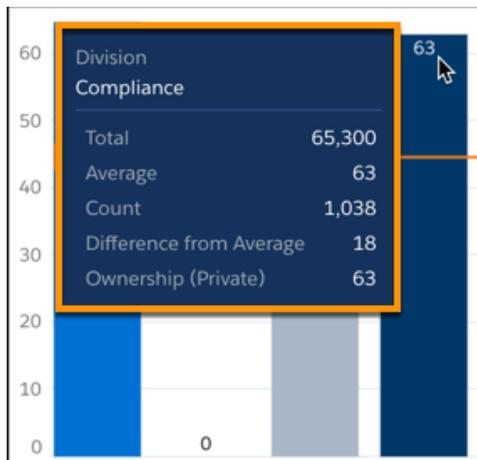
Explore Details from the Insight Description

To learn more about an insight, click the applicable hyperlink in the insight description.
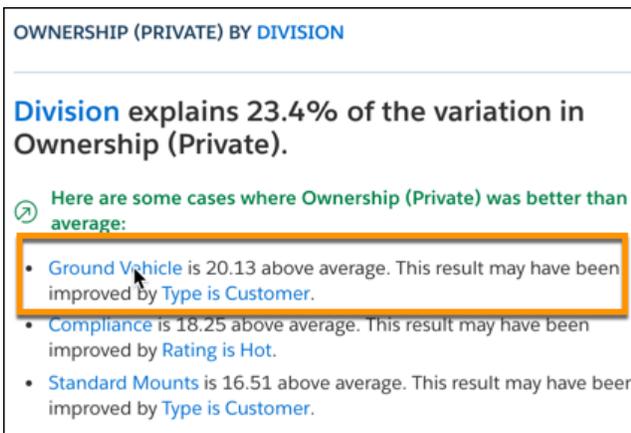
Einstein redraws the chart based on your selection.

Explore Insights for a Variable

Explore how an explanatory variable is related to your story's goal.

When you create a new story version or open a story, Einstein shows you a list of insights, starting with the most statistically significant ones. These are the primary insights in your story. They are descriptive insights that help you explore, at an overview level, what factors contribute to the outcome. These descriptive insights are similar to what you see in business intelligence (BI), visualization, and reporting products.

This initial, unfiltered list contains insights across all your data. To focus your exploration and delve deeper into insights, select any variable from the Variables panel. It's common to begin with variables that show higher correlations to the outcome.

📝 **Note:** Keep in mind that correlation is not causation. Correlation merely describes the strength of association between variables, not whether they causally affect each other. It's up to you to determine whether a correlation indicates anything more than just a statistical link.

1. Open a Story on page 1634.

2. In the Variables panel, select a variable you want to investigate.
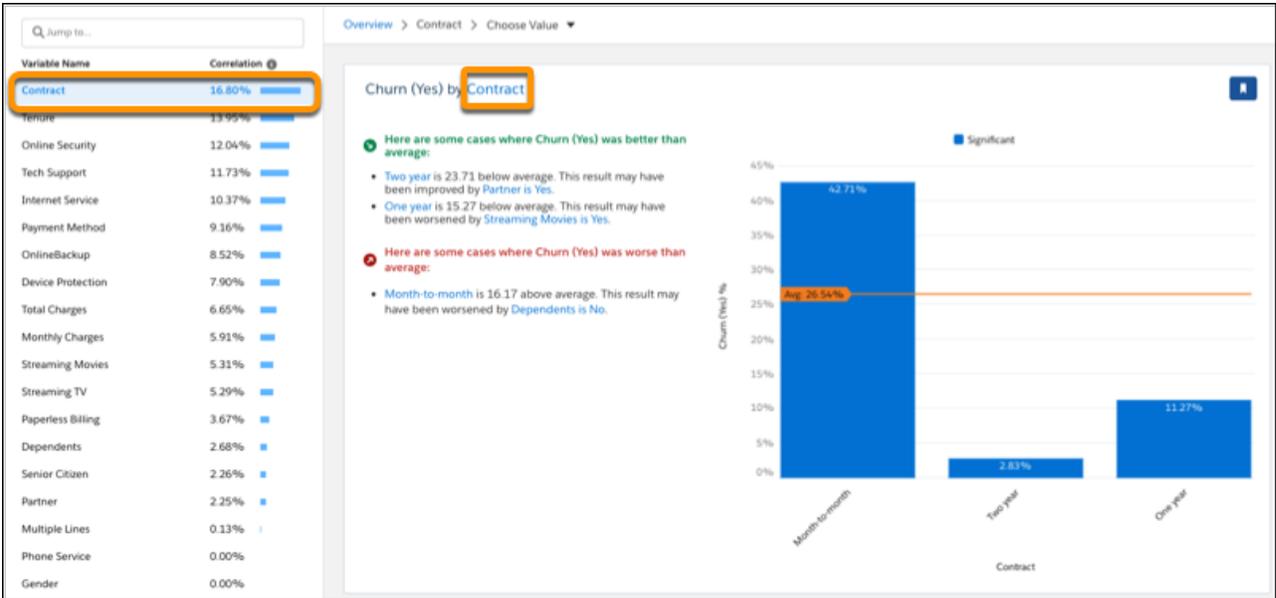
- The insights list shows insights that are associated with the selected variable only.
- The filter selector shows the variable you're investigating.

- The first insight in the list shows how the selected variable is associated with the outcome. It represents a summary of all values associated with the variable.
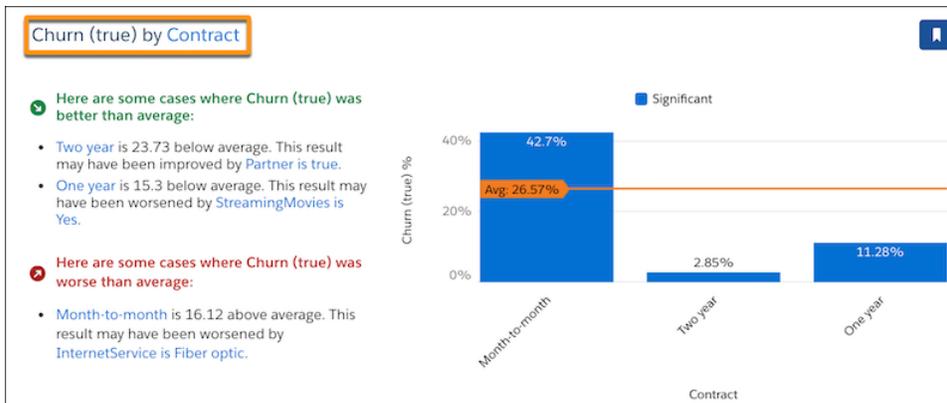
- Subsequent insights in the list show how combinations of values and conditions are associated with the outcome.
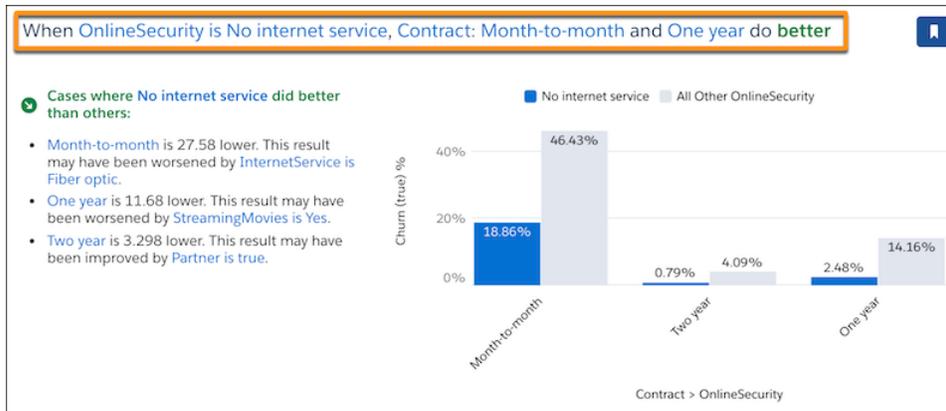
3. Scroll through the list of descriptive insights and examine any that interest you.

Explore a First-Order Insight for a Numeric Variable

When you select a numeric explanatory variable, Einstein displays a bar chart with the numeric variable along the horizontal axis.

Explore a First-Order Insight for a Text Variable

When you select a text (categorical) variable, Einstein displays a bar chart with the text variable along the horizontal axis.

Explore a First-Order Insight for a Date Variable

When you select a date variable, Einstein displays a chart with the date variable along the horizontal axis.

Explore a Second-Order Insight

A second-order insight shows how the combination of two explanatory variables explains variation in the outcome variable. In second-order analysis, the combined impact of both variables together on the outcome is sometimes called the interaction effect.

Explore a First-Order Insight for a Numeric Variable

When you select a numeric explanatory variable, Einstein displays a bar chart with the numeric variable along the horizontal axis.

The following example shows a descriptive insight with a numeric explanatory variable.

To help orient you to what this chart reveals, refer to the headline on the left. In the chart, generally look for blue bars, starting from left to right.

Chart Axes

Axis	Description
Horizontal	Explanatory variable (number) for this insight. Each bar represents a value (or bucket) of the explanatory variable. Values are sorted numerically from left to right. ✏️ **Note:** For numeric variables with low cardinality (ten or fewer observations), Einstein Discovery displays numbers along the x-axis instead of ranges (buckets).
Vertical	Outcome variable for this story.

Chart Elements

Number	Element	Description
1	Global Average	The orange line represents the global average of all variables.
2	Bucket of the Explanatory Variable	Each vertical bar represents a value or bucket (range) of the *Explanatory Variable*. • Blue signifies a bucket that is statistically significant. • Gray signifies a bucket that is statistically insignificant. • The height of the bar shows the average *Outcome Variable* for this bucket relative to the **Global Average**.

Number	Element	Description
		Hover-over details:
		• **_Explanatory Variable_** and **_Bucket_**
		• **Total**: Sum of all _Outcome Variable_s for this bucket.
		• **Count**: Number of observations in this bucket.
		• **Difference from Average**: How much this bucket's average differs from the global average (**Average** - **Global Average**).
		• **_Outcome Variable_**: Average _Outcome Variable_ for this bucket (**Total** / **Count**).
		• **Global Average**: Global average of all variables.

Explore a First-Order Insight for a Text Variable

When you select a text (categorical) variable, Einstein displays a bar chart with the text variable along the horizontal axis.

The following example shows a descriptive insight with a categorical explanatory variable.

To help orient you to what this chart reveals, refer to the headline on the left. In the chart, generally look for blue bars, starting from left to right.

Chart Axes

Axis	Description
Horizontal	Explanatory variable (categories) for this insight. Each bar represents a category of the explanatory variable. Categories are sorted by descending frequency from left to right (unless otherwise selected in Story Setup).

Axis	Description
	Categories containing the highest frequency of observations are shown to the left, and the categories with the lowest frequency of observations are shown to the right.
Vertical	Outcome variable for this story.

Chart Elements

Number	Element	Description
1	Global Average	The orange line represents the global average of all variables.
2	Category of the Explanatory Variable	Each vertical bar represents a category of the *Explanatory Variable*. • Blue signifies a category that is statistically significant. • Gray signifies a category that is statistically insignificant. • The height of the bar shows the average *Outcome Variable* for this category relative to the **Global Average** line. In this example, **Division=Standard Hardware** has an average CLV that is the most below the global average. Hover-over details: • ***Explanatory Variable*** and ***Category*** • **Total**: Sum of all *Outcome Variable*s for this category. • **Count**: Number of observations in this category. • **Difference from Average**: How much this category average differs from the global average (**Average** - **Global Average**). • ***Outcome Variable***: Average *Outcome Variable* for this category (**Total** / **Count**). • **Global Average**: Global average of all variables.
3	Category with the Highest Average	Category with the highest average.

Explore a First-Order Insight for a Date Variable

When you select a date variable, Einstein displays a chart with the date variable along the horizontal axis.

The following example shows a descriptive insight with a date explanatory variable.
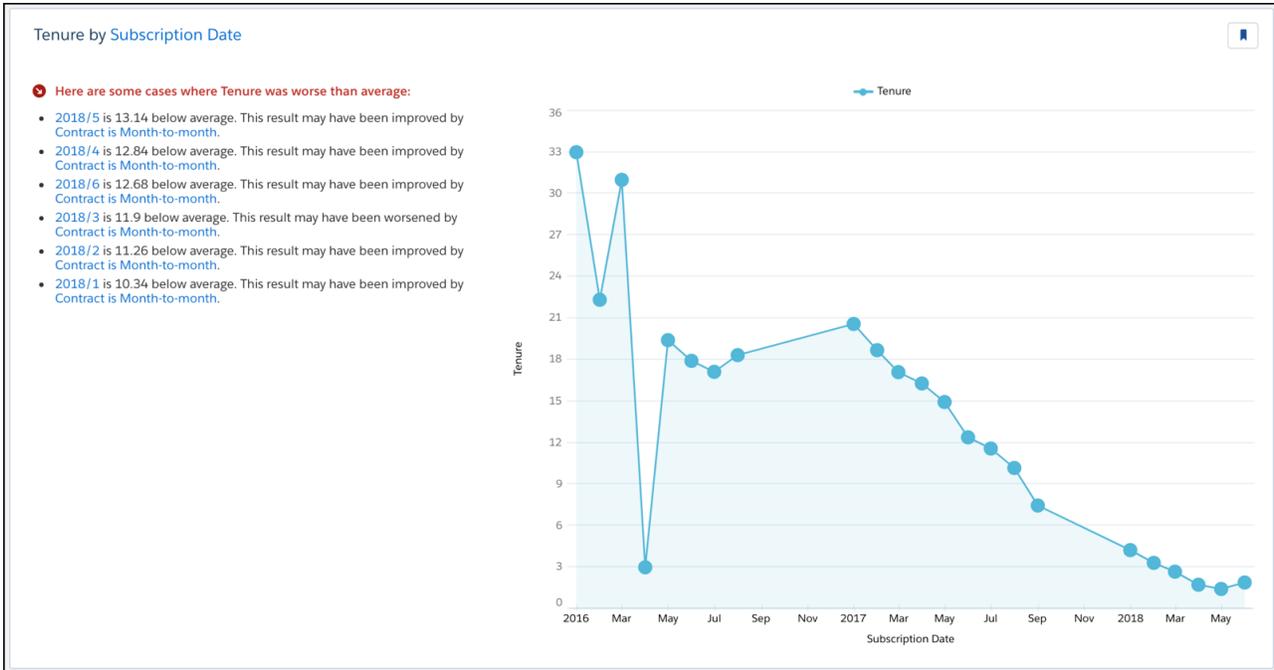
To help orient you to what this chart reveals, refer to the headline on the left. In the chart, generally look for blue bars, starting from left to right.

Chart Axes

Axis	Description
Horizontal	*Explanatory Variable* (date) for this insight. Sorted chronologically from left to right.
Vertical	*Outcome Variable* for this story.

Hover-Over Details

To see statistical details, hover over a bar in the chart.

📝 Note: The word *Condition* refers to the what's described in the insight heading.

Field	Description
Total	Sum of *Outcome Variable* for all observations that meet the *Condition*.
Count	Number of observations that meet the *Condition*.
Different from Average	The average for *Condition* minus the global average.
Outcome Variable	Average of *Outcome Variable* for all observations that meet the *Condition* (**Sum** / **Count**).
Global Average	Global average of all variables.

Explore a Second-Order Insight

A second-order insight shows how the combination of two explanatory variables explains variation in the outcome variable. In second-order analysis, the combined impact of both variables together on the outcome is sometimes called the interaction effect.

The following example shows a second-order insight.

To help orient you to what this chart reveals, refer to the headline on the left. In the chart, generally look for blue bars, starting from left to right.

Chart Axes

Axis	Description
Horizontal	Explanatory variables for this insight. Each pair of vertical bars represents a category of one explanatory variable. Within each pair, the vertical bar on the left represents a single category of the other explanatory variable . The vertical on the right represents all other categories of the other explanatory variable.
	Categories are sorted by descending frequency from left to right (unless otherwise selected in Story Setup). Categories containing the highest frequency of observations are shown to the left, and the categories with the lowest frequency of observations are shown to the right.
Vertical	Outcome variable for this story.

Chart Elements

Number	Element	Description
1	Legend	Describes the meaning of the vertical bars in the pairs. • **Category** - The left bar is blue when it is statistically significant or dark gray when it is statistically insignificant. • **All Other Categories** - The right bar is light gray. To determine whether the left bar is significant, Einstein performs the following tests: • For the numeric use case (numeric outcome variables), Einstein performs a two-sample unpaired Student's t-test (no assumption of equal variance) for each bar, testing it against its complement (such as **Type is Consulting** and **Type is NOT Consulting**). • For the classification use case (binary outcomes), Einstein performs a Chi-Square Goodness of Fit test to compare the observed number of the desired outcome to the expected number based on the complement. If the p-value of the test is below 0.01, the left bar is colored blue. Otherwise, it is colored a dark gray. Blue bars, then, are statistically significantly different from their complement (that their average would be so far from their complement is unlikely to have happened randomly).
2	Paired Vertical Bar	Each pair represents a category of the explanatory variable that appears along the horizontal axis. With each pair: • The left bar represents one category of another *Explanatory Variable*. • The right bar represents all other categories of this *Explanatory Variable*. • The height of each vertical bar shows the average *Outcome Variable* for this category or categories. Hover over a bar to see additional information.
	Left Bar Popup	• ***First Explanatory Variable*** and ***Category*** • ***Second Explanatory Variable*** and ***Category*** • **Difference in Average for Other Buckets**: The differences between the average of records that are left bar and the right bar (left bar **Average** - right bar **Average**). • **Total**: Sum of all *Outcome Variable*s for this category. • **Count**: Number of observations of this category. • **Difference from Average**: How much this category average differs from the global average (**Average** - **Global Average**). • ***Outcome Variable***: Average *Outcome Variable* for this category (**Total** / **Count**). • **Global Average**: Global average of all variables.
	Right Bar Popup	• ***First Explanatory Variable*** and ***Category*** • ***Second Explanatory Variable*** and ***All Other Categories*** • **Total**: Sum of all *Outcome Variable*s for this combination of categories. • **Count**: Number of observations within this combination of categories. • **Difference from Average**: How much this combination of categories average differs from the global average (**Average** - **Global Average**). • ***Outcome Variable***: Average *Outcome Variable* for this category (**Total** / **Count**). • **Global Average**: Global average of all variables.

Compare a Category or Bucket With the Global Average

Explore how a category or bucket relates to a variable's global average. Isolating a value helps you better understand how it relates to the story's goal.

To compare a value to

1. Open a Story on page 1634.

2. In the Variables panel, select a variable you want to investigate.

3. In the Filter Selector, choose the value you want to compare with the global average.
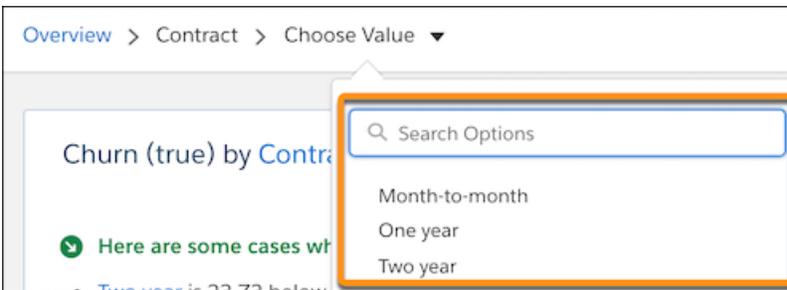
Einstein shows a summary of the comparison between the selected value and the global average.
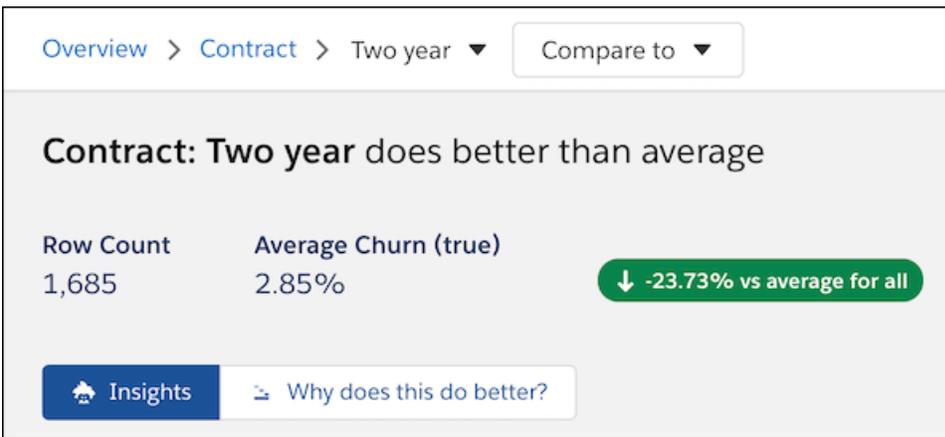
The insights list shows related insights you can explore.

4. To dive deeper, click **Why does this do better?** or **Why does this do worse?** as appropriate. To learn more, see Explore Why a Value Does Better or Worse Than Average on page 1668.

Explore Why a Value Does Better or Worse Than Average

Learn why a category or bucket does better or worse than the global average.

📝 Note: The term *why* refers to a *high statistical correlation*, not necessarily a causal relationship.

1. Open a Story on page 1634.

2. In the Variables panel, click a variable.

3. From the Filter Selector, choose a value.

4. Depending on your selections, click **Why does this do better?** or **Why does this do worse?**.

> 📝 Note: If this option is not available on the toolbar, check to see whether the story was created using the **Insights Only** option (see Select the Story Type on page 1626).
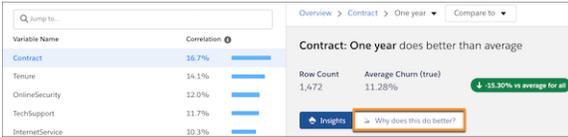
#	Area	Description
1	Title	Drivers of {outcome} when {selected value}.
2	Factor Explanations	Describe how each factor contributed to the predicted outcome.

#	Area	Description
3	Waterfall Chart	Visualization of the factors, how they contributed to the predicted outcome, and in comparison to the global outcome. • X-axis represents the contribution to the story's outcome. • Y-axis represents the drivers (explanatory variables) of the outcome variable.

The waterfall chart contains the following elements.

Number	Element	Description
1	Global Average	The gray bar represents the global average of all variables. Hover-over details: • **Drivers of *Outcome*: Global Average** • **Global Mean**: Mean *Outcome* for all variables in the dataset. • **Global Count**: Number of observations for all variables in the dataset. • **Contribution to Outcome**: Same as the **Global Mean**.
2	Category (First Impact)	Category for this insight. Represents the focus of this investigation. Hover-over details: • **Drivers of *Outcome*** when *Category* occurs. • **Impact**: The amount of change on *Outcome* that the model attributes to this *Category*. • **Coefficient**: Represents the impact this condition has on the outcome variable. The coefficient quantifies the change in the mean of the outcome variable when there is a one-unit shift in the explanatory variable, assuming all other variables in the model remain constant.

Number	Element	Description
		• **Precluded Sum**: Sum of *Outcome* for this *Category* without the effect of precluded terms (observations that are not this *Category*). • **Frequency**: How much of the entire dataset this *Category* represents. • **Conditional Frequency**: How much of this subset the *Category* represents. • **Precluded Count**: Number of terms that were precluded from this insight (terms that are not this *Category*). • **Contribution to Outcome**: Same as **Impact**.
3	Next 4 Impacts	Terms interacting with *Category*. Hover-over details: • **Drivers of *Outcome*** when *Condition* occurs. • **Impact**: The amount of change on *Outcome* that the model attributes to this *Condition*. • **Coefficient**: Represents the impact this condition has on the outcome variable. The coefficient quantifies the change in the mean of the outcome variable when there is a one-unit shift in the explanatory variable, assuming all other variables in the model remain constant. • **Frequency**: How much of the entire dataset this *Condition* represents. • **Conditional Frequency**: How much of this subset the *Condition* represents. • **Contribution to Outcome**: Same as **Impact**.
4	Small Terms Related To	The aggregated effect of all terms interacting with *Category* that do not appear in the other bars above. Hover-over details: • **Combined Impact**: Net impact on the *Outcome* of all the combined terms. • **Terms Combined**: Number of terms combined. • **Contribution to Outcome**: Same as **Combined Impact**.
	Other Impacts	Terms that are not specific to *Category* but that still occur more or less often. Hover-over details: • **Drivers of *Outcome*** when *Condition* occurs. • **Impact**: The amount of change on *Outcome* that the model attributes to this *Condition*. • **Coefficient**: Represents the impact this condition has on the outcome variable. The coefficient quantifies the change in the mean of the outcome variable when there is a one-unit shift in the explanatory variable, assuming all other variables in the model remain constant. • **Frequency**: How much of the entire dataset this *Condition* represents. • **Conditional Frequency**: How much of this subset the *Condition* represents. • **Contribution to Outcome**: Same as **Impact**.
5	Unrelated Small Contributors	Unrelated small contributors. To see additional information, hover over this bar. Hover-over details: • **Combined Impact**: Net impact on the *Outcome* of all the unrelated small contributors. • **Terms Combined**: Number of terms combined into the unrelated small contributors. • **Contribution to Outcome**: Same as **Combined Impact**.
6	Unexplained	Represents the unexplained variables on the *Outcome*. To see the contribution to outcome for unexplained variables, hover over this bar.

Number	Element	Description
7	Category Average	The blue bar represents the average for this category. Hover-over details: • **Drivers of *Outcome*: Average when *Category*** • **Global Mean**: Mean *Outcome* for *Category*. • **Global Count**: Number of observations for *Category*. • **Contribution to Outcome**: Same as **Global Mean**.

Compare Categories or Buckets

Compare categories or buckets to learn how they relate to your story's goal. For example, you can compare the sales performance between two different territories (such as "North America" and "Europe"). Einstein Discovery displays a waterfall chart to help you visualize the comparison between territories. Comparative insights help you determine whether the patterns you focus on are real and not accidents of the data.

To compare categories or buckets:

1. Open a Story on page 1634.

2. In the Variables panel, select a variable you want to investigate.

3. In the Filter Selector, choose the first value you want to compare.
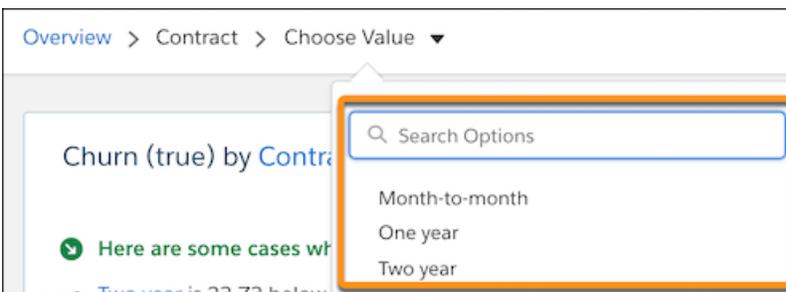
4. In the Filter Selector, choose the second value you want to compare.
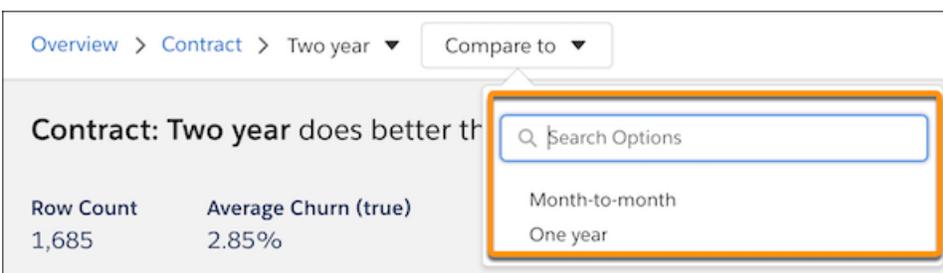
Einstein shows a summary and waterfall chart of the comparison between the selected values.
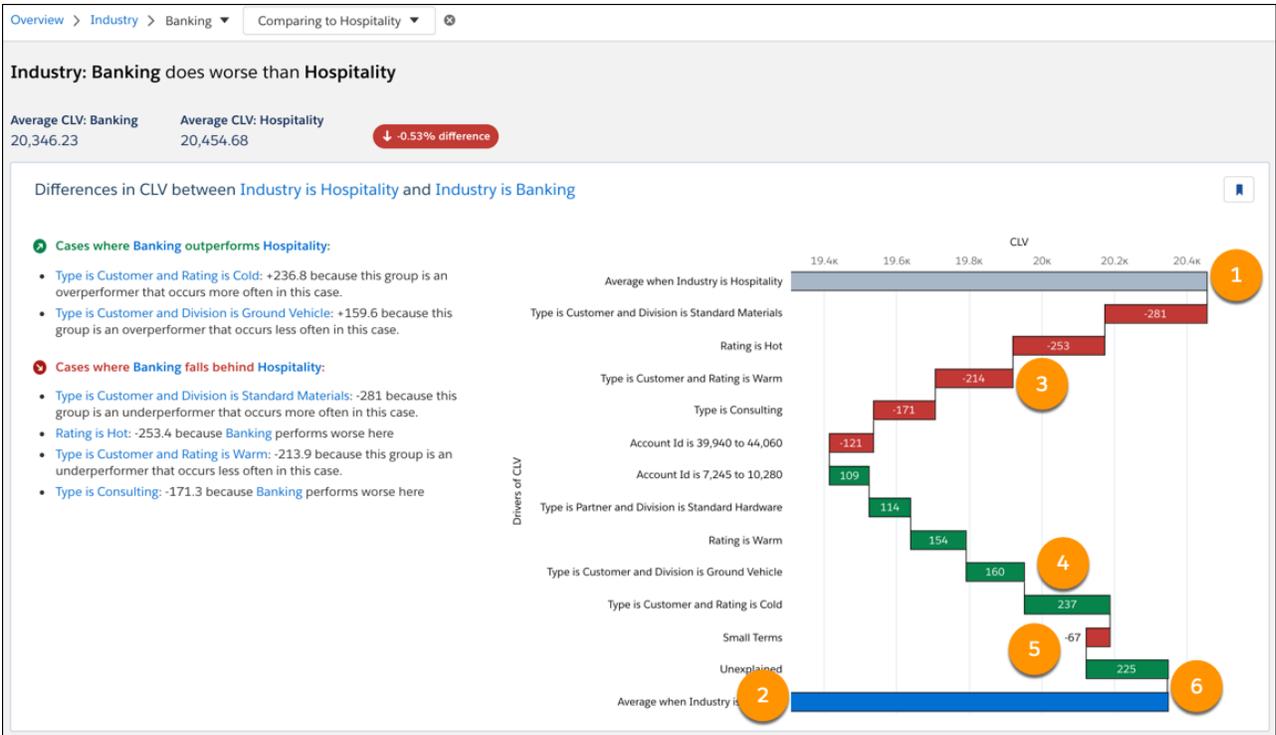
Chart Axes

Axis	Description
Horizontal	Outcome variable for this story.
Vertical	Drivers of the outcome variable.

Chart Elements

Number	Element	Description
1	First Explanatory Variable	The gray bar represents the first (left-most) explanatory variable selected in the story toolbar. Hover over this bar to see additional information. **Drivers of _Outcome_** when _First Explanatory Variable_ occurs **Global Mean**: Mean for the selected _First Explanatory Variable_ **Global Count**: Number of observations for the selected _First Explanatory Variable_ **_Outcome Variable_**: Average _Outcome Variable_ for this bucket (**Total** / **Global Count**).
2	Second Explanatory Variable	The blue bar represents the second explanatory variable selected in the story toolbar. Hover over this bar to see additional information. **Drivers of _Outcome_** when _Second Explanatory Variable_ occurs **Global Mean**: Mean for the selected _Second Explanatory Variable_ **Global Count**: Number of observations for the selected _Second Explanatory Variable_

Number	Element	Description
		• **Outcome Variable**: Average *Outcome Variable* for this bucket (**Total** / **Global Count**).
3 & 4	• (3) Cases where falls behind • (4) Cases where outperforms	• Green bars reflect cases in which explanatory variables have an favorable effect on the outcome. A favorable effect moves you closer to your story's goal, such as increasing opportunity wins when the goal is to maximize opportunity wins. • Red bars reflect cases in which explanatory variables have an unfavorable effect on the outcome. An unfavorable effect moves you away from your story's goal, such as increasing customer churns when the goal is to minimize customer churns. Hover over a bar to see additional information. Some bars display the following information: • **Drivers of Outcome** when *Explanatory Variable* and *Other Condition* occurs • **Left Base Impact**: How much this term impacts the first selected group. • **Left Coefficient**: Model coefficient for the interaction between this term and the first selected group. • **Left Frequency**: How often this term occurs in the first selected group. • **Right Coefficient**: Model coefficient for the interaction between this term and the second selected group. • **Left Interaction Impact**: Impact of the interaction between this variable and the first selected group, on the first group. • **Right Interaction Impact**: Impact of the interaction between this variable and the second selected group, on the second group. • **Base Coefficient**: The model coefficient for just this term. • **Difference**: The estimated difference in average between the first and second groups that can be attributed to the term and its interactions with the groups. • **Right Frequency**: How often this term occurs in the second selected group. • **Right Base Impact**: How much this term impacts the second selected group. • **Outcome**: Net effect on *Outcome*.
5	Small Terms	Represents the aggregate of terms that are too small to include in the other categories. Therefore, these terms are combined into a single set of statistics. Hover over this bar to see additional information. • **Combined Impact**: Net impact on the *Outcome* of all the combined terms. • **Terms Combined**: Number of terms combined. • **Outcome**: Net effect on *Outcome*.
6	Unexplained	Represents the unexplained variables on the *Outcome*. Hover over this bar to see the effect of unexplained variables on the outcome.

