

Analyze Your Data

Salesforce, Spring '17





CONTENTS

ANALYTICS					. 1
Wave Analytics					. 1
Reports and Dashboards					56
WAVE ANALYTICS,	REPORTS,	AND DA	ASHBOARDS	RESOURCES 72	27
INDEX					29

ANALYTICS

Wave Analytics

Salesforce Wave Analytics is a cloud-based platform for connecting data from multiple sources, creating interactive views of that data, and sharing those views in apps. It's a better way to distribute insight to business users so they can understand and take action on changing information.

Work with Wave	
© Get Started Using Wave	Wave Analytics Glossary Trailhead: Wave Analytics Basics Trailhead: Wave Dashboard Navigation
Collaborate and Take Action from Wave	Share a Visualization Share Wave with Communities
Go Mobile with Wave	Wave Analytics for iOS Wave Analytics for Android Trailhead: Wave Mobile Exploration

Enable Wave for Your Org	
Set Up Wave Access PDF	Wave Analytics Security Implementation Guide PDF Trailhead: Wave Analytics Administration Basics
Enable Wave Features	Wave Analytics Limits Wave Analytics Limitations
Manage Wave Datasets PDF	External Data API Guide External Data Format Reference Trailhead: Wave Data Integration Basics

Produce Wave Assets	
Explore Your Data	Visualize Data with Charts Trailhead: Wave Desktop Exploration Trailhead: Smart Data Discovery Basics

Produce Wave Assets	
Build Wave Dashboards	Trend Data in Wave Add Widgets to the Dashboard Trailhead: Wave Dashboard Designer Basics Trailhead: Create a Product Pipeline Dashboard with Wave Charts
© Embed and Customize Dashboards	Extended Metadata (XMD) Reference Bindings Developer Guide for Wave Dashboards Dashboard JSON Reference
© Organize and Share Wave Assets	
Develop on the Wave Platform	Wave REST API Developer's Guide SAQL Reference Wave Migration, Packaging, and Distribution

Accelerate with Prebuilt Wave Apps	
1 The Sales Wave Analytics App	Trailhead: Sales Wave App
The Service Wave Analytics App	Trailhead: Service Wave App
Event Monitoring Wave App	Trailhead: Event Monitoring
Wave for B2B Marketing App	

Find Insights in Your Data with Einstein Data Discovery	
© Get Started Using Einstein Data Discovery	Einstein Data Discovery Setup Guide Trailhead: Einstein Data Discovery Basics

Learn Wave Analytics with Trailhead

Trailhead is the fun and free way to learn Salesforce. Now you can get acquainted with Wave Analytics through Trailhead, using a special Wave-enabled Developer Edition org.



For the Wave Analytics trails, you can't use an old Developer Edition org. You must sign up for the special Developer Edition that comes with a limited Analytics Cloud Wave Platform license and contains sample data required for the Wave trails.

You'll find the sign up for the special Developer Edition org in these trails:

- Explore with Wave Analytics: http://trailhead.salesforce.com/trail/wave_analytics_explorer
- Build and Administer Wave Analytics: https://trailhead.salesforce.com/trails/wave_analytics_enable_and_produce
- Accelerating Analytics with Wave Apps: http://trailhead.salesforce.com/trail/wave_analytics_apps

SEE ALSO:

Wave Analytics Limits

Get Started Using Wave

With just a few clicks, you can run Wave apps, dashboards, and lenses.

IN THIS SECTION:

Open Wave Analytics

Go to Wave Analytics by opening the app menu or app launcher in the Salesforce header and then clicking **Wave Analytics**. Alternatively, access your personal Analytics home from a tab within Salesforce.

Learn Your Way Around Wave

The Wave Analytics tab is your home base for exploring your data.

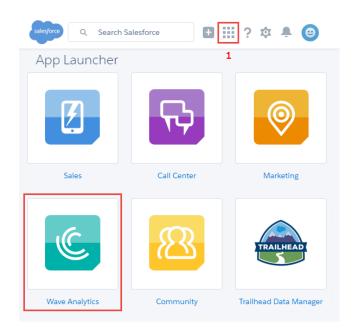
Explore with Wave Assets

Introduce yourself to the Wave Analytics way of data exploration.

Open Wave Analytics

Go to Wave Analytics by opening the app menu or app launcher in the Salesforce header and then clicking **Wave Analytics**. Alternatively, access your personal Analytics home from a tab within Salesforce.

The method for opening Wave Analytics depends on the user experience enabled in your org. In Lightning Experience, click the app picker (1), and then click the **Wave Analytics** app.



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

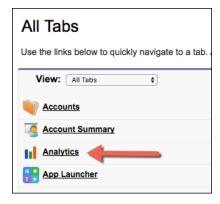
To access Wave Analytics:

"Use Wave Analytics"

In Salesforce Classic, select **Wave Analytics** from the app picker.



You can also get started with Wave from the **Analytics** tab in your Salesforce org. If you don't see the tab, your administrator sets it up like any new tab.



Learn Your Way Around Wave

The Wave Analytics tab is your home base for exploring your data.

IN THIS SECTION:

Pin Apps to Your Analytics Home

Wave apps are where you run your dashboards, and you can pin your favorite apps to your personal Analytics home page.

Run Apps

Start your Wave app to run and present its dashboards.

Track Notifications

The lower part of the Analytics home page tracks the progress on the notifications that you've set.

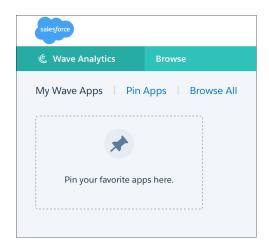
Browse Your Wave Assets

To view or search organized lists of apps, dashboards, and lenses, click **Browse All** on the home page.

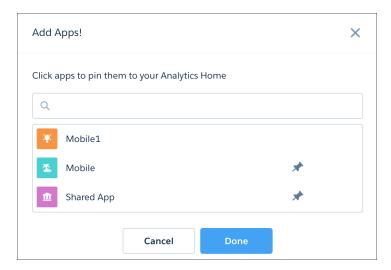
Pin Apps to Your Analytics Home

Wave apps are where you run your dashboards, and you can pin your favorite apps to your personal Analytics home page.

The first time you click the Wave Analytics tab, you'll see tiles for your three most recently used Wave apps. If you've unpinned all your apps, you're prompted to add apps to the home page.



Click **Pin Apps** to select the apps that you want to access regularly. Or to unpin an app, click it to remove the pin.



Apps appear on your Wave Analytics home in the order that you pinned them.



Note: If you don't have any apps yet, go to Create an App.

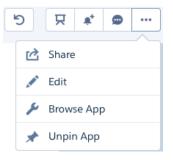
Run Apps

Start your Wave app to run and present its dashboards.

Click an app, either one that's pinned on your Wave Analytics home page or one listed under the Browse tab. Clicking an app opens it in presentation mode, displaying the first dashboard on your navigation list.

To open the navigation panel, click the navigation icon (). The panel lists the available dashboards. Click the icon again to close the panel.

When the navigation panel is open, click a dashboard to start it. Dashboards let you explore live, interactive views of your data. Go ahead and click the widgets, toggles, and charts. When you want to return the dashboard to its initial state, click the refresh icon (). In addition, from the running app you can:

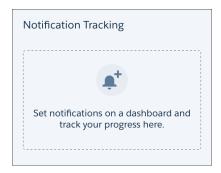


- Present full screen
- Set and manage notifications
- Annotate the dashboard

- Share the dashboard
- Open the dashboard in the dashboard designer in order to edit it (if you have permission and access)
- Browse the contents of the app
- Unpin the app from the home page (if you have permission and access).

Track Notifications

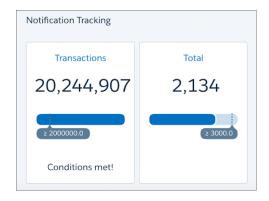
The lower part of the Analytics home page tracks the progress on the notifications that you've set.



Click an app on your Wave Analytics home page and navigate to a dashboard with KPIs. The KPI numbers are actually dashboard widgets that you can use to set notifications. Hover over the number, locate the dropdown menu, and select **Set Notification**.

To learn how to set notifications, see Set Smart Notifications for Your Most Important Business Metrics.

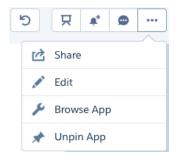
All your active notifications appear on your Analytics home page. The notifications tracking gauges are designed to show the progress toward the conditions set in your notification.



Browse Your Wave Assets

To view or search organized lists of apps, dashboards, and lenses, click **Browse All** on the home page.

You can also view the contents of a running app by selecting **Browse App** from its menu.



Wave's Browse tab gives you options to view all assets, sort by asset, and search. Lists of assets are arranged in most recently used order. Click **Show More** next to an asset to see the full list.

Click the list view icon () to view assets with their descriptions and additional information. Click the tile view icon () to go back to tiles with thumbnail images of dashboards and lenses.

In list view, relevant timestamps are provided, depending on the asset. For lenses and dashboards, Modified is the last time the asset was edited, and Refresh is the last time the asset's dataset was updated. If a dashboard contains components that reference more than one dataset, the Refresh timestamp will reflect the most recently updated dataset.

Explore with Wave Assets

Introduce yourself to the Wave Analytics way of data exploration.

IN THIS SECTION:

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 Wave Analytics

Wave collects and organizes your data in datasets, lenses, dashboards, and apps. These collections, also known as "Wave 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 Wave 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 Wave 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!

SEE ALSO:

Explore Your Data

Build Wave Dashboards

Put It All Together with Wave Apps

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 Wave Analytics

Wave collects and organizes your data in datasets, lenses, dashboards, and apps. These collections, also known as "Wave 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 Wave Analytics retrieves information from the source data.

A *measure* is a quantitative value, 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 Wave Analytics, 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.

• 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 Wave Analytics returns an incorrect result.

A *dimension* is a qualitative value, like region, product name, and model number. 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 Wave 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 Wave Analytics 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.

IN THIS SECTION:

Key Elements of a Wave Dashboard

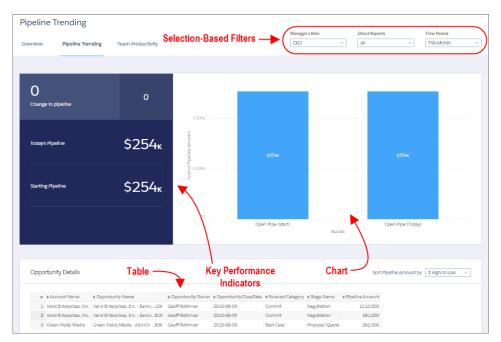
A dashboard isn't just for reading. It's for carrying on a conversation based on one or more datasets that matter to your company. Don't just look at the pictures—use the tools on the dashboard to dive deeper into the data that you care about. Go ahead, click around, and explore!

SEE ALSO:

Build a Dashboard with the Wave Dashboard Designer Build a Dashboard with the Classic Designer

Key Elements of a Wave 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 Wave 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. There are different types of selection-based filter widgets: date, list, range, and toggle.

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.

Work with Wave

Personalize your own Wave experience and collaborate with Wave Analytics apps with features such as notifications, annotations, presentation mode, and downloading filtered data from Wave. In addition, custom menus in lenses and dashboards allow you to take action in Salesforce directly from Wave.

IN THIS SECTION:

Set Smart Notifications for Your Most Important Business Metrics

With notifications, Wave works for you, when and where you choose. Attach notification criteria to number widgets in dashboards, and select when to run the queries. You can see notifications in Wave, Lightning Experience, and email. Notifications are available in dashboards built in the Wave dashboard designer.

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 Wave 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.

Download Images and 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).

Incorporate Visualizations through Chatter Posts and Sharing Links

Share a visualization with your colleagues by posting to Chatter, or by getting its unique URL. A Chatter post provides an image and a link to the asset—lens, dashboard, or app—in Wave Analytics. Colleagues with the link and access to the asset can drill in and explore the information that's presented.

Take Control of Your Salesforce Experience with Wave Custom Actions Menus

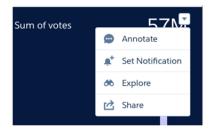
Wave custom actions menus in charts and tables let you take advantage of Salesforce actions—such as posting to Chatter, creating an event or opportunity, or updating a record—and go directly to Salesforce records.

Set Smart Notifications for Your Most Important Business Metrics

With notifications, Wave works for you, when and where you choose. Attach notification criteria to number widgets in dashboards, and select when to run the queries. You can see notifications in Wave, Lightning Experience, and email. Notifications are available in dashboards built in the Wave dashboard designer.

To set up a Wave notification, follow these steps.

- 1. Locate the number widget on your dashboard that shows the KPI you want to be notified about. Optionally, you can use the dashboard's filters to narrow the query against your data and create a personalized KPI.
- 2. Select **Set Notification** from the drop-down list.



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

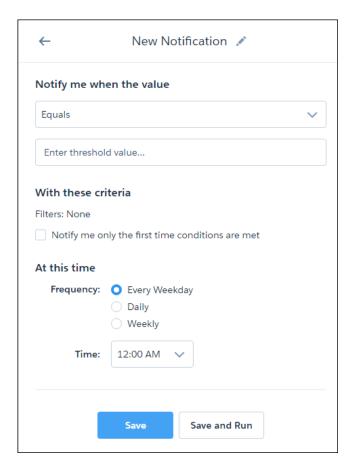
To set notifications on Wave Analytics dashboard widgets:

"Use Wave Analytics"

3. In the New Notification panel, set the value that you want to know about as soon as your widget query returns it.

For example, say that you want to know when your team has reached 75% of quota. You'd click **Set Notification** on the Quota Attainment number widget. In the New Notification panel, you'd select **Equals or is greater than** from the drop-down list and enter 75 for the threshold value.

4. Select the frequency and time when you want the query to run.



5. Test your notification on the spot with **Save and Run**, or simply save it. No administrator setup is required.

When the conditions you've set for the KPI are met, you're notified in the app and via email. In Wave and Lightning Experience, the bell icon () informs you of notifications.

In Wave, the drop-down list from the bell includes the option to manage Wave notifications. You can also manage notifications on the dashboard. The bell icon with the plus (*) opens the list of notifications and shows numbers in blue (*) on widgets that have notifications attached. In the All Notifications panel, each notification has a drop-down action list for editing and deleting the notification.



Take note of these considerations for Wave notifications.

• You can set up to five notifications per user.

- Notifications are available only for dashboards designed in the Wave dashboard designer.
- Notifications are available only from number widgets.
- For queries with groupings, all results are evaluated for the notification criteria, not just the first row (which is the value shown in the widget).

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.

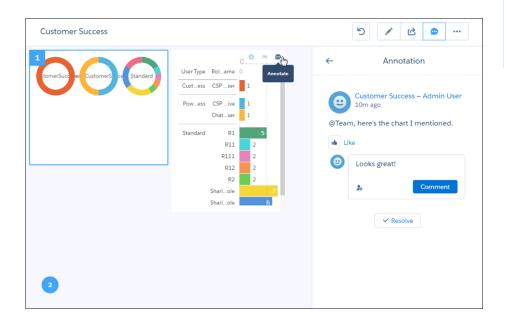


Note: Annotations are supported only in dashboards built using the Wave dashboard designer.

To create annotations, select Annotate from the drop-down list on a widget, or click the speech bubble icon () in the dashboard control bar. Widgets with annotations show a blue square with the number of open annotations.

Wave 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 an annotation in a blue square or circle to open it for comments.



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 annotate Wave Analytics dashboard widgets:

"Use Wave Analytics"

Although you can @mention anyone in your org, the security you have set up controls access to the dashboard.

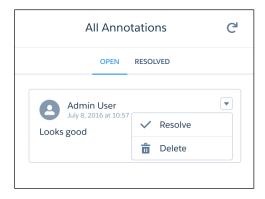
Wave dashboard annotations are natively integrated with Chatter, and comments on dashboards also appear as Chatter posts. The

Annotation panel includes the option to post a screenshot of the current state of the dashboard. On an open annotation, you can refresh the feed or close the annotation.



← Widget Annotations C

In the dashboard panel, each annotation has a drop-down action list for resolving or deleting the annotation.



You receive all posts on a dashboard if you click its + Follow button. When you see annotations in Chatter, click the dashboard image to navigate directly to the annotation on the dashboard in Wave.

SEE ALSO:

Enable Annotations on Dashboard Widgets

Present Live, Interactive Dashboards

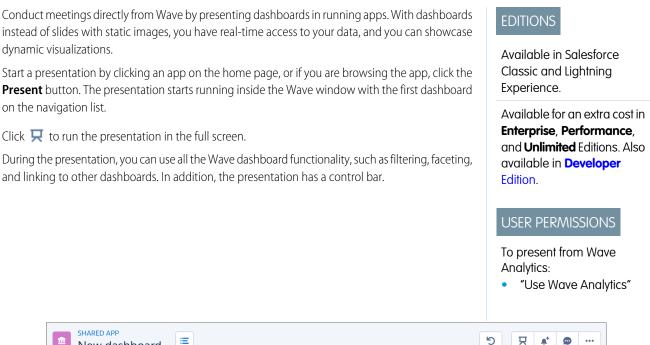
instead of slides with static images, you have real-time access to your data, and you can showcase dynamic visualizations. Start a presentation by clicking an app on the home page, or if you are browsing the app, click the **Present** button. The presentation starts running inside the Wave window with the first dashboard on the navigation list.

Click 😾 to run the presentation in the full screen.

SHARED APP

New dashboard

During the presentation, you can use all the Wave dashboard functionality, such as filtering, faceting, and linking to other dashboards. In addition, the presentation has a control bar.



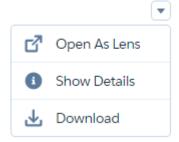
You can share, download, access the dashboard designer, set notifications, and collaborate in a dashboard's feed with annotations.

To move to another dashboard, click is to open the navigation panel and select the next dashboard to present. Click is again to close the navigation panel.

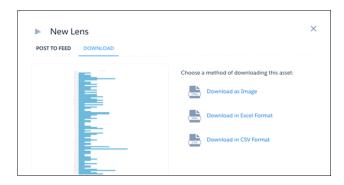
Download Images and Filtered Data

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 guery (or step).

Select **Download** from the drop-down menu on a dashboard widget.



Or, from a lens, select **Share** and then **Download**.



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 download screenshots and data in tabular format through the Wave Analytics user interface:

 "Download Wave Analytics Data"

When downloading filtered data, the number of rows is limited by the query. The default limit per query is 10,000 rows of data. To set a different value, use the SAQL limit statement.

SEE ALSO:

Enable Downloading from Wave Analytics

Incorporate Visualizations through Chatter Posts and Sharing Links

Share a visualization with your colleagues by posting to Chatter, or by getting its unique URL. A Chatter post provides an image and a link to the asset—lens, dashboard, or app—in Wave Analytics. Colleagues with the link and access to the asset can drill in and explore the information that's presented.

1. Click Share.



- 2. Click the tab for the sharing method that you want: **Post to Feed**, which posts to Chatter an image and link to the lens, dashboard, or app; or **Get URL**, which is 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 that posted images are public.
 - **b.** If you're getting the unique URL, copy the link, and then paste it wherever you want to share it.
- **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 Wave Analytics 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.

Take Control of Your Salesforce Experience with Wave Custom Actions Menus

Wave custom actions menus in charts and tables let you take advantage of Salesforce actions—such as posting to Chatter, creating an event or opportunity, or updating a record—and go directly to Salesforce records.

With custom actions menus, you can use quick actions that have been added to Salesforce objects—directly from Wave 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 for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To view visualizations:

"Use Wave Analytics"

EDITIONS

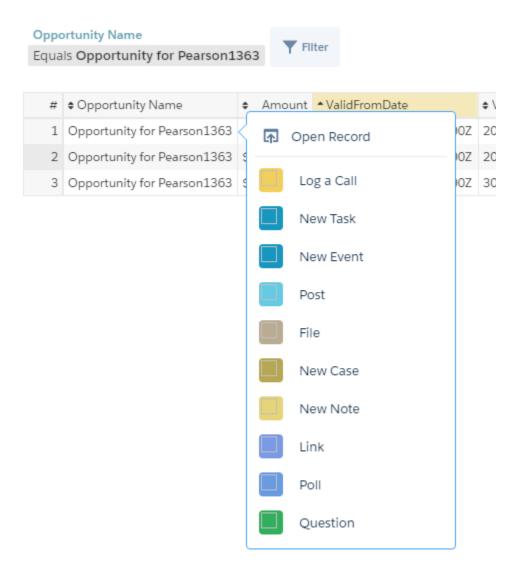
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

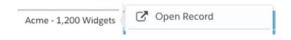
To use custom actions menus:

"Use Wave Analytics"



You can access actions menus from the following parts of charts and tables in Wave:

• Value labels on charts (such as "Opportunity for Pearson" in the image above).



• Legend value labels in table cells.



• Values in table cells.

Analytics Go Mobile with Wave



Hover over the labels to see if an actions menu is available. Click the value, cell, or legend displays the menu, to see the menu's contents and select from the available actions or open a record.

Developers or administrators edit the Extended Metadata (XMD) file for a dataset to create menus. To learn more, see Enable Actions and Links with Custom Actions Menus in Lenses and Dashboards

Go Mobile with Wave

Find resources for Wave Analytics for Android and iOS.

Mobile Topics	Resources
Get the Apps	iTunes Google Play
Log In and Get Started	Wave Analytics for iOS Wave Analytics for Android
Optimize Dashboard Layouts for Mobile	Generate Unique Wave Dashboard Layouts for Different Devices Mobile Dashboard Layouts JSON Reference
Administer Wave Mobile	Connected App for Wave Analytics on Mobile
Know the Limitations	Limits for Using Wave Analytics on iPhone® and iPad® Limits for Using Wave Analytics on Android Devices

Wave Platform Setup

Set up your organization to use the Salesforce Wave Analytics platform by enabling Wave and creating and assigning permission sets.

The information here describes how to set up your organization to use the Wave platform. Read Prebuilt Wave Apps to get started with and set up Sales Wave Analytics, Service Wave, and other apps.

Each Analytics Cloud - Wave Analytics Platform license is a single-user license that provides access to the Salesforce Wave Analytics cloud. It includes single Analytics Cloud - Sales Wave Analytics App and Analytics Cloud - Service Wave Analytics App single-user licenses. The table shows data storage limits for each license. If you require more data, you can purchase Analytics Cloud - Additional Data Rows, which entitles you to an additional 100 million rows.

License	Limit
Analytics Cloud - Wave Analytics Platform	100 million rows

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

License	Limit
Analytics Cloud - Sales Wave Analytics App	25 million rows when used without Analytics Cloud - Wave AnalyticsPlatform license. Use of Sales Wave app license does not increase data limit for platform license
Analytics Cloud - Service Wave Analytics App	25 million rows when used without Analytics Cloud - Wave AnalyticsPlatform license. Use of Service Wave app license does not increase data limit for platform license
Analytics Cloud - Additional Data Rows	100 million rows

Important:

- Wave Analytics license data storage limits are contractual, not technical. Licensee agrees to strictly monitor its total number of data rows.
- If you purchased a Wave Analytics Platform license before October 20, 2015 with Analytics Cloud Builder or Analytics Cloud Explorer permission set licenses: Read Set up the Wave Analytics Platform With Licenses Purchased Before October 20, 2015.
 If you're migrating users from Builder or Explorer licenses to the new Analytics Cloud Wave Analytics Platform license: See Migrating From Wave Licenses Purchased Before 10/20/2015 to New Wave Platform Licenses before you start the setup process for those users.
- If you disable Wave Analytics, user permissions are removed from each defined permission set. If you re-enable Wave Analytics later, you must define the permission sets again.