Explore Predictions and Improvements

Explore predicted outcomes and suggested ways to improve those predicted outcomes. Perform interactive, "what if" analyses and change feature selections to see prediction scores, top prediction factors, and top improvements to enhance prediction scores.

About Predictions and Improvements

In Einstein Discovery, a *prediction* is a probabilistic value of a future outcome, as calculated by a predictive model. Predictions are similar to the kind of regression or machine-learning analysis that data scientists conduct using advanced analytics tools. Knowing the most influential predictor variables can, for example, help a sales or marketing manager choose where to invest budget and which areas are less important. You can explore which variable values maximize or minimize your desired outcome.

An *improvement* is a suggested action, based on prescriptive analytics, that a user can take to improve the likelihood of a desired outcome. Improvements are associated with *actionable variables*, which are explanatory variables that people can control. For example, you can determine which marketing campaign or sales strategy to use for a given customer. Einstein Discovery recommends specific actions for changing predictors and seeing the impact on the prediction.

Explore What Could Happen

To simulate and explore future outcomes:

1. Open a Story on page 1634.

2. On the Story Toolbar, click **Predictions** (1).
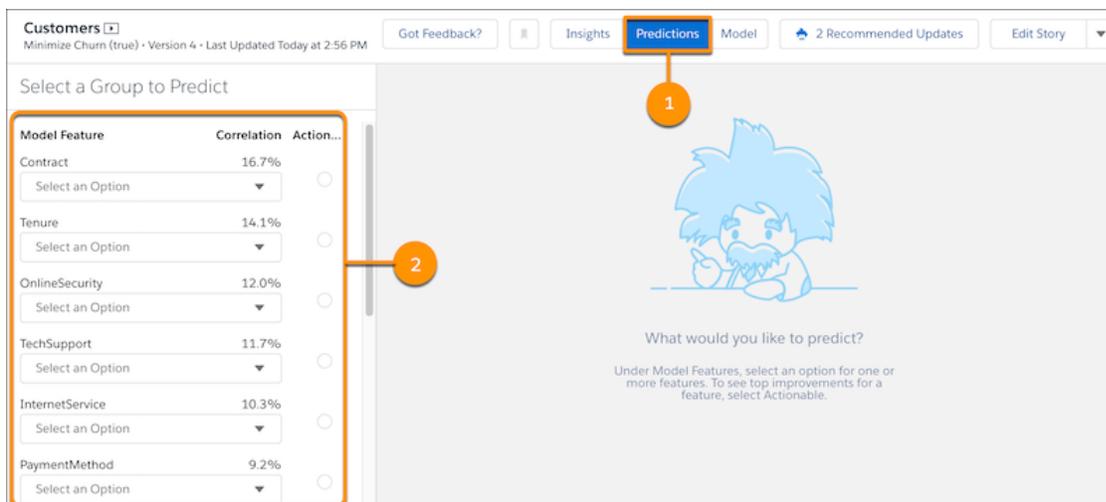
> Note: If **Insights** is not available on the toolbar, check to see whether the story was created using the **Insights Only** option (see Select the Story Type on page 1626).

3. Under **Select a Group to Predict** (2), select an option from one or more features.
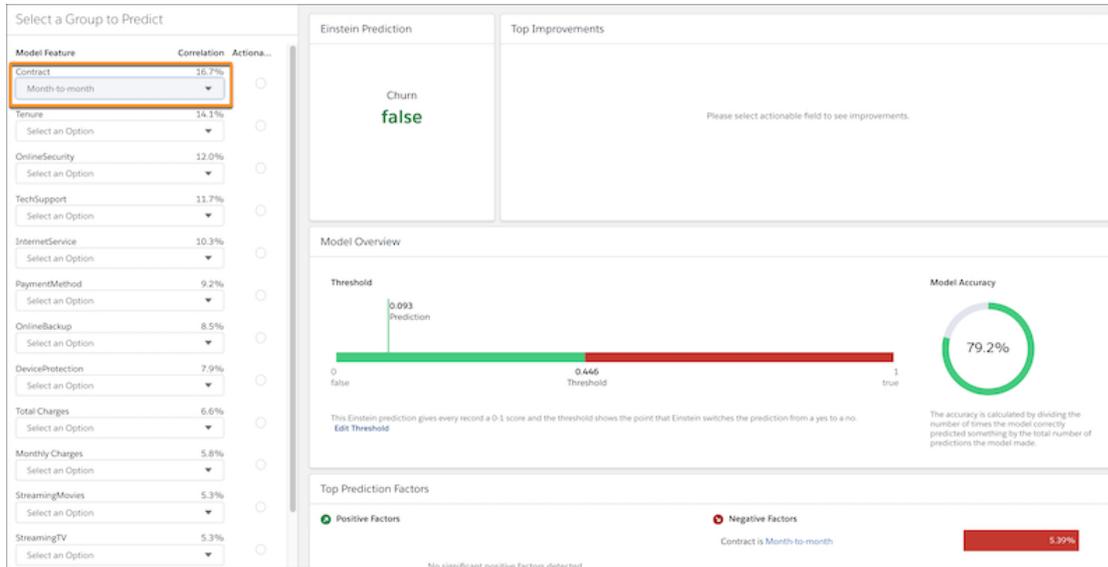
Field	Description
Einstein Prediction	Prediction score for your selections. Probabilistic calculation of the outcome based on the story's model.
Top Improvements	Suggested actions that you can take to improve the predicted outcome.
Model Overview	Metrics that describe the quality of the model. For details, see Explore Model Metrics on page 1682.
Top Prediction Factors	Explanatory variables, favorable and unfavorable, that are most strongly associated with the predicted outcome.
Insights	Any insights associated with your prediction. 📝 Note: A shield icon indicates that the variable is flagged as sensitive.

4. To see recommendations for a given feature, select **Actionable**.
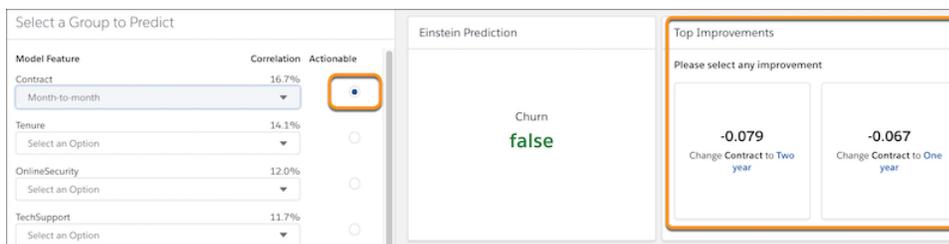
If there are ways to statistically improve the predicted outcome for the selected feature, Einstein displays them in **Top Improvements**.

Bookmark an Insight in a Story

Bookmarking an insight allows you to export it to a Quip document. It also makes it easier to quickly find it in the Insights List when the Bookmark filter on the toolbar is disabled.

If you click the Bookmark Filter (▚) icon on the Story toolbar to enable it, the Insights list displays only those insights that have been bookmarked.

- To bookmark an insight, click its Bookmark (▚) icon to enable it. The color changes to blue.
- To undo the bookmark for an insight, click its Bookmark (▚) icon again to disable it. The color changes to white.

Get a Lens View of an Insight

Show the data in an insight as a lens.

To drill down into details of an insight associated with a text field:

1. Click the **Explore** button.
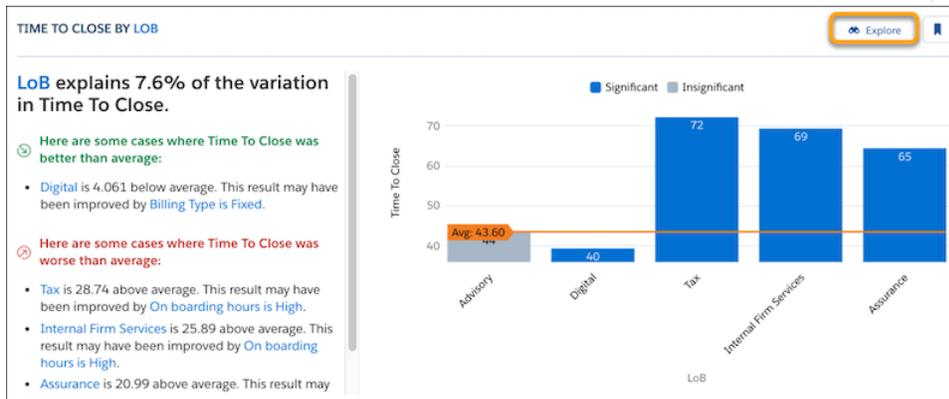
Einstein shows a Lens view of the insight.

2. Explore the lens view. To learn more, see View Your Data in a Lens on page 1062.

Export and Share Insights

Export story insights to Quip so that you can share them with others.

To export story insights to Quip:

1. In your story, bookmark the insights you want to export. See Bookmark an Insight in a Story on page 1677.

2. From the Story Toolbar, click the dropdown and then click **Publish to Quip**.

3. When prompted, click **Open Quip Story**.
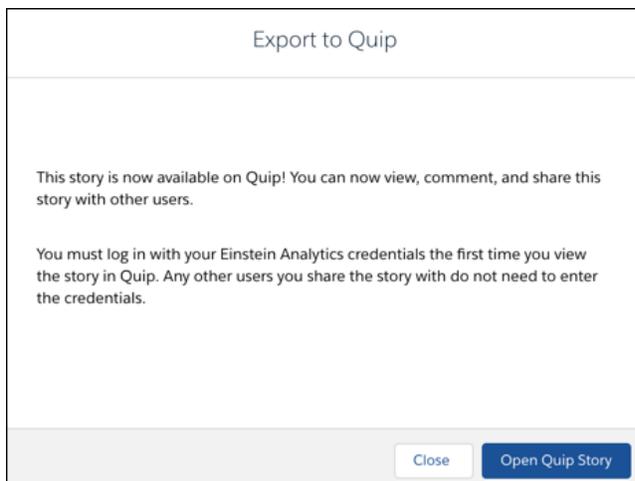
4. When prompted, type in a name for the Quip file in the **Name** field.

5. Click the **Folder** dropdown list to choose a location to save the Quip file.

6. Click the **Save in...** button.

EDITIONS

Available in Salesforce
Classic and Lightning
Experience.

Available with Tableau CRM,
which is available for an
extra cost in **Enterprise**,
Performance, and
Unlimited Editions. Also
available in **Developer
Edition**.

USER PERMISSIONS

To export a story:
- Share Einstein Discovery
  Stories

Einstein Discovery opens the Quip document containing the pinned insight(s) you exported.

Build, Deploy, and Manage Models

Models represent the predictive and prescriptive analytics portion of a story. Einstein Discovery uses models to predict outcomes and suggest ways in which to improve predicted outcomes. You can build, evaluate, compare, deploy, and manage these models.

The following articles describe how to work with models.

About Models

A *model* (also called a *predictive model*) is the sophisticated, custom algorithm that Einstein Discovery uses to predict a particular outcome. A model accepts inputs (one or more explanatory variables) and produces outputs (a predicted outcome, top factors, and improvements). When you create a story version with **Create Predictive Model** enabled, Einstein Discovery generates its associated model automatically.

Explore Model Metrics

Model metrics reveal quality measures and associated details for a model. Use model metrics to evaluate a model's ability to predict an outcome. When ready, you then deploy a model to Salesforce to predict outcomes in production.

Deploy Models

Deploy a model so that you can use it to make predictions and improvements.

Compare Models

Compare metrics for multiple models side by side to see how they stack up against each other. For example, comparing segments in your data can reveal the most important variables in each segment.

Manage Prediction Definitions and Models

Use the Model Manager to view, configure, and manage prediction definitions and models that have been deployed in your org.

About Models

A *model* (also called a *predictive model*) is the sophisticated, custom algorithm that Einstein Discovery uses to predict a particular outcome. A model accepts inputs (one or more explanatory variables) and produces outputs (a predicted outcome, top factors, and improvements). When you create a story version with **Create Predictive Model** enabled, Einstein Discovery generates its associated model automatically.

Models and Predictive Analytics

Based on data mining, machine learning, and predictive statistical modeling, *predictive analytics* is the practice of predicting future outcomes based on a comprehensive analysis of past outcomes. Einstein Discovery uses the model to generate diagnostic insights, predictions, and improvements.

Terminology

Refer to the following terminology when working with models.

Term	Definition
model	The sophisticated, custom algorithm that Einstein Discovery generates based on a comprehensive, statistical understanding of past outcomes. Einstein Discovery uses models to predict future outcomes. A model accepts the values of one or more predictor variables as input and produces a predicted outcome as output, along with (optionally) top factors and improvements. Einstein Discovery creates a model automatically whenever you create a new story version with **Create Predictive Model** enabled.

Term	Definition
predictor	An explanatory variable that a model accepts as input in order to calculate a prediction. Predictors are also known as predictor variables or independent variables.
prediction	A derived value, produced by a model, that represents a possible future outcome. You can think of a prediction as the output of a predictive model that is based on the inputs of predictor variables that the model accepts.
top predictor	A condition that most significantly drives the predicted outcome. A condition is a data value associated with a variable. In Einstein Discovery, a predictor consists of one or two conditions.
improvement	A suggested action that a user can take to improve the likelihood of a desired outcome. Improvements are associated with actionable variables, which are variables over which users can possibly control or influence, such as the shipping method or a subscriber's membership level. By taking the actions that Einstein suggests, users can increase their chances of having a more favorable outcome.
prediction definition	A container object in Einstein Discovery associated with one or more models. If a prediction definition contains multiple models, then each model produces predictions for a different segment of the data. A prediction definition can contain up to ten active models.
segmentation	Involves deploying models that target different segments (subsets) of your data. For example, suppose your data contains large, medium, and small customers, and your company organization is oriented around customer size to address the specialized needs of each group. You could build and deploy separate models for large, medium, and small customers to address the unique characteristics of each group. You define segments using filters that specify conditions for each group. Segmentation involves prediction definitions with multiple models.
prediction column	In a Tableau CRM dataset, the column where Einstein Discovery stores prediction values returned from the model.

Types of Models

Einstein Discovery uses two types of models. The model type depends on the field used for the outcome variable in your story.

Use Case	Description	Regression Type
Classification Use Case	Categorical (text) fields (classifications) contain only two qualitative values. Examples include variables that are either true or false, public or private, churned or not churned, and so on. These fields separate your data into two distinct groups. Predicting a categorical field is a *binary classification problem* with its own set of metrics to measure model quality.	logistic regression
Numeric Use Case	Numeric fields (measures) can contain many different types of values. Predicting a number field is a *regression problem* with its own set of metrics to measure model quality.	linear regression

Considerations When Working With Models in Einstein Discovery

- For a numeric variable, if Einstein Discovery finds low cardinality (ten or fewer unique values) during analysis, the data type for this variable in the generated model is text rather than numeric.

Explore Model Metrics

Model metrics reveal quality measures and associated details for a model. Use model metrics to evaluate a model's ability to predict an outcome. When ready, you then deploy a model to Salesforce to predict outcomes in production.

To view model metrics, on the Story toolbar, click **Model**. The metrics that are visible in the Model Metrics tabs depend on the use case (classification or numeric) for the outcome variable in your story.

Metrics for Classification Use Cases

The classification use case is based on text (categorical) variables with binary outcomes. The Model Metrics tabs show quality statistics associated with logistic regression models.

Metrics for Numeric Use Cases

The numeric use case is based on outcomes that are numeric variables. The Model Metrics tabs show quality statistics associated with linear regression models.

Implement Recommended Updates

If Einstein Discovery detects possible improvements in your data during validation, it displays a message under **Validation Results** in the Overview tab and prompts you to **Review Updates**.

Metrics for Classification Use Cases

The classification use case is based on text (categorical) variables with binary outcomes. The Model Metrics tabs show quality statistics associated with logistic regression models.

The following tabs display Model Metrics.

Overview Tab for Classification Use Cases

The Overview tab shows an at-a-glance summary of your model's validation results.

Model Evaluation Tab for Classification Use Cases

The Model Evaluation tab provides information about a model's performance, gains and lift, cross-validation results, and coefficent values.

Threshold Evaluation Tab for Classification Use Cases

The Threshold Evaluation tab helps you optimize the threshold value for a model. The threshold value tells your model how to classify a binary outcome. If the calculated probability is above the threshold value, Einstein classifies the outcome one way (such as True or Positive). If the calculated probability is below the threshold value, Einstein classifies the outcome the other way (such as False or Negative).

Prediction Examination Tab for Classification Use Cases

The Prediction Examination tab allows you to interact with the model metrics and see how they stack up against actual outcomes.

R Code Tab

The R Code tab contains the transformations made to your Tableau CRM dataset and scoring code. Inspect the code to examine the underlying details of the model that Einstein Discovery produced.

SEE ALSO:

Explore Model Metrics

Overview Tab for Classification Use Cases

The Overview tab shows an at-a-glance summary of your model's validation results.

Path to Deployment

Area	Description
Review Model Accuracy	AUC (Area Under the Curve) represents the rate of correct classification by a logistic model. An AUC of 0.5 means that the model performs no better than random guessing. An AUC of 1.0 means that the model correctly classifies data 100% of the time, which can indicate data leakage. Click **View Model Evaluation** to see the Model Evaluation Tab for Classification Use Cases on page 1684.
Set a Threshold	Shows the current threshold of the model. Click **View Threshold Evaluation** to go to the Threshold Evaluation Tab for Classification Use Cases on page 1692.
Pre-deploy Checks	Summarizes the results of validation tests on the story data: • Recommended updates to the story. Click **View** to Implement Recommended Updates on page 1705. • Cross-Validation Test • Accountability Alerts

Training Data and the Model

Area	Description
Distribution of Outcome Variable	Shows the distribution of values (count and range) for the outcome variable in the training data.

Area	Description
Top Predictors	Lists the top predictors (explanatory variables) for this model and their correlation with the outcome variable.

SEE ALSO:

> Metrics for Classification Use Cases

Model Evaluation Tab for Classification Use Cases

The Model Evaluation tab provides information about a model's performance, gains and lift, cross-validation results, and coeffecient values.
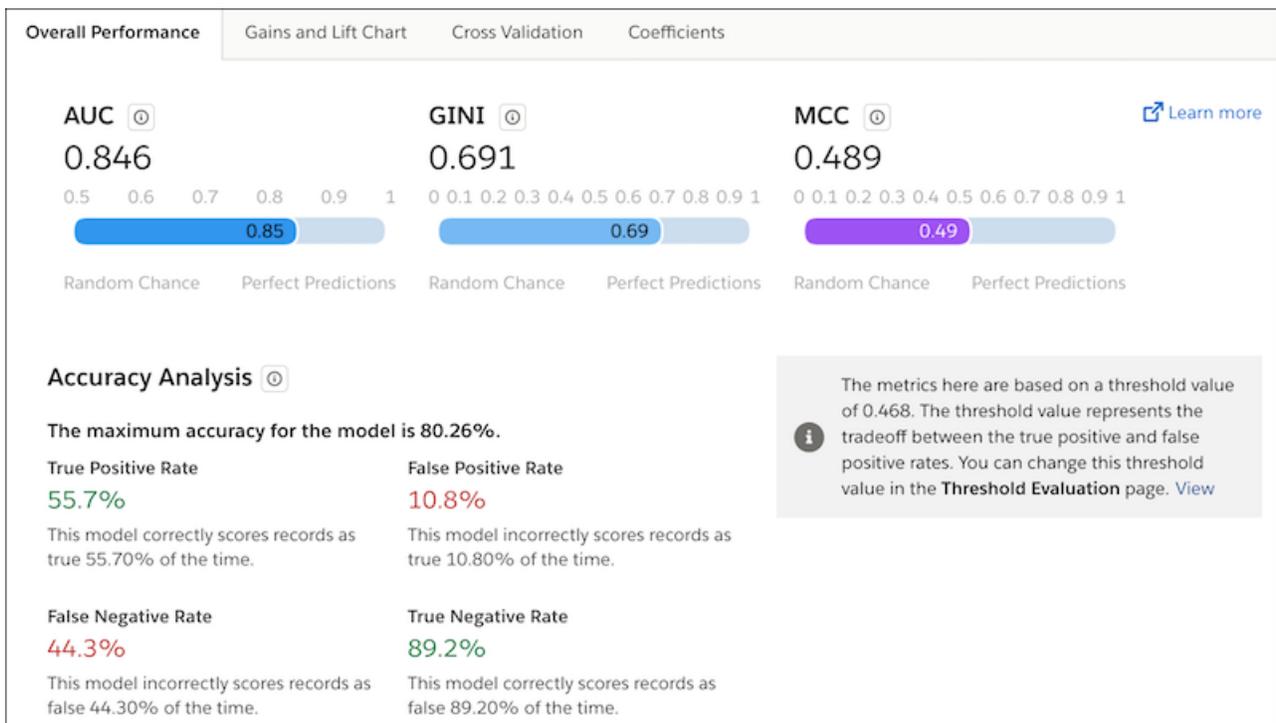
For model evaluation details, click a subtab.

Overall Performance Tab for Classification Use Cases

The Overall Performance tab shows key metrics for model quality.

Gain and Lift Charts for Classification Use Cases

This tab shows two charts - Cumulative Gains and Cumulative Capture - that help you evaluate your model's ability to predict outcomes.

Cross-Validation Tab for Classification Use Cases

To test a model's ability to make predictions, Einstein Discovery uses k-fold cross-validation, a process that reduces sampling bias when validating a model. This tab summarizes the results of the cross-validation process for this model, as well as some of the underlying computational details.

A model uses coefficients to calculate a prediction for a specific observation. You can filter the list of coefficients and also download the data in a CSV file.

SEE ALSO:

Metrics for Classification Use Cases

Overall Performance Tab for Classification Use Cases

The Overall Performance tab shows key metrics for model quality.

In the Overview Tab for Classification Use Cases on page 1682, click **Advanced Overview** to see summary metrics for the model.
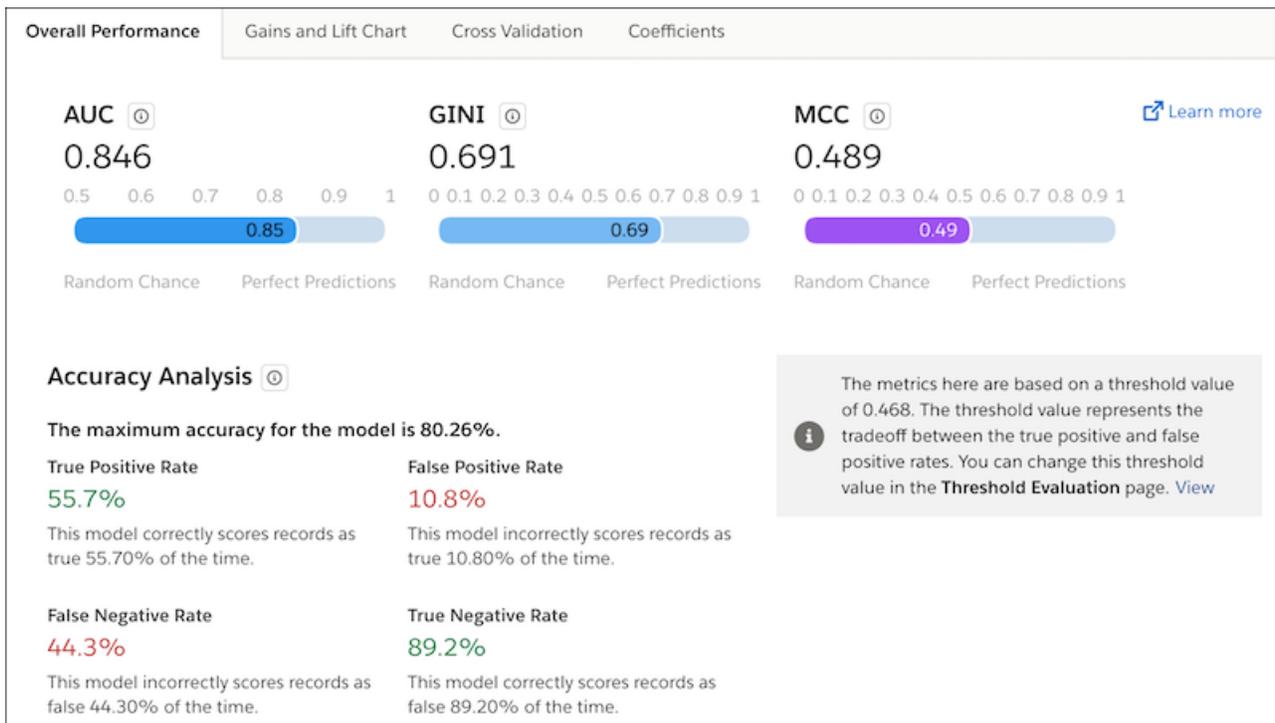
Summary Metrics

Metric	Description
AUC	Area Under the Curve. Measures the logistic model's rate of correct classification. AUC is frequently used to evaluate model quality in classification use cases. Range: • 0.5 means that the model performed no better than random guessing. • 1.0 means that the model correctly classified the data 100% of the time. An AUC of 1.0 is suspect because it can indicate data leakage: the data used to train your model includes one or more columns that contain the information that you are trying to predict.

Metric	Description
GINI	GINI Index. Measures how closely this classification model performs to a theoretically best possible model. Range: - 0 means that the model performed no better than random guessing. - 1 means that the predictions matched observations exactly (and should be viewed with skepticism).
MCC	Matthews Correlation Coefficient. Measures the quality of a classification model. Provides a more even representation of the four parts of the confusion matrix than other classification metrics. In contrast, accuracy and the F1 score can be misleading when one class is predicted much more accurately than another in a classification use case. Range: - -1 means that the model wrongly predicted the opposite class every time. - 0 means that the model performed no better than random guessing. - +1 means that the model correctly predicted the class every time.

Accuracy Analysis

Accuracy measures the proportion of outcomes that the model predicted correctly.

Formula:

Accuracy = (TP + TN) / Total # of Predictions

where

- **TP** represents the number of true positives (positive prediction with a positive result)
- **TN** represents the number of true negatives (negative prediction with a negative result)
- **Total # of Predictions** that the model made (both correct and incorrect)

Metric	Description
The maximum accuracy for this model is	The maximum accuracy for this model.
True Positive Rate	This model correctly scores records as true n% of the time.
True Negative Rate	This model correctly scores records as false n% of the time.
False Positive Rate	This model incorrectly scores records as true n% of the time.
False Negative Rate	This model incorrectly scores records as false n% of the time.
Threshold Value	The threshold value represents the tradeoff between the true positive and false positive rates. You can adjust this threshold setting in the Threshold Evaluation Tab for Classification Use Cases on page 1692.

SEE ALSO:

 Metrics for Classification Use Cases

Gain and Lift Charts for Classification Use Cases

This tab shows two charts - Cumulative Gains and Cumulative Capture - that help you evaluate your model's ability to predict outcomes.

Cumulative Gains Chart

The Cumulative Gains chart plots gain against the cumulative data fraction for this model.
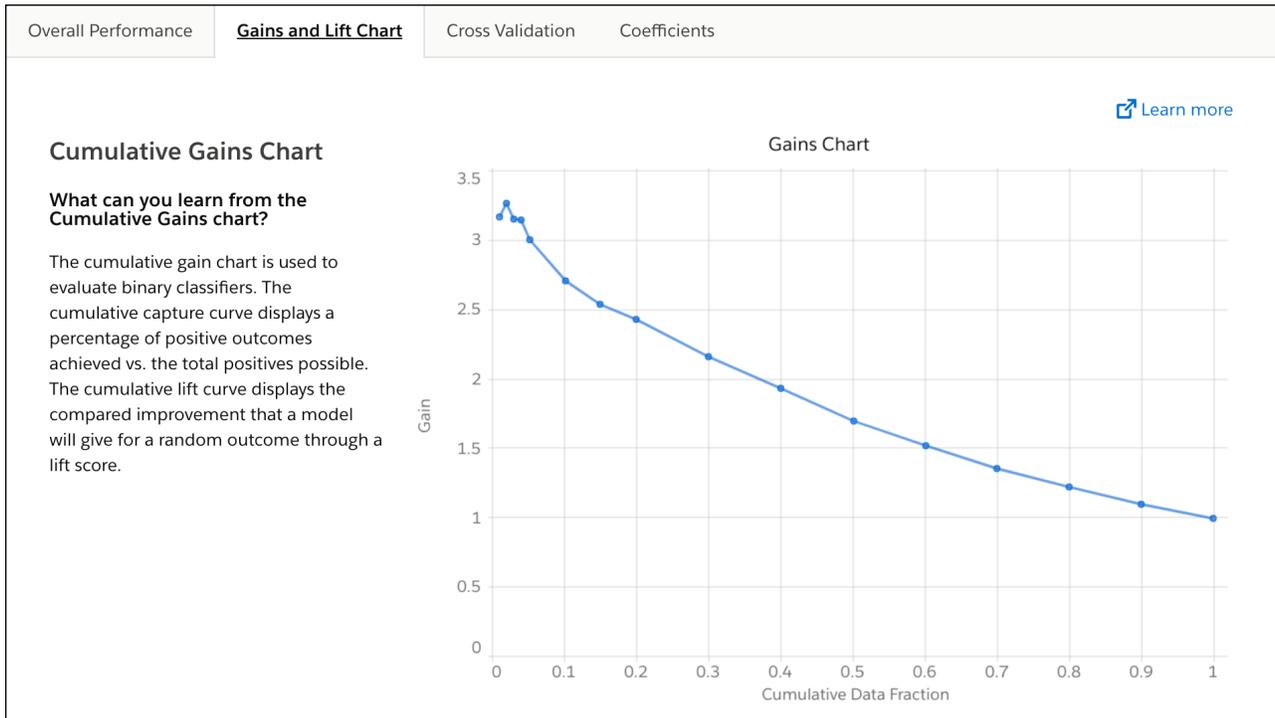
The Cumulative Gains displays a plot of:

- **Cumulative Data Fraction** (X-axis): Percentile. For example, a cumulative data fraction of 0.1 equates to the top decile, or the 10% of records with the highest scores.
- **Gain** (Y-axis).

Hover over a data point to see details about the X and Y coordinates.

Cumulative Capture Rate Chart

The cumulative capture rate chart shows the lift curve as a cumulative percentage of positive outcomes achieved versus the total positives possible. This model prioritizes the high probability cases, resulting in more positive outcomes in the initial actions.
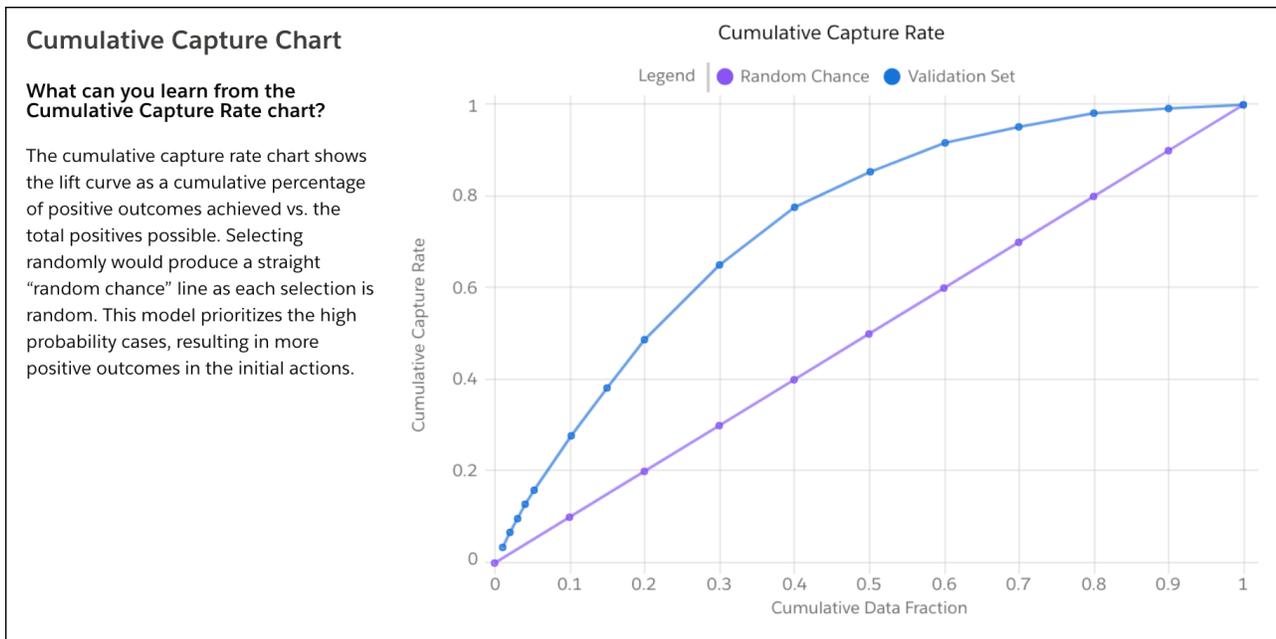
The Cumulative Capture Chart displays a plot of:

- **Cumulative Data Fraction** (X-axis): Percentile. For example, a cumulative data fraction of 0.1 equates to the top decile, or the 10% of records with the highest scores.
- **Cumulative Capture Rate** (Y-axis): The percent of all true cases captured by that percentile. For example, a 0.20 cumulative capture rate at a 0.10 cumulative data fraction means that 20% of all of your TRUE cases are found in the top 10% of model scores.

Data plots represent the following information:

- **Random Chance**: data resulting from random selection (no model)
- **Validation Set**: data derived from the model against the data in the Tableau CRM dataset that Einstein Discovery used to validate the model's performance

Hover over a data point to see details about the X and Y coordinates.