Wave Platform Setup Process

Follow these basic steps to set up your organization to use the Wave Analytics platform.

- 1. Enable Wave for your organization.
- **2.** Define the main Wave user types in your organization.
- **3.** Create permission sets to group related user permissions according to the needs of each user type.
- **4.** Assign a permission set to an individual user, which also *auto-assigns* the Analytics Cloud Wave Analytics Platform permission set license to that user.





You can assign a Wave permission set license along with any of the following Salesforce user licenses:

- Force.com (app subscription)
- Force.com (one app)
- Full CRM
- Salesforce Platform
- Salesforce Platform One

For best results, follow the steps for setting up the Wave Analytics platform in the order shown here.

IN THIS SECTION:

1. Learn About Wave Analytics Cloud Permission Set Licenses and User Permissions

Wave Analytics permission set licenses enable the permissions to explore data with and manage the Salesforce Wave Analytics platform. They also let Salesforce Community users view Wave dashboards.

2. Identify Wave Analytics Platform User Types

Identifying types of Wave platform users helps assure that the setup process meets your team's analytics needs.

3. Enable Wave Analytics and Create Permission Sets

After Salesforce provisions you with the Wave Analytics platform license, enable Wave and create permissions.

4. Assign Wave Analytics Permission Sets to Users

Assign permission sets to one or more users—either one at a time, or in bulk—to give them access to Wave functionality.

5. Complete Setting up the Wave Analytics Platform

Enhance the Wave platform user experience and fine-tune Wave access to Salesforce data with optional setup procedures.

SEE ALSO:

Set up the Wave Analytics Platform With Licenses Purchased Before October 20, 2015
Migrating From Wave Licenses Purchased Before 10/20/2015 to New Wave Platform Licenses

Learn About Wave Analytics Cloud Permission Set Licenses and User Permissions

Wave Analytics permission set licenses enable the permissions to explore data with and manage the Salesforce Wave Analytics platform. They also let Salesforce Community users view Wave dashboards.

- Each Wave platform user needs an Analytics Cloud Wave Analytics Platform permission set license to use the Wave platform.

 Creating a permission set and assigning it to a user *auto-assigns* the Wave platform permission set license to that user. You can also manually assign permission set licenses to users.
- Community users need an Analytics Cloud Wave Community Users permission set license to view Wave apps shared via dashboards embedded in Visualforce pages.

The Analytics Cloud - Wave Analytics Platform permission set license enables the following permissions.

User Permission	What It Enables
"Use Wave Analytics"	Accessing Wave Analytics. Viewing and exploring datasets. Viewing and sharing lenses and dashboards.
"Create and Edit Wave Analytics Dashboards"	Creating, editing, and deleting Wave Analytics dashboards.
"Create Wave Analytics Apps"	Creating, editing, deleting, and sharing Wave Analytics applications.
"Upload External Data to Wave Analytics"	Uploading external data to Wave Analytics to create a dataset. Monitoring dataflow and system jobs in the data monitor.
"Edit Wave Analytics Dataflows"	Editing, starting, stopping, and rescheduling the dataflow. Creating, editing, and running dataset recipes. Monitoring dataflow and system jobs in the data monitor. 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 Wave Analytics.

"Manage Wave Analytics"	Accessing all Wave Analytics features. Provides Wave administrator-level capabilities.
"Download Wave Analytics Data"	Downloading screenshots and data in tabular format through the Wave Analytics user interface.



Note: When you create permission sets for Wave users, selecting any Wave Analytics permission automatically enables the "Use Wave Analytics" 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 Wave Analytics" permission, the user must also have Viewer access on an app to view its datasets, lenses, and dashboards.

The Analytics Cloud - Wave Community Users permission set license enables the "View Wave Analytics on Communities Pages" permission. That permission enables external users to view Wave dashboards embedded in VisualForce pages in their communities. See Enable Wave Analytics for Communities on page 27.

Identify Wave Analytics Platform User Types

Identifying types of Wave platform users helps assure that the setup process meets your team's analytics needs.

The Wave Analytics Platform license includes the permissions that enable users in your organization to import, work with, and explore data using the Wave platform. The license also includes permissions that let administrators manage Wave. To set up Wave in the most useful way, think through the different ways people access Wave 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 Wave dashboards, lenses, and datasets and occasionally import data to Wave.
- Administrators or managers, who access to the full range of Wave functionality to create apps, dashboards, lenses, and datasets, edit data, and otherwise customize the Wave experience.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

Accordingly, when you set up the Wave 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 Wave.
- A "manage" permission set that's reserved for a select few administrators and managers and contains permissions to use all Wave features

You can set up Wave Analytics 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 Wave Analytics and Create Permission Sets

After Salesforce provisions you with the Wave Analytics platform license, enable Wave and create permissions.

Enable the Wave Platform

To use the Wave platform, first enable it for your organization.

1. In the Salesforce Setup menu, under Administer, click **Analytics Cloud | Getting Started**.

2. Click Enable Analytics.

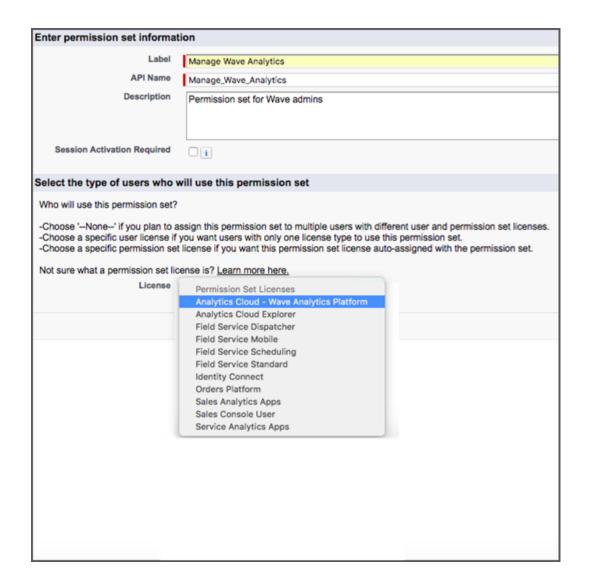
Create Permission Sets

The next step in the setup process is to create permission sets made up of Wave user permissions. We show you how to create two permission sets:

- Manage Wave permission set, for users who create and administer Wave.
- View Wave permission set, for users who explore app dashboards and datasets.

Mote:

- Customers who purchased Wave before October 20, 2015: See Set up the Wave AnalyticsPlatform With Licenses Purchased Before October 20, 2015 before following the instructions here.
- The Manage Wave Analytics permission set enables the equivalent functionality provided by the previous Wave Analytics Builder license (purchased before October 20, 2015). If you have a Builder license or you're migrating to the current Wave Analytics platform license, follow these steps to create a permission set with the same functionality.
- The View Wave Analytics permission set enables the equivalent functionality provided by the previous Wave Analytics Explorer license (purchased before October 20, 2015). If you have an Explorer license or you're migrating to the current Wave Analytics 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 Wave Analytics or View Wave Analytics 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.** Select the permission set license to associate with this permission set by using the License drop-down option. In this case, select the Analytics Cloud Wave Analytics Platform permission set license.



If you select a specific permission set license, any user assigned to the permission set is *auto-assigned* the permission set license. If you leave the default of --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 Wave platform license. Then click **Edit**.
- **6.** Do one of the following.
 - If you're creating the Manage Wave Analytics permission set, select "Manage Wave Analytics" user permissions and click Save.
 You've successfully created a permission set that enables access to all Wave features. You don't have to select any of the other individual permissions. Assign this permission set sparingly because it provides access to all Wave features, many of which are inappropriate for most analytics users.
 - If you're creating the View Wave Analytics permission set, select "Use Wave Analytics Cloud" and "Upload External Data to Wave Analytics" user permissions and click **Save**. You've successfully created a permission set that enables a user to view the Wave 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 Wave.

Assign Wave Analytics Permission Sets to Users

Assign permission sets to one or more users—either one at a time, or in bulk—to give them access to Wave functionality.



Walk Through It: assign a permission set

- 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
 Wave Analytics and Create Service Wave Permission Sets on page 22, select either the Manage Wave Analytics or View Wave Analytics permission sets.
- 3. Click Manage Assignments and click Add Assignments.
- **4.** Select the users who need access to Wave, and click **Done**. When you assign the permission set to users, Salesforce *auto-assigns* the Analytics Cloud Wave Analytics Platform *permission set license* to those users.

Assign the Manage Wave Analytics permission set sparingly, since it provides access to all Wave features. Assign it only to users who administer or manage the Wave platform.

Complete Setting up the Wave Analytics Platform

Enhance the Wave platform user experience and fine-tune Wave access to Salesforce data with optional setup procedures.

IN THIS SECTION:

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 Wave Analytics for Communities

Securely share Wave Analytics apps with community partner and customer users.

Enable Downloading from Wave Analytics

Enable users to download the results from lens explorations and dashboard widgets as images (.png), Microsoft® Excel® (.xls), or comma-separated values (.csv) files.

Grant Users Access to the Wave Dashboard Designer and Classic Designer

To grant users access to either dashboard designer, assign each of them the "Create and Edit Wave Analytics Dashboards" user permission. For users to access the Wave dashboard designer, the **Enable Wave Dashboard Designer** setup option must also be enabled, which it is by default.

Enable Annotations on Dashboard Widgets

Annotate dashboard widgets with comments posted in the dashboard and in Chatter.

Enable the Wave REST API

You can access Wave Analytics datasets and lenses programmatically using the Wave REST API. Selecting this option overrides the "APIEnabled" permission for individual users and gives all Wave users (including Community users) access to the Wave API.

Connected App for Wave Analytics on Mobile

Install the Salesforce Wave Analytics 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.

USER PERMISSIONS

To assign a permission set to users:

"Assign Permission Sets"

Wave Analytics Security Implementation Guide

Wave Analytics 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 Wave Analytics

Wave Analytics requires access to Salesforce data when extracting the data and also when the data is used as part of row-level security. Wave Analytics gains access to Salesforce data based on permissions of two internal Wave Analytics users: Integration User and Security User.

Set Up a Whitelist of Trusted Sites for Embedded Wave Dashboards

Specify your trusted sites in the Wave Analytics whitelist, you can include embedded Wave dashboards in websites and apps outside of Salesforce servers.

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.

- (1) 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 Wave 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.

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 modify settings:

Enable Wave Analytics for Communities

Securely share Wave Analytics apps with community partner and customer users.



Note: Only users with a Customer Community Plus or Partner Community license can use this feature. This feature is supported in communities but not in portals.

To set up Wave Analytics for sharing with Communities, complete the following steps.

- 1. From Setup, enter Wave Analytics in the Quick Find box, then select **Settings**. Select Enable Wave Analytics for Communities.
- **2.** Assign your community members the Analytics Cloud Wave Community Users permission set license.
- **3.** Create a permission set that includes the "View Wave Analytics on Communities Pages" permission and assign it to your community members.

EDITIONS

Available in: Salesforce Classic

Available for an extra cost in: **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in: **Developer** Edition

USER PERMISSIONS

To modify settings:

 "Manage Wave Analytics"

To view Wave dashboards and lenses:

"View Wave Analytics on Communities Pages"

Enable Downloading from Wave Analytics

Enable users to download the results from lens explorations and dashboard widgets as images (.png), Microsoft® Excel® (.xls), or comma-separated values (.csv) files.

To set up users to download data from Wave, complete the following steps.

- 1. From Setup, enter *Permission Sets* in the Quick Find box, and then select **Permission Sets** under Manage Users.
- **2.** Select the Analytics Cloud Permission Set where you want to add the permission, and then select **System Permissions**.
- 3. Select Download Wave Analytics Data.
- 4. Click Save.

SEE ALSO:

Download Images and Filtered Data

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 modify settings:

Grant Users Access to the Wave Dashboard Designer and Classic Designer

To grant users access to either dashboard designer, assign each of them the "Create and Edit Wave Analytics Dashboards" user permission. For users to access the Wave dashboard designer, the **Enable Wave Dashboard Designer** setup option must also be enabled, which it is by default.

If Wave dashboard designer is disabled, complete the following tasks to enable it.

- 1. From Setup, enter Wave Analytics in the Quick Find box, then select Settings.
- 2. Select Enable Wave Dashboard Designer, and then click Save.
- **3.** For each user that builds dashboards, assign the "Create and Edit Wave Analytics Dashboards" user permission to them.

Each user has a choice whether to use the classic designer or Wave dashboard designer for each dashboard.

SEE ALSO:

Build a Dashboard with the Wave Dashboard Designer

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 access the Wave dashboard designer or classic designer:

 "Create and Edit Wave Analytics Dashboards"

Enable Annotations on Dashboard Widgets

Annotate dashboard widgets with comments posted in the dashboard and in Chatter.



Note: Annotations are supported only in dashboards built with the Wave dashboard designer.

To enable annotations, complete the following steps.

- From Setup, enter Feed Tracking in the Quick Find box, and then select Feed
 Tracking under Chatter.
- 2. Scroll down and select Wave Asset.
- 3. Select Enable Feed Tracking.
- 4. Click Save.

SEE ALSO:

Collaborate with Dashboard Annotations

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 modify settings:

Enable the Wave REST API

You can access Wave Analytics datasets and lenses programmatically using the Wave REST API. Selecting this option overrides the "APIEnabled" permission for individual users and gives all Wave users (including Community users) access to the Wave API.

- 1. From Setup, enter Wave Analytics in the Quick Find box, then select **Settings**.
- 2. Select Enable the Wave API for all Wave users.
- 3. Click Save.

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 modify settings:

 "Manage Wave Analytics"

Connected App for Wave Analytics on Mobile

Install the Salesforce Wave Analytics 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 https://login.salesforce.com/packaging/installPackage.apexp?p0=04tB0000000DKTA.
- 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.** After clicking **Done**, click the Salesforce Wave Analytics App Package and then **View Components**.
- **4.** Select iOS or Android. To configure policies and other settings, click **Edit**. When you are 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.

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 and configure Connected App:

Attribute Key	Mobile clients can share	Mobile Device
SHARE_IMAGE_AIRDROP	Images via AirDrop	iOS
SHARE_IMAGE_ CAMERAROLL	Images from Camera Roll	iOS
SHARE_IMAGE_ OPEN	The image that's currently open on the mobile device	iOS

Attribute Key	Mobile clients can share	Mobile Device
SHARE_IMAGE_ S1	Images in Salesforce1	iOS
SHARE_IMAGE_ EMAIL	Images through email	iOS
SHARE_LINK_NFC	Links via Near Field Communication (NFC)	Android
SHARE_LINK_BLUETOOTH	Links over Bluetooth	Android
SHARE_LINK_AIRDROP	Links via AirDrop	iOS
SHARE_LINK_ CLIPBOARD	The link that's currently in the clipboard of the mobile device	iOS and Android
SHARE_LINK_S1	Links in Salesforce1	iOS and Android
SHARE_LINK_EMAIL	Links through email	iOS and Android
SPOTLIGHT_SEARCH	Mobile clients can see Wave Analytics 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 Wave-enabled org, from Setup, enter *Connected Apps* in the Quick Find box, then select **Connect Apps OAuth Usage**.

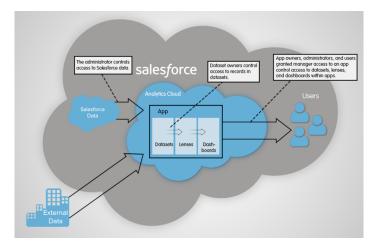
You can block user sessions, revoke individual users, and drill in to user details. For more information, see Monitoring Usage for a Connected App.

Note: For certificate-based authentication, users must set their desktop browsers to allow cookies.

For additional 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.

Wave Analytics Security Implementation Guide

Wave Analytics 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 Wave
 Analytics 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: Wave Analytics doesn't synchronize with sharing that's configured in Salesforce. Instead, you configure security separately in Wave Analytics. Wave Analytics provides a robust row-level security feature that enables you to model many different types of access controls on datasets.

For complete information about implementing Wave Analytics security, see Wave Analytics Security Implementation Guide.

Salesforce Data Access in Wave Analytics

Wave Analytics requires access to Salesforce data when extracting the data and also when the data is used as part of row-level security. Wave Analytics gains access to Salesforce data based on permissions of two internal Wave Analytics users: Integration User and Security User.

Wave Analytics 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, Wave 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 guery the dataset using that predicate.

IN THIS SECTION:

Control Access to Salesforce Objects and Fields

Wave Analytics 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.

Control Access to Salesforce Objects and Fields

Wave Analytics 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.

USER PERMISSIONS

To clone a user profile:

"Manage Profiles and Permission Sets"

To edit object permissions:

- "Manage Profiles and Permission Sets"
 - AND
 - "Customize Application"

Set Up a Whitelist of Trusted Sites for Embedded Wave Dashboards

Specify your trusted sites in the Wave Analytics whitelist, you can include embedded Wave dashboards in websites and apps outside of Salesforce servers.

To set up a Wave Analytics whitelist, complete the following steps.

- From Setup, enter Whitelist in the Quick Find box, and then select Whitelist under Wave Analytics.
- 2. On the Whitelist for Embedding Wave Dashboards page, click **New**.
- **3.** On the Whitelisted Site Edit page, enter the information for the site you want to add to the Wave whitelist.
- 4. Click Save.

Set up the Wave Analytics Platform With Licenses Purchased Before October 20, 2015

If you purchased Wave Analytics Builder or Explorer licenses before October 20, 2015, you can continue to use your licenses to set up the Wave plaform for users in your organization or you can migrate to the new single-user Wave Analytics platform license.

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 modify settings:

 "Manage Wave Analytics"

- Note: This information applies only to customers who purchased Wave Analytics licenses before October 20, 2015, and continue to use the Analytics Cloud Builder or Analytics Cloud Explorer permission set licenses after that without migrating to the new Analytics Cloud Wave Platform license. If you purchased Wave on or after October 20, 2015, go to Wave Platform Setup.
- (1) Important: If you're migrating from the previous Analytics Cloud Builder or Analytics Cloud Explorer platform licenses to the single-user Wave Analytics platform license, read Migrating From Wave Licenses Purchased Before 10/20/2015 to New Wave Platform Licenses.

The Analytics Cloud Wave Platform License

Analytics Cloud Wave Platform license purchased before October 20, 2015, enable you to turn on Wave Analytics for your organization. Each previous Wave Platform license is contractually limited to a maximum of 400 users. If you need more than 400 users, you can purchase additional Wave Platform licenses.

Each previous Wave Platform license enables you to:

- Store up to 250,000,000 rows in your final registered datasets.
- Run up to 50 concurrent gueries in Wave Analytics.

The Analytics Cloud Permission Set Licenses

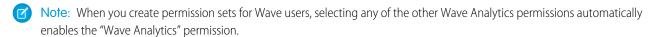
Each user needs an Analytics Cloud permission set license to use the product. The permission set license defines what user permissions can be assigned to the user. The administrator assigns permission set licenses to users.

As part of Wave Analytics platform licenses sold before October 20, 2015, Salesforce provides the following permission set licenses:

- Analytics Cloud Builder permission set license enables users to access all Wave Analytics features.
- Analytics Cloud Explorer permission set license enables users to upload external data to Wave Analytics and explore data.

Here are Wave Analytics user permissions included with each permission set license.

User Permission	Included in Explorer or Builder Permission Set License?	What It Enables
"Create and Edit Wave Analytics Dashboards"	Builder only	Creating and editing Wave Analytics dashboards.
"Create Wave Analytics Apps"	Builder only	Creating and sharing Wave Analytics applications.
"Edit Wave Analytics Dataflows"	Builder only	Downloading, uploading, starting, stopping, and rescheduling the dataflow. Viewing dataflow and system jobs in the data monitor.
"Manage Wave Analytics"	Builder only	Accessing all Wave Analytics features. Provides Wave administrator-level capabilities.
"Upload External Data to Wave Analytics"	Builder and Explorer	Uploading external data to Wave Analytics to create a dataset. Viewing dataflow and system jobs in the data monitor.
"Use Wave Analytics"	Builder and Explorer	Using Wave Analytics, and viewing the datasets, lenses, and dashboards that the user has permission to view. Automatically enabled when you select any other Wave permission.



You can assign previous Wave Analytics permission set licenses along with any of the following Salesforce user licenses:

- Force.com (app subscription)
- Force.com (one app)
- Full CRM
- Salesforce Platform
- Salesforce Platform One

SEE ALSO:

Migrating From Wave Licenses Purchased Before 10/20/2015 to New Wave Platform Licenses

Migrating From Wave Licenses Purchased Before 10/20/2015 to New Wave Platform Licenses

To complete migration from previous Analytics Cloud Builder and Analytics Cloud Explorer licenses purchased before October 20, 2015, to the new Analytics Cloud - Wave Analytics Platform license, you may need to set up Wave users in your organization with new permission sets.

When Salesforce provisions your organization with the new Analytics Cloud - Wave Analytics Platform license, you may have to set up some users with the new license.

- Builder license users. Users who access Wave with the Analytics Cloud Builder permission
 set license should be able to continue to use the Wave Analytics platform without any additional
 setup. The Builder license is simply renamed Analytics Cloud Wave Analytics Platform, and you'll
 see that permission set license in Salesforce Setup.
- 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 Analytics Cloud Wave Platform 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.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

If users in your organization require you to set them up with the new Analytics Cloud - Wave Platform license—even if they used Wave with a previous license—follow the steps in Wave Platform Setup on page 19. You need to assign them the new permission set license, create one or more new permission sets, and assign the permission sets to users.

Update the Sales Wave Analytics Quota (Target) Dataset

If you answer "yes" to the configuration wizard question about whether your business tracks quotas at the user level, you need to update the Sales Wave Analytics Quota dataset so the app includes quotas data for your team.

When you create Sales Wave Analytics, a Quota (Target) dataset is created along with the app. To explore how team members are tracking to their monthly, quarterly, or annual quotas in Sales Wave Analytics, you need to update the Quota dataset by completing the following steps. You can also watch the video Create the Sales Wave App, Part 2: Upload the Quota CSV File and Schedule a Dataflow (Lightning Experience), which covers the steps described here.