Cumulative Data Fraction	0.4
	(6.89% of 5.81 for Validation Set)
Cumulative Capture Rate	0.78
	(8.87% of 8.76 for Validation Set)

SEE ALSO:

    Metrics for Classification Use Cases

Cross-Validation Tab for Classification Use Cases

To test a model's ability to make predictions, Einstein Discovery uses *k*-fold cross-validation, a process that reduces sampling bias when validating a model. This tab summarizes the results of the cross-validation process for this model, as well as some of the underlying computational details.

Overall Performance	Gains and Lift Chart	**Cross Validation**	Coefficients

4-Fold Cross-Validation Results ⓘ

⬀ Learn more

Metric Name		Training Set	Validation Set	Fold 1	Fold 2	Fold 3	Fold 4
Number of Rows	ⓘ	7043	7043	1764	1714	1774	1791
AUC	ⓘ	0.835	0.826	0.822	0.826	0.83	0.831
GINI	ⓘ	0.67	0.652	0.643	0.652	0.659	0.662
Log Loss	ⓘ	0.427	0.436	0.44	0.427	0.434	0.442
Mean Per Class Error	ⓘ	0.242	0.243	0.246	0.249	0.242	0.236

Model Validation Methodology

Einstein Discovery conducts k-fold cross-validation (k=4) on your model. This process involves the following steps:

1. Randomly divide all the observations in the Tableau CRM dataset into four separate partitions of equal size.

2. Conduct four test passes (folds) in which three of the partitions serve as the training set and one partition serves as the test set.

   📝 Note: After completing the four test passes, each partition has served once as the validation set and three times as part of the training set.

3. For each fold, compile model metrics.

4. Take the average of the four folds for an overall score.

Cross-Validation Columns

The following table describes the columns in the Cross-Validation tab.

Column Name	Description
Metric Name	Name of the metric.
Training Set	Metrics for the set of observations in the Tableau CRM dataset that Einstein Discovery uses to train the model.
Validation Set	Metrics for the set of observations in the Tableau CRM dataset that Einstein Discovery uses to validate the predictions generated by the trained model.
Fold #1	Metrics for the first fold.
Fold #2	Metrics for the second fold.
Fold #3	Metrics for the third fold.
Fold #4	Metrics for the fourth fold.

Cross-Validation Metrics

The following table describes the metrics in the Cross-Validation tab.

Metric Name	Description
Number of Rows	Total number of observations. The meaning of a value varies per column. • For the **Training Set** and **Validation Set** columns, the numbers are the same. This value represents the total number of observations in the entire dataset used in the creation of the story. • For the **Fold #**1 through **Fold #4** columns, this value represents how many observations fell in that fold (approximately 25% of the entire dataset).
AUC	Area Under the Curve. Represents the rate of correct classification by a logistic model. AUC is the most reasonable metric to use for classification use cases. Range: • 0.5 is randomly guessing • 1.0 means that the model correctly classified the data 100% of the time (which is suspicious)
GINI	GINI Index. Quantifies how close the obtained model performs to a theoretically best possible model. Range: • 0 means that the model is no better than random. • 1 means that the predictions match observations exactly (view with skepticism)
Log Loss	Logarithmic Loss. Measures model performance on a scale of 0 to 1, where 0 represents a perfect model (the predicted probability correctly matches actual observations 100%). The less the predicted probability correctly matches actual observations (lower performance), the higher the log loss. Log loss considers uncertainty in model performance.
Mean Per Class Error	Measures how often the predictions are wrong. Lower values mean the predictions are wrong less often, and therefore the model is better at making predictions.

SEE ALSO:

Metrics for Classification Use Cases

Coefficients Tab for Classification Use Cases

A model uses coefficients to calculate a prediction for a specific observation. You can filter the list of coefficients and also download the data in a CSV file.

Metric	Description
Number of Variables	Number of variables in the model.
Number of Variables Including Interactions	Number of variables plus interactions (two-variable pairs) in the model.
Intercept	The expected mean value of Y when all $X = 0$.
Value	Variable or variable pair used in the model.
Coefficient	Number that represents the impact that an explanatory variable (or a pair of explanatory variables) has on the outcome variable. The coefficient quantifies the change in the mean of the outcome variable when there is a one-unit shift in the explanatory variable(s), assuming all other variables in the model remain constant.
Scaled Importance	Importance relative to other coefficients, starting with a maximum importance of 1.

You can:

- Filter the list by typing text in the search box.
- Click **Download CSV** to download the model's coefficient values to a CSV file.

SEE ALSO:

[Metrics for Classification Use Cases](#)

Threshold Evaluation Tab for Classification Use Cases

The Threshold Evaluation tab helps you optimize the threshold value for a model. The threshold value tells your model how to classify a binary outcome. If the calculated probability is above the threshold value, Einstein classifies the outcome one way (such as True or Positive). If the calculated probability is below the threshold value, Einstein classifies the outcome the other way (such as False or Negative).

Threshold Evaluation   ↗ Learn more

The threshold in a classification model is a value ranging from 0 to 1. It represents the tradeoff between the true positive and false positive rates.

ROC Curve ⓘ

Legend | ● Model ● No Model

Number of Rows   **AUC** ⓘ
7043   0.826

Controls

Threshold Controls

Threshold Value   0.514

Optimize a specific metric

[Custom ▼]

Cost Ratio ⓘ

Set this and Einstein can help select the optimal threshold for your model

[No cost ratio set]   [Apply]

Metric	Description
ROC Curve	Receiver Operating Characteristic Curve. Displays the performance measurement at various threshold settings. ROC is a probability curve and AUC (Area Under the Curve) represents the degree or measure of separability. This chart shows how well the model is able to distinguish between classes. • **Y-axis**: True Positive Rate: **TPR = TP / (TP + FN)** • **X-axis**: False Positive Rate>: **FPR = FP / (FP + TN)** • **Model** (blue line) • **No Model** (gray line)—the same as random chance

Metric	Description
Controls	You can set an optimal threshold that represents the cutoff for the two outcomes you are predicting. To change the selected threshold value: • **Threshold Value**: Drag the slider to set the threshold. • **Optimize a Specific Metric**: Select a common metric from the list. • **Cost Ratio**: To let Einstein Discovery pinpoint an optimized threshold, specify a cost ratio (the ratio between the false positives and false negatives). The **Threshold Value** value reflects your selection. In the ROC graph, the blue dot moves to the corresponding location on the Actual Model line that represents the threshold value along the ROC curve.
Number of Rows	Number of rows in the training data.
AUC	Area Under the Curve. Represents the rate of correct classification by a logistic model. An AUC of 0.5 means that the model performs no better than random guessing. An AUC of 1.0 means that the model correctly classifies data 100% of the time, which can indicate data leakage.

SEE ALSO:

[Metrics for Classification Use Cases](#)

Prediction Examination Tab for Classification Use Cases

The Prediction Examination tab allows you to interact with the model metrics and see how they stack up against actual outcomes.

Examination of the Training Dataset

The table displays a random sample of 100 rows of data (observations) in the Tableau CRM dataset used to train the model. Columns in the table represent columns in the dataset.

Einstein Prediction

Select a row in the table to display details about the prediction that the model generated for that observation.

Metric	Description
Predicted Outcome	Outcome that the model predicted.
Actual Outcome	Outcome that actually occurred (observed).
Score	Prediction score for this observation.
Threshold	The threshold value for the model. The threshold represents the tradeoff between the true positive and false positive rates.
Optimize a Specific Metric	To maximize the threshold for a common metric, select that metric from the list.
Top factors	Shows the features that have the biggest contribution to the outcome.

SEE ALSO:

Metrics for Classification Use Cases

R Code Tab

The R Code tab contains the transformations made to your Tableau CRM dataset and scoring code. Inspect the code to examine the underlying details of the model that Einstein Discovery produced.

Note: The sole purpose of showing the R code is for model transparency. It is not intended to be modified or executed.

If you want to copy the R Code text to another editor for viewing, click **Copy to Clipboard**.

SEE ALSO:

[Metrics for Classification Use Cases](#)

Metrics for Numeric Use Cases

The numeric use case is based on outcomes that are numeric variables. The Model Metrics tabs show quality statistics associated with linear regression models.

The following tabs display Model Metrics.

[Overview Tab for Numeric Use Cases](#)

The Overview tab shows an at-a-glance summary of your model's validation results.

[Model Evaluation Tab for Numeric Use Cases](#)

The Model Evaluation tab provides information about a models' performance, cross-validation results, and coefficent values.

The Prediction Examination tab allows you to interact with the model metrics and see how they stack up against actual outcomes.

SEE ALSO:

Overview Tab for Numeric Use Cases

The Overview tab shows an at-a-glance summary of your model's validation results.

Path to Deployment

Area	Description
Review Model Accuracy	R^2 measures a regression's model's ability to explain variation in the outcome. R^2 ranges from zero (random chance) to one (perfect model). In general, the higher the R^2, the better the model predicts outcomes. Click **View Model Evaluation** to see the Model Evaluation Tab for Numeric Use Cases on page 1697.

Area	Description
Pre-deploy Checks	Summarizes the results of validation tests on the story data: - Recommended updates to the story. Click **View** to Implement Recommended Updates on page 1705. - Cross-Validation Test - Accountability Alerts

Training Data and the Model

Area	Description
Distribution of Outcome Variable	Shows the distribution of values (count and range) for the outcome variable in the training data.
Top Predictors	Lists the top predictors (explanatory variables) for this model and their correlation with the outcome variable.

SEE ALSO:

Metrics for Numeric Use Cases

Model Evaluation Tab for Numeric Use Cases

The Model Evaluation tab provides information about a models' performance, cross-validation results, and coefficent values.
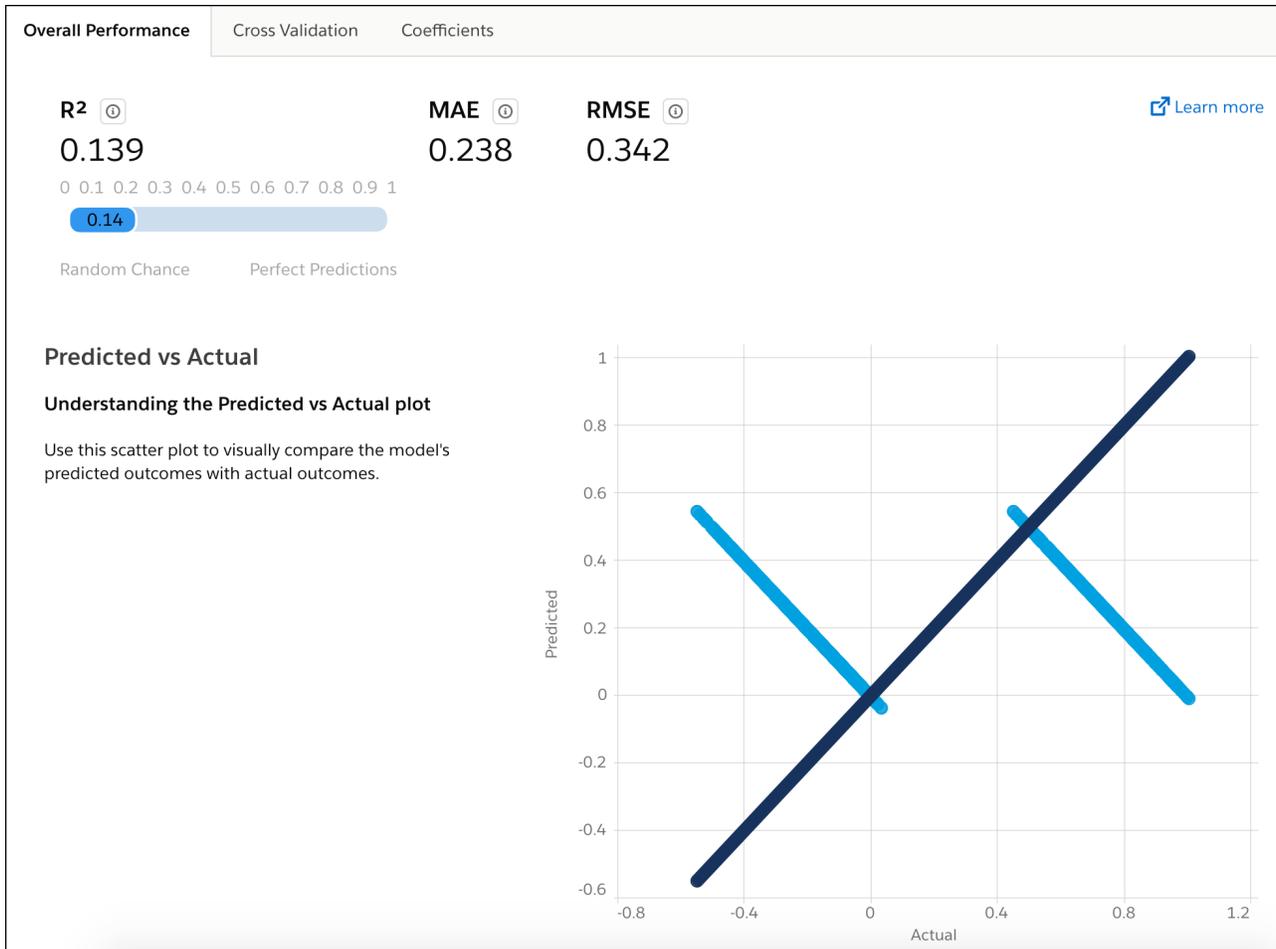
For model evaluation details, click a subtab.

[Overall Performance Tab for Numeric Use Cases](#)

The Overall Performance tab shows metrics for key indicators and model quality.

[Cross-Validation Tab for Numeric Use Cases](#)

To test a model's ability to make predictions, Einstein Discovery uses *k*-fold cross-validation, a process that reduces sampling bias when validating a model. This tab summarizes the results of the cross-validation process for this model, as well as some of the underlying computational details.

[Coefficients Tab for Numeric Use Cases](#)

A model uses coefficients to calculate a prediction for a specific observation. You can filter the list of coefficients and also download the data in a CSV file.

SEE ALSO:

[Metrics for Numeric Use Cases](#)

Overall Performance Tab for Numeric Use Cases

The Overall Performance tab shows metrics for key indicators and model quality.

On the Model Evaluation Tab for Numeric Use Cases on page 1697, click **Overall Performance**.

Summary

Metric	Description
R^2	R^2 measures the model's ability to explain variation in the outcome, which is an indicator of how predictive the model is. Range: • 0 means that the model is not able to explain any variability in the outcome. • 1 means that the model explains all of the variability.
MAE	Mean Absolute Error. Measures the absolute difference between the actual value and the prediction. All differences are weighted equally in this average, which means that it is not as sensitive to outliers as MSE.
RMSE	Root Mean Squared Error. Represents the square root of MSE (Mean Squared Error, which is the average squared error of the model's predictions). RMSE measures the difference between the values predicted by the model and the observed values. You can think of RMSE as the "standard deviation of errors".

Predicted vs Actual

Use this scatter plot to visually compare the model's predicted outcomes with actual outcomes.
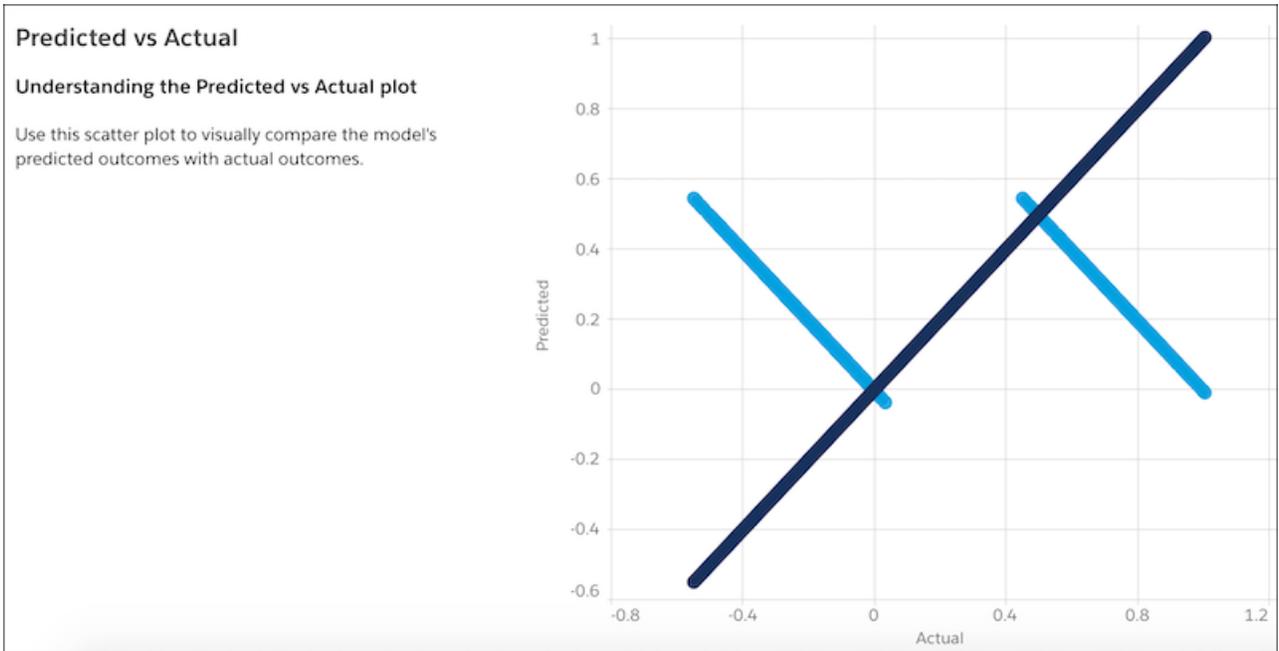
Normal QQ Plot for Standardized Residuals

For regression models, one of the key assumptions is that the residual errors for the outcome variable are normally distributed. Use the QQ (quantile-quantile) plot to quickly check this assumption and determine whether and how residual errors depart from normality
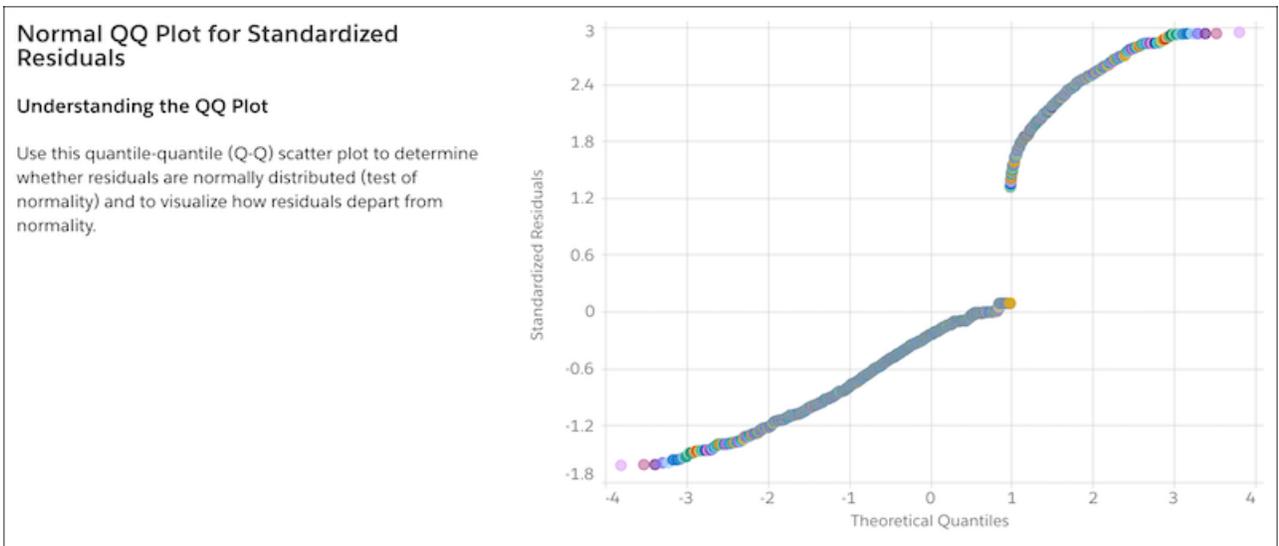
If the QQ plot shows your residual errors to be approximately linear, then you can be confident that your model satisfies the normal distribution assumption.

SEE ALSO:

Cross-Validation Tab for Numeric Use Cases

To test a model's ability to make predictions, Einstein Discovery uses *k*-fold cross-validation, a process that reduces sampling bias when validating a model. This tab summarizes the results of the cross-validation process for this model, as well as some of the underlying computational details.

On the , click **Cross Validation**.

Metric Name		Training Set	Validation Set	Fold 1	Fold 2	Fold 3	Fold 4
Number of Rows	ⓘ	7043	7043	1764	1714	1774	1791
R²	ⓘ	0.148	0.139	0.13	0.113	0.164	0.151
MSE	ⓘ	0.116	0.117	0.114	0.12	0.114	0.119
RMSE	ⓘ	0.34	0.342	0.337	0.347	0.338	0.345
MAE	ⓘ	0.236	0.238	0.23	0.24	0.238	0.242
RMSLE	ⓘ	0.239	0.24	0.237	0.243	0.238	0.241
Residual Deviance	ⓘ	815.38	824.08	200.76	205.93	202.83	213.06
Mean Residual Deviance	ⓘ	0.116	0.117	0.114	0.12	0.114	0.119
Null Deviance	ⓘ	956.83	956.97	230.92	232.23	242.6	251.21
Null Degrees of Freedom	ⓘ	7042	7042	1763	1713	1773	1790
Residual Degrees of Free	ⓘ	6969	6969	1690	1640	1700	1717

Title above table: **4-Fold Cross-Validation Results** ⓘ   ⬈ Learn more

Tabs: Overall Performance | **Cross Validation** | Coefficients

Model Validation Methodology

Einstein Discovery conducts *k*-fold cross-validation (*k*=4) on your model. This process involves the following steps:

1. Randomly divide all the observations in the Tableau CRM dataset into four separate partitions of equal size.

2. Conduct four test passes (folds) in which three of the partitions serve as the training set and one partition serves as the test set.

   📝 Note: After completing the four test passes, each partition has served once as the validation set and three times as part of the training set.

3. For each fold, compile model metrics.

4. Take the average of the four folds for an overall score.

Model Metrics Columns

The following table describes the columns in the Model Metrics area of the Cross-Validation tab.

Column Name	Description
Metric Name	Name of the metric.
Training Set	Metrics for the set of observations in the Tableau CRM dataset that Einstein Discovery used to train the model.
Validation Set	Metrics for the set of observations in the Tableau CRM dataset that Einstein Discovery used to validate the predictions generated by the trained model.
Fold #1	Metrics for the first fold.
Fold #2	Metrics for the second fold.
Fold #3	Metrics for the third fold.
Fold #4	Metrics for the fourth fold.

Model Metrics Rows

The Cross-Validation tab shows the following metrics for a model.

Metric Name	Description
Number of rows	Total number of observations. The meaning of a value varies per column. • For the **Training Set** and **Validation Set** columns, the numbers are the same. This value represents the total number of observations in the entire dataset used in the creation of the story. • For the **Fold #**1 through **Fold #4** columns, this value represents how many observations fell in that fold (approximately 25% of the entire dataset). • For the **Average** column, this value represents the average of the folds, which is the number of observations in the dataset divided by 4.
R^2	R^2 measures the model's ability to explain variation in the outcome, which is an indicator of how predictive the model is. Range: • 0 means that the model is not able to explain any variability in the outcome (random chance). • 1 means that the model explains all of the variability (perfect model).
MSE	Mean Squared Error. Measures the average squared error of the model's predictions. MSE computes the square difference between the observed (actual) outcome and the predicted values, and then averages them.
RMSE	Root Mean Squared Error. Represents the square root of MSE. RMSE measures the difference between the values predicted by the model and the observed (actual) values. You can think of this value as the "standard deviation of errors."
MAE	Mean Absolute Error. Measures the absolute difference between the actual value and the prediction. All differences are weighted equally in this average which means that it is not as sensitive to outliers as MSE.
RMSLE	Root-Mean-Squared Logarithmic Error. Compare with RMSE: • RMSLE is less sensitive to outliers than RMSE. RMSE penalizes only for the overall scale of the error, while RMSLE penalizes for the scale of the error relative to the actual value.

Metric Name	Description
	• RMSLE penalizes under-predictions more than over-predictions. This makes RMSLE more useful than RMSE when you want to err on the side of over-estimating. For example, to optimize stock levels of non-perishables, you prefer a surplus over potentially running out of stock.
Residual Deviance	Measures how well the developed model performs on your dataset. • Just as with Null Deviance, lower values mean better fit. Thus, if Residual Deviance is low, the model does well fitting the data. • The larger the gap between Residual Deviance and Null Deviance, the better the model is doing at picking up the complexities in the data.
Mean Residual Deviance	Measures how well the model performs on your dataset. Lower values indicate a better fit.
Null Deviance	Measures how well an extremely simple model would perform on your dataset. Lower values mean better fit. Thus, if Null Deviance is low, your data is not particularly complex.
Null Degrees of Freedom	Represents the Chi-Square distributions of Null Deviance.
Residual Degrees of Freedom	Represents the Chi-Square distributions of Residual Deviance.

SEE ALSO:

Metrics for Numeric Use Cases

Coefficients Tab for Numeric Use Cases

A model uses coefficients to calculate a prediction for a specific observation. You can filter the list of coefficients and also download the data in a CSV file.

On the , click **Coefficients**.

Metric	Description
Number of Variables	Number of variables in the model.
Number of Variables Including Interactions	Number of variables plus interactions (two-variable pairs) in the model.
Intercept	The expected mean value of Y when all X = 0.
Value	Variable or variable pair used in the model.
Coefficient	Number that represents the impact that an explanatory variable (or a pair of explanatory variables) has on the outcome variable. The coefficient quantifies the change in the mean of the outcome variable when there is a one-unit shift in the explanatory variable(s), assuming all other variables in the model remain constant.
Scaled Importance	Importance relative to other coefficients, starting with a maximum importance of 1.

You can:

- Filter the list by typing text in the search box.
- Click **Download CSV** to download the model's coefficient values to a CSV file.

SEE ALSO:

[Metrics for Numeric Use Cases](#)

Prediction Examination Tab for Numeric Use Cases

The Prediction Examination tab allows you to interact with the model metrics and see how they stack up against actual outcomes.

Examination of the Training Dataset

The table displays a random sample of 100 rows of data (observations) in the Tableau CRM dataset used to train the model. Columns in the table represent columns in the dataset.

Einstein Prediction

Select a row in the table to display details about the prediction that the model generated for that observation.

Metric	Description
Predicted Outcome	Outcome that the model predicted.
Actual Outcome	Outcome that actually occurred (observed).
Top factors	Features with the biggest contribution to the outcome.

SEE ALSO:

Implement Recommended Updates

If Einstein Discovery detects possible improvements in your data during validation, it displays a message under **Validation Results** in the Overview tab and prompts you to **Review Updates**.

1. To see the suggested improvements, click **Review Updates** in the Overview tab.

   📝 Note: The **Review Updates** button is clickable only if Einstein Discovery found opportunities to improve your story. Otherwise, the button is grayed out.

Einstein Recommends Improvements in 3 Areas

Duplicates   Dominant Values   Recommended Buckets

Multiple fields are providing the same information (for example NAME and ID). Keeping the most descriptive (for example NAME) will ease interpretation and allow additional insights to surface.

MonthOfChurn and Churn
MonthOfChurn and Churn represent the same data. Keeping the most descriptive field will ease interpretation and allow additional insights to surface.
- ◯ Retain MonthOfChurn
- ◯ Retain Churn
- ⦿ Do Nothing

StreamingMovies and OnlineSecurity
StreamingMovies and OnlineSecurity are 70.8% similar and explain 3.5% and 4.4% of the variation respectively. Together they explain no additional variation.
- ◯ Retain StreamingMovies
- ◯ Retain OnlineSecurity
- ⦿ Do Nothing

StreamingMovies and DeviceProtection
StreamingMovies and DeviceProtection are 73.6% similar and explain 3.5% and 3.3% of the variation respectively. Together they explain no additional variation.
- ◯ Retain StreamingMovies
- ◯ Retain DeviceProtection
- ⦿ Do Nothing

StreamingMovies and InternetService
StreamingMovies and InternetService are 71.6% similar and explain 3.5% and 7% of the variation respectively. Together they explain no additional variation.
- ◯ Retain StreamingMovies
- ◯ Retain InternetService
- ⦿ Do Nothing

Cancel   Customize Story   Create New Story

---

2. In each tab, accept any improvements you want to implement in your story.

3. Click **Create New Story**.

For more information, see Improve a Story on page 1648.

Deploy Models

Deploy a model so that you can use it to make predictions and improvements.

📝 Note: Before you deploy a model:

- Consider reviewing Model Metrics to assess its quality.
- For the classification use case (binary outcomes based on text fields), consider setting the threshold level in the Threshold Evaluation Tab for Classification Use Cases on page 1692.
- If you want this model to use automated prediction fields, before you deploy it, read the deployment requirements in Display Einstein Predictions Using Automated Prediction Fields on page 1751.

To deploy a model, open it, click **Deploy Model** (either from the Model Overview screen or from the dropdown on the story toolbar), and then complete the following steps.

1. Deploy to a New or Existing Prediction Definition
   Choose whether to deploy this model to a new prediction definition or to an existing prediction definition that you select.

2. Select How To Deploy the Model
   If you deploy this model to a new prediction, select the Salesforce object you want to associate with the prediction.

3. Map Dataset Fields to Fields in the Salesforce Object

   Define the mapping between fields in the Tableau CRM dataset and fields in the Salesforce object.

4. Configure the Prediction Field

   Choose whether you want automated prediction fields.

5. Configure Segmentation Filters

   Choose whether to use the model to get predictions on all data or on just a segment (subset) of the data. For example, you can focus on a specific product model or a group of customers. A prediction definition can contain multiple models in which each model produces predictions for a different segment.

6. Configure Improvements

   Einstein Discovery improvements are suggested actions that users can take to improve predicted outcomes. To activate improvements, enable one or more variables as actionable in your model. Actionable variables represent factors in the data that people can control, such as deciding which marketing campaign to use for a particular customer.

7. Configure Model Performance Monitoring

   Model accuracy compares a model's predicted outcomes with actual (observed) outcomes. If you want Einstein to monitor the performance of this model, you must tell it how to recognize actual outcomes (data that are not expected to change).

8. Review Your Selections and Deploy the Model

   Review your deployment settings before deploying the model.

Deploy to a New or Existing Prediction Definition

Choose whether to deploy this model to a new prediction definition or to an existing prediction definition that you select.

1. Select one of the following options:

Option	Description
Deploy model as a new prediction	Create a prediction definition with this model. Optionally, change the default **Prediction Title**.
Add model or replace a model in an existing prediction	Deploy this model to an existing prediction definition that you select. You can add a model or you can replace a deployed model.
	A prediction definition is a container object that consists of one or more models. For example, you can have two different win rate models for your opportunities - one for small business and one for enterprises. Each model can be deployed to the same prediction definition. Segmentation filters can be used to choose the correct model for the record that is being scored. Later, you can use the Model Manager to change the order of the models in this prediction definition.

2. To deploy to an existing prediction definition, select it from the list:

If you want to replace a deployed model, select **Replace model** and select it from the list.

3. Do one of the following:

- If you deploy to a new prediction definition, click **Add Object**.
- If you deploy to an existing prediction definition, click **Map Fields** and skip to Map Dataset Fields to Fields in the Salesforce Object on page 1709.

Select How To Deploy the Model

If you deploy this model to a new prediction, select the Salesforce object you want to associate with the prediction.
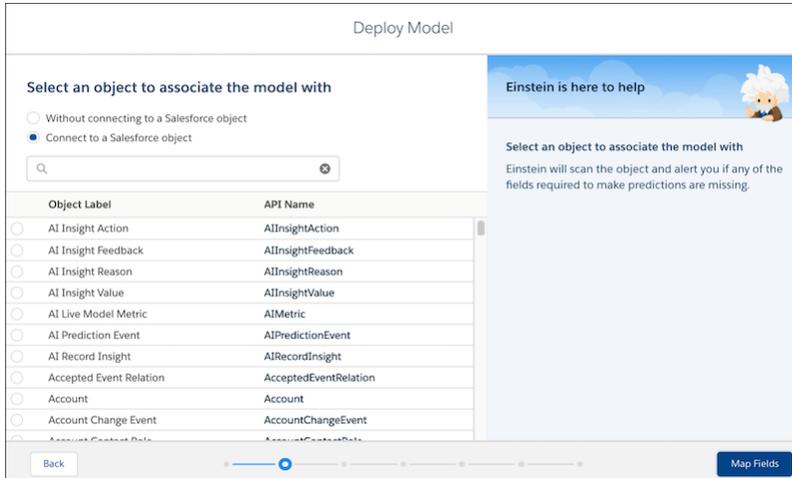
> **Note:** If this model does not depend on a Salesforce object, select **Without connecting to a Salesforce object** and proceed
> to Configure Segmentation Filters on page 1712. Skip selecting a Salesforce object only if you intend to use the model for the
> following use cases:
>
> - programmatic predictions via the Einstein Prediction Service , or
> - dataset scoring via the prediction Transformation component in Tableau CRM dataflows.
>
> In all other cases, selecting a Salesforce object during deployment is required.

1. Select the target Salesforce object to which you want to add this model.

   Einstein Discovery scans the object and alerts you whether you're missing any of the fields required to make predictions. If there are
   missing fields, deploying the model requires more development work by the Salesforce Admin to expose the predictions, as described
   in Display Einstein Predictions Using Custom Fields (Deprecated) on page 1784. If possible, consider selecting a different target
   Salesforce object.