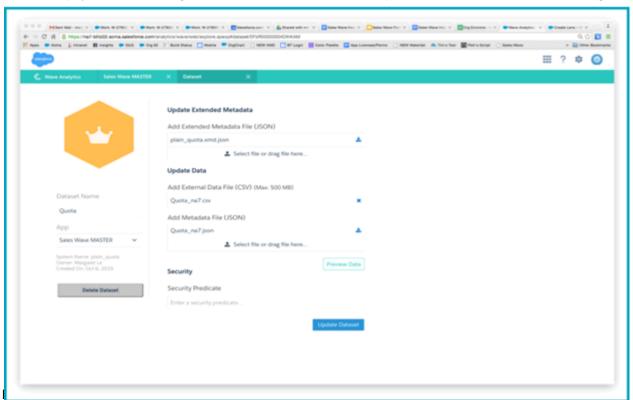
Note: When you work with .CSV files you want to import to Sales Wave Analytics, create and open them using only a UTF-8-compliant text editor. Opening them in Microsoft Excel or other spreadsheet software reformats .CSV files and makes them unusable in Sales Wave Analytics

USER PERMISSIONS

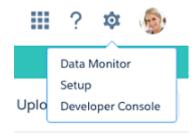
To create and manage Wave apps:

- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"
- Note: If your business doesn't track quotas at the user level, you can skip this step. Complete the step only if you answer "yes" to the configuration wizard question about whether your business tracks quotas at the user level.
- (1) 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

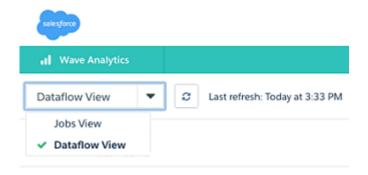
- 1. Create a .CSV file to include the fields just described, that is StartDate (in yyyy-mm-dd format, QuotaAmount, OwnerName, Username.. For an example, see Sales Wave Analytics Example .CSV File.
- **2.** Save the file to a location that you can easily remember.
- 3. In Salesforce, go to the Wave Analytics home page and find the Quota (Target) dataset.
- **4.** Hover over the dataset, then click **Edit**.
- 5. Salesforce displays the dataset editing screen for the Quota dataset. From the Update Data section, select Select file or drag file



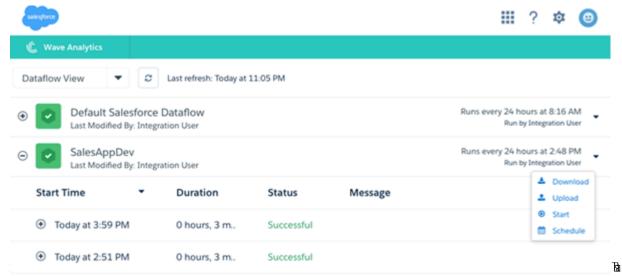
- 6. In the dialog box that opens, navigate to the .CSV file you created in Step 1, and double-click it.
- 7. Click Update Dataset.
- **8.** If your fiscal period is different than calendar period, that is if it starts another date besides January 1, you also need to update the Quota Metadata file.
 - a. Copy the JSON from Sales Wave Analytics App 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 Wave metadata, the numeral "0" stands for January, "1" stands for February, and so on up to "11," which stands for December.
 - **c.** Save the file to your desktop and upload it to Sales Wave Analytics by dragging it to **Select file or drag file here...** in the Add Metadata File (JSON) section of the dataset editing screen.
 - d. Click Update Dataset.
- **9.** After uploading your quota data, re-run the dataflow to update the dashboards.
 - **a.** Click the Gear menu at the upper right of the Sales Wave screen and select Data Monitor.



b. Select Dataflow view from the menu at the top left of the Data Monitor screen.



c. Find your app; you may have to scroll down the page. Open the menu located on the far right of the screen next to the app icon and name, and click



it. The dataflow assures that Sales Wave has your company's latest sales data. You can learn more about dataflows from Schedule the Sales Wave App Daily Dataflow.

IN THIS SECTION:

Sales Wave Analytics Example .CSV File

Here's an example of the .CSV file you create to update the Sales Wave Analytics Quota (Target) dataset.

Sales Wave Analytics App Quota Dataset JSON File

Use this file to update the Sales Wave Analytics Quota dataset with your fiscal year start date if it's not January 1.

Sales Wave Analytics Example .CSV File

Here's an example of the .CSV file you create to update the Sales Wave 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

Create the file using a UTF-8 compliant text editor, and do not open the file with Microsoft Excel or another spreadsheet application. Field names are case-sensitive and must appear in your file exactly as shown here.

See Update the Sales Wave Analytics Quota (Target) Dataset on page 400.



Example:

```
OuotaAmount, StartDate, OwnerName, Username
5000000,2014-01-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-02-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-03-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-04-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-05-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-06-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-07-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-08-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-09-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-10-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-11-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-12-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-01-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-02-01, Nathan Sinha, user18.esm4nmdicvqc@yourDomain.com
5000000,2014-03-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-04-01, Nathan Sinha, user18.esm4nmdicvqc@yourDomain.com
5000000,2014-05-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-06-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-07-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-08-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-09-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-10-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-11-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-12-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
```

Sales Wave Analytics App Quota Dataset JSON File

Use this file to update the Sales Wave 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 Wave Analytics following the instructions in Update the Sales Wave Analytics Quota (Target) Dataset.

```
{
    "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"
                }
        }
   ]
}
```

Data Integration

You can integrate Salesforce data and external data into Wave Analytics to enable users to explore and visualize the data with explorer and designer. *External data* is data that resides outside of Salesforce, such as data from outside applications and spreadsheets.

When you load data into Wave Analytics, you load it into datasets. A dataset is a collection of related data that is stored in a denormalized, yet highly compressed form.

You can use the following methods to create datasets in Wave Analytics.

	Dataflow JSON	Dataset Builder	Upload User Interface	External Data API	Wave Connector
Data source	Salesforce objects; existing datasets	Salesforce objects	External data	External data	Microsoft Excel
Can join external and Salesforce data?	Yes	No	No	No	No
Graphical user interface?	No (JSON)	Yes	Yes	No (programmatic access)	Yes
Can create multiple datasets at once?	Yes	No	No	No	No
Supports incremental extraction?	Yes	No	No	Yes	No
Data refresh method	Automatic	Automatic	Manual	Manual	Manual
Can filter records?	Yes	No	No	No	No
Can generate new columns when creating datasets?	Yes (delta and dimension columns)	No	No	No	No
Can override metadata?	Yes	No	Yes	Yes	No

IN THIS SECTION:

Datasets

A dataset is a collection of related data that is stored in a denormalized, yet highly compressed form. For each platform license, your organization can have a maximum of 250 million rows of data stored for all registered datasets combined.

Create Datasets with a Dataflow

You can use a dataflow to create one or more datasets based on data from Salesforce objects or existing datasets.

Transformations for Wave 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.

Create a Dataset with the Dataset Builder

You can use the dataset builder to create a single dataset based on data from one or more Salesforce objects. The dataset builder generates and appends the associated JSON to the dataflow definition file. The dataset is created the next time the dataflow runs. The data in the dataset refreshes each time the dataflow runs. You can also edit the dataflow definition file to add transformations that manipulate the dataset.

Optimize Your Dataflows with Replication

Replication decouples the extract of Salesforce data from your dataflows, letting you run it on a separate schedule. By scheduling these extracts ahead of time, your dataflow has less to do and runs faster. To lighten the load even more, replication extracts Salesforce data incrementally by default, meaning that only data that's changed gets extracted.

Wave Connector for Excel Data

The Salesforce Wave Connector makes it easy to import data from Microsoft Excel 2013 to Wave Analytics.

Create a Dataset with External Data

You can either upload external data through the user interface or through the External Data API to create a dataset. When you upload an external data file (in .csv, .gz, or .zip format), you can also provide a metadata file. A metadata file contains metadata attributes that describe the structure of the data in the external data file. If you upload a .csv from the user interface, Wave Analytics automatically generates the metadata file, which you can preview and change. If you do not provide a metadata file, Wave Analytics imports all external data file columns as dimensions.

Edit a Dataset

You can edit a dataset to change the dataset name, app, security predicate, or extended metadata (XMD) file associated with the dataset. For datasets created from an external data file, you can also upload a new external data file or metadata file to update the data or metadata.

Delete a Dataset

Delete unnecessary datasets from your My Private App or in shared apps on which you have at least Editor access. Removing datasets reduces clutter and helps you avoid reaching your org's limit for rows across registered datasets.

Prepare Datasets with Recipes

A dataset recipe is a user-interface tool that lets you take data from your existing datasets and replications, prepare it, and output the results to a new dataset. Use a recipe to combine data from multiple datasets or replications, bucket the data, add formula fields, and cleanse the data by transforming field values. You can remove fields and filter rows that you don't need, before you create the target dataset.

Row-Level Security for Datasets

If a Wave Analytics user has access to a dataset, they have access to all records in the dataset, by default. However, you can implement row-level security on a dataset to restrict access to records. Some records might contain sensitive data that shouldn't be accessible by everyone.

Predicate Expression Syntax for Datasets

You must use valid syntax when defining the predicate expression.

Datasets

A dataset is a collection of related data that is stored in a denormalized, yet highly compressed form. For each platform license, your organization can have a maximum of 250 million rows of data stored for all registered datasets combined.

Wave Analytics applies one of the following types to each dataset field:

Date

A date can be represented as a day, month, year, and, optionally, time. You can group, filter, and perform math on dates.

Dimension

A dimension is a qualitative value, like region, product name, and model number. Dimensions are handy for grouping and filtering your data. Unlike measures, you can't perform math on dimensions. To increase query performance, Wave Analytics indexes all dimension fields in datasets.

Measure

A *measure* is a quantitative value, like revenue and exchange rate. You can do math on measures, such as calculating the total revenue and minimum exchange rate.

For each dataset that you create, you can apply row-level security to restrict access to records in the dataset.



Attention: Before you create a dataset, verify that the source data contains at least one value in each column. Columns with all null values won't be created in datasets and can't be referenced in dataflows, lenses, or dashboards. Consider providing a default value for null values, like "n/a" or "empty."

IN THIS SECTION:

Numeric-Value Handling in Datasets

Wave Analytics internally stores numeric values in datasets as long values. For example, it 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."

Null Measure Handling in Datasets

When you create or update a dataset, through your dataflow or a CSV upload for example, blank measure values in your data need to be replaced with concrete values. Wave uses the default values specified in your dataflow or CSV metadata file to replace blank values. When null measure handling is enabled, you can specify these defaults using the special "null" value. When no default value is specified, Wave replaces blanks with null values.

Enable Null Measure Handling

Null measure handling lets you specify "null" default values in your dataflows and CSV uploads. When no default value is specified, Wave replaces blanks with null values in your datasets.

Date Handling in Datasets

When Wave Analytics loads dates into a dataset, it breaks up each date into multiple fields, such as day, week, month, quarter, and year, based on the calendar year. For example, if you extract dates from a CreateDate field, Wave Analytics generates date fields such as CreateDate_Day and CreateDate_Week. If your fiscal year differs from the calendar year, you can enable Wave Analytics to generate fiscal date fields as well.

Dataflow JSON

You can use the dataflow to create one or more datasets based on data from Salesforce objects or existing datasets. 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. With a dataflow, you can manipulate the extracted data and override the metadata before you load it into a dataset. The dataflow runs on a daily schedule to continually refresh the data.

Dataset Builder

Use the dataset builder to create a single dataset based on data from one or more related Salesforce objects. With the dataset builder you simply point and click to identify and select related Salesforce objects.

Wave Connector for Excel Data

The Salesforce Wave Connector makes it easy to import data from Microsoft Excel 2013 to Wave Analytics.

Upload External Data from the User Interface

You can use the upload user interface to create a single dataset based on external data. You can upload an external data file in a .csv, .gz, or .zip format. To refresh the data, you can overwrite the data in the dataset by uploading a new external data file.

External Data API

You can use the External Data API to create a single dataset based on external data in the .csv format. 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.

Numeric-Value Handling in Datasets

Wave Analytics internally stores numeric values in datasets as long values. For example, it 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."

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 might receive unexpected results. For example, if you try to load the value 3.7E-16 with a scale of 16 into a dataset, Wave Analytics tries to store the value as 3700000000000000. However, because this value exceeds the maximum, Wave Analytics fails to load the entire record. In addition, if you perform a query that aggregates measures—like sum or group by—and the resulting value exceeds the maximum, the value overflows and Wave Analytics returns an incorrect result.

Null Measure Handling in Datasets

When you create or update a dataset, through your dataflow or a CSV upload for example, blank measure values in your data need to be replaced with concrete values. Wave uses the default values specified in your dataflow or CSV metadata file to replace blank values. When null measure handling is enabled, you can specify these defaults using the special "null" value. When no default value is specified, Wave replaces blanks with null values.



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 on page 45.