2. Click **Map Fields**.

Map Dataset Fields to Fields in the Salesforce Object

Define the mapping between fields in the Tableau CRM dataset and fields in the Salesforce object.
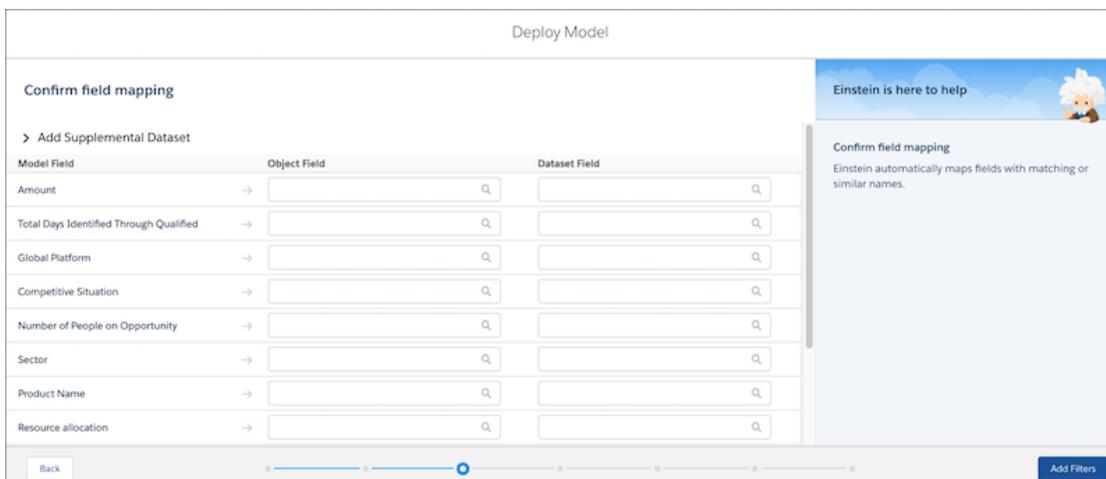
1. If you want to include external data or computed fields in queries associated with this model, click **Add Supplemental Dataset** and fill in the following fields.

Field	Description
Selected Salesforce Object	Name of the selected Salesforce object. Read only.
Object ID Field in Selected Salesforce Object	Name of the Object ID field in the selected Salesforce object. Einstein automatically selects a default field. If you want to change this setting, click the **x** and select a different field. 📝 Note: Einstein displays object fields to which you have access. If you don't see the object ID field you expect, check the field-level security settings for your user account to ensure that you have at least read access to that field in the Salesforce object.
Supplemental Dataset	Name of the supplemental dataset that contains the additional data to use when scoring records. 📝 Note: Adding a dataset will use only Salesforce sharing rules. Any sharing rules on the dataset will be ignored.
Record ID Field in Supplemental Dataset	Record ID field in the selected supplemental dataset that corresponds to the selected Object ID field in the Salesforce object.

Use supplemental datasets for predictive models that require information that is not found in Salesforce, such as explanatory variables stored in outside data sources, or computed fields. For example, to score an Opportunity record, a model can require the task total, which is a calculated field. Using a dataflow, you can calculate and save the task total value in a supplemental Tableau CRM dataset. Then, when Einstein scores a Salesforce record, it retrieves this calculated value from the row of the supplemental dataset that corresponds to the record in the Salesforce object.

📝 Note: Predictions that connect to a dataset depend on the frequency with which the dataset is refreshed.

2. For a field in the dataset, select its corresponding field in the Salesforce object.

📝 Note:

- If you want to deploy this model using automated prediction fields (see Display Einstein Predictions Using Automated Prediction Fields on page 1751), you must map *all* fields.
- If you deploy this model using no automated prediction fields, then you do not need to map all fields.
- If you want to use this model for the prediction Transformation on page 925 in Tableau CRM data flows, do not map *any* fields.

Field	Description
Model Field	Name of the field in the model.
Object Field	Name of the field in the Salesforce object.
Dataset Field	Name of the corresponding field in the supplemental dataset, if applicable.

3. Click **Config Prediction Field**.

Configure the Prediction Field

Choose whether you want automated prediction fields.

1. Select one of the following options.

Option	Description
No prediction field (automatic predictions will not be enabled)	Do not store results, or store results in a custom field that you configure later. Select this option if you do not use automated prediction fields. This kind of prediction is run programmatically via the Einstein Prediction Service on page 1754 using REST API calls or implementations based on the Einstein Discovery Managed Package described in Display Einstein Predictions Using Custom Fields (Deprecated) on page 1784.
Create a new prediction field from label	Have Einstein create the prediction field using the label you specify. Einstein stores prediction results in this prediction field.
Use an existing prediction field	Have Einstein store prediction results in an existing prediction field using the developer name you specify.

> **Note:** After you deploy this model, you can change these settings using the Model Manager. See Configure the Prediction Field for a Prediction Definition on page 1729.

2. Click **Add Filters**.

Configure Segmentation Filters

Choose whether to use the model to get predictions on all data or on just a segment (subset) of the data. For example, you can focus on a specific product model or a group of customers. A prediction definition can contain multiple models in which each model produces predictions for a different segment.

1. If you want to add filters, select **Yes, focus on a segment (Advanced)** and specify one or more filters.

A filter defines a condition that selects the records to include in a segment. For each filter, select the field and operator (such as Equals, Contains, or Starts with), and then specify a value.

> **Note:**
> - Filter expressions are case sensitive. The value you specify must be an exact match. For example, "software" and "Software" are different matches. If your data contains variations in capitalization, consider cleansing your data during data prep to standardize patterns of uppercase and lowercase characters. For instructions on preparing data and data integration options, see Integrate Data into Tableau CRM Datasets on page 597.
> - Einstein displays object fields to which you have access. If you don't see an object field you expect, check the field-level security settings for your user account to ensure that you have at least read access to that field in the Salesforce object.

2. Click **Add Variables**.

Configure Improvements

Einstein Discovery improvements are suggested actions that users can take to improve predicted outcomes. To activate improvements, enable one or more variables as actionable in your model. Actionable variables represent factors in the data that people can control, such as deciding which marketing campaign to use for a particular customer.

📝 **Note:** After you deploy this model, you can change these settings using the Model Manager. See

📝 **Note:** Einstein displays variables associated with object fields to which you have access. If you don't see a variable you expect, check the field-level security settings for your user account to ensure that you have at least read access to that field in the Salesforce object.

1. To have Einstein Discovery suggest ways in which to improve a predicted outcome, select one or more variables in the list. Select only variables that can be controlled. Do not select any fields that cannot be controlled, such as a customer's name or postal code. Einstein Discovery suggests actions only for the fields selected here.

2. If you want, you can customize the text that Einstein Discovery uses to suggest actions to users. If one or more variables are enabled for improvements, select **Specify custom text** and define the text you want Einstein Discovery to use when prompting users for improvements.

> **Note:** Custom text is not localized. Einstein Discovery displays the text exactly as you specify it here.

For each enabled variable, you can select one of three options:

Option	Description
Use default text	Displays the default Einstein Discovery improvement text for this value.
Do not show	Hides improvement text for this value. Use this when a suggested improvement does not apply.
Use custom text	Displays the custom improvement text you specify here.

Configure Text Variables

You can define custom text for individual values in a text variable. For example, for a ServicePlan variable, you can define one message for "Trial and "Basic" ("Upgrade to Premium Plan"), and another for "None" ("Sign Up for a Service Plan"):

For each text value that you want to customize, click **Add**, select **Use custom text**, and specify the text you want. You can include the following placeholders inside the message text:

Placeholder	Description
`[field_name]`	Variable display name.
`[value]`	Suggested value or range.

> **Note:** The **Any Value** option applies to any text value other than those customized above.

Configure Numeric Variables

You can define custom text for increasing or decreasing a numeric variable. Select **Use custom text** and specify the text you want. You can include the following placeholders inside the message text:

Placeholder	Description
[field_name]	Variable display name.
[value]	Suggested value or range.
[diff]	Difference between the current value and the suggested value or range.

For bucketed numbers, you can also specify:

Placeholder	Description
[value_low]	Lower edge of the suggested range.
[value_high]	Higher edge of the suggested range.
[min_diff]	Difference between [value_low] and the current value.
[max_diff]	Difference between [value_high] and the current value.

3. Click **Define Terminal State**.

Configure Model Performance Monitoring

Model accuracy compares a model's predicted outcomes with actual (observed) outcomes. If you want Einstein to monitor the performance of this model, you must tell it how to recognize actual outcomes (data that are not expected to change).

An actual outcome represents data that has reached its *terminal state*, such as historical data that is not expected to change. For example, after an order ships, the ship date and the number of items shipped are finalized.

1. Define one or more conditions under which your model's outcome field has attained its terminal state. That way, Einstein Discovery knows which outcomes to include in the performance analysis. For each condition, select the field and operator (such as Equals, Contains, or Starts with), and then specify a value.

> **Note:**
> - Filter expressions are case sensitive. The value you specify must be an exact match. For example, "software" and "Software" are different matches. If your data contains variations in capitalization, consider cleansing your data during data prep to standardize patterns of uppercase and lowercase characters. For instructions on preparing data and data integration options, see Integrate Data into Tableau CRM Datasets on page 597.
> - Einstein displays object fields to which you have access. If you don't see an object field you expect, check the field-level security settings for your user account to ensure that you have at least read access to that field in the Salesforce object.
>
> To view the performance for this model, display it in the Model Manager. See View a Prediction Definition on page 1723.

> **Note:** After you deploy this model, you can change these settings using the Model Manager. See Configure Performance Monitoring for a Prediction Definition on page 1730.

2. Click **Review**.

Review Your Selections and Deploy the Model

Review your deployment settings before deploying the model.

1. Verify your selections.

    If you want to change anything, click **Back** to return to a previous screen and change the selection.

2. Click **Deploy**. Einstein deploys this model to the target Salesforce object and notifies you whether it succeeded.

3. Click **Done**.

After a model is deployed, you can:

- Compare Models on page 1717
- Manage Prediction Definitions and Models on page 1719
- Predict Outcomes on page 1750

Compare Models

Compare metrics for multiple models side by side to see how they stack up against each other. For example, comparing segments in your data can reveal the most important variables in each segment.

To compare models:

1. In Model Metrics, click **Compare Models**.

2. Select two or three models to compare.

3. Click **Show Comparison**.
Einstein Discovery compares the selected models, and then opens a new tab showing an overview of the results.

Name	Description
Story Name	Name of the story associated with the model.
Dataset	Name of the dataset associated with the story.
Model Type	Type of model. Example: RegressionGLM
Created Date	Story version date.

4. To view comparative metrics, click the **Metrics** tab.

Model Comparison
Outcome: Maximize CLV

Overview   **Metrics**   Data

Model Metrics

R²	0.18334125003445345	0.18909698051039658	0.2085012754485962
Observation Count	10000	9894	9898
Null Deviance	203845148908.28497	144919729634.24127	146394971112.77625
Residual Degrees of Freedom	9933	9846	9824
RMSLE	NaN	0.18780866440781588	0.1835500034885883
Mean Residual Deviance	16646748.80801361	11875414.255139796	11705849.826190885
MAE	2330.937842793735	2081.9337631929457	2065.107442960079
Residual Deviance	166467488080.1361	117495348640.35315	115864501579.63737
RMSE	4080.0427458561767	3446.072293951448	3421.3812746010763
AIC	194792.02554947062	189349.0256503093	189335.18867841942
Null Degrees of Freedom	9999	9893	9897
MSE	16646748.80801361	11875414.255139796	11705849.826190885

5. To view the top correlations in the data, click the **Data** tab.

Model Comparison
Outcome: Maximize CLV

Overview   Metrics   **Data**

Model Metrics

Story Name	Maximize CLV		Maximize CLV		Maximize CLV	
Top Correlations						
	Division	14.2%	Division	18.1%	Division	18%
	Account Id	9.12%	Type	3.99%	Account Id	11.7%
	Type	3.25%	Industry	2.4%	Type	3.96%
	Rating	2.06%	Rating	2.28%	Industry	2.39%
	Industry	1.56%	AccountScore	0.685%	Rating	2.28%
	AccountScore	0.724%	BillingState	0.278%	AccountScore	0.685%
	BillingState	0.145%	StartDate	-0.000853%	BillingState	0.28%
	StartDate	0.0522%	Ownership	-0.00706%	CloseDate	0.0102%
	Ownership	-0.00339%			Ownership	-0.0079%
	CloseDate	-0.0233%			StartDate	-0.0151%

Manage Prediction Definitions and Models

Use the Model Manager to view, configure, and manage prediction definitions and models that have been deployed in your org.

Start Model Manager
You launch Model Manager from the Analytics panel in Tableau CRM Studio.

Manage Prediction Definitions
Use the Model Manager to manage prediction definitions in your org.

Configure Model Performance Alerts
Models drift when characteristics in the real world data diverge significantly from the training data used to build them. Einstein can send out alert notifications if your model performance falls below configured thresholds. Significant increases in prediction error rates, missing values, or out-of-bounds values can trigger notifications. You can set alert thresholds, schedule weekly or monthly performance evaluation cycles, and specify who should receive notifications.

Manage Models

Use the Model Manager to manage the models associated with your prediction definitions.

Package and Distribute Your Prediction Definitions and Models

Einstein Discovery supports packaging for prediction definitions and models. You can promote prediction definitions from a sandbox to a production environment. You can distribute prediction definitions using managed or unmanaged packages.

SEE ALSO:

Get Predictions in Tableau Calculated Fields

Start Model Manager

You launch Model Manager from the Analytics panel in Tableau CRM Studio.

1. In the Analytics panel, click the **Model Manager** tab.
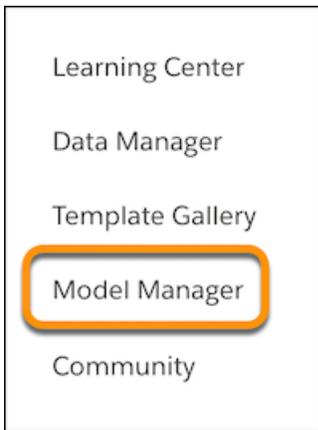
2. Review the statistics that summarize all deployed predictions within your org.

Column	Description
Active/Total Models	• Total number of deployed models that are active within the org. An active model is a valid model that can be used to get predictions. • Total number of deployed models (active + inactive) within the org. A model becomes inactive manually (see Enable or Disable a Model on page 1747) or if an internal error occurred during model deployment.
Total Predictions	Total number of predictions made within the org. Applies only to models that use automated prediction fields. The chart shows the daily number of predictions made within the past ten days.
Warnings	Total number of warnings associated with the predictions within the org. Applies only to models that use automated prediction fields. Warnings occur when column mappings are missing or if data is out of bounds (not found in the training set). The chart shows the daily number of predictions made within the past ten days.

3. Scroll the list of prediction definitions. Each row represents a prediction definition that contains one or more models.
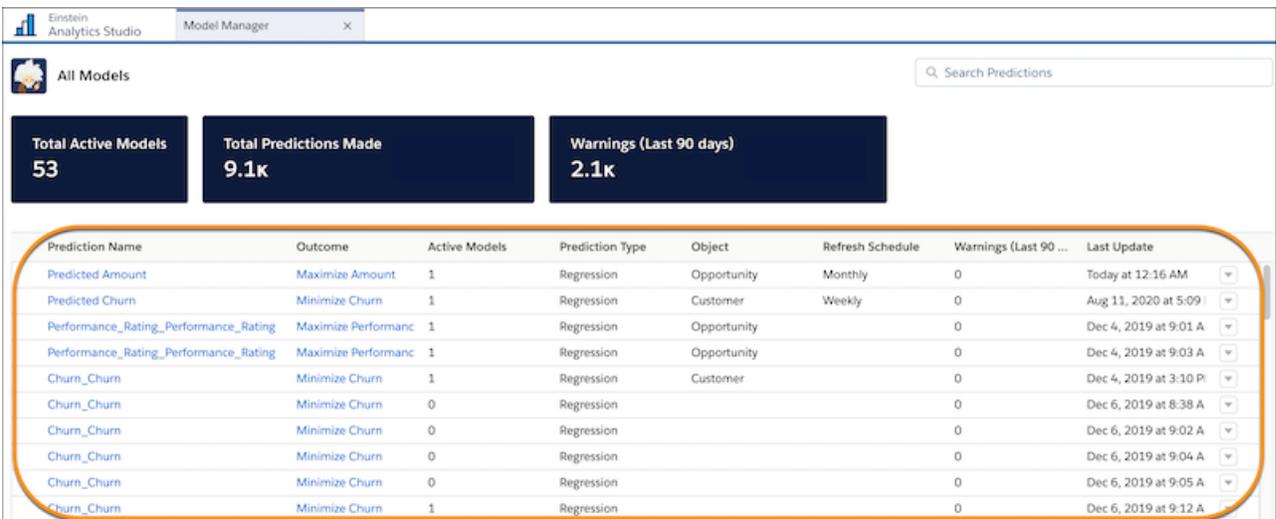
Column	Description
Prediction Name	Name of the prediction definition.
Outcome	Outcome variable for the prediction definition, and whether to minimize or maximize the outcome.

1721

Column	Description
Warnings	Number of warnings associated with the prediction definition. Warnings occur when column mappings are missing or if data is out of bounds (not found in the training set). If a prediction definition has warnings, an alert image appears to its left.
Active Models	Number of active models associated with the prediction definition.
Prediction Type	• **Binary Classification** predictions are associated with text fields (classification use cases) with to possible outcomes • **Regression** predictions are associated with number fields (numeric use cases)
Object	Salesforce object associated with the prediction definition.
Refresh Schedule	Frequency of scheduled automatic refresh.
Warnings (Last 90 days)	Warnings within the most recent 90 days.
Last Update	Timestamp of the last update to the prediction definition.

Manage Prediction Definitions

Use the Model Manager to manage prediction definitions in your org.

View a Prediction Definition

You can view a prediction definition deployed in your org.

Configure Automatic Model Refresh for a Prediction Definition

Use automatic refresh to schedule regular, unattended model updates on a weekly or monthly basis.

Configure the Prediction Field for a Prediction Definition

You can write prediction scores automatically to selected Salesforce fields. Writing prediction scores requires that you deploy a model for which you can map all model fields back to Salesforce fields. You can use an existing AI Prediction field or create one (and specify the label).

Configure Performance Monitoring for a Prediction Definition

If you want Einstein Discovery to display performance monitoring charts for a prediction definition, you must explicitly configure performance monitoring.

Score Records in Bulk

You can score predictions on multiple records at a time. For example, use bulk scoring to refresh all prediction scores after you deploy an updated model. You can also run bulk scoring on historical data to see how well your model performs. With bulk scoring, you can score all records, a segment of the records, or records that have not reached the terminal state.

Analyze Prediction Accuracy with the Einstein Accuracy Analytics App

The Einstein Accuracy Analytics app helps you monitor how well your models predict actual outcomes over time.

View a Prediction Definition

You can view a prediction definition deployed in your org.

To view a prediction definition:

1. Open Model Manager.

2. In the prediction definitions list, click the name of the prediction definition you want to view, or click **View** in the dropdown.

> 📝 **Note:** When you open a prediction definition, its Prediction ID appears at the end of the URL in the browser. For example, if the URL is:
>
> ```
> https://xyz.lightning.force.com/wave/wave.app?tsid=0x0x0x0x0x0x0x0#goal/1OID00000004CAkOBX
> ```
>
> then the Prediction ID is `1OID00000004CAkOBX`.

If a prediction definition is being monitored for performance, you first see a performance metrics graph that shows summary statistics and model performance statistics over time.

Prediction performance compares the predicted value with the actual outcome for records that have reached a terminal state (for example, IsClosed == true for an Opportunity).

- For binary classification predictions, **Average Accuracy** shows the cumulative total accuracy for all models under the current prediction definition. Each data point on the chart represents the daily accuracy per model from when the automatic prediction was first enabled.

- For regression predictions, **Average RMSE** (Root Mean Squared Error) shows the average RMSE for all models under a prediction definition. RMSE computes the standard deviation of the difference between predicted and actual outcomes. Each datapoint on the chart represents the daily average RMSE per model from when the automatic prediction was first enabled.

Model Manager also displays the **Total # of Predictions** and any **Performance Warnings**.

To set up performance monitoring, see Configure Performance Monitoring for a Prediction Definition on page 1730.

3. To view prediction definition settings, click the **Settings** tab.

Column	Description
Prediction Field	Custom field for automated predictions. For instructions, see Configure the Prediction Field for a Prediction Definition on page 1729.
Performance Definition	Settings used to calculate model performance. For more information, see Configure Performance Monitoring for a Prediction Definition on page 1730.

Column	Description
Advanced	• **Prediction ID**: Unique, system-assigned ID for this prediction definition. • **Developer Name**: Name of the developer associated with this prediction definition.

Configure Automatic Model Refresh for a Prediction Definition

Use automatic refresh to schedule regular, unattended model updates on a weekly or monthly basis.

Over time, a deployed model's performance can drift. The model becomes less accurate in predicting outcomes due to changes in the business environment, data, or requirements. To combat drift, refresh your model periodically.

1. View a Prediction Definition on page 1723 in Model Manager.
2. Click the **Model Refresh** subtab.
3. Click **Enable Automatic Refresh**.

Model Manager

Prediction: **Predicted Amount** ✏

1 Model

Overview   **Model Refresh**   Settings

Automatic Model Refresh   **Next scheduled refresh:** Not enabled   Enable Automatic Refresh
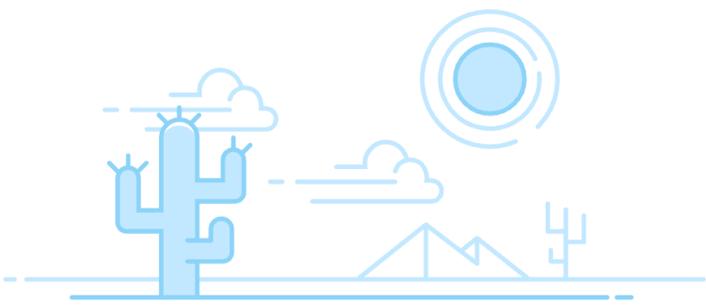
No history to show

Check back after the next scheduled refresh

4. Configure schedule settings.

Field	Description
Refresh Frequency	Monthly or Weekly.
Day of the Week / Month	Day of the week or month, depending on the refresh frequency.
Start Time	Start time for the refresh job.

5. Configure refresh settings.

Field	Description
Models to Refresh	Select each model you want to refresh.
Refresh Warning Threshold	Specify a threshold (percent) for a refresh warning.
Re-score records after refresh	Enable to re-score records after refresh.

6. Configure recipients of notification messages. Click **Edit Recipients**, add or update recipients in the subscribers list, and close the modal when finished.

7. Click **Save**.
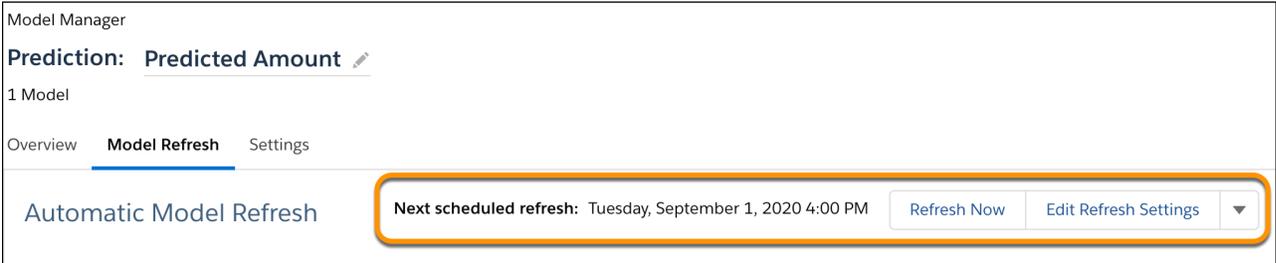
After you schedule a refresh job:

- To refresh the model on demand, click **Refresh Now** and click **Continue** to confirm that you want to run the refresh job.
- To disable a refresh job, click the dropdown and choose **Disable Automatic Refresh**.

Whenever the refresh job completes, subscribed recipients receive an email notification, and the Model Refresh page displays the job history, as shown in the following example.
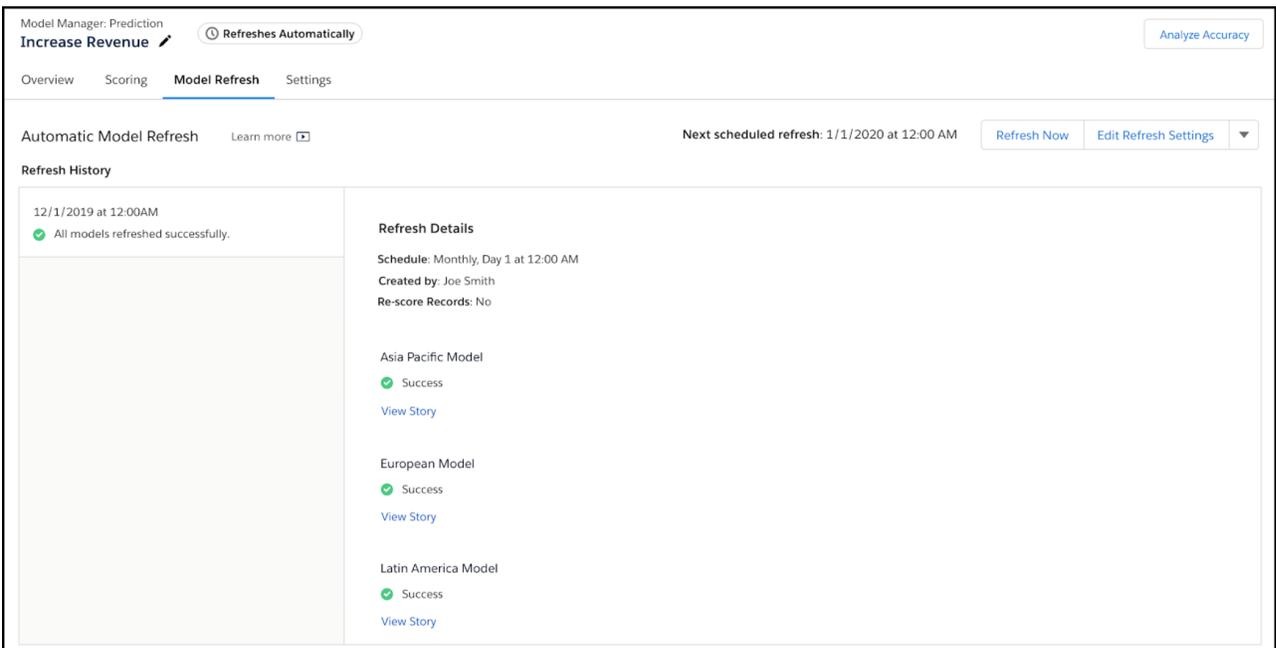
Configure the Prediction Field for a Prediction Definition

You can write prediction scores automatically to selected Salesforce fields. Writing prediction scores requires that you deploy a model for which you can map all model fields back to Salesforce fields. You can use an existing AI Prediction field or create one (and specify the label).

📝 Note: These settings can also be configured during model deployment. See Configure the Prediction Field on page 1711.

To configure the Prediction Field in a prediction definition:

1. View a Prediction Definition on page 1723 in Model Manager.

2. Click the **Settings** tab.

3. Under **Prediction Field**, select one of the following options.

Metric	Description
No prediction field (automatic predictions will not be enabled)	Do not store results in a prediction fields, or store results in a custom field that you configure later. Select this option if you do not use automated prediction fields. This kind of prediction is run programmatically via the Einstein Prediction Service on page 1754 using REST API calls or implementations based on the Einstein Discovery Managed Package described in Display Einstein Predictions Using Custom Fields (Deprecated) on page 1784.
Create a new prediction field from label	Have Einstein create the prediction field using the label you specify. Einstein stores prediction results in this prediction field.
Use an existing prediction field	Have Einstein store prediction results in an existing prediction field using the developer name you specify.

4. Click **Save Settings**.

Configure Performance Monitoring for a Prediction Definition

If you want Einstein Discovery to display performance monitoring charts for a prediction definition, you must explicitly configure performance monitoring.

To measure model performance, Einstein Discovery compares *predicted* outcomes with *actual* outcomes. An actual outcome represents data that has reached its *terminal state*, such as historical data that is not expected to change. For example, after an order ships, the ship

date and the number of items shipped are finalized. You define the conditions under which data has reached a terminal state so that Einstein Discovery knows which data to include in its performance analysis.

> Note: These settings can also be configured during model deployment. See Configure Model Performance Monitoring on page 1715.

To configure performance monitoring:

1. View a Prediction Definition on page 1723 in Model Manager.

2. Click the **Settings** tab.

3. Under **Performance Definition**, select an **Outcome Field** to use for determining terminal state.

4. Select **Yes, define conditions for a terminal state**.

5. Specify one or more conditions.

- For each condition, select a field, select an operator (**Equals** or **Not Equals**), and set a value.

  > 📝 Note:
  >
  > – Filter expressions are case sensitive. The value you specify must be an exact match. For example, "software" and "Software" are different matches. If your data contains variations in capitalization, consider cleansing your data during data prep to standardize patterns of uppercase and lowercase characters. For instructions on preparing data and data integration options, see Integrate Data into Tableau CRM Datasets on page 597.
  >
  > – Einstein displays object fields to which you have access. If you don't see an object field you expect, check the field-level security settings for your user account to ensure that you have at least read access to that field in the Salesforce object.

- To add multiple conditions, click **Add More**.
- To delete a condition, click the trash icon next to it.

6. Click **Save Settings**.

Score Records in Bulk

You can score predictions on multiple records at a time. For example, use bulk scoring to refresh all prediction scores after you deploy an updated model. You can also run bulk scoring on historical data to see how well your model performs. With bulk scoring, you can score all records, a segment of the records, or records that have not reached the terminal state.

To view a model:

1. View a Prediction Definition on page 1723 in Model Manager.

2. Click the **Scoring** tab.
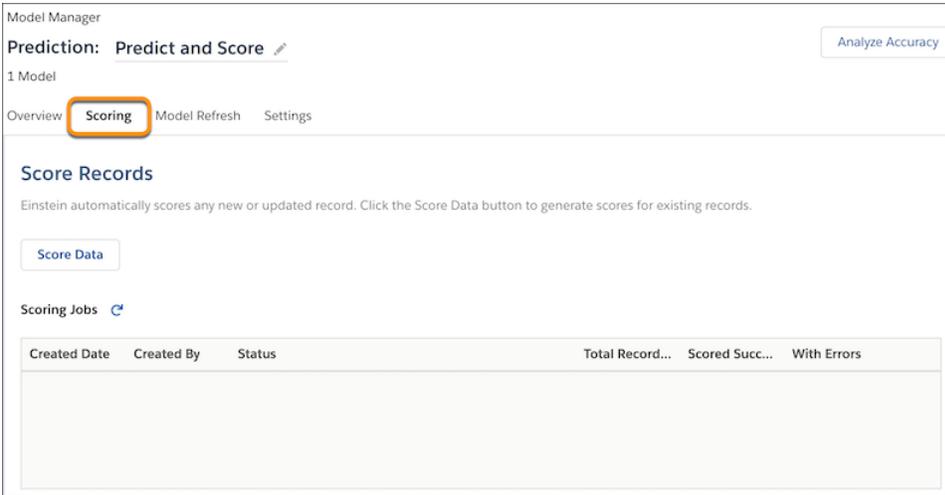
3. Click **Score Data**.
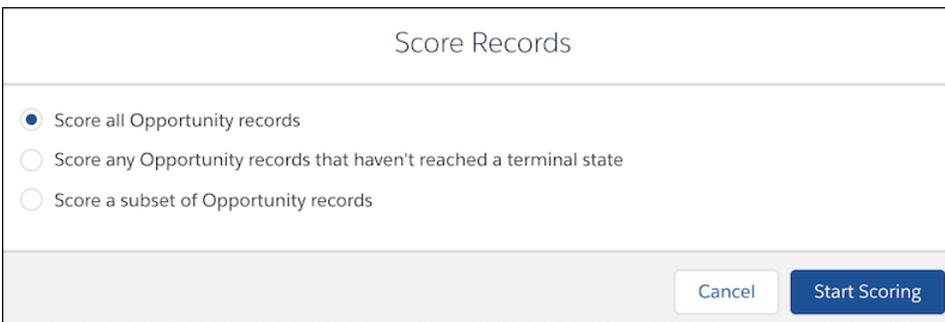
4. Select the scope of the records you want to score.

Column	Description
Score all *ObjectName* records	Score all records.
Score any *ObjectName* records that haven't reached a terminal state	Score any records that have not yet reached a terminal state.
Score a subset of *ObjectName* records	Select a subset of records (segment). Specify one or more filter conditions for the records you want score.
	✎ Note: • Filter expressions are case sensitive. The value you specify must be an exact match. For example, "software" and "Software" are different matches. If your data contains variations in capitalization, consider cleansing your data during data prep to standardize patterns of uppercase and lowercase characters. For instructions on preparing data and data integration

Column	Description
	options, see Integrate Data into Tableau CRM Datasets on page 597.
	• Einstein displays object fields to which you have access. If you don't see an object field you expect, check the field-level security settings for your user account to ensure that you have at least read access to that field in the Salesforce object.

5. Click **Start Scoring**.

6. To verify that you want to proceed, click **Continue**.

   Model Manager creates a new scoring job and displays scoring progress.

When finished, Model Manager displays the results of the scoring job.

Column	Description
Created Date	Date when the scoring job was created.
Created By	User who created the scoring job.
Status	Status of the scoring job.
Total Records Selected	Number of records selected for scoring.
Scored Successfully	Number of records that were scored successfully.
With Errors	Number of records with scoring errors.

> **Note:** If the daily predictions limit is reached in your org, active scoring jobs are paused, then resumed the next day.