Null measure handling is important because it affects the results of certain calculations. Without null measure handling, Wave replaces blanks in your data with zeros, and you can't specify nulls as default values in your dataflows. Consider this example of data with customer satisfaction scores, where some customers have not responded.

Source Data

Customer	CSAT
Customer 1	3
Customer 2	
Customer 3	4
Customer 4	
Customer 5	5
Average CSAT	4
Min CSAT	3

Dataset with Zeros for Blanks

Customer	CSAT
Customer 1	3
Customer 2	0
Customer 3	4
Customer 4	0
Customer 5	5
Average CSAT	2.4
Min CSAT	0

Dataset with Nulls for Blanks

Customer	CSAT
Customer 1	3
Customer 2	null
Customer 3	4
Customer 4	null
Customer 5	5
Average CSAT	4
Min CSAT	3

Calculated values such as average and minimum are correct in the source data, but when blank values are replaced with zeros, the resulting calculations are incorrect. Here the zeros are included in the calculation. When the calculations are made using null measures, the average and minimum calculations are correct.

Null measure handling allows the use null values in these dataflow transformations.

append	Wave can append datasets with different measure fields. For example, if one dataset has an Amount field that is not present in a second dataset, the new dataset created after the append has an Amount field containing nulls in rows from the second dataset. Without null measure handling, appending datasets is not possible when measure fields are different.
augment	When the key in the left dataset is null or has no match in the right dataset, Wave appends columns from the right dataset and inserts nulls for dimensions and measures. Without null measure handling, Wave inserts zeros for measures.
computeExpression	The defaultValue attribute for a computed field accepts a value of null. For example:
	<pre>"SAQLNode": { "action": "computeExpression", "parameters": { "source": "extractOpp", "mergeWithSource": true, "computedFields": [{</pre>
computeRelative	The defaultValue attribute for a computed field accepts a value of null. When a value can't be calculated, such as when previous () or next () has no value, Wave uses this default. For example:
	<pre>"computeTrending": { "action": "computeRelative", "parameters": { "partitionBy": ["OpportunityId"], "orderBy": [</pre>

dim2mea The measureDefault parameter accepts a value of null. Wave uses this value when it can't convert a dimension to a valid numeric value. When no default is specified, Wave 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 field accepts a value of null. When no default is specified, wave 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 Wave can update non-null values with nulls. Update



Note: The delta transformation is not 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 45.

Enable Null Measure Handling

Null measure handling lets you specify "null" default values in your dataflows and CSV uploads. When no default value is specified, Wave replaces blanks with null values in your datasets.



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, first contact Salesforce Customer Support to evaluate your org and determine what additional steps you might need to take.

USER PERMISSIONS

To enable null measure handling:

"Customize Application"

Additional 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.

When Salesforce makes null measure handling available in your org, you can enable it from Setup. Enter *Wave Analytics* in the Quick Find box, then click **Settings**. Select **Enable null measure handling in Wave 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"

- 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.

"Calculate_Delta": { "action": "delta", "parameters": { "dimension": "OpportunityId", "epoch": "CreatedDate_day_epoch", "inputMeasure": "Amount", "outputMeasure": "DeltaAmount", "source": "Extract_Opportunity_History" } },

computeRelative **and** computeExpression **Transformations**

```
"Compute Previous": {
    "action": "computeRelative",
    "parameters": {
      "partitionBy": [
        "OpportunityId"
      ],
      "orderBy": [
        {
          "name": "CreatedDate",
          "direction": "asc"
        }
      ],
      "computedFields": [
        {
          "name": "PrevAmount",
          "expression": {
            "sourceField": "Amount",
            "offset": "previous()",
            "default": "current()"
        }
      ],
```

delta Transformation	computeRelative and computeExpression Transformations
	<pre>"source": "Extract_Opportunity_History" } }, "Compute_Delta": { "action": "computeExpression", "parameters": { "computedFields": [</pre>

SEE ALSO:

computeRelative Transformation computeExpression Transformation

Date Handling in Datasets

When Wave Analytics loads dates into a dataset, it breaks up each date into multiple fields, such as day, week, month, quarter, and year, based on the calendar year. For example, if you extract dates from a CreateDate field, Wave Analytics generates date fields such as CreateDate_Day and CreateDate_Week. If your fiscal year differs from the calendar year, you can enable Wave Analytics to generate fiscal date fields as well.

Wave Analytics generates the following date fields.

Field Name	Field Type	Description
<date field="" name="">_Second</date>	Text	Number of seconds. If the date contains no seconds, value is '0.'
<date field="" name="">_Minute</date>	Text	Number of minutes. If the date contains no minutes, value is '0.'
<date field="" name="">_Hour</date>	Text	Number of hours. If the date contains no hours, value is '0.'
<date field="" name="">_Day</date>	Text	Day of the month.

Field Name	Field Type	Description
<date field="" name="">_Week</date>	Text	Week number in calendar year.
<date field="" name="">_Month</date>	Text	Month number in calendar year.
<date field="" name="">_Quarter</date>	Text	Quarter number in calendar year.
<date field="" name="">_Year</date>	Text	Calendar year.
<date field="" name="">_Week_Fiscal</date>	Text	Week number in fiscal year.
<date field="" name="">_Month_Fiscal</date>	Text	Month number in fiscal year.
<date field="" name="">_Quarter_Fiscal</date>	Text	Quarter number in fiscal year.
<date field="" name="">_Year_Fiscal</date>	Text	Fiscal year.
<date field="" name="">_sec_epoch</date>	Numeric	Number of seconds that have elapsed since January 1, 1970 (midnight UTC).
<date field="" name="">_day_epoch</date>	Numeric	Number of days that have elapsed since January 1, 1970 (midnight UTC).

You can set metadata attributes to control how dates are loaded into datasets and to enable Wave Analytics to generate fiscal date fields. You set the metadata attributes in the sfdcDigest transformation parameters for Salesforce data or in the metadata file for external data.



Important: Before loading dates from an external data file, ensure that you review the date format requirements here. Also, ensure that the column names in the external data file do not conflict with the generated date field names. For example, if you load a CSV with column Create_Date, Wave Analytics generates the Create_Date_Year field in the dataset. If the CSV also had a field named Create Date Year, Wave Analytics would throw an error because the names conflict.

Fiscal Periods in Wave Analytics

If the calendar and fiscal year differ, you can enable Wave Analytics to generate the fiscal date fields in the dataset in addition to calendar date fields. To enable Wave Analytics to generate fiscal date fields, set the fiscalMonthOffset attribute to a value other than '0'. You set this attribute for each date column for which you want to generate fiscal date fields. If you set the offset to '0' or you do not specify a value, Wave Analytics does not generate any fiscal date fields.

Additionally, to configure the fiscal periods, set the following metadata attributes for each date column:

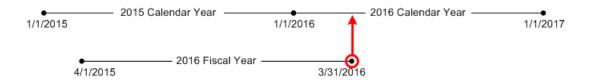
fiscalMonthOffset

In addition to enabling the generation of fiscal date fields, this attribute also determines the first month of the fiscal year. You 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'.

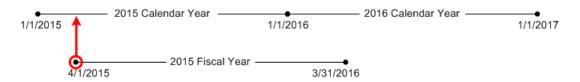
isYearEndFiscalYear

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.

To see how this works, let's look at a couple of examples. If isYearEndFiscalYear = true (or you do not 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 Wave Analytics

For each date loaded into a dataset, Wave Analytics 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 Wave Analytics is January 1 - January 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.

Dataflow JSON

You can use the dataflow to create one or more datasets based on data from Salesforce objects or existing datasets. 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. With a dataflow, you can manipulate the extracted data and override the metadata before you load it into a dataset. The dataflow runs on a daily schedule to continually refresh the data.

Wave Analytics provides a default dataflow that contains some sample transformation logic. This dataflow is just a sample that you must configure before running it.



To configure the dataflow, you add transformations to the dataflow definition file. A *dataflow definition file* is a JSON file that contains transformations that represent the dataflow logic. You can add transformations to determine what data to extract, how to transform datasets, and which datasets to register to make available for queries.

After you configure the dataflow, you upload the new dataflow definition file to Wave Analytics.

By default, the dataflow doesn't run automatically. To start running the dataflow on the schedule, you must manually start the dataflow first. After the first job runs, the dataflow job runs on the daily schedule. The dataflow runs on a daily schedule to capture the latest changes to Salesforce data and changes in the dataflow logic.

You can also stop, reschedule, and monitor dataflow jobs.

SEE ALSO:

Create Datasets with a Dataflow

Dataset Builder

Use the dataset builder to create a single dataset based on data from one or more related Salesforce objects. With the dataset builder you simply point and click to identify and select related Salesforce objects.

After you select the data to include in the dataset, the dataset builder generates and appends the associated JSON to the dataflow definition file. The dataset is created the next time the dataflow runs. The data in the dataset refreshes each time the dataflow runs.

SEE ALSO:

Create a Dataset with the Dataset Builder

Wave Connector for Excel Data

The Salesforce Wave Connector makes it easy to import data from Microsoft Excel 2013 to Wave Analytics.

The Wave Connector is available as an app for Excel 2013 on the desktop and Excel Online in Office 365. The Connector is available as an app from the Microsoft Apps for Office store or your organization's private app catalog. After you install the Connector just point and click to import data from Excel to Salesforce.

IN THIS SECTION:

Install the Wave Connector Excel App

The Wave Connector app gives users a fast, easy way to import data from Excel 2013 into Salesforce Wave Analytics...

Upload External Data from the User Interface

You can use the upload user interface to create a single dataset based on external data. You can upload an external data file in a .csv, .qz, or .zip format. To refresh the data, you can overwrite the data in the dataset by uploading a new external data file.

When Wave Analytics 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.

For external data, Wave Analytics infers metadata about each column of data in the external data file unless you specify different metadata attributes in a metadata file. A *metadata file* is a JSON file that describes the structure of an external data file. For example, you can use a metadata file to explicitly set the field type and default value for a specific column of external data. If no metadata file is provided when you upload external data, Wave Analytics treats every column as a dimension and sets the field type to 'Text.' This impacts the type of queries that can be placed on the dataset because you can't perform mathematical calculations on dataset columns with a Text field type. You can only perform mathematical calculations on dataset columns with a Numeric field type.

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.



Note: Wave temporarily stores the uploaded CSV and metadata files for processing only. After a dataset is created, Wave purges the files.

SEE ALSO:

Create a Dataset with External Data

External Data API

You can use the External Data API to create a single dataset based on external data in the .csv format. 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.

For more information about the External Data API, see the Wave Analytics External Data API Developer Guide.

Create Datasets with a Dataflow

You can use a dataflow to create one or more datasets based on data from Salesforce objects or existing datasets.

IN THIS SECTION:

1. Design the Dataflow

Before you start creating the dataflow definition file in the .json format, 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.

2. Configure the Dataflow

Configure the dataflow based on your dataflow design. You can configure the dataflow to extract data, transform datasets based on your business requirements, and register datasets that you want to make available for queries. To configure the dataflow, add transformations to the dataflow definition file.

3. Start and Stop a Dataflow

You can manually start a dataflow job to load the data into datasets immediately. You can also stop the job while it's running. You can run a maximum of 24 dataflow jobs during a rolling 24-hour period.

4. Monitor a Dataflow Job

Use the monitor in the data manager to monitor dataflow jobs to ensure that they complete successfully or to troubleshoot them if they fail.

5. Reschedule and Unschedule a Dataflow

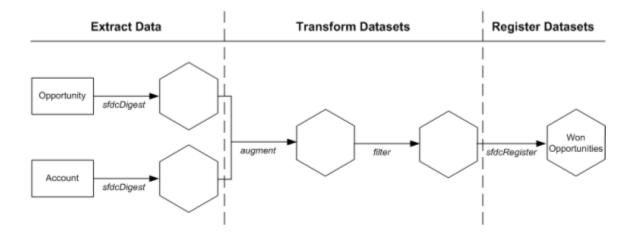
After you run a dataflow job for the first time, it runs on a daily schedule, by default. You can change the time of the daily schedule or schedule it to run on an hourly interval. You might change the schedule to ensure that the data is available by a particular time or to run the job during non-business hours. You can also unschedule a dataflow.