To manage scoring jobs:

- Click **Cancel** to cancel a scoring job in progress.
- Click **Run Again** to run a scoring job again.
- Click **Delete** to delete a scoring job.

Analyze Prediction Accuracy with the Einstein Accuracy Analytics App

The Einstein Accuracy Analytics app helps you monitor how well your models predict actual outcomes over time.

Use insights from the Accuracy Analytics app to fine-tune your models and produce better predictions. Set up the app in the Model Manager, then run it to view a dashboard with accuracy metrics and a trend chart. They help you assess the accuracy of any prediction configured for performance monitoring. You can monitor prediction accuracy for both logistic and linear regressions.

Create the Accuracy Analytics App for a Prediction Definition

To analyze accuracy for a prediction, you first set up Accuracy Analytics, which you use to analyze and display prediction accuracy.

View the Accuracy Analytics App for a Prediction Definition

After creating Accuracy Analytics, view its dashboard to see prediction accuracy metrics and a chart.

Create the Accuracy Analytics App for a Prediction Definition

To analyze accuracy for a prediction, you first set up Accuracy Analytics, which you use to analyze and display prediction accuracy.

Prerequisites for Models and Prediction Definitions

Before you set up Accuracy Analytics:

- Create the story using the **Insights & Predictions** story type. For instructions, see Select the Story Type on page 1626.
- Deploy the model to a prediction definition. For instructions, see Deploy Models on page 1706.
- Deploy the model using automated prediction fields. For instructions, see Configure the Prediction Field on page 1711.

  > **Note:** All fields in the model must be mapped to fields in the Salesforce object. Mappings must be to the primary object (no mapping to features in child objects).

- Performance monitoring must be configured for the prediction definition. For instructions, see Configure Model Performance Monitoring on page 1715 to configure the terminal state for your data.
- The prediction definition must have at least one active model. It can have multiple models. Scoring occurs when records in the associated Salesforce object are created and edited. For each prediction definition, only one model is used to score a record. For instructions, see Change the Model Evaluation Order in a Prediction Definition on page 1747.

Set Up the Accuracy Analytics App

1. View a Prediction Definition on page 1723 in Model Manager.
   If your prediction definition meets the prerequisites, you see the **Analyze Accuracy** button in the upper right corner of your screen. Click this button to create an app or open an existing app.

2. Click **Analyze Accuracy**.
   The Model Manager prompts you with details about the model:

Field	Description
Deployed Object	Salesforce object associated with this prediction definition.
Outcome Field	Outcome field for the model.
Predicted Field	Prediction field used to store the prediction value.
Prediction Model Type	One of the following values: • **Linear** for numeric fields • **Logistic** for text fields (classification use case)

3. Click **Continue**.

The Model Manager displays progress while creating the Accuracy Analytics app and its associated dataset and data flow.
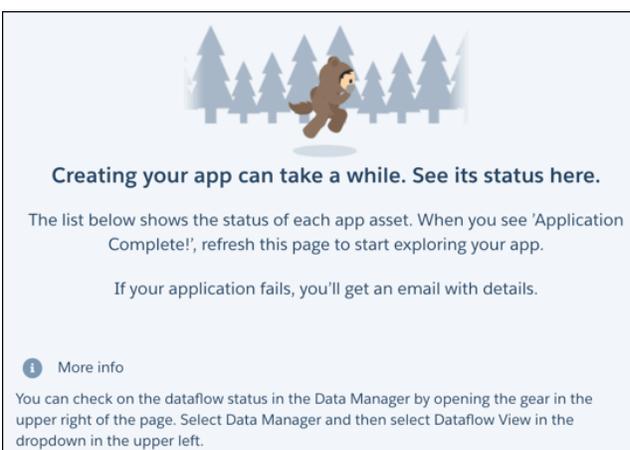
When finished, the Model Manager displays Accuracy Analytics with a dashboard, dataset, and dataflow.

View the Accuracy Analytics App for a Prediction Definition

After creating Accuracy Analytics, view its dashboard to see prediction accuracy metrics and a chart.

Update Data in the Dataset (Optional)

When you create Accuracy Analytics, its dataflow populates the dataset with a snapshot of the predictions for the Salesforce object. To get the latest data for your dashboard, refresh the data in the dashboard's dataset by running the associated dataflow. You can run the dataflow in several ways:

- In the Data Manager, run the dataflow manually. For instructions, see Run a Dataflow Manually on page 977.
- In the Data Manager, schedule the dataflow to run periodically. For instructions, see Schedule a Dataflow to Run Automatically on page 978.
- In the Model Manager, reconfigure the app and save changes, which triggers the dataflow to run.

View Prediction Accuracy

To open an Analyze Accuracy App for a prediction definition:

1. View a Prediction Definition on page 1723 in Model Manager.
   If your prediction definition meets the prerequisites, you see the **Analyze Accuracy** button in the upper right corner of your screen. Click this button to open an existing app.

2. Click **Analyze Accuracy**.
   If found, the app opens.
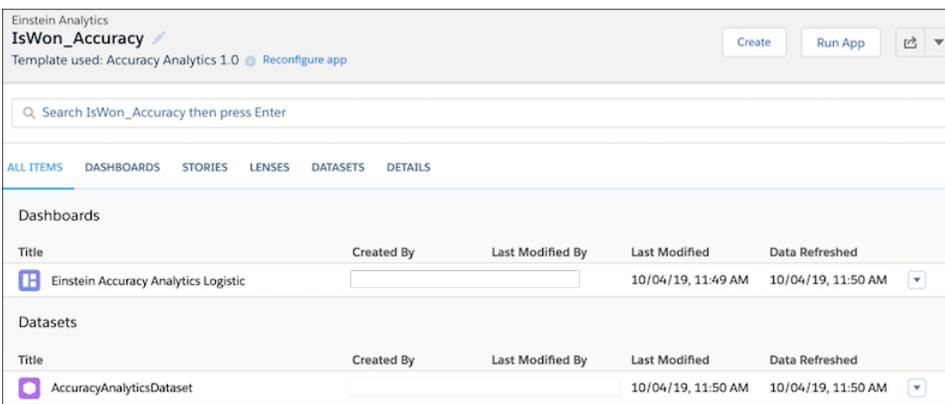
3. Click **Run App**.
   Einstein runs the App and displays its dashboard.

   Dashboard for Logistic Regression
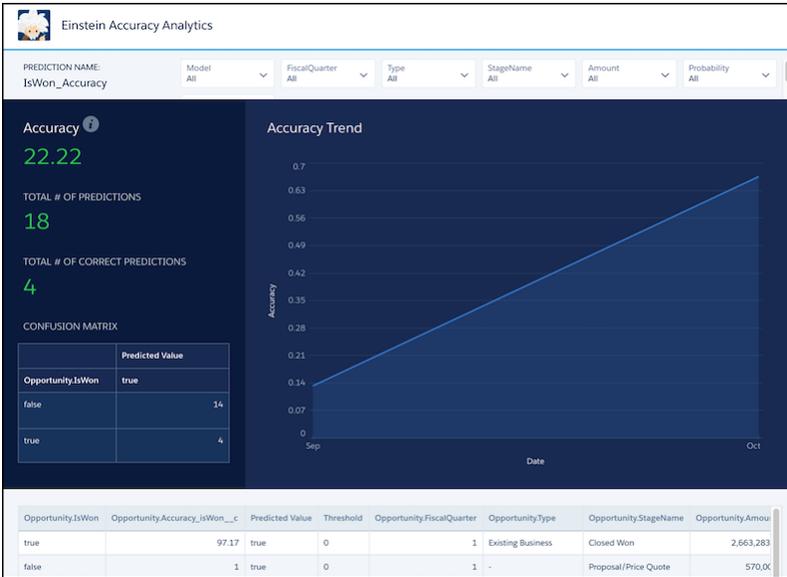
Area	Description
Accuracy	Accuracy of this prediction definition. Number of correct predictions divided by the total number of predictions. Excludes nulls in the predicted or actual field.
Total # of Predictions	Number of all predictions made.
Total # of Correct Predictions	Number of predictions that were correct.
Confusion Matrix	Used to evaluate the trade-offs between different error types based on the threshold value. It displays how many times the model correctly and incorrectly classifies a record as true and false at the associated threshold.
Details Table	Each row represents a scored record (such as an Opportunity) with the predicted value, actual value, and other details.
Accuracy Trend Chart	Shows prediction accuracy (x-axis) over time (y-axis).

Dashboard for Linear Regression
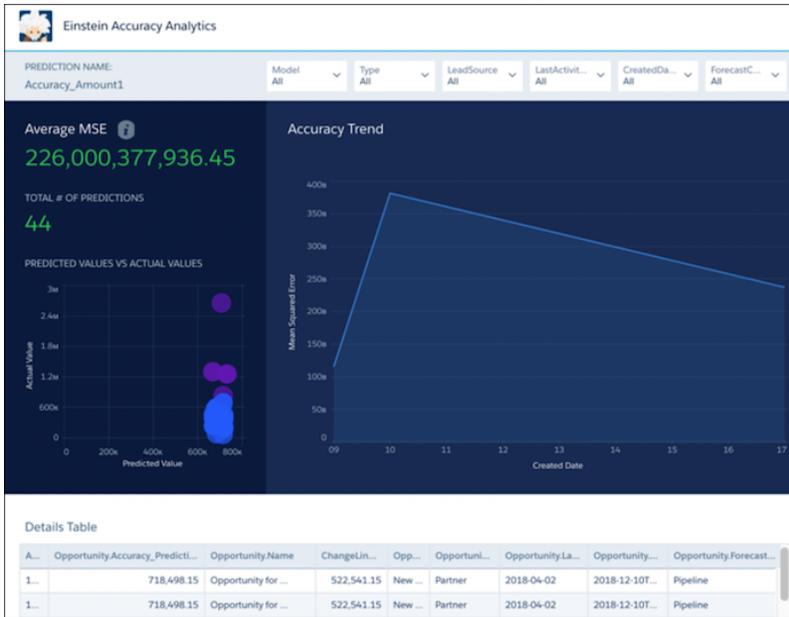
Area	Description
Average MSE	Measures the average squared error of the prediction definition's predictions. Computes the square difference between the observed (actual) outcome and the predicted values, and then averages them.
Total # of Predictions	Number of all predictions made.
Predicted Values vs Actual Values	Residuals plot chart that reveals the robustness of your model. A residual represents the difference between the actual value and the model's predicted value.
Details Table	Each row represents a scored record (such as an Opportunity) with the predicted value, actual value, and other details.
Accuracy Trend Chart	Shows prediction average MSE (x-axis) over time (y-axis).

4. If you want, drill down into the data by filtering columns.

Configure Model Performance Alerts

Models drift when characteristics in the real world data diverge significantly from the training data used to build them. Einstein can send out alert notifications if your model performance falls below configured thresholds. Significant increases in prediction error rates, missing values, or out-of-bounds values can trigger notifications. You can set alert thresholds, schedule weekly or monthly performance evaluation cycles, and specify who should receive notifications.

To configure performance alerts:

1. In Model Manager, view a prediction definition that uses automated prediction fields.
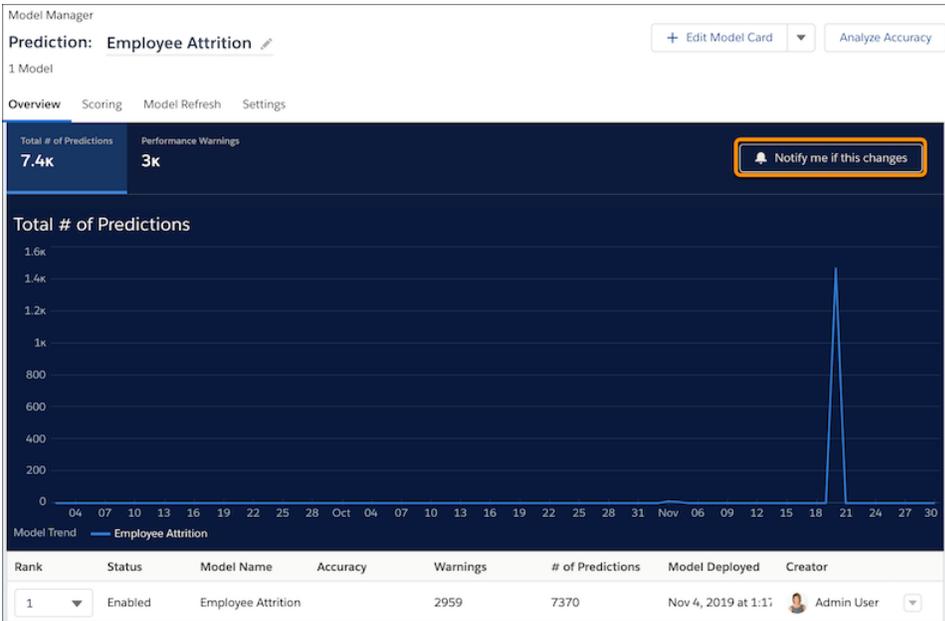
2. Click **Notify me if this changes**.

3. Enable and configure any of the alert conditions.

Condition	Alert Threshold
Prediction error rate increases	📝 Note: This option requires that your model's terminal state is configured to enable accuracy monitoring. Trigger an alert when the Root Mean Squared Error (RMSE) in the live data increases the selected percentage in the training data.
Too many missing values	Trigger an alert when the number of records with missing values exceeds the selected percentage.
Too many out-of-bounds values	Trigger an alert when the number of records with out-of-bounds values exceeds the selected percentage. An out-of-bounds value is a value not found in the training data used to train the model.
Notify recipients only the first time this condition is reached	Enable if you don't want subsequent notifications.

4. For **When should model performance be evaluated?**, configure how frequently to evaluate the model: **Weekly** or **Monthly**, and day of the week or month, and the start time.

5. For **Who should be notified?**, click **Edit Recipients** and specify the email addresses of notification recipients.

You can add users and groups.

- To add a group, select **Groups**, click in the **Group name** box, select a group, and click **Add**.

- To add a user, select **Users**, click in the **User name** box, select a user, and click **Add**.

To close this modal, click the **X** in the upper right corner.

6. Click **Save**.

Manage Models

Use the Model Manager to manage the models associated with your prediction definitions.

View a Model
You can view a model deployed in your org.

You can update model settings that were previously set during model deployment.

A model can undergo many updates over the course of its lifecycle. Factors such as altered business circumstances, new and better data, and improved story or model settings can motivate the creation and deployment of a new model version. To help you keep track of all these versions, examine a model's version history so that you know exactly when it was updated and by whom, and whether it is scheduled for an upcoming refresh job. For models that aren't performing as expected, you can revert to a previous version that performs better. To investigate the underlying settings associated with a particular model version, you can easily retrieve and examine the story version on which it's based.

Deployed models are enabled by default. You can disable a model if you want to hide it from Salesforce but still keep it for future use. A disabled model no longer provides scores to Salesforce objects. You can later enable it as needed.

You can change the order in which models are evaluated in a prediction definition.

You can view the story associated with a model.

You can delete a model you no longer need.

View a Model

You can view a model deployed in your org.

1. on page 1723 in Model Manager.
   The **Overview** tab displays one or more models associated with the selected prediction definition.

Each row represents a separately deployed model. For example, under the same prediction definition, you can see different models for different segments. One segment can target won opportunities while another can target lost opportunities.

Column	Description
Rank	Order in which models are executed under a prediction, with 1 being the first to execute. Only one model is used to score against a record. If a prediction has multiple models, then the model in the sequence with the first matching filter is used to score the record.
Status	Indicates whether the model is active (**Enabled**) or not (**Disabled**).

Column	Description
Model Name	Name of the model.
Accuracy	Logistic models only. Cumulative total accuracy for this model.
RMSE	Linear models only. RMSE (Root Mean Squared Error) for this model, which is the same as the square root of the models MSE (Mean Squared Error). RMSE measures the difference between the values predicted by the model and the observed (actual) values. You can think of this value as the "standard deviation of errors."
Warnings	Number of warnings associated with the model. Warnings occur when column mappings are missing or if data is out of bounds (not found in the training set).
# of Predictions	Total number of predictions made by this model within the org.
Model Deployed	Date and time when the model was deployed.
Creator	User who created the model.

2. To view details for a particular model, choose **View** from the dropdown.

- **Performance** tab: Displays model performance when performance is configured for the model.
- **Settings** tab: Displays the model threshold, actionable variables, field mappings, and segmentation filters.

Edit Model Settings

You can update model settings that were previously set during model deployment.

To edit model settings:

1. View a Prediction Definition on page 1723 in Model Manager.

2. Choose **Edit Settings** from the dropdown associated with the model you want to edit.

3. On the **Model** subtab, change the settings you want.

Column	Description
Model Name	Change the model name.
Threshold	For the classification use case, you can update the threshold value for a model. The threshold value tells your model how to classify a binary outcome. To learn more, see Threshold Evaluation Tab for Classification Use Cases on page 1692.
Want to focus on a particular segment in your dataset?	Choose whether to use the model to get predictions on all data, or on just a segment (subset) of the data. To learn more, see Configure Segmentation Filters on page 1712.
Confirm Field Mapping	If applicable, verify and change the mapping between the model, Salesforce object field, and dataset column. To learn more, see Map Dataset Fields to Fields in the Salesforce Object on page 1709.

4. Click the **Improvements** tab to enable improvements for one or more variables for your model. You can also specify custom text for the suggested improvement that Einstein Discovery displays to users. To learn more, see Configure Improvements on page 1713

5. Click **Save Changes**.

View and Manage a Model's Update History

A model can undergo many updates over the course of its lifecycle. Factors such as altered business circumstances, new and better data, and improved story or model settings can motivate the creation and deployment of a new model version. To help you keep track of all these versions, examine a model's version history so that you know exactly when it was updated and by whom, and whether it is scheduled for an upcoming refresh job. For models that aren't performing as expected, you can revert to a previous version that performs better. To investigate the underlying settings associated with a particular model version, you can easily retrieve and examine the story version on which it's based.

1. View a Prediction Definition on page 1723 in Model Manager.

2. View a model.

3. In the side panel, click the **History** tab.
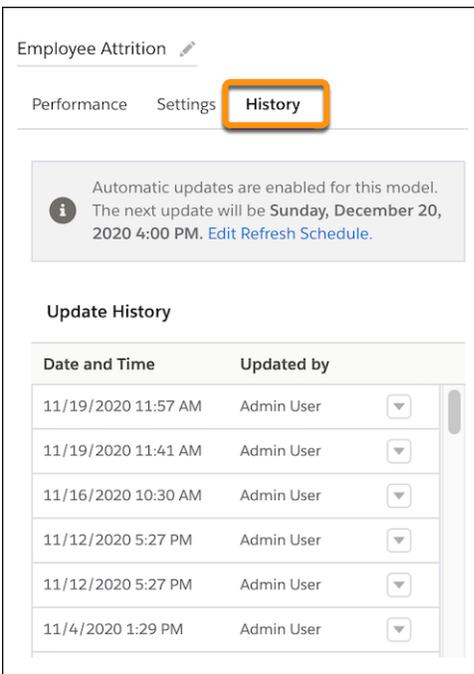
4. To view the story version associated with this model version, in the **Update History** list, click the menu and choose **View Story Version**.

5. To reset the model to a different version, in the **Update History** list, click the menu and choose **Reset Model To here**.

Enable or Disable a Model

Deployed models are enabled by default. You can disable a model if you want to hide it from Salesforce but still keep it for future use. A disabled model no longer provides scores to Salesforce objects. You can later enable it as needed.

1. View a Prediction Definition on page 1723 in Model Manager.

2. Choose **Disable** or **Enable** from the dropdown associated with the model you want to change.

Change the Model Evaluation Order in a Prediction Definition

You can change the order in which models are evaluated in a prediction definition.

When a record is modified and saved in a Salesforce object, Einstein checks the filter conditions in each model, starting with first model in this list. When Einstein finds a filter that matches the record, it applies that model to the record and stops looking at subsequent models. Only the first filter that matches the model is applied.

To change to evaluation order of the models in the prediction definition, select numbers in the **Rank** column.

View the Story Associated with a Model

You can view the story associated with a model.

1. View a Prediction Definition on page 1723 in Model Manager.

2. Click the dropdown associated with the model and choose **View Story**.

   Einstein Discovery opens the story.

Delete a Model

You can delete a model you no longer need.

> ✏️ **Note:** You cannot undelete a deleted model.

1. View a Prediction Definition on page 1723 in Model Manager.

2. Click the dropdown associated with the model and choose **Delete**.

   The model is removed from the list.

Package and Distribute Your Prediction Definitions and Models

Einstein Discovery supports packaging for prediction definitions and models. You can promote prediction definitions from a sandbox to a production environment. You can distribute prediction definitions using managed or unmanaged packages.

Promote Prediction Definitions and Models To Production

Build and test your models in a sandbox environment and, when it's production-ready, promote it to your production environment using change sets.

Distribute Prediction Definitions and Models With Managed and Unmanaged Packages

You can build and train models using your own data, then distribute them to others—within or outside your company—using managed or unmanaged packages. That way, customers enjoy the benefits of your models' predictions and improvements without needing the data to train the models themselves. Einstein Discovery supports both first-generation packaging (1GP) and second-generation packaging (2GP).

Promote Prediction Definitions and Models To Production

Build and test your models in a sandbox environment and, when it's production-ready, promote it to your production environment using change sets.

1. From Setup, in the Quick Find box, enter `change set`, and select **Outbound Change Sets**.

2. To create an outbound change set, click **New**, enter a name and description, then click **Save**.

3. To add the prediction to the change set, next to **Change Set Components**, click **Add**.
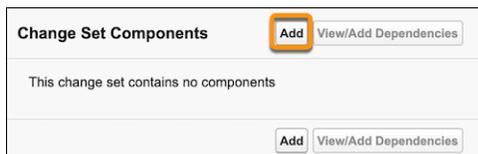
4. Select the type of component you want to deploy—either **Discovery Prediction** (a prediction definition that contains one or more models) or **Discovery Model** (an individual model).
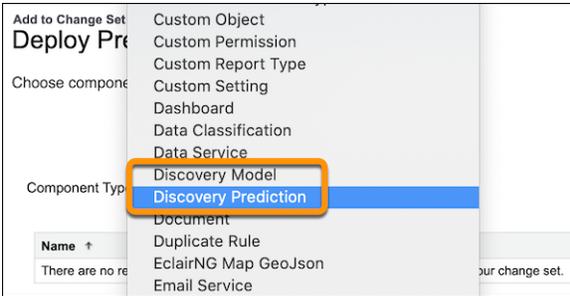
5. Select one or more deployed prediction definitions or models from the list, and then click **Add to Change Set**.
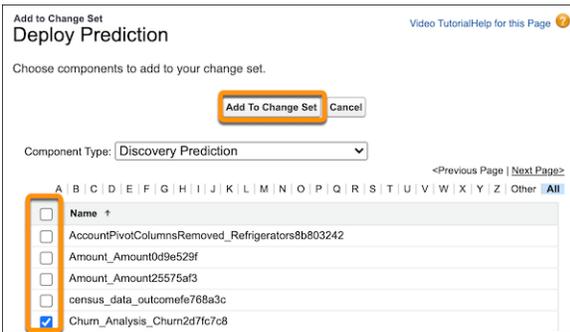
6. Add any models on which the prediction definition depends by clicking **View/Add Dependencies**, selecting any available models, and then clicking **Add to Change Set**.
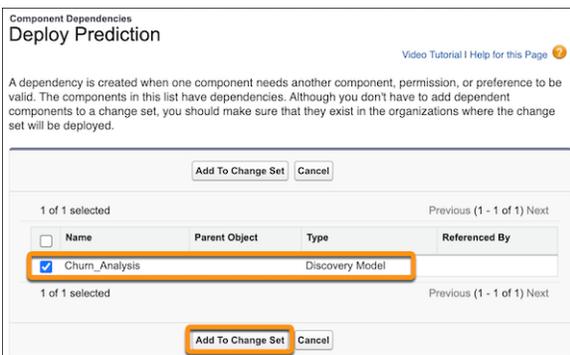
Now the prediction definition and any associated models appear in the components list.
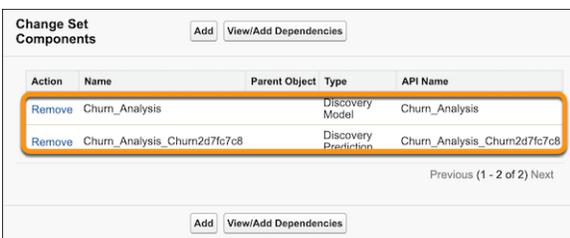
> **Note:** The change set includes the model but not the story used to create the model. A model's originating story version is not uploaded with the change set.

7. Upload the change list to the target production environment. Then go to that environment, navigate to **Inbound Change Sets**, find the change set you uploaded, and then deploy it just as for any other change set.
To learn more, see Change Sets in Salesforce help.

Distribute Prediction Definitions and Models With Managed and Unmanaged Packages

You can build and train models using your own data, then distribute them to others—within or outside your company—using managed or unmanaged packages. That way, customers enjoy the benefits of your models' predictions and improvements without needing the data to train the models themselves. Einstein Discovery supports both first-generation packaging (1GP) and second-generation packaging (2GP).

To add prediction definitions and models to a package:

1. In Package Manager, create a package.

2. In the **Components** tab, click **Add**.

3. In the **Component Type** list, select **Discovery Prediction** or **Discovery Model**.
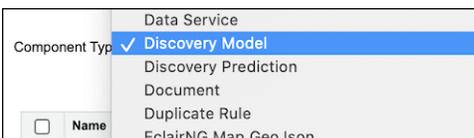
4. Select one or more prediction definitions or models from the list.

5. Click **Add Component**

To learn more about managing packages, see the following articles in Salesforce help:

- Work with Packages
- Distribute Your Apps

Predict Outcomes

After a model is deployed to Salesforce, Einstein Discovery can use it to predict outcomes and provide recommendations in different ways. You can display predictions in Lightning record pages and Experience Cloud sites pages. You can get predictions in Tableau CRM data flows. You can retrieve predictions programmatically using the Einstein Prediction Service via REST or Apex API calls.

Display Einstein Predictions Using Automated Prediction Fields
Prediction fields are a special type of custom field that only Einstein can write to. When automated prediction fields are selected during model deployment, Einstein Discovery can save predictions directly to Salesforce records for use in reports and list views. Only the prediction is written, not the reasons for the prediction or the recommended ways to improve the prediction.

Add Einstein Predictions to a Lightning Page
After a model is deployed, you can use it to display predictions and recommendations for a standard or custom object on a Lightning Experience record page.

Einstein Prediction Service
After deploying models with Einstein Discovery, use the Einstein Prediction Service API to embed your predictions into any website or application.

Get Predictions in Apex

Use the `ConnectApi.SmartDataDiscovery.predict` API to get predictions in your Apex code.

Get Predictions in Process Automation Formulas

Use the Einstein Discovery PREDICT function to get predictions in your process automation formulas. Your automation logic can make decisions based on predicted outcomes from deployed Einstein Discovery predictions. For example, in an approval process, a formula can determine whether a predicted outcome meets a threshold required for automatic approval.

Display Einstein Predictions Using Custom Fields (Deprecated)

If you want more programmatic control over predictions than what you get with automated prediction fields, you can install the Einstein Discovery Managed Package and implement a solution based on custom fields in a Salesforce object.

SEE ALSO:

Discovery Predict Transformation: Get Einstein Discovery Predictions

prediction Transformation

Einstein Predictions

Display Einstein Predictions Using Automated Prediction Fields

Prediction fields are a special type of custom field that only Einstein can write to. When automated prediction fields are selected during model deployment, Einstein Discovery can save predictions directly to Salesforce records for use in reports and list views. Only the prediction is written, not the reasons for the prediction or the recommended ways to improve the prediction.

A prediction field works just like other custom number fields: you can choose which profiles have read access and which page layouts to include the field. In addition, you can add this field to any list view and to any report. The only requirement is that you configure row-level security for prediction fields.

Deploy the Model

After creating your Einstein Discovery story and validating that the model metrics look good, it's time to deploy the model. Complete the steps in Deploy Models on page 1706 according to the following requirements:

Screen in Deployment Wizard	Description
New prediction or add to existing	Required. If you choose **Deploy model as a new prediction**, you must choose a Salesforce object.
Map Fields	Required. *All* fields must be mapped from the model to corresponding fields in the Salesforce object. Field mapping is supported for any field on that object for which the prediction definition was deployed, or for any parent object.
Config Prediction Field	Required. Select one of the following options (required): • **Create a new prediction field from label** • **Use an existing prediction field** 📝 Note: These options require that *all* model fields are mapped to fields in the selected Salesforce object.
Add Filters	Optional. Use as needed.

Screen in Deployment Wizard	Description
Add Variables	Optional Use as needed.
Define Terminal State	Optional Use as needed.

After deployment, Einstein scores any newly created or update records in the Salesforce object and writes the score to the prediction field.

Assign Field-Level Security to the Prediction Field (Required)

Before you can see this prediction field in Reports, List Views or Page Layouts, you must assign field-level security just like you do for other custom fields. For instructions, see Manage Fields for a Specific Object

> 📝 **Note:** Prediction fields are *always* read only.

Assign Predictions to List Views and Custom Report Types

After you have set field-level security, you can add this field to existing custom report types and list views just like you do for other custom fields.

> 📝 **Note:** If you plan to add this field to a page layout, updating this field is an asynchronous process. You may not see immediate results after editing a record.

View Reasons and Improvements Associated with Predictions

Only the prediction value (or score) is written to the prediction field. If you also want to display the prediction reason and recommended improvements, add the Einstein Prediction Lightning component on a Lightning record page. For instructions, see Add Einstein Predictions to a Lightning Page on page 1752.

> 📝 **Note:** When opening a record, predictions are real-time queries. However, the results are not written anywhere.

Add Einstein Predictions to a Lightning Page

After a model is deployed, you can use it to display predictions and recommendations for a standard or custom object on a Lightning Experience record page.

After a model is deployed, you can use it to display predictions and recommendations for a standard or custom object on a Lightning Experience record page.

1. In the Lightning App Builder, open the record page to which you want to add a prediction.

2. Drop the Einstein Predictions component where you want it on the Lightning record page.

3. In the component configuration panel, use the **Prediction** lookup to find and select the prediction you want to embed in the community page. The Lightning App Builder displays a list of deployed models that match the object associated with this Lightning record page.

4. Optionally, configure other settings as well.

USER PERMISSIONS

To add the Einstein Predictions component to a Lightning page
- Customize Application

To view predictions on a page
- View Einstein Discovery Recommendations

Setting	Description
Prediction	Search for a list of any predictions (deployed models) to which you have access.

Setting	Description
Title	Descriptive label for the prediction.
Show title	Select this checkbox to show the title.
Prediction score unit	Unit of measure for the prediction. Examples: • Currency for money-based predictions • % for probability-based predictions • time interval for time-based predictions (hours, days, weeks, and so on)
Unit precedes score	Select this checkbox to show units preceding the prediction, such as currency symbols. By default, units follow the prediction.
Positive prediction label	Label to display when the prediction is higher than the threshold in a binary classification model. Examples: Win, Retain, Success, and so on.
Negative prediction label	Label to display when the prediction is lower than the threshold in a binary classification model. Examples: Loss, Churn, Fail, and so on.
Collapse details	Select this checkbox to show top predictors but hide other details.
Show recommendations	Select this checkbox to show Einstein Discovery recommendations on how to improve the predicted outcome.
Number of recommendations	Set the maximum number of recommendations returned on the prediction card.
Recommendation threshold percentage	Specify a number to display only recommendations that impact the predicted outcome by this percentage or higher.
Show values for predictors	Select this checkbox to show the impact value next to predictors and recommendations.
Show prediction warnings	Select this checkbox to show user warnings, such as: • missing fields that are required for the model • field values that are outside of the valid range used to build the predictive model
Set Component Visibility	Configure any component visibility rules, if needed, by defining filter criteria for a device, record field, or other field.

When a user views this record page, predictions are updated in real time. No writeback to Salesforce is needed.

Einstein Prediction Service

After deploying models with Einstein Discovery, use the Einstein Prediction Service API to embed your predictions into any website or application.

About the Einstein Prediction Service

After deploying models with Einstein Discovery, use the Einstein Prediction Service REST API to embed your predictions into any website or application.

Get Predictions

The Einstein Prediction Service provides a REST API endpoint to request a prediction.

Manage Prediction Definitions

The Einstein Prediction Service provides REST API endpoints to manage prediction definitions. A prediction definition specifies what the model is trying to predict and the Salesforce entity associated with the prediction. Each prediction definition has a unique id. Only certain attributes of a prediction definition can be modified.

Manage Models

The Einstein Prediction Service provides REST API endpoints to manage models. Each model has a unique id. A model is used to evaluate data and return a prediction and recommendations. These REST endpoints allow you to make updates to model metadata, but not update the actual predictive model.

Manage Prediction Jobs

The Einstein Prediction Service provides REST API endpoints to run bulk scoring jobs for a prediction. Bulk scoring jobs enable you to score predictions on multiple records at a time. For example, after deploying an updated model, use bulk scoring to refresh all prediction scores. You can also run bulk scoring on historical data to see how well your model performs. With bulk scoring, you can score all records, a segment of the records, or records that have not reached the terminal state.

Manage Model Refresh Jobs

The Einstein Prediction Service provides REST API endpoints to retrieve metadata for model refresh jobs.

About the Einstein Prediction Service

After deploying models with Einstein Discovery, use the Einstein Prediction Service REST API to embed your predictions into any website or application.

Programmatically Interact with Einstein Predictions and Models

The Einstein Prediction Service API enables you to programmatically run predictions, manage prediction definitions, and manage models.