Design the Dataflow

Before you start creating the dataflow definition file in the .json format, think about the dataflow design. Consider what data to make available for gueries, 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 250 million per platform license.
- 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.

Configure the Dataflow

Configure the dataflow based on your dataflow design. You can configure the dataflow to extract data, transform datasets based on your business requirements, and register datasets that you want to make available for queries. To configure the dataflow, add transformations 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 Wave Analytics.

- 1. In Wave Analytics, click the gear icon () and then click **Data Manager**. The data manager opens on the Monitor tab, with the Jobs view selected by default.
- 2. Select Dataflow View.
- **3.** To download the dataflow definition file, click **Download** in the actions list (1).

EDITIONS

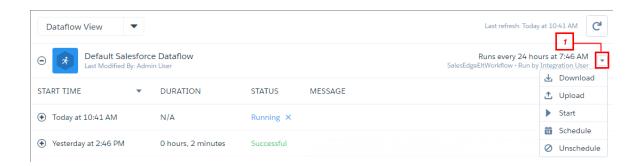
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To edit the dataflow definition file:

 "Edit Wave Analytics Dataflows"



4. Make a backup copy of the dataflow definition file before you modify it.

Wave Analytics 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.

For example, based on the design in the previous step, you can add the following transformation nodes:

```
{
   "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": [
```

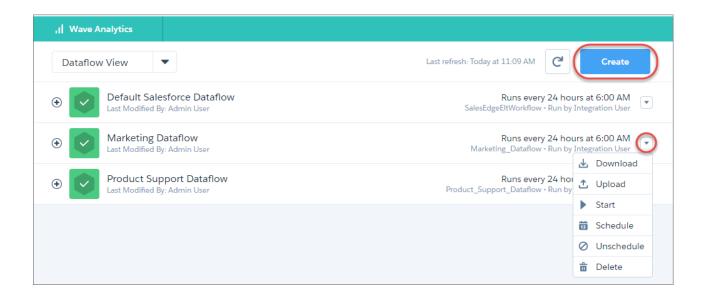
```
"Name"
         1
   },
   "Transform Filter Opportunities": {
      "action": "filter",
      "parameters": {
         "filter": "StageName:EQ:Closed Won",
         "source": "Transform Augment OpportunitiesWithAccountDetails"
   },
   "Register_Dataset_WonOpportunities": {
      "action": "sfdcRegister",
      "parameters": {
         "alias": "WonOpportunities",
         "name": "WonOpportunities",
         "source": "Transform Filter Opportunities"
   }
}
```

- 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. Wave Analytics determines the order in which to process the transformations by traversing the dataflow to determine the dependencies among them.
- (1) 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" are not 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 tool 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 list (1) to upload the updated dataflow definition file.
 - Note: Uploading the dataflow definition file does not affect any running dataflow jobs and does not automatically start the dataflow job.

You can now start the dataflow on demand or wait for it to run on the schedule. Users cannot query the registered datasets until the dataflow runs.

If you have enabled replication, you can create multiple dataflows, in addition to the default dataflow, and configure them in the same way. Look out for the **Create** button in the Monitor tab of the data manager after you enable replication. See Optimize Your Dataflows with Replication.



Note: Your org can have up to 10 dataflows. This includes the default dataflow, and Wave app dataflows such as Sales Wave and Service Wave. Keep in mind that each dataflow run counts towards your limit of 24 dataflow runs per day.

Start and Stop a Dataflow

You can manually start a dataflow job to load the data into datasets immediately. You can also stop the job while it's running. You can run a maximum of 24 dataflow jobs during a rolling 24-hour period.

- Note: By default, the dataflow doesn't run automatically. To start running the dataflow on the schedule, you must manually start the dataflow first. After the first job runs, the dataflow job runs on the daily schedule.
- 1. In Wave Analytics, click the gear icon () and then click **Data Manager**. The data manager opens on the Monitor tab, with the Jobs view selected by default.
- 2. Select Dataflow View.

EDITIONS

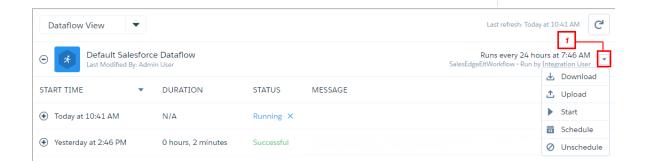
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To start a dataflow job:

 "Edit Wave Analytics Dataflows"



3. Click **Start** in the actions list (1) to start the dataflow job.

The dataflow job is added to the job queue. The **Start** button is grayed out while the dataflow job runs.

4. After the job completes, Wave Analytics 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.
- 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.

You can monitor the dataflow job in the data monitor to determine when dataflow completes. After the dataflow completes successfully, refresh the Home page to view the registered datasets.

Monitor a Dataflow Job

Use the monitor in the data manager to monitor dataflow jobs to ensure that they complete successfully or to troubleshoot them if they fail.

The Dataflow view of the monitor shows the status, start time, and duration of the last 10 dataflow jobs and retains the last 7 days of history. To help you troubleshoot a failed job, you can view error messages about the job, view the run-time details about every transformation that is processed, and download error logs where available.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

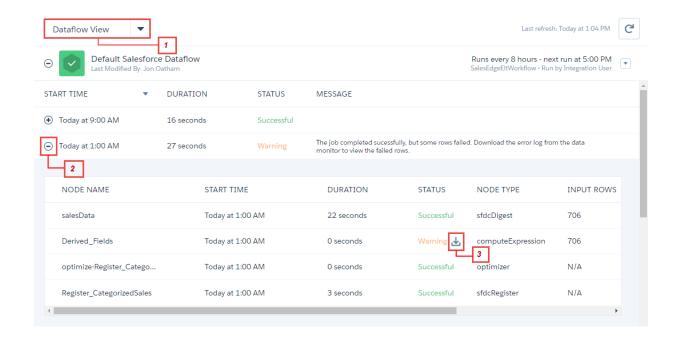
USER PERMISSIONS

To access the data monitor:

 "Edit Wave Analytics Dataflows", "Upload External Data to Wave Analytics", or "Manage Wave Analytics"

To download an error log:

 "Edit Wave Analytics Dataflows" and "View All Data"



- 1. In Wave Analytics, click the gear icon () and then click **Data Manager**. The data manager opens on the Monitor tab, with the Jobs view selected by default.
- 2. Select Dataflow View (1).
- Click to see the latest status of a job.

 Each job can have one of the following 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) and review 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 users with the "View All Data" permission can download error logs.
- 6. If there's a problem with the dataflow logic, edit the dataflow definition file, upload it, and then run the dataflow again.

Reschedule and Unschedule a Dataflow

After you run a dataflow job for the first time, it runs on a daily schedule, by default. You can change the time of the daily schedule or schedule it to run on an hourly interval. You might change the schedule to ensure that the data is available by a particular time or to run the job during non-business hours. You can also unschedule a dataflow.

- In Wave Analytics, click the gear icon () and then click **Data Manager**.
 The data manager opens on the Monitor tab, with the Jobs view selected by default.
- 2. Select Dataflow View.

EDITIONS

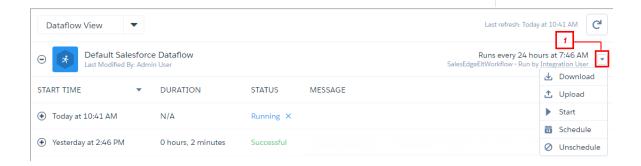
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

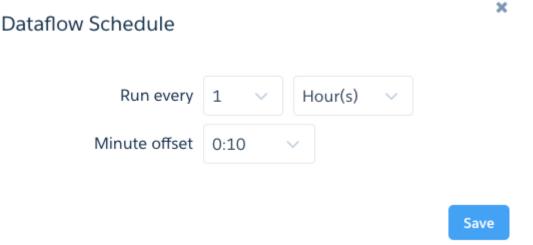
USER PERMISSIONS

To schedule or unschedule a dataflow job:

 "Edit Wave Analytics Dataflows"



- **3.** Click **Schedule** in the actions list (1) to reschedule the dataflow job. The default daily schedule and start time appears.
- **4.** To continue running it once per day, select the time that you'd like to start the dataflow job.
- **5.** To run it on an hourly basis, select **Hour(s)** and then specify the interval details. Minute offset is the number of minutes past the hour to run the dataflow.



Hourly scheduled dataflows run relative to a fixed base time (00:00 UTC). For example, if you schedule a dataflow to run every 6 hours with a 15-minute offset, the dataflow is scheduled to run at 00:15 UTC, 06:15 UTC, 12:15 UTC, and 18:15 UTC, each day. The first dataflow run occurs at the next scheduled time. So, if you change the schedule at 04:00 UTC, the dataflow runs at the next scheduled time (06:15 UTC), and then every 6 hours thereafter. To keep things simple, the scheduler specifies the time of the first dataflow run (in your time zone), after you save the schedule.

6. Click Save.

The Dataflow Schedule pop-up box specifies the next time, in your local time zone, that the dataflow runs.

- 7. Click Done.
- **8.** To stop a dataflow from running on a schedule, select **Unschedule** in the action list (1).

Transformations for Wave 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 gueries.

IN THIS SECTION:

append Transformation

The append transformation combines records from multiple datasets into a single dataset.

augment Transformation

The augment transformation augments an input dataset by combining columns 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, Wave 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.

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.

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.

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.

SEE ALSO:

Create Datasets with a Dataflow

append Transformation

The append transformation combines records from multiple datasets into a single dataset.

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 definition file to merge the existing datasets into a single dataset.

```
"Extract SalesTargets 2014Jan": {
  "action": "edgemart",
  "parameters": { "alias": "SalesTargets 2014Jan" }
},
"Extract SalesTargets 2014Feb": {
   "action": "edgemart",
   "parameters": { "alias": "SalesTargets 2014Feb" }
},
"Extract SalesTargets 2014Mar": {
   "action": "edgemart",
  "parameters": { "alias": "SalesTargets 2014Mar" }
},
"Append SalesTargets 2014Quarter1": {
   "action": "append",
   "parameters": {
      "sources": [
         "Extract SalesTargets 2014Jan",
         "Extract SalesTargets 2014Feb",
         "Extract SalesTargets 2014Mar"
      ]
},
"Register AllSalesTargets": {
   "action": "sfdcRegister",
   "parameters": {
      "alias": "AllSalesTargets",
      "name": "AllSalesTargets",
      "source": "Append SalesTargets 2014Quarter1"
   }
```

After you create the single dataset, you can use date filters to analyze the sales targets by month, quarter, or year.

IN THIS SECTION:

append Parameters

When you define an append transformation, you set the action attribute to append and specify the parameters.

append Parameters

When you define an append transformation, you set the action attribute to append and specify the parameters.

The following table describes the input parameters.

Parameter	Required?	Value
sources	Yes	Nodes in the dataflow definition file that identify the datasets that you want to
		merge.

SEE ALSO:

append Transformation

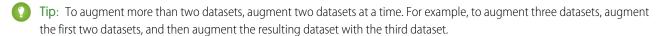
augment Transformation

The augment transformation augments an input dataset by combining columns 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.

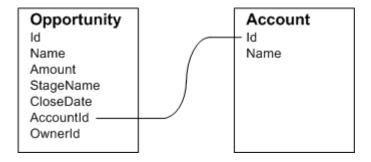
When you create the transformation, you identify each input dataset as the left or right dataset and specify the relationship between them. Wave Analytics combines all the columns of the left dataset with only the specified columns from the right dataset. (Keep in mind that each dataset can't have more than 5,000 columns.) Wave Analytics 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. For example:

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.