Prerequisites

Before you use the Einstein Prediction Service:

- Take the Einstein Prediction Service Trailhead module.
- Deploy the model you want to use for predictions. For instructions, see Deploy a Model on page 1706.
- Use your favorite REST client. If you want, use Workbench to test the API. To use Workbench, set up access to it in your org. You can log in to a hosted instance of Workbench at https://workbench.developerforce.com/login.php. For more information, see the Workbench documentation at https://github.com/forceworkbench/forceworkbench/wiki.

  Note: Workbench is a free, open source, community-supported tool (see the Help page in Workbench). Salesforce provides a hosted instance of Workbench for demonstration purposes only. Salesforce recommends that you do not use this hosted

instance of Workbench to access data in a production database. If you want to use Workbench for your production database, you can download, host, and configure it using your own resources. You can download Workbench from https://github.com/forceworkbench/forceworkbench.

- If you're not using Workbench, you need to have a connected app that handles Salesforce authentication for your REST client. To learn more, take the Connected App Basics Trailhead module.

Running Einstein Prediction Service API Commands

If you are using Workbench for your REST client:

1. Select API Version 50.0 (supported for this release).

2. Agree to the terms of service.

3. Log in.

4. Jump to REST Explorer.

Now you are ready to run API commands. The base URI is **/services/data/v50.0**. Simply select the method (GET, POST, and so on), append your endpoint to the base URI, specify the request body as needed, and then click **Execute**.

Get Predictions

The Einstein Prediction Service provides a REST API endpoint to request a prediction.

Prediction Request

```
POST /smartdatadiscovery/predict
```

POST Request Body

In the request body, you specify the prediction definition to use and the rows of data that you want to score. You can specify rows in one of three available formats:

Format	Description
Records	Salesforce record Ids associated with the subscribedEntity of the prediction definition (retrieved using a SOQL query).
RawData	A two-dimensional array of row values in which each row is a comma-separated list of values.
RecordOverrides	Array of objects containing the Salesforce record Ids. Optionally, override or append values in individual records with an array of row values (in which each row is a comma-separated list of values).

When you run a prediction, Salesforce applies the model specified in the prediction definition to the set of records and returns a prediction score for each record. If you specify 3 records, for example, you get 3 predictions in the order in which the records were specified in the request.

Request Body Attributes

Name	Type	Description
rows	object	Required when `type` is `RawData` or `RecordOverrides`. • `RawData`: Two-dimensional array of row values. • `RecordOverrides`: Array of objects containing the Salesforce record Ids. Optionally, override or append individual records with an array of row values.
type	String	Type of input data. Specifies how row data is represented in the request body. One of the following values: • `Records` Represent rows using Salesforce record Ids associated with the subscribedEntity of the prediction definition. • `RawData`: Represent rows within a two-dimensional array of row values. • `RecordOverrides`: Represent records using Salesforce record Ids. Optionally, override or append individual records with an array of row values.
records	object	Required when `type` is `Records`. Comma-separated list of Salesforce record Ids associated with the subscribedEntity of the prediction definition. Maximum: 200 records.

settings	object	Optional settings to control output. For example, you can specify the maximum number of prescriptions (improvements) to display:
		
		``
		"settings" : {
		    "maxPrescriptions" : 10 }
		``

columnNames	object	Required when `type` is `RawData` or `RecordOverrides`. Comma-separated list of column names representing the columns that the model analyzes. These columns were selected during

Name	Type	Description
		the story setup process. For more information, see Create a Story on page 1623.
predictionDefinition	String	Prediction definition Id. The Model Manager displays prediction definition Ids (see View a Prediction Definition on page 1723). You can also retrieve a list of available prediction definition Ids using the following API request:
| | | ```
GET /smartdatadiscovery
/predictiondefinitions
``` |

Sample Request when type is Records

The following example shows a request body in which `type` is `Records`. The records attribute provides a comma-separated list of Salesforce record Ids associated with the subscribedEntity of the prediction definition. You can specify up to 200 Salesforce record Ids. Two records are specified in the following example.

```
{
    "predictionDefinition": "1ORRM00000000304AA",
    "type": "Records",
    "records": ["006RM000002bEfiYAE", "006RM000002bEflYAE" ]
}
```

Sample Request when type is RawData

The following example shows a request body in which `type` is `RawData`. It names five columns and specifies two records with five data values each.

```
{
  "predictionDefinition": "0OR1H000000Gma9WAC",
  "type": "RawData",
  "columnNames": ["StageName","CloseDate","Account.BillingCountry","IsClosed","IsWon"],
  "rows": [
    ["Prospecting","2020-06-30","USA","false","false"],
    ["Qualification","2020-08-30","EMEA", "false","false"]
  ]
}
```

Sample Request when type is RecordOverrides

The following example shows a request body in which `type` is `RecordOverrides`. When specifying row data:

- Each Salesforce record Id represents a record in the subscribedEntity associated with the prediction definition.
- (Optional) Each row, where specified, contains the values to override or append to the data specified in the associated record.

You can specify up to 200 entries (record entries plus row entries) in a request. For example, if you have 120 record entries, you can override up to 80 record entries with row entries. The following example specifies two columns with two record entries and two overrides.

```
{
  "predictionDefinition": "0OR1H000000Gma9WAC",
  "type": "RecordOverrides",
  "columnNames": ["StageName", "CloseDate"],
  "rows": [
```

```
     {
         "record": "0061H00000dnhQEQAY",
          "row": ["Prospecting", "2020-06-30"]
     },

     {
          "record": "0061H00000dnhPzQAI",
          "row": ["Qualification", "2020-08-30"]
     }

  ]
}
```

Sample Request for Predictive Factors and Improvements

Starting in 50.0, this API returns a single prediction value by default. To request prediction factors and improvements, you must ask for them explicitly in the request body. The following code snippet specifies `settings` to request prediction factors and improvements.

```
{
    "predictionDefinition": "1ORB0000000TNYIOA4",
    "type": "Records",
    "records": ["006B0000002wvCtIAI"],
    "settings": {"maxPrescriptions": 3,
        "maxMiddleValues": 3,
        "prescriptionImpactPercentage": 87
    }
}
```

In this example:

- `maxPrescriptions` specifies the maximum number of improvements (3) to return in the response
- `maxMiddleValues` specifies the number of top predictors (3) to return in the response
- `prescriptionImpactPercentage` specifies the threshold filter (minimum % improvement for the outcome, which in this example is 87%) needed for the improvement to be returned in the response

To learn more about these settings and see these elements on an example Lightning page, see

POST Response

Example POST Response

```
{
  "predictionDefinition" : "1ORRM0000000030",
  "predictions" : [ {
    "model" : {
      "id" : "1OtRM000000002b0AA"
    },
    "prediction" : {
      "baseLine" : 799315.4282959097,
      "importWarnings" : {
        "mismatchedColumns" : [ ],
        "missingColumns" : [ "OpportunityAge", "Account.Owner.UniqueUserName",
"Account.Industry", "Account.AccountSource", "Account.Owner.Name", "Account.BillingCountry",
```

```
  "Name" ],
        "outOfBoundsColumns" : [ ]
      },
      "middleValues" : [ {
        "columns" : [ {
          "columnName" : "Has Line Item",
          "columnValue" : "true"
        } ],
        "value" : 543763.66105859
      } ],
      "other" : 0.0,
      "smallTermCount" : 0,
      "total" : 1343079.0893544997
    },
    "prescriptions" : [ ],
    "status" : "Success"
  }, {
    "model" : {
      "id" : "1OtRM000000002b0AA"
    },
    "prediction" : {
      "baseLine" : 799315.4282959097,
      "importWarnings" : {
        "mismatchedColumns" : [ ],
        "missingColumns" : [ "OpportunityAge", "Account.Owner.UniqueUserName",
"Account.Industry", "Account.AccountSource", "Account.Owner.Name", "Account.BillingCountry",
 "Name" ],
        "outOfBoundsColumns" : [ ]
      },
      "middleValues" : [ {
        "columns" : [ {
          "columnName" : "Has Line Item",
          "columnValue" : "true"
        } ],
        "value" : 543763.66105859
      } ],
      "other" : 0.0,
      "smallTermCount" : 0,
      "total" : 1343079.0893544997
    },
    "prescriptions" : [ ],
    "status" : "Success"
  } ],
  "settings" : {
    "maxPrescriptions" : 0,
    "maxMiddleValues" : 0,
    "prescriptionImpactPercentage" : 0
  }
}
```

POST Response Body Attributes

Name	Type	Description
predictionDefinition	String	Prediction Definition Id used to make the prediction.

Name	Type	Description
predictions	Array of PredictionItemRepresentation	Details for all predictions returned.
settings	PredictionSettingsRepresentation	Settings used for improvements.

PredictionItemRepresentation

Name	Type	Description
model	PredictionModelRepresentation	Details of the model that was used to make the prediction.
prediction	PredictionDetailRepresentation	Details of the prediction, including prediction value, predictive factors, and warnings.
prescription	Array of PrescriptionDetailRepresentation	Details of how actionable variables could be used to improve the prediction.
status	String	Success or Error

PredictionModelRepresentation

Name	Type	Description
id	String	Id of the model used to make the prediction.

PredictionDetailRepresentation

Name	Type	Description
baseline	number	Baseline of the prediction result.
importWarnings	PredictionWarningsRepresentation	Prediction warnings, if any. Example warnings: Out of Bounds, Missing Values.
middleValues	Array of PredictionFactorsRepresentation	Top predictive factors used to explain this prediction.
other	number	Subset of the total prediction that is explained using small terms.
smallTermCount	number	Count of small term predictive factors used to make this prediction.
total	number	Final prediction number. Calculated as the sum of `baseline` + `middleValues` + `other` (no need to manually calculate this).

PredictionWarningsRepresentation

Name	Type	Description
mismatchedColumns	Array of String	Deprecated. This property is always blank.
missingColumns	Array of String	List of column names that were required by the predictive model but were either not included in the prediction request, or the values were null.
outOfBoundsColumns	Array of PredictionFieldValueRepresentation	List of fields whose inputs fall outside of the boundaries of how the model was trained.

PredictionFieldValueRepresentation

Name	Type	Description
columnName	string	Name of the column passed into the prediction.
columnValue	string	String representation for the column value passed into the prediction.

PredictionFactorsRepresentation

Name	Type	Description
columns	array of PredictionFieldValueRepresentation	Column values(s) used that impact the prediction.
value	number	Impact of the column value(s) to the prediction score.

PrescriptionDetailRepresentation

Name	Type	Description
columns	array of PredictionFieldValueRepresentation	Column name and value that can be changed to improve the prediction. Although this is an array, only a single column name-value pair shows up here.
value	number	How much the total prediction would increase if the value was updated.

PredictionSettingsRepresentation

Name	Type	Description
maxPrescriptions	number	Maximum number of prescriptions that have been returned. If this is -1, then all prescriptions have been returned.
maxMiddleValues	number	Maximum number of top predictors that have been returned.

Name	Type	Description
prescriptionImpactPercentage	number	Impact threshold for which prescriptions have been returned. • If this is 0, then all prescriptions have been returned. • If this is 5, for example, then only prescriptions that will improve the prediction by at least 5% have been returned.

Manage Prediction Definitions

The Einstein Prediction Service provides REST API endpoints to manage prediction definitions. A prediction definition specifies what the model is trying to predict and the Salesforce entity associated with the prediction. Each prediction definition has a unique id. Only certain attributes of a prediction definition can be modified.

Get Available Prediction Definitions

```
GET /smartdatadiscovery/predictionDefinitions
```

The following code shows an example response:

```
HTTP/1.1 200 OK
Date: Tue, 15 Sep 2020 21:58:01 GMT
Strict-Transport-Security: max-age=31536000; includeSubDomains
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
X-Robots-Tag: none
X-B3-TraceId: d16d33c8d031db6e
X-B3-SpanId: d16d33c8d031db6e
X-B3-Sampled: 0
Cache-Control: no-cache,must-revalidate,max-age=0,no-store,private
Set-Cookie: BrowserId=hf3WNPeeEeqhuDcd5jEV6A; domain=.salesforce.com; path=/; expires=Wed,
 15-Sep-2021 21:58:01 GMT; Max-Age=31536000
Content-Type: application/json;charset=UTF-8
Vary: Accept-Encoding
Content-Encoding: gzip
Transfer-Encoding: chunked


{
  "nextPageUrl" : null,
  "predictionDefinitions" : [ {
    "countOfActiveModels" : 1,
    "countOfModels" : 1,
    "createdBy" : {
      "id" : "005B0000001nz1lIAA",
      "name" : "MyUserName",
     "profilePhotoUrl" : "https://c.gus.content.force.com/profilephoto/729B00000008pwV/T"

    },
    "createdDate" : "2020-04-22T01:25:19.000Z",
    "id" : "1ORB000000000bOOAQ",
    "label" : "Maximize CLV - first version",
```

```
    "lastModifiedBy" : {
      "id" : "005B0000001nz1lIAA",
      "name" : "MyUserName",
      "profilePhotoUrl" : "https://c.gus.content.force.com/profilephoto/729B00000008pwV/T"

    },
    "lastModifiedDate" : "2020-04-22T01:25:19.000Z",
    "modelsUrl" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB000000000bOOAQ/models",

    "name" : "Maximize_CLV_first_version7c48e8db",
    "outcome" : {
      "goal" : "Maximize",
      "label" : "CLV",
      "name" : "CLV"
    },
    "predictionType" : "Regression",
    "status" : "Enabled",
    "url" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB000000000bOOAQ"
  }, {
    "countOfActiveModels" : 1,
    "countOfModels" : 1,
    "createdBy" : {
      "id" : "005B0000001nz1lIAA",
      "name" : "MyUserName",
      "profilePhotoUrl" : "https://c.gus.content.force.com/profilephoto/729B00000008pwV/T"

    },
    "createdDate" : "2020-01-13T21:28:06.000Z",
    "id" : "1ORB00000004CAkOAM",
    "label" : "Won_Won",
    "lastModifiedBy" : {
      "id" : "005B0000001nz1lIAA",
      "name" : "MyUserName",
      "profilePhotoUrl" : "https://c.gus.content.force.com/profilephoto/729B00000008pwV/T"

    },
    "lastModifiedDate" : "2020-04-14T20:37:21.000Z",
    "modelsUrl" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB00000004CAkOAM/models",

    "name" : "Won_Wone329300b",
    "outcome" : {
      "goal" : "Maximize",
      "label" : "Won",
      "name" : "Won"
    },
    "predictionType" : "Classification",
    "status" : "Enabled",
    "url" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB00000004CAkOAM"
  }, {
    "countOfActiveModels" : 2,
```

```
      "countOfModels" : 2,
      "createdBy" : {
        "id" : "005B0000001nz1lIAA",
        "name" : "MyUserName",
        "profilePhotoUrl" : "https://c.gus.content.force.com/profilephoto/729B00000008pwV/T"

      },
      "createdDate" : "2020-01-17T00:24:33.000Z",
      "id" : "1ORB00000004CApOAM",
      "label" : "CLV Prediction",
      "lastModifiedBy" : {
        "id" : "005B0000001nz1lIAA",
        "name" : "MyUserName",
        "profilePhotoUrl" : "https://c.gus.content.force.com/profilephoto/729B00000008pwV/T"

      },
      "lastModifiedDate" : "2020-01-17T00:24:33.000Z",
      "modelsUrl" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB00000004CApOAM/models",

      "name" : "CLV_Prediction3d72828a",
      "outcome" : {
        "goal" : "Maximize",
        "label" : "CLV",
        "name" : "CLV"
      },
      "predictionType" : "Regression",
      "status" : "Enabled",
      "url" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB00000004CApOAM"
  }, {
      "countOfActiveModels" : 1,
      "countOfModels" : 1,
      "createdBy" : {
        "id" : "005B0000001nz1lIAA",
        "name" : "MyUserName",
        "profilePhotoUrl" : "https://c.gus.content.force.com/profilephoto/729B00000008pwV/T"

      },
      "createdDate" : "2020-02-04T08:28:14.000Z",
      "id" : "1ORB00000004CCCOA2",
      "label" : "CLV_CLV",
      "lastModifiedBy" : {
        "id" : "005B0000001nz1lIAA",
        "name" : "MyUserName",
        "profilePhotoUrl" : "https://c.gus.content.force.com/profilephoto/729B00000008pwV/T"

      },
      "lastModifiedDate" : "2020-02-04T08:28:14.000Z",
      "modelsUrl" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB00000004CCCOA2/models",

      "name" : "CLV_CLVa7cf508c",
      "outcome" : {
```

```
      "goal" : "Maximize",
      "label" : "CLV",
      "name" : "CLV"
    },
    "predictionType" : "Regression",
    "status" : "Enabled",
    "url" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB00000004CCCOA2"
  } ],
  "totalSize" : 4,
  "url" : "/services/data/v50.0/smartdatadiscovery/predictiondefinitions"
}
Requested in 0.431 sec
Workbench 50.0.0
```

Get Metadata for a Prediction Definition

```
GET /smartdatadiscovery/predictiondefinitions/{predictionDefinitionId}
```

The following code shows an example response.

```
HTTP/1.1 200 OK
Date: Thu, 17 Sep 2020 16:27:43 GMT
Strict-Transport-Security: max-age=31536002; includeSubDomains
Public-Key-Pins-Report-Only: pin-sha256="9n0izTnSRF+W4W4JTq51avSXkWhQB8duS2bxVLfzXsY=";
pin-sha256="5kJvNEMw0KjrCAu7eXY5HZdvyCS13BbA0VJG1RSP91w=";
pin-sha256="njN4rRG+22dNXAi+yb8e3UMypgzPUPHlv4+foULwl1g="; max-age=86400; includeSubDomains;
 report-uri="https://a.forcesslreports.com/hpkp-report/00DB0000000K2Uzm";
Expect-CT: max-age=86400,
report-uri="https://a.forcesslreports.com/Expect-CT-report/00DB0000000K2Uzm"
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
X-Robots-Tag: none
X-B3-TraceId: da9949518cdecb56
X-B3-SpanId: da9949518cdecb56
X-B3-Sampled: 0
Cache-Control: no-cache,must-revalidate,max-age=0,no-store,private
Set-Cookie: BrowserId=tsMTvvkCEeqU9WHivUXHfA; domain=.salesforce.com; path=/; expires=Fri,
 17-Sep-2021 16:27:43 GMT; Max-Age=31536000
Content-Type: application/json;charset=UTF-8
Vary: Accept-Encoding
Content-Encoding: gzip
Transfer-Encoding: chunked


{
  "countOfActiveModels" : 1,
  "countOfModels" : 1,
  "createdBy" : {
    "id" : "005B0000004iaa7IAA",
    "name" : "Admin User",
    "profilePhotoUrl" :
"https://einsteinprediction123--c.documentforce.com/profilephoto/729B0000000EmIX/T"
  },
```

```
  "createdDate" : "2020-03-31T01:18:51.000Z",
  "id" : "1ORB000000000JuOAI",
  "label" : "ItalyInfo",
  "lastModifiedBy" : {
    "id" : "005B00000051RBqIAM",
    "name" : "YourUserName",
    "profilePhotoUrl" :
"https://einsteinprediction123--c.documentforce.com/profilephoto/729B00000003ARf/T"
  },
  "lastModifiedDate" : "2020-08-18T23:09:25.000Z",
  "modelsUrl" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB000000000JuOAI/models",

  "name" : "ItalyInfo9461190a",
  "outcome" : {
    "goal" : "Minimize",
    "label" : "Total Cases",
    "name" : "totale_casi"
  },
  "predictionType" : "Regression",
  "pushbackField" : {
    "label" : "mypredField",
    "name" : "Custom_Opportunity__c.mypredField__c"
  },
  "refreshConfig" : {
    "isEnabled" : true,
    "recipientList" : [ {
      "displayName" : "YourUserName",
      "id" : "005B00000051RBqIAM",
      "type" : "User"
    } ],
    "schedule" : {
      "dayInWeek" : "wednesday",
      "frequency" : "monthlyrelative",
      "nextScheduledDate" : "2020-10-08T02:00:00.000Z",
      "time" : {
        "hour" : 19,
        "timeZone" : {
          "gmtOffset" : -7.0,
          "name" : "Pacific Daylight Time",
          "zoneId" : "America/Los_Angeles"
        }
      },
      "weekInMonth" : "first"
    },
    "shouldScoreAfterRefresh" : false,
    "userContext" : {
      "id" : "005B00000051RBqIAM"
    },
    "warningThresholdPercentage" : 0.05
  },
  "status" : "Enabled",
  "subscribedEntity" : "Custom_Opportunity__c",
```

```
  "url" : "/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB000000000JuOAI"
}
```

The `refreshConfig` section describes model refresh jobs for this prediction. Model refresh jobs are configured in Model Manager as described in Configure Automatic Model Refresh for a Prediction Definition on page 1725.

Create a Prediction Definition

```
POST /smartdatadiscovery/predictiondefinitions
```

Update a Prediction Definition

```
PATCH /smartdatadiscovery/predictiondefinitions/{predictionDefinitionId}
```

Delete a Prediction Definition

```
DELETE /smartdatadiscovery/predictiondefinitions/{predictionDefinitionId}
```

Manage Models

The Einstein Prediction Service provides REST API endpoints to manage models. Each model has a unique id. A model is used to evaluate data and return a prediction and recommendations. These REST endpoints allow you to make updates to model metadata, but not update the actual predictive model.

Get Available Models

```
GET /smartdatadiscovery/predictiondefinitions/{predictionDefinitionId}/models
```

The following code shows an example response:

> 📝 **Note:** If you don't have a My Domain deployed in your org, your URL format is different. If you have a My Domain deployed and enhanced domains are enabled in your org, your URL format is different. The Stabilize URLs in Visualforce, Experience Builder, Site.com studio, and content files My Domain setting also affects this format. For details, see My Domain URL Formats in Salesforce Help.

```
HTTP/1.1 200 OK
Date: Thu, 30 Jan 2020 17:34:12 GMT
Strict-Transport-Security: max-age=31536000; includeSubDomains
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
X-Robots-Tag: none
Cache-Control: no-cache,must-revalidate,max-age=0,no-store,private
Set-Cookie: BrowserId=uqyLl0OGEeq2RPW2iPHbvQ; domain=.salesforce.com; path=/; expires=Fri,
 29-Jan-2021 17:34:12 GMT; Max-Age=31536000
Content-Type: application/json;charset=UTF-8
Vary: Accept-Encoding
Content-Encoding: gzip
Transfer-Encoding: chunked


{
  "models" : [ {
```

```
    "actionableVariables" : [ {
      "label" : "Type",
      "name" : "Type",
      "type" : "Text"
    }, {
      "label" : "Ownership",
      "name" : "Ownership",
      "type" : "Text"
    }, {
      "label" : "Rating",
      "name" : "Rating",
      "type" : "Text"
    }, {
      "label" : "Division",
      "name" : "Division",
      "type" : "Text"
    }, {
      "label" : "AccountScore",
      "name" : "AccountScore",
      "type" : "Text"
    } ],
    "createdBy" : {
      "id" : "005B0000002zz1lIAA",
      "name" : "MyUserName",
      "profilePhotoUrl" :
"https://MyDomainName--c.documentforce.com/profilephoto/729B00000009ttx/T"
    },
    "createdDate" : "2020-01-17T00:24:35.000Z",
    "fieldMappingList" : [ {
      "modelField" : {
        "label" : "CloseDate",
        "name" : "CloseDate",
        "type" : "Date"
      }
    }, {
      "modelField" : {
        "label" : "Industry",
        "name" : "Industry",
        "type" : "Text"
      }
    }, {
      "modelField" : {
        "label" : "StartDate",
        "name" : "StartDate",
        "type" : "Date"
      }
    }, {
      "modelField" : {
        "label" : "Ownership",
        "name" : "Ownership",
        "type" : "Text"
      }
    }, {
      "modelField" : {
```

```
          "label" : "Type",
          "name" : "Type",
          "type" : "Text"
        }
    }, {
      "modelField" : {
        "label" : "Rating",
        "name" : "Rating",
        "type" : "Text"
      }
    }, {
      "modelField" : {
        "label" : "BillingState",
        "name" : "BillingState",
        "type" : "Text"
      }
    }, {
      "modelField" : {
        "label" : "Division",
        "name" : "Division",
        "type" : "Text"
      }
    }, {
      "modelField" : {
        "label" : "AccountScore",
        "name" : "AccountScore",
        "type" : "Text"
      }
    } ],
    "filters" : [ ],
    "id" : "1OtB00000004CApKAM",
    "label" : "CLV",
    "lastModifiedBy" : {
      "id" : "005B0000002zz1lIAA",
      "name" : "MyUserName",
      "profilePhotoUrl" :
"https://MyDomainName--c.documentforce.com/profilephoto/729B00000009ttx/T"
    },
    "lastModifiedDate" : "2020-01-17T00:24:35.000Z",
    "model" : {
      "id" : "1OTB000000000ajOAA"
    },
    "modelType" : "Regression",
    "name" : "CLV",
    "predictionDefinitionUrl" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB00000004CApOAM",
    "sortOrder" : 0,
    "status" : "Enabled",
    "url" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB00000004CApOAM/models/1OtB00000004CApKAM"

  }, {
    "actionableVariables" : [ {
      "label" : "Industry",
```

```
      "name" : "Industry",
      "type" : "Text"
    }, {
      "label" : "Type",
      "name" : "Type",
      "type" : "Text"
    }, {
      "label" : "Ownership",
      "name" : "Ownership",
      "type" : "Text"
    }, {
      "label" : "Rating",
      "name" : "Rating",
      "type" : "Text"
    }, {
      "label" : "Division",
      "name" : "Division",
      "type" : "Text"
    } ],
    "createdBy" : {
      "id" : "005B0000002zz1lIAA",
      "name" : "MyUserName",
      "profilePhotoUrl" :
"https://MyDomainName--c.documentforce.com/profilephoto/729B00000009ttx/T"
    },
    "createdDate" : "2020-01-17T00:26:14.000Z",
    "fieldMappingList" : [ {
      "modelField" : {
        "label" : "CloseDate",
        "name" : "CloseDate",
        "type" : "Date"
      }
    }, {
      "modelField" : {
        "label" : "Industry",
        "name" : "Industry",
        "type" : "Text"
      }
    }, {
      "modelField" : {
        "label" : "StartDate",
        "name" : "StartDate",
        "type" : "Date"
      }
    }, {
      "modelField" : {
        "label" : "Ownership",
        "name" : "Ownership",
        "type" : "Text"
      }
    }, {
      "modelField" : {
        "label" : "Type",
        "name" : "Type",
```

```
        "type" : "Text"
      }
    }, {
      "modelField" : {
        "label" : "Rating",
        "name" : "Rating",
        "type" : "Text"
      }
    }, {
      "modelField" : {
        "label" : "BillingState",
        "name" : "BillingState",
        "type" : "Text"
      }
    }, {
      "modelField" : {
        "label" : "Division",
        "name" : "Division",
        "type" : "Text"
      }
    }, {
      "modelField" : {
        "label" : "AccountScore",
        "name" : "AccountScore",
        "type" : "Text"
      }
    } ],
    "filters" : [ ],
    "id" : "1OtB00000004CAqKAM",
    "label" : "CLV",
    "lastModifiedBy" : {
      "id" : "005B0000002zz1lIAA",
      "name" : "MyUserName",
      "profilePhotoUrl" :
"https://MyDomainName--c.documentforce.com/profilephoto/729B00000009ttx/T"
    },
    "lastModifiedDate" : "2020-01-17T00:26:14.000Z",
    "model" : {
      "id" : "1OTB000000000ajOAA"
    },
    "modelType" : "Regression",
    "name" : "CLV",
    "predictionDefinitionUrl" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB00000004CApOAM",
    "sortOrder" : 1,
    "status" : "Enabled",
    "url" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB00000004CApOAM/models/1OtB00000004CAqKAM"

  } ],
  "totalSize" : 2,
  "url" :
"/services/data/v50.0/smartdatadiscovery/predictiondefinitions/1ORB00000004CApOAM/models"
}
```

Get Metadata for a Model

```
GET /smartdatadiscovery/predictiondefinitions/{predictionDefinitionId}/models/{modelId}
```

Delete a Model

```
DELETE /smartdatadiscovery/predictiondefinitions/{predictionDefinitionId}/models/{modelId}
```

SEE ALSO:

[My Domain URL Formats](#)

Manage Prediction Jobs

The Einstein Prediction Service provides REST API endpoints to run bulk scoring jobs for a prediction. Bulk scoring jobs enable you to score predictions on multiple records at a time. For example, after deploying an updated model, use bulk scoring to refresh all prediction scores. You can also run bulk scoring on historical data to see how well your model performs. With bulk scoring, you can score all records, a segment of the records, or records that have not reached the terminal state.

Bulk scoring jobs are configured in Model Manager. To set up Bulk Scoring jobs, see [Score Records in Bulk](#) on page 1732.

📝 **Note:** If the daily predictions limit is reached in your org, active scoring jobs are paused, then resumed the next day.

Run a Bulk Scoring Job for a Prediction

```
POST /smartdatadiscovery/predict-jobs
```

POST Request Body

In the request body, specify the prediction definition Id associated with the bulk scoring request job, along with a user-defined label, and optional filter settings.

```
{
   "predictionDefinition":{ "id":"{{predictionDefinitionId}}" },
   "label":"{{label}}",
   "useTerminalStateFilter" : false,
   "filters":{
      "filters":[
         {
            "fieldName":"Opportunity.Name",
            "values":[ "My Opportunity" ],
            "operator":"Equal"
         }
      ]
   }
}
```

Request Body Attributes

Name	Type	Description
id	String	Prediction definition Id. The Model Manager displays prediction definition Ids, as described in [View a Prediction Definition](#) on

Name	Type	Description
		page 1723. You can also retrieve a list of available prediction definition Ids using the following API request: `GET /smartdatadiscovery /predictiondefinitions`
label	String	User-defined label that identifies this version of the job.
filters	object	Optional filters to score a portion of the records. If no filters are specified, then all records are scored. If specified, you can use one type of filter, either `useTerminalStateFilter` or `filters`, but not both.
useTerminalStateFilter	Boolean	Whether to score records in the terminal state. • `true` - score records that are *not* in terminal state • `false` (default) - score all records, whether a record is in terminal state or not
filters	object	One or more filters to score a subset of records.

Get Scoring Jobs

```
GET smartdatadiscovery/predict-jobs
```

Manage Model Refresh Jobs

The Einstein Prediction Service provides REST API endpoints to retrieve metadata for model refresh jobs.

Model refresh jobs are configured in Model Manager. To set up model refresh jobs, see Configure Automatic Model Refresh for a Prediction Definition on page 1725.

Get Prediction Refresh Jobs

```
GET /smartdatadiscovery/refresh-jobs
```

The following code shows an example response.

```
{
  "refreshJobs" : [ {
    "createdBy" : {
      "id" : "005B0000006DdetIAC",
      "name" : "YourUserName",
      "profilePhotoUrl" :
"https://wavepm--c.gus.content.force.com/profilephoto/729B0000000Eqe5/T"
    },
    "createdDate" : "2020-09-09T16:16:39.000Z",
    "endTime" : "2020-09-09T16:20:02.000Z",
    "id" : "1OXB00000008OIPOA2",
    "refreshTarget" : {
      "id" : "1ORB0000000TNXyOAO"
```

```
    },
    "refreshTasksUrl" :
"/services/data/v50.0/smartdatadiscovery/refresh-jobs/1OXB00000008OIPOA2/refresh-tasks",
    "startTime" : "2020-09-09T16:16:41.000Z",
    "status" : "Success",
    "type" : "UserTriggered",            "url" :
"/services/data/v50.0/smartdatadiscovery/refresh-jobs/1OXB00000008OIPOA2"
  } ],
  "totalSize" : 1,
  "url" : "/services/data/v50.0/smartdatadiscovery/refresh-jobs"
}
```

Name	Type	Description
`"refreshTarget" : { "id" :}`	String	One of the following: • prediction definition id—scheduled jobs are defined at the prediction definition level • deployed model id—users can refresh an individual failed deployed model manually
`status`	String	Status of the job.
`type`	object	One of the following: • `Scheduled` • `UserTriggered`—a user triggered the refresh on demand in Model Manager

Get Prediction Refresh Job Details

```
GET smartdatadiscovery/refresh-jobs/{refreshJobId}
```

The following code shows an example response.

```
{
  "refreshJobs" : [ {
    "createdBy" : {
      "id" : "005B0000006DdetIAC",
      "name" : "YourUserName",
      "profilePhotoUrl" :
"https://wavepm--c.gus.content.force.com/profilephoto/729B0000000Eqe5/T"
    },
    "createdDate" : "2020-09-09T16:16:39.000Z",
    "endTime" : "2020-09-09T16:20:02.000Z",
    "id" : "1OXB00000008OIPOA2",
    "refreshTarget" : {
      "id" : "1ORB0000000TNXyOAO"
    },
    "refreshTasksUrl" :
"/services/data/v50.0/smartdatadiscovery/refresh-jobs/1OXB00000008OIPOA2/refresh-tasks",
    "startTime" : "2020-09-09T16:16:41.000Z",
    "status" : "Success",
```

```
    "type" : "UserTriggered",
    "url" : "/services/data/v50.0/smartdatadiscovery/refresh-jobs/1OXB00000008OIPOA2"
  }
```

Get Prediction Refresh Job Task Details

```
GET smartdatadiscovery/refresh-jobs/{refreshJobId}/refresh-tasks
```

The following code shows an example response.

```
{
  "refreshTasks" : [ {
    "createdBy" : {
      "id" : "005B0000006DdetIAC",
      "name" : "YourUserName",
      "profilePhotoUrl" :
"https://wavepm--c.gus.content.force.com/profilephoto/729B0000000Eqe5/T"
    },
    "createdDate" : "2020-09-09T16:16:39.000Z",
    "endTime" : "2020-09-09T16:19:54.000Z",
    "id" : "1OxB00000008OIKKA2",
    "refreshTarget" : {
      "id" : "1OtB0000000TNdAKAW",
      "label" : "IsWon",
      "name" : "IsWon"
    },
    "refreshedAIModel" : {
      "id" : "1OTB0000000PDeBOAW"
    },
    "source" : {
      "story" : {
        "id" : "1Y3B00000004HOCKA2"
      },
      "storyVersion" : {
        "id" : "9B4B00000008UrEKAU"
      }
    },
    "startTime" : "2020-09-09T16:16:45.000Z",
    "status" : "Success"
  } ],
  "totalSize" : 1,
  "url" :
"/services/data/v50.0/smartdatadiscovery/refresh-jobs/1OXB00000008OIPOA2/refresh-tasks"
}
```

Name	Type	Description
refreshTarget	String	Target deployed model that was refreshed.
refreshedAIModel	String	Id of the resulting AI model that was deployed.
source	object	Specific story version and story Id associated with this refresh model.

Get Predictions in Apex

Use the `ConnectApi.SmartDataDiscovery.predict` API to get predictions in your Apex code.

You can programmatically run predictions in Apex using the `ConnectApi.SmartDataDiscovery.predict` method. There are two types of input:

- record IDs
- record IDs plus calculated fields that are not stored in the record

This function accepts an input of up to 200 record IDs or raw data rows, along with a prediction definition ID. It returns up to 200 predictions of type `ConnectApi.SmartDataDiscoveryPrediction`.

> 📝 **Note:** Before you use the Apex Predict API, deploy the model you want to use for predictions. For instructions, see Deploy Models on page 1706.

For more information about Apex code, see the Apex Developer Guide.

API Usage Limits

- `ConnectApi.SmartDataDiscovery.predict` can be called up to 2,000 times per user per hour.
- Each `ConnectApi.SmartDataDiscovery.predict` can handle up to 200 records in a single call.
- `ConnectApi.SmartDataDiscovery.predict` can be called no more 50,000 times per day (24 hours) per org.

Batch Apex and Callouts for Models Using Supplemental Datasets

For batch Apex, if you are using a model associated with a supplemental data set (see Map Dataset Fields to Fields in the Salesforce Object on page 1709), be sure to follow the requirements for callouts described in Callout Limits and Limitations in the Apex documentation.

Get Predictions Using the Record ID as Input

Syntax

```
public ConnectApi.SmartDataDiscoveryPrediction
ConnectApi.SmartDataDiscovery.predict(ConnectApi.SmartDataDiscoveryPredictInputRecords
input)
```

SmartDataDiscoveryPredictInputRecords class

Name	Type	Description
`predictionDefinition`	String	18-digit prediction definition Id.
`records`	List<Id>	List of records to be scored. These record Ids must be from the same object type associated with the deployed prediction definition.
`settings`	ConnectApi.SmartDataDiscoveryPredictSettings	Set the number of top predictors, the number of improvements, and the improvement impact threshold to return in the response.

Get Predictions Using the Record ID and Calculated Fields as Input

Syntax

```
public ConnectApi.SmartDataDiscoveryPrediction
ConnectApi.SmartDataDiscovery.predict(SmartDataDiscoveryPredictInputRecordOverrides input)
```

SmartDataDiscoveryPredictInputRecordOverrides class

Name	Type	Description
`predictionDefinition`	String	18-digit prediction definition Id.
`rows`	List<ConnectApi.SmartDataDiscoveryPredictInputRowObject>	Ordered List of row details, including the record Id and values for columns that are not mapped from Salesforce to the prediction definition.
`columnNames`	List<String>	Ordered List of column names that are not mapped.
`settings`	ConnectApi.SmartDataDiscoveryDiscoveryPredictSettings	Set the number of top predictors, the number of improvements, and the improvement impact threshold to return in the response.

ConnectApi.SmartDataDiscoveryPredictInputRowObject class

Name	Type	Description
`record`	String	18-digit Id for the Salesforce record that will get the score. The record Id must match the object type associated with the deployed prediction definition.
`row`	List<String>	Ordered List of the values that accompany the ordered list of `columnNames` provided above.

ConnectApi.SmartDataDiscoveryDiscoveryPredictSettings class

Name	Type	Description
`maxMiddleValues`	Integer	Maximum number of top predictors (0-3) to return in the response. The terms "top factor" and "top predictor" mean the same thing. Default: 0 (no top predictors returned).
`maxPrescriptions`	Integer	Maximum number of improvements (0-200) to return in the response. The terms "improvement" and "prescription" mean the same thing. Default: -1 (unlimited).
`prescriptionImpactPercentage`	Integer	Improvement impact threshold, specified as a percentage (0 to 100). If set to 20, for example, only improvements that affect the predicted outcome more than 20% are returned. Default: 0.

Output Classes

ConnectApi.SmartDataDiscoveryPrediction class

Name	Type	Description
predictions	List<ConnectApi.SmartDataDiscoveryPredictObject> or List<ConnectApi.SmartDataDiscoveryPredictErrorObject>	List of predictions or exceptions returned from the `predict` method.

ConnectApi.SmartDataDiscoveryPredictObject class

Name	Type	Description
prediction	ConnectApi.SmartDataDiscoveryPredict	Object containing all the prediction data.
prescriptions	List<ConnectApi.SmartDataDiscoveryPredictCondition>	List of improvements representing suggestions to improve the prediction. This is blank if no action variables were selected when deploying the prediction.

ConnectApi.SmartDataDiscoveryPredict class

Name	Type	Description
total	double	Final prediction value.
other	double	Unexplainable portion of the final prediction value.
baseLine	double	Baseline from where the prediction started. This is basically the average if there were no model.
middleValues	List<ConnectApi.SmartDataDiscoveryPredictCondition>	List of top predictors that contribute to the prediction.
importWarnings	List<ConnectApi.SmartDataDiscoveryImportWarning>	List of warnings associated with the prediction. If there are no warnings, this returns null.

ConnectApi.SmartDataDiscoveryPredictCondition class

Name	Type	Description
value	double	Value that this field / value combination contributes to the final prediction.
columns	List<ConnectApi.SmartDataDiscoveryPredictColumn>	Column information.

ConnectApi.SmartDataDiscoveryPredictColumn class

Name	Type	Description
columnName	String	Name of the column. Example: `Account Name`
columnValue	String	Value of the column. Example: `Acme`

Connectapi.SmartDataDiscoveryPredictImportWarnings class

Name	Type	Description
outOfBoundsColumns	List<ConnectApi.PredictOutOfBounds>	Columns + values that are out of bounds. These values were not used when training the model.
missingColumns	List<String>	List of column names that are required to make a prediction but are missing. Perhaps they are null?

Connectapi.SmartDataDiscoveryPredictOutOfBoundsFields class

Name	Type	Description
columnName	String	Name of the column that is out of bounds.
columnValue	String	Value of the column that is out of bounds. This means the model was not trained on this particular value. Example: The model was trained on Type = `"Manufacturing"` but received a value of `"Manufracturing"`.

Exceptions

If there are rows that caused the prediction method to fail, this call returns an exception rather than a prediction.

ConnectApi.SmartDataDiscoveryPredictErrorObject class

Name	Type	Description
message	String	Error message.

Code Example

This code example is designed for building test cases.

```
// Initialize the Prediction API Input Record
ConnectApi.SmartDataDiscoveryPredictInputRecords predictInput = new
ConnectApi.SmartDataDiscoveryPredictInputRecords();

// Set Prediction Definition
predictInput.predictionDefinition = '1ORB0000000TN1SOAW';

// Set Records
predictInput.records = new List<Id>{'006B0000004ApAZIA0'};

// Set Settings for Prediction
ConnectApi.SmartDataDiscoveryDiscoveryPredictSettings settings = new
ConnectApi.SmartDataDiscoveryDiscoveryPredictSettings();
settings.maxMiddleValues = 3;//[0-3] limit of 3
settings.maxPrescriptions = 3;
settings.prescriptionImpactPercentage = 20;
predictInput.settings = settings;
```

```
/////////////////////////////////////////////////////////////

// Generate Predictions
ConnectApi.SmartDataDiscoveryPrediction response =
ConnectApi.SmartDataDiscovery.predict(predictInput);
// Get Prediction
ConnectApi.SmartDataDiscoveryPredictObject prediction =
(ConnectApi.SmartDataDiscoveryPredictObject) response.predictions[0];
// Get Prescriptions (Improvements)
List<ConnectApi.SmartDataDiscoveryPredictCondition> prescriptions = prediction.prescriptions;

// Iterate over the first prescription
ConnectApi.SmartDataDiscoveryPredictCondition first_prescription = prescriptions[0];
Double value = first_prescription.value;
List<ConnectApi.SmartDataDiscoveryPredictColumn> columns = first_prescription.columns;

// Prescriptions usually have one-column information:
ConnectApi.SmartDataDiscoveryPredictColumn column_1 = columns[0];

String columnName = column_1.columnName;
String columnValue = column_1.columnValue;
String columnInputVal = column_1.inputValue;
/////////////////////////////////////////////////////////////

// Get Top Factors (Top Predictors)
List<ConnectApi.SmartDataDiscoveryPredictCondition> topFactors =
prediction.prediction.middleValues;
// Iterate over the first TopFactor
ConnectApi.SmartDataDiscoveryPredictCondition first_topFactor = topFactors[0];

Double tf_value = first_topFactor.value;
List<ConnectApi.SmartDataDiscoveryPredictColumn> tf_columns = first_topFactor.columns;

// Top Factors have multiple Columns, so it is recommended to iterate over the list
ConnectApi.SmartDataDiscoveryPredictColumn tf_column_1 = tf_columns[0];

String tf_columnName = tf_column_1.columnName;
String tf_columnValue = tf_column_1.columnValue;
String tf_columnInputVal = tf_column_1.inputValue;
/////////////////////////////////////////////////////////////

// Get Import Warnings
ConnectApi.SmartDataDiscoveryPredictImportWarnings importWarnings =
prediction.prediction.importWarnings;

// System Debug
System.debug('Total ' + prediction.prediction.total);
System.debug('Import Warnings ' +importWarnings);
```

Get Predictions in Process Automation Formulas

Use the Einstein Discovery PREDICT function to get predictions in your process automation formulas. Your automation logic can make decisions based on predicted outcomes from deployed Einstein Discovery predictions. For example, in an approval process, a formula can determine whether a predicted outcome meets a threshold required for automatic approval.

PREDICT Function in Process Automation Formulas

The PREDICT function is available when defining formulas associated with:

- Next Best Action
- validation rules
- screen, headless, and invocable flows
- processes (in Process Builder)
- workflow rules
- approval processes
- predefined field values
- field update actions
- default values

To learn more, see Calculate Field Values With Formulas and the PREDICT section in Formula Operators and Functions I–Z.

> **Note:**
>
> - The PREDICT function is not supported in formula-based fields on Salesforce objects.
> - Getting predictions from Einstein Discovery predictions requires the View Einstein Discovery Recommendations system permission. To learn more, see Learn About Einstein Discovery Permissions and Permission Sets on page 1593.

Use the PREDICT Function in a Formula

1. Create or edit an approval process, flow, process (in Process Builder), workflow rule, or Next Best Action.
2. In the formula builder, under **Functions**, select **Advanced**, select **PREDICT**, and click **Insert Selected Function**. The following example shows the default entry for PREDICT in a formula for an approval process.

The PREDICT function returns an Einstein Discovery prediction for a record based on the specified record ID or for a list of fields and their values. It uses the following syntax:

```
PREDICT(PredDefId, [recordId] | [field, value, ...])
```

where

Parameter	Description
PredDefId	The Prediction Definition ID of a deployed prediction in your Salesforce org.
[recordId]	The recordId of the record for which you want a prediction. If specified, the PREDICT function returns a prediction for the data values in that record.
[field, value, ...]	List of field names and values. Be sure to provide all the fields that the prediction requires as input. If specified, the PREDICT function returns a prediction for the data values you provided.

3. Define the prediction you want to use in this formula. The syntax for *PredDefId* must match the following pattern:

```
$SmartDataDiscovery.PredictionDefinitions.<predictionDevName>.Id
```

Select *PredDefId*, click **Insert Field**, select **SmartDataDiscovery > Prediction Definitions**, select an available prediction, select **Prediction Definition Id**, and click **Insert**.

4. Specify the record ID or provide a list of fields and associated values.

   The following example shows the formula for an approval process. It evaluates to TRUE if the predicted outcome for the given record is less than the specified discount amount (`Discount__c`).

5. Click **Save**.

Example Formulas

Example with recordId

```
PREDICT($SmartDataDiscovery.PredictionDefinitions.Recommende_kFjqS_1370.Id, Id)
```

This example calls the PREDICT function and passes a prediction definition and recordId.

Example with List of Fields and Values

```
PREDICT($SmartDataDiscovery.PredictionDefinitions.Recommende_kFjqS_1370.Id,
        'Customer_Typec', Text(Customer_Typec), 'List_Price__c', List_Price__c)
```

This example calls the PREDICT function and passes a prediction definition and list of fields with associated values.

Display Einstein Predictions Using Custom Fields (Deprecated)

If you want more programmatic control over predictions than what you get with automated prediction fields, you can install the Einstein Discovery Managed Package and implement a solution based on custom fields in a Salesforce object.

> **Note:** This feature is deprecated. The Einstein Discovery Managed Package simply provides a wrapper around Apex code. To use predictions in Apex, see Get Predictions in Apex on page 1776.

Alternatively, consider using automated prediction fields (see Display Einstein Predictions Using Automated Prediction Fields).

Install the Einstein Discovery Managed Package (Deprecated)

Your Salesforce admin must install the Salesforce managed package to enable Einstein Discovery recommendations in a Salesforce Object.

Create Custom Fields in Salesforce to Display Recommendations

You create custom fields in Salesforce to display the outcome, explanation, and prescription information imported from Einstein Discovery.

Connect Einstein Discovery to Your Custom Fields

To import the recommendations, you connect your custom fields with Einstein Discovery.

Use Process Builder to Score Your Records

You use Process Builder to create a process to track scores.

Score Records in Bulk

You can bulk score using Workbench, Salesforce Data Loader, or the Salesforce Developer Console.

Create an Apex Trigger

The Apex trigger is fired when a Salesforce object's records are inserted or updated.

Install the Einstein Discovery Managed Package (Deprecated)

Your Salesforce admin must install the Salesforce managed package to enable Einstein Discovery recommendations in a Salesforce Object.

> **Note:** This feature is deprecated. The Einstein Discovery Managed Package simply provides a wrapper around Apex code. To use predictions in Apex, see Get Predictions in Apex on page 1776.

Before installing the managed package, Einstein Discovery must be enabled in your Salesforce org. For more information, see the Einstein Discovery setup documentation.

1. In your browser, navigate to https://appexchange.salesforce.com/appxListingDetail?listingId=a0N3A00000FOm9nUAD
   You see the Einstein Discovery Writeback page.

2. Click **Get it Now**.

   If prompted, enter your login information.

3. On the Einstein Discovery Writeback page, select **Install for Admins Only**.

4. Click **Install**.

5. When you see a message indicating that the installation is complete, click **Done**.
   In Setup, on the Installed Packages page, you see a package named **Einstein Discovery Writeback**.

Create Custom Fields in Salesforce to Display Recommendations

You create custom fields in Salesforce to display the outcome, explanation, and prescription information imported from Einstein Discovery.

1. From Setup, select **Customize** and then click the Salesforce object that you want to add the fields to.

2. Click **Add a custom field**.

3. Add a field in which to store the Einstein Discovery outcome information.

    a. In the Custom Fields & Relationships section, click **New**.

    b. Select **Number**, and click **Next**.

    c. Enter a field label. For example, `Discovery Outcome`. The field name is filled in automatically. Click **Next**.

    d. On the Establish field-level security page, specify the field's access settings for each profile. Only a trigger is allowed to update data in the custom field, so select the **Visible** checkbox and the **Read-Only** checkbox on the applicable profiles. Click **Next**.

    e. Choose which page layouts display the field. Click **Save & New**.

4. Add a field in which to store the Einstein Discovery explanation information.

    a. Select **Text Area (Long)**, and click **Next**.

    b. Enter a field label. For example, `Discovery Explanation`. Click **Next**.

    c. On the Establish field-level security page, specify the field's access settings for each profile. Only a trigger is allowed to update data in the custom field, so select the **Visible** checkbox and the **Read-Only** checkbox on the applicable profiles. Click **Next**.

    d. Choose which page layouts display the field. Click **Save & New**.

5. Add a field in which to store the Einstein Discovery prescription information.

    a. Select **Text Area (Long)**, and click **Next**.

    b. Enter a field label. For example, `Discovery Prescription`. Click **Next**.

    c. On the Establish field-level security page, specify the field's access settings for each profile. Only a trigger is allowed to update data in the custom field, so select the **Visible** checkbox and the **Read-Only** checkbox on the applicable profiles. Click **Next**.

    d. Choose which page layouts display the field. Click **Save**.

The three new custom fields are now in the Custom Fields & Relationships list. They are also visible on the object's detail page, depending on the page layout. Next, put the fields in their own section in the page layout.

Connect Einstein Discovery to Your Custom Fields

To import the recommendations, you connect your custom fields with Einstein Discovery.

1. From Setup, select **Develop** then select **Custom Settings**.

2. Next to Einstein Discovery - Write Back, click **Manage**, and then click **New**.

3. Fill out the information. Use field names, not field labels.

Table 16: Fields

Field	Description
Name	This integration name is used when the Apex trigger references the custom setting data. Note the name because you use it when you create the Apex trigger. You use this value to replace `[NameOfCustomSetting]` in the sample code.
Commentary Field Name	The name of the field you created to store the Einstein Discovery explanation information. Example: `Discovery_Explanation__c`.
Object	The Salesforce object where you added the custom fields. Example: `Case`.

Field	Description
Outcome Field Name	The field label that you created. Example: *Discovery_Outcome__c*.
Prediction Definition Id	The ID of the Einstein Discovery prediction definition from which you want to import the recommendations. To get the Prediction Definition ID, go into the Model Manager, open a prediction, and copy its Prediction ID from the URL in your browser. For instructions, see View a Prediction Definition on page 1723.
Prescription Field Name	The name of the prescription field that you created. Example: *Discovery_Prescription__c*.
Hide Scoring Warnings	When scoring records, fields with out of bounds values as well as fields that are missing from the model will show as warnings within the prediction explanation. If you want to suppress these warnings from users, check this option to hide warnings.
Disable Scoring	If you want to disable scoring for any reason, check this option. You can do this if, for example, you want to bulk load data and don't want to score it right away.
Batch Update Query	The batch Apex classes included in the package use this query to determine which records to score in bulk. Leave this empty if you do not use batch Apex to score existing records.

4. Click **Save**.

Use Process Builder to Score Your Records

You use Process Builder to create a process to track scores.

> 📝 **Note:** To display Einstein Discovery Predictions in a Salesforce object, you can use *either* Process Builder or an Apex trigger (see Create an Apex Trigger on page 1791). Both approaches require that you install the Einstein Discovery managed package (see Install the Einstein Discovery Managed Package (Deprecated) on page 1785).

To begin, start by creating a new process for the object type that will contain the scores:

1. To launch Process Builder, from **Setup**, search for and then select **Process Builder**.

2. To create a new process, click **New**.

3. In the New Process window, enter a **Process Name**, click the down arrow for **The process starts when**, select **A record changes**, and then click **Save**.

4. In the Process Builder, click **Add Object** and select the object where Einstein Discovery will write the scores. Under **Start the process**, select **when a record is created or edited**, and then click **Save**.

5. Click **Add Criteria**, specify the Criteria name, and choose the criteria you want to use to score your data. If there are no criteria, simply choose **No criteria—just execute the actions!** When finished, click **Save**.

6. Under **Immediate Actions**, click **Add Action**.

7. For **Action Type**, select **Apex**.

8. Specify an **Action Name** (for example, **Score Records**).

9. For **Apex Class**, select **Einstein Discovery - Score Records**.

10. For **Record ID**, select **Field Reference** and set the value as the record Id.

11. For **Value**, click **Find a Field**, enter **Master Record ID**, and click **Choose**.

12. For **Custom Setting Name**, verify that **Type** is **String**, and set the value as **[NameOfCustomSetting]**.

13. **Optional**: If you were not able to map all of the fields to your object and you would like to calculate these values using formula fields:

   a. Click **Add Row**, click **Find a Variable** and select **Column Names (Comma Separated)**, verify that the type is **String**, and under **Value**, specify the column names where you would like to manually calculate the values.

   b. Click **Add Row**, select **1st Value**, select the **Type**, and enter a **Value**. Repeat this step for all column names entered in the previous step, and make sure to choose the next value sequential variable (**2nd Value**, **3rd Value**, and so on).

   An example of an process that could be used to score Accounts could look like this:

14. Click **Save**.

Score Records in Bulk

You can bulk score using Workbench, Salesforce Data Loader, or the Salesforce Developer Console.

Bulk Score Using Workbench or Salesforce Dataloader

This approach is the fastest way to bulk score because it uses the Bulk API's ability to score batches of data in parallel. However, it requires some manual steps.

1. Run a query to select Ids from the object you want to bulk score. Download that list as a CSV. Limit the query to no more than 400K rows of data (which is the maximum that can be processed per user per hour). If there is more than one score per record, decrease the row limit by a factor of 2 for each additional score calculated per record.

2. Check Dataloader or Workbench settings:

- Ensure that **Bulk API** is enabled.
- Ensure that the Bulk API is running in **Parallel Mode**. Note that Parallel Mode is the default for the Bulk API. Just verify that the **Enable serial mode for Bulk API** setting unchecked.
- Start with a Batch size between 2000 and 5000. This will determine how many batches will be scored in parallel.

3. Perform an update on the CSV file of IDs downloaded in step 1.

Bulk Score Using the Salesforce Developer Console

The Einstein Discovery Writeback managed package includes a Batch APEX class and a Scheduled APEX Class that you can use to bulk score records in a more automated fashion. This approach requires fewer manual steps, but its run-time execution takes longer because it scores each batch one at a time (serial versus parallel).

The following APEX classes are included:

BatchScoreRecords

This class takes the Batch Update Query defined in the custom settings, limits data based on API limits, breaks the data into batches, and processes them.

This class takes one input, **Row Limit**, which specifies the number of rows to process in this transaction. Set this to 400,000 if there is only one score per record. Decrease the row limit by a factor of 2 for each additional score calculated per record.

To invoke this class on a one-time basis, simply open up the Salesforce Developer Console, navigate to Debug Open Execute Anonymous Window, and type the following:

```
Database.executeBatch(new ed_insights.BatchScoreRecords(('<CONFIG_NAME>', <ROW LIMIT>,
<BATCH SIZE>));
```

where:

- *<CONFIG_NAME>*: name of the custom setting to use
- *<ROW LIMIT>*: number of rows to process
- *<BATCH SIZE>*: size of each batch

  > **Note:** You might need to adjust this if you run into CPU timeouts.

ScheduleBatchScoreRecords This class can be used to schedule batch scoring. This is useful if there are more than 400K records and you just want to automate the scoring of the records over time. Specify the following:

```
System.schedule('ScoreHourly', '0 0 * * * ?', new
ed_insights.ScheduleBatchScoreRecords(('<CONFIG_NAME>', <ROW LIMIT>, <BATCH SIZE>));
```

where:

- *<CONFIG_NAME>*: Config name that will contain the query used for batch scoring.
- *<ROW LIMIT>*: Number of rows to process in this transaction. Set this to 400,000 if there is only one score per record. Decrease the row limit by a factor of 2 for each additional score calculated per record.
- *<BATCH SIZE>*

  > **Note:** You might need to adjust this if you run into CPU timeouts.

To invoke this class on a one time basis, simply open up the Salesforce Developer Console, navigate to **Debug Open Execute Anonymous Window**, and type the following:

Troubleshooting

- **APEX CPU Timeout**: This error might occur due to batch sizes. Simply adjust the batch size to a smaller number and try again.
- **Intermittent Failures of 200 records**: When using Bulk API in parallel mode, batches are invoking the scoring API concurrently. If more than five batches invoke the API at the exact same time, a concurrency limit will be encountered, causing an entire 200-row chunk of IDs to fail. Simply wait for the entire set of data to complete and go back and score only the records that failed.

Create an Apex Trigger

The Apex trigger is fired when a Salesforce object's records are inserted or updated.

An Apex trigger calls the API that generates the prediction.

> **Note:** To display Einstein Discovery Predictions in a Salesforce object, you can use *either* an Apex trigger or Process Builder (see Use Process Builder to Score Your Records on page 1787). Both approaches require that you install the Einstein Discovery managed package (see Install the Einstein Discovery Managed Package (Deprecated) on page 1785).

Add a trigger to the object that Einstein Discovery is making the recommendations on. In addition, add a trigger to related objects when the story uses their fields to calculate the prediction. For example, supposed you want to set up predictions on the Deal object. Set a trigger on the Deal object itself. Set a trigger on the Account object when any of its fields affect the prediction.

The following steps show the different types of triggers you can use.

1. From Setup, select **Customize** and then click the object that you want to add the trigger to.

2. Click **Triggers** and then click **New**.

3. To define your trigger, enter Apex code similar to this sample code.

   a. Replace `[ObjectName]` with the name of the object that you are adding the trigger to.

   b. Replace `[NameOfCustomSetting]` with the integration name you created on the Einstein Discovery Integration information page.

   Apex trigger for single object with no data transformation

Use this example if the Salesforce object contains all of the fields that the Einstein Discovery story uses to make a prediction. In this case, all of the data that the story requires exists in one object. Also, the data in the object is in the same format and has the same field name as the variables in the story.

```
trigger SetDealPrediction on [ObjectName] (after insert, after update) {
   if(System.isFuture()) return;
   if(ed_insights.CheckRecursive.runOnce()) {
   // custom Settings' name
   String CONFIG_NAME = '[NameOfCustomSetting]';
   ed_insights.TriggerHandler.insertUpdateHandle(CONFIG_NAME);
   }
}
```

Apex trigger with data transformation

Use this example to perform some transformation on the data in the Salesforce object. The transformation in the Apex trigger converts the data in the Salesforce object into a format that the Einstein Discovery story can use.

```
trigger SetDealPrediction on [ObjectName] (after insert, after update)
{
    if(System.isFuture()) return;

    List<Map<String, String>> fieldValues = new List<Map<String, String>>();

    // Iterate through all records that are supposed to be processed by this trigger
    // Only need to populate the fields that are NOT mapped to the object.
    // Fields mapped to the object are automatically queried via the recordID

    for ([objectName] o: Trigger.new) {

    // the field-value Map to be passed to the setPredition() method
        Map<String, String> fieldStringMap = new Map<String, String>();

        fieldStringMap.put('Id', o.Id); //Must Have this or scoring will fail
        fieldStringMap.put('<model field 1>',<value 1>);
        fieldStringMap.put('<model field 2>',<value 2>);
        fieldValues.add(fieldStringMap);

 }

    // make sure there are fieldValues populated; otherwise, skip this
    if(fieldValues.size()>0)
    {
        ed_insights.TriggerHandler.insertUpdateHandleForFieldValues(CONFIG_NAME,
JSON.serialize(fieldValues));
    }
  }
}
```

4. Make sure that **Is Active** is selected.

5. Click **Save**.

When records of this type are created or updated, the trigger fires and gets the recommendations from Einstein Discovery.

Get Predictions in Tableau

Einstein Discovery in Tableau provides predictions and improvements for Tableau data in worksheets, calculated fields, and flows.

[Introduction to Einstein Discovery in Tableau](#)

Einstein Discovery in Tableau brings trusted, real-time predictions and improvements to your Tableau data in a no-code required, rapid iteration environment.

[Get Predictions in Tableau Dashboards](#)

Integrate on-demand, interpretable predictions from Einstein Discovery natively in Tableau dashboards. With the Einstein Discovery dashboard extension, you can embed dynamic predictions in worksheets and explore "what if" prediction scenarios with parameters.

[Get Predictions in Tableau Calculated Fields](#)

It's easy to get Einstein Discovery predictions in calculated fields. In Model Manager, simply select a prediction definition, generate a Tableau script, and then copy and paste the script into a Tableau calculated field.

[Embed Einstein Predictions in Tableau Flows](#)

Use the Prediction step in Tableau Prep to include Einstein Discovery predictions directly into your flows. Enrich your flow output with predictions and, optionally, improvements and top factors.

Introduction to Einstein Discovery in Tableau

Einstein Discovery in Tableau brings trusted, real-time predictions and improvements to your Tableau data in a no-code required, rapid iteration environment.

Integration Prerequisites

Einstein Discovery in Tableau is supported in Tableau version 21.1 or later. Integration requires:

Requirement	Description
License	One of the following licenses: • Einstein Discovery in Tableau license • Tableau CRM Plus license • Einstein Predictions license These licenses are available for an extra cost.
Salesforce user account	Account that is configured to access Einstein Discovery. If you use the Einstein Discovery in Tableau license, then your user account must have the **View Einstein Discovery Recommendations Via Connect API** system permission assigned to it. If you use either the Tableau CRM Plus license or Einstein Predictions license: • To get predictions using already deployed Einstein Discovery predictions, the account must have the **View Einstein Discovery Recommendations** system permission assigned to it. • To build, deploy, and manage predictions in Einstein Discovery, the account must have the **Manage Einstein Discovery** permission assigned to it. To configure user accounts, see Set Up Einstein Discovery on page 1593.
Tableau user account	For setup instructions, see your Tableau documentation.

Integration Overview

Refer to the following resources for an overview of Einstein Discovery in Tableau.

- Product Overview: Introducing Einstein Discovery in Tableau
- Video: Introducing Einstein Discovery in Tableau
- Blog: Bringing AI predictions to Tableau with Einstein Discovery
- Blog: Einstein Discovery in Tableau – Try it for Yourself
- Video: Introducing Tableau Business Science | A new class of AI-powered analytics
- Blog: What is Tableau Business Science?
- White Paper: Introducing Tableau Business Science

Tableau Help Resources

Refer to the following resources for detailed instructions regarding Einstein Discovery in Tableau.

- Tableau Desktop help: What's New (Authoring): Integrate predictions from Einstein Discovery into Tableau
- Tableau Desktop help: Integrate Einstein Discovery Predictions in Tableau (Authoring)
- Tableau Online help: Configure Einstein Discovery Integration
- Tableau Server help: What's New: Integration of Einstein Discovery Predictions
- Tableau Server help: Configure Einstein Discovery Integration
- Tableau Prep help: Add Einstein Discovery Predictions to your flow

Getting Started with Tableau

If you're new to Tableau, try these learning resources:

- Overview: Get started with Tableau
- Tutorial: Tutorial: Get Started with Tableau Desktop
- Tableau Desktop help: Get Started
- Trail: Tour the Tableau Environment
- Trail: Get Started with Web Authoring in Tableau Online
- Trail: Get Started with Data Visualization in Tableau Desktop

SEE ALSO:

Get Predictions in Tableau

Get Predictions in Tableau Dashboards

Integrate on-demand, interpretable predictions from Einstein Discovery natively in Tableau dashboards. With the Einstein Discovery dashboard extension, you can embed dynamic predictions in worksheets and explore "what if" prediction scenarios with parameters.

Tableau Documentation for the Einstein Discovery Dashboard Extension

This topic provides an overview of the Einstein Discovery Dashboard Extension. For detailed instructions on setting up and using the extension in Tableau dashboards, refer to the following topics in Tableau Desktop help:

- Explore Predictions in Tableau with the Einstein Discovery Dashboard Extension

- [Requirements for Access](#)

Tableau Source Data

In Tableau, there are two types of sources of data you can use with the extension:

- **Worksheets** allow you to get predictions for one or more rows of data (bulk predictions). You can get bulk predictions for up to 50,000 rows of data at a time.
- **Parameters** allow you to conduct interactive, "what if" predictive analysis on a single set of input values.

The source data you use in Tableau - whether in a worksheet or in parameters - must provide all the data elements that the Einstein Discovery prediction requires to predict the outcome you want. The data must match the granularity that the Einstein Discovery prediction expects. For example, if the prediction expects sales per individual order, then your Tableau data must be in the form of individual orders, not aggregated into monthly totals for all orders.

Use the Einstein Discovery Dashboard Extension in Tableau Dashboards

In Tableau, as an author, create worksheets and the dashboard, and then add the Einstein Discovery dashboard extension to the dashboard.

When configuring or using the extension, if prompted, log into Salesforce and allow access from Tableau.

Next, configure the extension by selecting a deployed Einstein Discovery prediction definition and data source details.

Map model fields to source data in the dashboard.

Configure other settings, if you want, such as whether to retrieve top predictors and improvements.

After the settings are configured, dashboard viewers and users can click different marks in the Tableau visualization to explore predictions on the underlying data. Here is an example Einstein Prediction panel:

SEE ALSO:

[Get Predictions in Tableau](#)

Get Predictions in Tableau Calculated Fields

It's easy to get Einstein Discovery predictions in calculated fields. In Model Manager, simply select a prediction definition, generate a Tableau script, and then copy and paste the script into a Tableau calculated field.

To embed Einstein Discovery predictions in Tableau calculated fields:

1. Log into Salesforce with a user account that has the **Manage Einstein Discovery** permission assigned to it.

2. Open Tableau CRM Analytics Studio.

3. Open Model Manager.

4. [View a Prediction Definition](#) on page 1723.

5. Click the **Settings** tab.

6. From the **Edit Settings** menu, choose **Create Tableau Table Calculation**.

Model Manager generates a script containing the ID of the prediction definition you want to use, the columns (variables) used in the model, and the corresponding source columns in Tableau (which you can edit later in Tableau).

7. Click **Copy to Clipboard**.

8. In your Tableau worksheet, paste the script generated from Model Manager into the Calculation Editor for the target calculated field.

9. If needed, replace the column names inside each ATTR or SUM function with the name of the matching column in your Tableau data, then click **OK**.

   When the script runs, it sends a prediction request (with the input data) to the prediction definition in Salesforce. Einstein Discovery returns the results to Tableau, where it appears in your worksheet or dashboard.

   📝 Note: For detailed Tableau instructions, refer to Einstein Discovery in Pass Expressions with Analytics Extensions in Tableau Desktop help.

SEE ALSO:

Manage Prediction Definitions and Models

Get Predictions in Tableau

Embed Einstein Predictions in Tableau Flows

Use the Prediction step in Tableau Prep to include Einstein Discovery predictions directly into your flows. Enrich your flow output with predictions and, optionally, improvements and top factors.

Tableau Documentation for the Prediction Step in Tableau Prep

This topic provides an overview of the Prediction step supported in Tableau Prep version 2021.1.3 or later. For detailed instructions on setting up and using the Predict step in your flows, refer to the following topic in Tableau Prep help: Add Einstein Discovery Predictions to your flow.

Add a Prediction Step to a Flow

In Tableau Prep, as a creator, open a flow, and then add a Prediction step.

Next, connect to Salesforce with a user account that can access Einstein Discovery.

From the list of deployed prediction definitions in Salesforce, select one to use to generate predictions. Fields for prediction values are added to the data source automatically.

Map model fields to flow fields and configure other settings as needed.

When the flow is run, the Prediction step gets a prediction for each row of data and saves the prediction value, and optionally other information, to the flow output.

SEE ALSO:

Get Predictions in Tableau

Modify and Enrich Salesforce Data with Data Pipelines

Salesforce Data Pipelines is a high-performance data platform that can clean, transform, and enrich large volumes of data at scale. Use Salesforce Data Pipelines to enrich and modify Salesforce data without needing third-party tools or taking data outside your trusted Salesforce environment. Unlike external ETL tools and data warehousing solutions, Data Pipelines is built natively into your Salesforce CRM, ensuring that updates are fast and secure. No more round-tripping data through expensive, fragile, lower-performing external systems.

📝 **Note:** Data Pipelines isn't available to Government Cloud customers.

If your CRM data is scattered among multiple systems, use Data Pipelines to consolidate it into one system, like Salesforce. Bring in data from external services, like Snowflake, Amazon S3, and more. To gain access to data in external systems, use the prebuilt connectors—there are a ton of them. For systems that don't have a connector, export the data as a CSV, and then upload it.

Get Familiar with Salesforce Data Pipelines

Before you start using Salesforce Data Pipelines, learn what it can do and get familiar with its tools.

What Can I Do with Data Pipelines?

Before Data Pipelines, companies often resorted to exporting data, enriching it in external systems, and then writing the results back into Salesforce–often a clunky and expensive approach. But now, with Data Pipelines, do it all natively, securely, and more quickly in Salesforce. Data Pipelines simplifies the process of consolidating, aggregating, and calculating results based on both CRM data and external data, ensuring rapid updates that give your CRM users the latest and most complete picture of customer activity.

With Data Pipelines, you can:

- Transform and aggregate Salesforce data quickly and at scale. Use clicks, not code to perform simple-to-complex calculations on your data. Calculate totals and create aggregations to help your employees understand customer activity across millions of records.

- Enrich your Salesforce data with external data to complete the picture of customer activity. Data Pipelines provides numerous connectors that make it easy to access the data held in external systems. Use the connectors to connect and regularly sync external data from these systems, and then prepare the external data to fit into Salesforce objects.

- Complete data projects faster. No need to implement complex external infrastructure for common data sync and clean-up needs. Using Data Pipelines, the typical data preparation tasks can be configured faster and without the need to purchase or integrate multiple external tools.

- Improve your data quality. Use the machine learning and clean-up tools available in Data Pipelines to clean your data.

To illustrate the power of Data Pipelines, let's check out a sample use case. Your organization wants to provide premium service to larger accounts. To enable sales and customer support reps to see at-a-glance how large customers are, you'd like to display the lifetime spend for each account on Salesforce's Account details page. To also gauge how happy your customers are with your business, you previously sent out a Google Analytics survey. You'd now like to display the general sentiment of each customer's comment on the details page as well. With lots of customers, you're hoping there's an easy way to determine the sentiment of 20,000 comments. Well, you're in luck—Salesforce Data Pipelines to the rescue! To calculate total lifetime spend, you use Data Pipelines to build a recipe that aggregates all opportunity amounts for each account. The recipe pulls this data from the Salesforce objects, Opportunity and Opportunity Line Item. You also pull in customer survey comments from Google Analytics. The recipe uses machine learning to automatically detect the general sentiment of each customer's comment and attaches a sentiment to each account. The recipe writes the results to a custom Salesforce object.

You then build a Lightning widget in the Account details page to display the total lifetime spend and sentiment for each account.

Did you notice that there was no mention of code? That's right, Salesforce Data Pipelines provides an intuitive user interface that allows you to make simple clicks to tackle complex tasks.

Get Oriented with the Data Pipelines Tools

Data Pipelines uses Data Manager and Data Prep tools to accomplish tasks. Before you get your hands dirty, get familiar with these tools.

Data Manager for Data Pipelines

After you've set up Data Pipelines, use Data Manager to build, run, and monitor jobs that extract, prepare, and load data into Salesforce. Then you can surface the results anywhere, like in Salesforce fields or Lightning components.

Data Prep for Data Pipelines

Data Prep provides an intuitive, visual interface that allows you to easily point-and-click your way to build recipes that prepare data and load it into a target.

Data Manager for Data Pipelines

After you've set up Data Pipelines, use Data Manager to build, run, and monitor jobs that extract, prepare, and load data into Salesforce. Then you can surface the results anywhere, like in Salesforce fields or Lightning components.

To open the Data Manager app, select **Data Manager** from the app picker.

The Data Manager consists of multiple tabs.

Note: If you purchased Tableau CRM and Data Pipelines, use the toggle to switch between the two. For example, to view and create recipes for Data Pipelines, select **Data Pipelines** first.

Use these Data Manager tabs to perform various tasks.

Tab	Usage
Connections	Create, edit, and delete connections to different source and target systems. A connection enables recipes to access data in that system. Data Pipelines provides out-of-the-box connectors that allow you to quickly create connections.
Data	View data sources (datasets and connected objects) that you can use in recipes. Upload CSV files to get access to other data sources that don't have prebuilt connectors.
Recipes	View, edit, delete, and run recipes. You can also access Data Prep to create a recipe from this tab.
Monitor	Monitor and troubleshoot jobs, like recipes that are currently running.

Tab	Usage
Limits	Monitor Data Pipeline usage and limits. You can also view this information on the Getting Started page in setup. The limits vary based on your licenses.

Data Prep for Data Pipelines

Data Prep provides an intuitive, visual interface that allows you to easily point-and-click your way to build recipes that prepare data and load it into a target.

Watch a Demo: ▶ Introducing Data Prep (English Only)

Data Prep shows a visual representation of the recipe, showing how you are preparing your data and where to write the results.

Set Up Salesforce Data Pipelines

Set up your organization to use Data Pipelines. Enable Data Pipelines and grant users access to it.

Data Pipelines Requirements
This section provides requirements for using Data Pipelines.

Assign the Permission Set License and Permission Set to Users
To enable a user to access Data Pipelines features, grant them the Use Data Pipelines Add On User Settings permission set license and the Data Pipelines Add On permission set.

Enable Data Pipelines and Features
To use Data Pipelines, first enable it for your organization.

EDITIONS

Available in Lightning Experience.

Available with Salesforce Data Pipelines (a feature of Tableau CRM), which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions.

Learn About Data Pipelines User Permissions

Users with the Data Pipelines Add On User Settings permission set license can be assigned all Data Pipelines user permissions. To assign all Data Pipelines user permissions, assign them the Data Pipelines Add On permission set.

Data Pipelines Limits

Review the limits for Data Pipelines. If needed, you can purchase additional licenses to increase the base limits documented below.

Data Pipelines Limitations

Data Pipelines uses some features originally built for Tableau CRM, including Data Prep and Data Manager. As a result, you might see some Tableau CRM options that aren't supported in Data Pipelines.

Data Pipelines Requirements

This section provides requirements for using Data Pipelines.

Supported Salesforce Licenses and Editions

Review the required Salesforce editions and licenses to ensure that Data Pipelines works in your org and can be used by your users.

Each Data Pipelines user must also have one of these Salesforce user licenses.

- Lightning Platform (app subscription)
- Lightning Platform (one app)
- Full CRM
- Salesforce Platform
- Salesforce Platform One

Data Pipelines supports Enterprise, Performance, and Unlimited editions.

Supported Browsers

Data Pipelines supports all browsers supported by Lightning Experience, with the following exceptions.

- Data Pipelines isn't supported on Apple® Safari®.
- Data Prep isn't supported on Internet Explorer 11.

Browser Zoom

Browser zoom settings other than 100% aren't supported for Data Pipelines.

Screen Resolution

The minimum screen resolution required to support all Salesforce features is 1024 x 768 pixels.

Assign the Permission Set License and Permission Set to Users

To enable a user to access Data Pipelines features, grant them the Use Data Pipelines Add On User Settings permission set license and the Data Pipelines Add On permission set.

1. From Setup, enter `Users` in the Quick Find box, then select **Users**.

2. Click the user's name.

3. Hover over Permission Set License Assignments and click **Edit Assignments**.

4. To enable the permission set license, select **Data Pipelines Add On User Settings** and click **Save**.

> **Note:** The user must have one of these Salesforce user licenses to assign them the Data Pipelines Add On User Settings permission set license.
>
> - Lightning Platform (app subscription)
> - Lightning Platform (one app)
> - Full CRM
> - Salesforce Platform
> - Salesforce Platform One

5. Hover over Permission Set Assignments and click **Edit Assignments**.

> **Tip:** For efficiency, you can assign a permission set to groups of users.

6. To assign all user permissions available in the Data Pipelines Add On User Settings permission set license to the user, select the Data Pipelines Add On User option, click the Add button, and then click **Save**.

Enable Data Pipelines and Features

To use Data Pipelines, first enable it for your organization.

1. From Setup, enter `Data Pipelines` in the Quick Find box, then select **Getting Started**.

> **Note:** To see this option, you must be assigned the Use Data Pipelines Add On User Settings permission set license and the Data Pipelines Add On permission set.

2. To enable Data Pipelines, select the **Data Pipelines** option.

3. To enable users to write data to targets, select the output connectors.

For example, select Salesforce to write data to Salesforce. If you don't enable output connectors, users can write data to datasets only.

SEE ALSO:

Connect to Your Data Sources and Targets

Learn About Data Pipelines User Permissions

Users with the Data Pipelines Add On User Settings permission set license can be assigned all Data Pipelines user permissions. To assign all Data Pipelines user permissions, assign them the Data Pipelines Add On permission set.

Data Pipelines comes with the following user permissions.

User Permission	What It Enables
Add Tableau CRM Remote Connections	Add connections to access data from external data sources.
Edit Dataset Recipes	Create, edit, and run recipes. Monitor jobs in Data Manager. Doesn't enable editing security predicates in existing recipes, or running and scheduling recipes based on datasets that have security predicates.
Edit Tableau CRM Dataflows	Edit, delete, and use remote connections; add and remove connected objects; run and schedule data sync; create, edit, delete, run, schedule, and monitor recipes. Use discretion when assigning this user permission because it enables access to all Salesforce object data to which the Integration User has access. See Salesforce Data Access in Tableau CRM.
Upload External Data to Tableau CRM	Upload external data to Data Pipelines to create a dataset. Monitor jobs in Data Manager.
Use Data Pipelines Base	Open Data Manager and query datasets using the API. At minimum, a user must be assigned this user permission to use Data Pipelines.

The Edit Dataset Recipes and Edit Tableau CRM Dataflows user permissions enable some of the same features. To help distinguish when to use each user permission, check out Data Prep Access Based on Your User Type.

Data Pipelines Limits

Review the limits for Data Pipelines. If needed, you can purchase additional licenses to increase the base limits documented below.

> ⊡ **Note:** If you purchased Data Pipelines and Tableau CRM, each has dedicated limits that are handled separately and enforced independently. For example, datasets created by recipes created with Tableau CRM don't affect the Data Pipelines limit on total rows for all datasets.

This table lists Data Pipelines org-level limits.

Limit	Value	Can Add On?	Details
Total rows in all datasets	100 million	Yes	
Rows processed	100 million/month	Yes	
Recipe Job Runtime	30 hours/month	Yes	Excludes queue time. All jobs started after time is exceeded are blocked until the next calendar month (GMT).
Concurrent recipes	1	Yes, the maximum is 2.	
CSV data written to Salesforce	1 GB/day	Yes	
Data written to Salesforce	10 GB/day	Yes	
Data written to Tableau Hyper (Tableau Online)	10 GB/day	Yes	
Data written to Amazon S3 and Snowflaked (combined)	10 GB/day	Yes	Both output connectors share this limit.
Queries	25,000/day	Yes	
Assignable User Licenses	3	Yes	
Maximum file size for all CSV uploads in a rolling 24-hour period	20 GB	No	
Number of recipes	20	Yes	This limit is shared between Tableau CRM and Data Pipelines if you purchased both. Both licenses add to the pool.
Number of connected objects	50	Yes, the maximum is 150.	This limit is shared between Tableau CRM and Data Pipelines if you purchased both. Both licenses add to the pool. The 150-maximum applies to Data Pipelines only.

This table lists all service protection limits. We've added service protection limits to ensure that we protect our service from excessive loads.

Limit	Value	Can Add On?
Concurrent API Calls	20	No
Concurrent Queries per Org	5	No
Concurrent Queries per User	5	No
API Calls per Hour per User	10,000	No

Limit	Value	Can Add On?
Maximum Number of Concurrent Data Sync Runs	2	No
Maximum Average Number of Values per Multivalue Field	100	No

Data Pipelines also enforces the API, dataset field, data sync, recipe, external data, query, and direct data limits enforced by Tableau CRM. For cases where Data Pipelines and Tableau CRM specify different limits for the same feature, the Data Pipelines limits takes precedence.

SEE ALSO:

Tableau CRM Limits

Data Pipelines Limitations

Data Pipelines uses some features originally built for Tableau CRM, including Data Prep and Data Manager. As a result, you might see some Tableau CRM options that aren't supported in Data Pipelines.

Data Pipelines doesn't support the following features:

- Sync out for Snowflake

    Note: Data Pipelines supports the Snowflake output connector.

- Sharing inheritance for datasets
- Packaging
- Event-based scheduling user interface

If you purchase Tableau CRM and Data Pipelines, you can't share recipes between them. If needed, create a copy in both environments.

Some Data Pipelines user interface labels, messages, and documentation aren't localized yet—they're available in English only.

To minimize documentation duplication between Tableau CRM and Salesforce Data Pipelines, you might see links to Tableau CRM help topics in the Data Pipelines online help. Although the topic mentions Tableau CRM only, it covers features available in Data Pipelines as well. To indicate when a link takes you to Tableau CRM content, we appended "(Tableau CRM help)" in the link.

Data Pipelines can support encryption to enhance security for Salesforce customers. It extends encryption capabilities to the data at rest that is stored in datasets (in the Salesforce file system). To enable Encryption in Salesforce Data Pipelines, you must be approved by the Tableau CRM Encryption Product Manager. Your org must have a Shield Platform Encryption tenant secret. For more information, see Tableau CRM Encryption.

Modify and Enrich Salesforce Data with Recipes

Use recipes to prepare large amounts of Salesforce and external data at scale before loading it into Salesforce objects. For example, you can define data preparation logic that combines data from two data sources and cleans up inconsistent date formats. Add nodes and transformations to a recipe to perform calculations as well as combine, transform, enrich, and clean your data. Use the smart transformations to predict missing values, detect sentiment of text, and forecast measures into the future. There's so much you can do with recipes.

To set up access to source data, create an input connection. When you create a connection, select the objects and columns to pull data from. You can add a filter to the connection to extract a subset of all rows. In the connection properties, you also specify a user account that determines what data the connection can access. For example, to access data in Amazon S3, specify an Amazon S3 user account. If the user account doesn't have access to an object, the connection can't pull data from that object.

After you create a connection, run the associated data sync job to extract the data from each selected object in the data source. To speed up the process of getting data, a data sync pulls data from the data source in advance and stores it in connected objects. After you run a data sync for the first time, a recipe can use the connected objects as sources.

To create a recipe, select the input data (connected objects or datasets), add data preparation logic to prepare that data, and specify the target to load the results into. For example, you can use a recipe to combine data from different sources, clean the data to make it consistent, and then load the results into a Salesforce object.

Run the recipes to prepare the data and load it into the targets. To continually refresh the data, schedule the sync and recipe jobs to run on a regular basis. To ensure that your recipes use the latest data, ensure that the data sync jobs complete before dependent recipes run.

This diagram shows a sample process for syncing Salesforce and external data into connected objects, and then using a recipe to prepare the synced data and load it into Salesforce objects.

Stage CSV Data in Datasets

You can also use CSV data in recipes. Because CSV files can't be used as sources in a recipe, you can upload the CSV file and write it to a dataset, and then use the dataset as a source for the recipe. When you upload the file, it's temporarily stored for processing only. After the dataset is created, the file is purged. If you want to use the file again later, keep a copy.

Connect to Your Data Sources and Targets

Use prebuilt connectors to quickly connect to data in your Salesforce orgs (local and external), apps, data warehouses, and database services. Data Pipelines offers different types: input and output connectors. Use an input connector to extract data from a data source. Use an output connector (beta) to load the recipe results into a target. To use an output connector, the org admin must first enable it in setup.

Run Sync to Synchronize Data in Connected Objects

In Data Manager, you can schedule sync to run automatically or manually run it. Run data sync manually the first time to make the data available to build recipes. Schedule subsequent syncs to regularly update the data. You schedule data sync for each connection, where all objects under the connection sync at the specified time, and not individual objects. To sync objects from the same data source on different intervals, create multiple connections to the data source, and set a unique schedule for each connection. To ensure that the latest source data is loaded into the targets, schedule data syncs to run before the corresponding recipes.

Prepare Data in a Recipe

Data Prep provides an intuitive, visual interface that allows you to easily point-and-click your way to build recipes that prepare data and load it into a target. Use the graph of a recipe to see at a glance where data comes from and how it flows through the recipe to the target. To validate the recipe as you build, preview how raw data is transformed at every step of the way.

Run Recipes to Get Data into Salesforce

In Data Manager, you can manually run them. When you run a recipe, Data Prep prepares the source data and then loads the results into the target based on the recipe definition. To update the target with the latest synced data, schedule data syncs to run before the corresponding recipes.

Monitor Sync and Recipes

Use Data Manager's Monitor tab to monitor and troubleshoot data sync and recipe jobs. The tab provides the status, job type, start time and duration for each job. To filter the jobs with a particular status, click the corresponding subtab, like Queued or Failed. You can also view error messages about a job.

Use Data Manager's Data tab to access data available for recipes. You can view datasets created by recipes that can be used as sources in other recipes. You can also view all connected objects.

Use Data Manager's Limits tab to monitor usage and to ensure you don't hit the limits. The limits vary based on your licenses. If needed, contact Salesforce to increase your org's limits.

Stage CSV Data in Datasets

You can also use CSV data in recipes. Because CSV files can't be used as sources in a recipe, you can upload the CSV file and write it to a dataset, and then use the dataset as a source for the recipe. When you upload the file, it's temporarily stored for processing only. After the dataset is created, the file is purged. If you want to use the file again later, keep a copy.

When you upload the file, Data Pipelines infers the metadata about each column in the CSV file. Metadata describes the structure of the data in the file, like the data type, precision, and scale. If you upload a CSV from the user interface, Data Pipelines automatically generates the metadata, which you can preview and change.

Before uploading a CSV file:

- Review the format requirements, like date formats, in the Analytics Cloud External Data Format Reference (Tableau CRM help).

- Ensure that the column names in the external data file don't conflict with the generated date column names. For example, if you load a CSV with column Create_Date, Data Pipelines generates the Create_Date_Year column in the dataset. If the CSV also has a column named Create_Date_Year, the Data Pipelines throws an error because the names conflict. For more information about auto-generated date columns in datasets, see Handle Date Values (Tableau CRM help).

1. To open Data Manager, select **Data Manager** from the app picker.

2. To open the Data tab, click **Data**.

3. Click **New Dataset**.

4. Click **Upload Files**, select the file, and then click **Open**.

5. Check that Data Pipelines has correctly identified the properties of your file.

Usually, Data Pipelines correctly identifies your file properties. If it doesn't, your data may not load correctly and you will see unexpected results when you preview the data on the next screen. For example, if we chose the wrong field delimiter, your columns might be off.

6. To edit the file properties, click **Edit** or upload a new metadata file.

The metadata file contains the file properties as well.

7. Click **Next**.

8. In the Dataset Label field, enter a name for the dataset.

   By default, Data Pipelines uses the file name as the dataset name. The name can't exceed 80 characters.

9. Select the app where the dataset will be created.

   By default, Data Pipelines selects your My Private App. To change an app, click the cross on it and select a different one.

10. Click **Next**.

    The Edit Column Attributes screen appears. Here, you can preview the data and change the attributes for each column.

11. To change a column's attributes, click the column in the list on the left.

Column attributes appear in the right panel. The column attributes that you see are determined by the column type.

> 🛑 **Important:** Data Pipelines detects the format for date fields based on a sample of values. If the sample contains values with unsupported formats or a mixture of formats, Data Pipelines sets the field type to Text. If you change the date format that Data Pipelines detects, rows with a different format will fail.
>
> Consider this example data.
>
> | Row | SIC Code | SIC Description | Last Updated |
> | --- | --- | --- | --- |
> | 1 | 1110 | Barley growing | 1/10/17 |
> | 2 | 1120 | Rice growing | 11/14/17 |
> | 3 | 1130 | Alliaceous vegetable growing | 1/1/17 |
>
> Data Pipelines detects the date format for the Last Updated field as $M/d/yy$. This format displays months and days below 10 without leading zeros, and years as 2 digits, as in 1/1/17. If you change the format to $MM/dd/yy$, rows 1 and 3 will fail because Data Pipelines expects the month and day parts of the date values to have 2 digits.

12. When you finish reviewing or editing column attributes, click **Next**.
Data Pipelines uploads the data, prepares and creates the dataset, and shows progress as it happens.

Connect to Your Data Sources and Targets

Use prebuilt connectors to quickly connect to data in your Salesforce orgs (local and external), apps, data warehouses, and database services. Data Pipelines offers different types: input and output connectors. Use an input connector to extract data from a data source. Use an output connector (beta) to load the recipe results into a target. To use an output connector, the org admin must first enable it in setup.

Watch a Demo: ▶ Connect to External Data (English Only)

1. To open Data Manager, select **Data Manager** from the app picker.

2. To open the Connections tab, click **Connections**.
   The Connect page opens. By default, you have a connection to your local Salesforce org only.

3. To connect to a different data source or create another connection to the same data source, click **Connect to Data**.

   By default, the Input Connections tab is selected. Keep this selection to connect to a recipe source. Select **Output Connections** to connect to a recipe target.

4. To add a connection, click **Add Connection**.

5. Select the connection type.

   For example, to connect to another Salesforce org (not your local org), select **Salesforce External Connector**.

6. Enter the connection details.

   The connection details vary by data source.

7. Save the connection.

8. After you receive the successful connection message, click **Continue**.
   The new connection is added to the connections list.

9. Click the connection to select an object and its fields from the data source.

10. In the Select an Object dialog box, select the object that you'd like to extract data from, and then click **Continue**.

11. Select the fields that you want to extract from the object and click **Continue**.

12. Review the sample records to ensure you're pulling the correct fields and the fields have the expected data.

13. To change the field attributes, like label and default value, click ✏ .

14. To filter the records that are extracted from the object, click ▼ .

15. To save the object and field selections, click **Save**.
The object appears under the connection type.

16. To change the field selections, attributes, or filter for an existing object, click the object name.

17. To add another object from this connection, click **Connect to Data**, select the same connection, and then select an object and its fields.

SEE ALSO:

Tableau CRM Help: Connect to Local Salesforce Data

Tableau CRM Help: Connect to Remote Data Outside of Your Salesforce Org

Run Sync to Synchronize Data in Connected Objects

In Data Manager, you can schedule sync to run automatically or manually run it. Run data sync manually the first time to make the data available to build recipes. Schedule subsequent syncs to regularly update the data. You schedule data sync for each connection, where all objects under the connection sync at the specified time, and not individual objects. To sync objects from the same data source on different intervals, create multiple connections to the data source, and set a unique schedule for each connection. To ensure that the latest source data is loaded into the targets, schedule data syncs to run before the corresponding recipes.

📝 **Note:** Data sync jobs don't count towards your daily recipe run limit.

1. To open Data Manager, select **Data Manager** from the app picker.

2. To open the Connections tab, click **Connections**.

3. To run or schedule the sync for all objects in a connection, or set notifications, click the drop down (⌄) next to the connection name and select the appropriate option.

> **Note:** The Sync Out option is reserved for later use. Don't use it.

4. To sync or disconnect an object, change the connection mode (for local Salesforce connection only), or change row-level sharing settings for an object, click the drop down (⬇) next to the object and select the appropriate option.

5. To change the connection mode, choose one of these options.

- **Incremental Sync** updates only rows that changed since the last sync. It's the fastest option. Pulls only new, updated, and deleted records to match the changes in the Salesforce object since the previous sync. Use this method to run faster syncs.

- **Periodic Full Sync** updates rows incrementally and periodically overwrites all rows with records in the Salesforce object. Runs an incremental sync on each scheduled sync. Also runs a full sync on the first sync that happens after Friday 11 PM in your org's time zone.

- **Full Sync** updates all rows with records in the Salesforce object. Pulls all records from the Salesforce object, and overwrites the records from the previous sync.

Prepare Data in a Recipe

Data Prep provides an intuitive, visual interface that allows you to easily point-and-click your way to build recipes that prepare data and load it into a target. Use the graph of a recipe to see at a glance where data comes from and how it flows through the recipe to the target. To validate the recipe as you build, preview how raw data is transformed at every step of the way.

Watch a Demo: ▶ Introducing Data Prep (English Only)

In Data Prep, a recipe consists of nodes. You can add different types of nodes to a recipe to bring in, manipulate, and write data to a target. Data Prep displays a graph of the recipe and its nodes in the Graph area.

EDITIONS

Available in Lightning Experience.

Available with Salesforce Data Pipelines (a feature of Tableau CRM), which is available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions.

A Data Prep graph doesn't show every transformation. It shows only inputs (source data), appends, aggregates, filters, joins, and the output (where the data is written). It also shows transforms, which are groups of transformations that change the raw data, such as concatenation and column type conversions. By hiding lower-level data changes, the graph provides a higher-level, easier-to-read view of the flow of data. We call each object shown in the graph a *node*. A recipe can have multiple Input, Append, Aggregate, Filter, Join, Transform, and Output nodes.

Although the graph doesn't show individual data transformations, you can select a Transform node (1) in the Graph area to see its transformations in the Details area (2).

You can select a transformation step in the left panel of the Details area to preview the results of that transformation in the Preview tab. Similarly, you can select any node in the graph to preview the results of that node.

When you run a recipe, each node applies its function on the input data, and then the results are written to the specified target.

Create a Recipe with Data Prep

Use a Data Prep recipe to clean, transform, and enrich data before loading the results into one or more targets. To ensure you're building the right logic, preview the results as you build it. When you're done, run the saved recipe to write the results to the targets.

Create a Recipe with Data Prep

Use a Data Prep recipe to clean, transform, and enrich data before loading the results into one or more targets. To ensure you're building the right logic, preview the results as you build it. When you're done, run the saved recipe to write the results to the targets.

Watch a Demo: ▶ Create a Data Prep Recipe (English Only)

1. To open Data Manager, select **Data Manager** from the app picker.

2. To open the Recipes tab, click **Recipes**.

USER PERMISSIONS

To manage and create a recipe:
- Edit Analytics Dataflows OR Edit Analytics Dataflows

3. To create a recipe, click **New Recipe**.

4. To choose the initial input data for the recipe, click **Select Data**.

5. Select the check box next to each dataset or connected object that you want to add as input for the recipe.

You can't use a trended dataset as input data for a recipe. If needed, you can add more data to the recipe later.

6. In the Selected Columns area on the right, choose which columns to include from the selected input data.

By default, all columns are included. If you added multiple data source objects, select each row, one at a time, to choose its columns. The Selected Columns area shows columns for the highlighted row only.

7. Click **Next**.

Data Prep adds a separate Input node to the graph for each input data selection.

8. To add a node to the recipe, click the Add Node button (⊕) between two nodes or at the end of the recipe, and then select the node type. To show the Add Node button between two nodes, hover the cursor on the connecting line.

Add nodes to transform, filter, aggregate, join, and append data.

9. To ensure you're adding the right data prep logic, preview the results of each node and transformation that you add to the recipe. To preview the results, select the node or transformation in the graph.

For example, selecting an Aggregate node shows the following preview results.

10. To edit or delete a node, select the node, click ⋯, and then select the appropriate action.

11. To edit or delete a transformation in a Transform node, select the Transform node, select the transformation step, and then click ✏ or 🗑 .

12. To split the recipe into two branches, hover the cursor over the node where the branch begins, click ⬧, and then select the node type to start the new branch.

You can branch a recipe to write results to multiple output nodes or to transform some of the rows before appending them together again later.

13. To connect nodes in the graph, drag the Add Node button (+) or Branch button ( ) on top of a node in the recipe graph, and then select an action, like Connect Nodes.

14. End the recipe with one or more Output nodes to tell the recipe where to write the results.

To write results to a service, like Salesforce or Amazon S3, your org admin must enable the output connector in setup and you must create an output connection in the Connect tab.

15. To save the recipe, click **Save**.

16. Enter a recipe name and description, and then click **Save** again.

17. To change the recipe name, click the Edit Recipe Name button (pencil icon) and enter a new label.

As you build recipes, feel free to share your feedback with us. Click **Got Feedback** in Data Prep to tell us what's on your mind.

SEE ALSO:

Tableau CRM Help: Nodes for Data Prep Recipes

Tableau CRM Help: Transformations for Data Prep Recipes

Tableau CRM Help: Preview Results in a Data Prep Recipe

Tableau CRM Help: Profile Columns to Understand Data in a Data Prep Recipe

Run Recipes to Get Data into Salesforce

In Data Manager, you can manually run them. When you run a recipe, Data Prep prepares the source data and then loads the results into the target based on the recipe definition. To update the target with the latest synced data, schedule data syncs to run before the corresponding recipes.

Use time-based scheduling to ensure that fresh data is available by a particular time or to run the job during non-business hours. You can schedule a recipe to run every 15 minutes, hourly, weekly, monthly, or on specific days of the week or dates.

1. To open Data Manager, select **Data Manager** from the app picker.

2. To open the Recipes tab, click **Recipes**.

3. To run a recipe, click the drop down ( ) next to the recipe and select **Run Now**.

> 📝 **Note:** To secure access to sensitive data in datasets and connected objects, you also need the Edit Analytics Dataflows user permission when working with recipes.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available for an additional cost in: **Enterprise**, **Performance**, and **Unlimited** Editions

USER PERMISSIONS

To run or schedule a recipe:
- Edit Analytics Dataflows OR Edit Analytics Dataflows

Monitor Sync and Recipes

Use Data Manager's Monitor tab to monitor and troubleshoot data sync and recipe jobs. The tab provides the status, job type, start time and duration for each job. To filter the jobs with a particular status, click the corresponding subtab, like Queued or Failed. You can also view error messages about a job.

1. To open Data Manager, select **Data Manager** from the app picker.

2. To open the Monitor tab, click **Monitor**.

3. To locate a recipe, search for the recipe by name in the top right corner.

4. To filter the jobs by status, select one of the subtabs at the top of the page.

If there's a problem with the recipe logic, edit the recipe and then run it again.

View Data Sources and Targets

Use Data Manager's Data tab to access data available for recipes. You can view datasets created by recipes that can be used as sources in other recipes. You can also view all connected objects.

To edit a dataset, click the drop down (⬇) for the dataset on the Dataset subtab and select **Edit**. You can change the app, security predicate, and metadata file of a dataset. You can set up actions on dataset fields. You can restore a previous version of a dataset. You can also delete the dataset.

On the Connected subtab, you can view all connected objects as well as their connection name and the last time they ran. To edit or delete an existing connection, click the drop down (⬇) and select **Edit** or **Delete**.

✏️ **Note:** Ignore the Live subtab. Live connections are not supported by Data Pipelines.

View Data Pipeline Usage

Use Data Manager's Limits tab to monitor usage and to ensure you don't hit the limits. The limits vary based on your licenses. If needed, contact Salesforce to increase your org's limits.

In Data Manager, the Limits tab shows org-level limits only.

To view other Data Pipelines limits, see the online help.

SEE ALSO:

Data Pipelines Limits

Query Datasets with the Query API Endpoint

If your recipes write results to datasets, you can query the datasets using the Query API endpoint. Although this endpoint is part of the Tableau CRM REST API, you can use it to query datasets built with Data Pipelines. This endpoint supports SAQL and SQL queries. You can use this endpoint to create an application that queries data in datasets.

📝 Note: Other Tableau CRM REST API endpoints aren't available without purchasing a Tableau CRM license.

SEE ALSO:

Tableau CRM Help: Query Resource (API Endpoint)

Tableau CRM Help: Analytics Cloud SAQL Reference

Tableau CRM Help: SQL for Tableau CRM Developer Guide

Tableau CRM Help: Analytics SDK Developer Guide

INDEX