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.

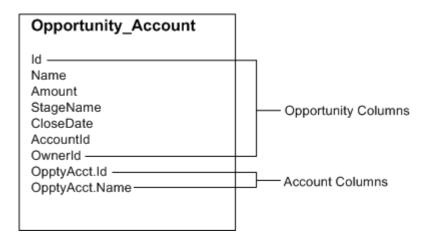


You create the following dataflow definition file.

```
{
    "Extract Opportunity": {
        "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" }
          }
   },
    "Augment OpportunitiesWithAccountDetails": {
        "action": "augment",
        "parameters": {
            "operation": "LookupSingleValue",
            "left": "Extract_Opportunity",
            "left key": [
                "AccountId"
            ],
            "relationship": "OpptyAcct",
            "right": "Extract_AccountDetails",
            "right key": [
                "Id"
            "right select": [ "Name" ]
        }
```

```
},
"Register_OpportunityDetails": {
    "action": "sfdcRegister",
    "parameters": {
        "alias": "Opportunity_Account",
        "name": "Opportunity_Account",
        "source": "Augment_OpportunitiesWithAccountDetails" }
}
```

After you run the dataflow, Wave Analytics creates and registers the Opportunity_Account dataset. It also adds the relationship as a prefix to all columns from the right dataset.



IN THIS SECTION:

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, Wave Analytics doesn't perform a lookup to match the record. Instead, Wave Analytics 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.



Price		
Prodid	Pricebook	UnitPrice
Prod1	Standard	1000
Prod2	Standard	450
null	Custom	800
Prod3	Standard	700

Wave Analytics doesn't match the last record because the product ID is null. Instead, Wave Analytics 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

Wave Analytics 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.



Price		
Prodid	Pricebook	UnitPrice
Prod1	Standard	1000
Prod2	Standard	450
	Custom	800
null	Standard	700

Wave Analytics 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, Wave Analytics 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.



Price		
Prodid	Pricebook	UnitPrice
Prod1	Standard	1000
Prod2	Standard	450
Prod3	Standard	700

Wave Analytics 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 left key doesn't have a match in the right dataset, Wave Analytics 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.



Price		
Prodid	Pricebook	UnitPrice
Prod4	Standard	1000
Prod5	Standard	450

Because no keys match, Wave Analytics 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 }
```

Handling Multiple Matches

If the left dataset has a one-to-many relationship with the right dataset, Wave Analytics might find multiple matches for a left record. What Wave Analytics 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. Wave Analytics randomly selects one row from the list of matched rows.



Note: Each time you run the dataflow, Wave Analytics 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		
ld	Name	
Prod1	Table	
Prod2	Chair	
Prod3	Bench	

Price			
Prodid	Pricebook	UnitPrice	
Prod1	Standard	1000	
Prod2	Standard	450	
Prod3	Custom	800	
Prod3	Standard	700	

Although there are multiple rows for Prod3 in the Price dataset, Wave Analytics randomly chooses one matching row and returns values based on that row. Here's the resulting dataset after the augment if Wave Analytics 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

Wave Analytics 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 }
```

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.
left_key Yes	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.

Parameter	Required?	Value
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, Wave 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: 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
- 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.
- Wave 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:

```
"saqlExpression": "Derived_A + \"world\""}
]
}
```

- You can choose whether the resulting dataset includes only the derived fields, or includes the input and derived fields.
- 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 ld 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",
            "saglExpression": "\"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" }
}</pre>
```

IN THIS SECTION:

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, Wave can replace null values in a field with a default value.

Parameter	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	Wave Analytics 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 Wave 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\")"
format	Yes (for Date fields only)	Format of the derived date field. For information about formats, see the External Data Format Reference.
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 between 1 and 18, inclusive. For text values: Value defaults to 255 characters, and must be between 1 and 32,000 characters, inclusive.

Parameter	Required?	Value
		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 between 0 and 17 characters, inclusive. 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.

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: 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 records by opportunity ID, and then chronologically sort records within each partition to correctly identify the previous and next values.



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.

OpportunityHistory Object					
OpportunityId	CreatedDate	StageName	Amount	CloseDate	
00630000003JrH6AAK	2005-11-23T20:24:09.000Z	Closed Won	36938	11/15/2005	
00630000003SYADAA4	2005-11-08T18:47:18.000Z	Present/Demo/Quote	4200	12/3/2005	
00630000003Yx9BAAS	2005-11-23T01:06:57.000Z	Present/Demo/Quote	71500	11/30/2005	
00630000003JsMcAAK	2005-10-28T18:12:06.000Z	Negotiation/Review	18750	11/15/2005	
00630000003Yx9BAAS	2005-11-25T18:03:46.000Z	Closed Won	5832	11/23/2005	
00630000003Yx9BAAS	2005-11-23T01:07:46.000Z	Present/Demo/Quote	7128	11/30/2005	
00630000003JrH6AAK	2005-11-03T21:53:36.000Z	Negotiation/Review	20148	11/15/2005	
00630000003SZAZAA4	2005-11-08T19:44:27.000Z	Present/Demo/Quote	141120	12/16/2005	
00630000003JrH6AAK	2005-10-17T22:28:33.000Z	Present/Demo/Quote	42900	11/15/2005	
00630000003JsMcAAK	2005-10-17T23:30:20.000Z	Negotiation/Review	18750	10/31/2005	
00630000003OG95AAG	2005-10-28T17:45:30.000Z	Present/Demo/Quote	52000	11/30/2005	

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.

Opportunity Created Date

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.

ValidToDate

Determines the date to which the state of the opportunity is valid. You can use this field to determine the time period for which the opportunity details are valid in each record.

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.

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.

```
},
"computeTrending": {
    "action": "computeRelative",
    "parameters": {
        "partitionBy": ["OpportunityId"],
        "orderBy": [
                "name": "CreatedDate",
                "direction": "asc"
        ],
        "computedFields": [
            {
                "name": "OpportunityCreatedDate",
                "expression": {
                    "sourceField": "CreatedDate",
                    "offset": "first()",
                    "default": "current()"
            },
                "name": "ValidToDate",
                "expression": {
                    "sourceField": "CreatedDate",
                    "offset": "next()",
                    "default": "3000-01-01T00:00:00.000Z"
            },
                "name": "AmountPrev",
                "expression": {
                    "sourceField": "Amount",
                    "offset": "previous()",
                    "default": "0"
            },
                "name": "CloseDatePrev",
                "expression": {
                    "sourceField": "CloseDate",
                    "offset": "previous()",
                    "default": "01/01/1970"
            }
        "source": "extractOppHistory"
},
"Register OppportunityHistory Dataset": {
    "action": "sfdcRegister",
    "parameters": {
        "alias": "SalesPipelineTrending",
        "name": "Sales Pipeline Trending",
```

```
"source": "computeTrending"
}
}
```

The dataflow runs and creates the following dataset.

Sales Pipeline Trending Dataset								
OpportunityId	CreatedDate	StageName	Amount	CloseDate	OpportunityCreatedDate	ValidToDate	AmountPrev	CloseDatePrev
00630000003JrH6AAK	2005-10-17T22:28:33.000Z	Present/Demo/Quote	429000	11/15/2005	2005-10-17T22:28:33.000Z	2005-11-03T21:53:36.000Z	0	1/1/1970
00630000003JrH6AAK	2005-11-03T21:53:36.000Z	Negotiation/Review	201480	11/15/2005	2005-10-17T22:28:33.000Z	2005-11-23T20:24:09.000Z	429000	11/15/2005
00630000003JrH6AAK	2005-11-23T20:24:09.000Z	Closed Won	369380	11/15/2005	2005-10-17T22:28:33.000Z	3000-01-01T00:00:00.000Z	201480	11/15/2005
00630000003JsMcAAK	2005-10-17T23:30:20.000Z	Negotiation/Review	187500	10/31/2005	2005-10-17T23:30:20.000Z	2005-10-28T18:12:06.000Z	0	1/1/1970
00630000003JsMcAAK	2005-10-28T18:12:06.000Z	Negotiation/Review	187500	11/15/2005	2005-10-17T23:30:20.000Z	3000-01-01T00:00:00.000Z	187500	10/31/2005
00630000003OG95AAG	2005-10-28T17:45:30.000Z	Present/Demo/Quote	520000	11/30/2005	2005-10-28T17:45:30.000Z	3000-01-01T00:00:00.000Z	0	1/1/1970
00630000003SYADAA4	2005-11-08T18:47:18.000Z	Present/Demo/Quote	42000	12/3/2005	2005-11-08T18:47:18.000Z	3000-01-01T00:00:00.000Z	0	1/1/1970
00630000003SZAZAA4	2005-11-08T19:44:27.000Z	Present/Demo/Quote	1411200	12/16/2005	2005-11-08T19:44:27.000Z	3000-01-01T00:00:00.000Z	0	1/1/1970
00630000003Yx9BAAS	2005-11-23T01:06:57.000Z	Present/Demo/Quote	715000	11/30/2005	2005-11-23T01:06:57.000Z	2005-11-23T01:07:46.000Z	0	1/1/1970
00630000003Yx9BAAS	2005-11-23T01:07:46.000Z	Present/Demo/Quote	71280	11/30/2005	2005-11-23T01:06:57.000Z	2005-11-25T18:03:46.000Z	715000	11/30/2005
00630000003Yx9BAAS	2005-11-25T18:03:46.000Z	Closed Won	58320	11/23/2005	2005-11-23T01:06:57.000Z	3000-01-01T00:00:00.000Z	71280	11/30/2005

Notice that Wave partitions the records by opportunity ID and then sorts the records in ascending order based on the CreatedDate field within each partition. Wave can now use the previous and next rows within each partition to determine changes in field values in the dataset.

IN THIS SECTION:

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.

Parameter	Required?	Value
		Example:
		<pre>"orderBy": [</pre>
computedFields	Yes	A list of definitions for derived fields.
		Example:
		<pre>"computedFields": [</pre>

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.
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.

Parameter	Required?	Value	
		Example:	
	<pre>"expression": { "sourceField": "CloseDate", "offset": "previous()", "default": "01/01/1970"</pre>		
		See expression.	

expression

The following table describes the attributes in the ${\tt expression}$ section.

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()"
		first() Gets the value from the first record in the partition, like the first CreateDate for an opportunity.
	Example:	
		<pre>"offset": "first()"</pre>
		next()
		Gets the value from the next record.
		Example:
		"offset": "next()"
		<pre>previous() Gets the value from the previous record. Example: "offset": "previous()"</pre>
	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.	

Parameter	Required?	Value
		? 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()"	

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.

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 Wave Analytics processes dates, it creates the following epoch date columns for each date processed:

Epoch Time Column	Description
<date_column_name>_sec_epoch</date_column_name>	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</date_column_name>	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

OpportunityId	CloseDate	StageName	Amount
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 (Opportunityld) 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 (Opportunityld), 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.

IN THIS SECTION:

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.

Parameter	Required?	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

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.

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, Wave Analytics 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"
}
}
```

IN THIS SECTION:

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.
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"
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.

0

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 1: 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"
}
```

IN THIS SECTION:

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

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.



Example: Let's look at an example. You would like 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 definition.

```
"Extract Opportunities": {
   "action": "sfdcDigest",
   "parameters": {
      "object": "Opportunity",
      "fields": [
         { "name": "Id" },
         { "name": "Name" },
         { "name": "Amount" },
         { "name": "StageName" },
         { "name": "CloseDate" },
         { "name": "AccountId" },
         { "name": "OwnerId" }
      1
},
"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"
    }
}
```

IN THIS SECTION:

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.

filter Parameters

When you define a filter transformation, you set the action attribute to filter and specify the parameters.

The following table describes the input parameters:

Parameter	Required?	Value
filter	Yes	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.

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.



Note: String comparisons in a filter expression are case-sensitive.

You can use the following types of filter expressions:

Filter Expression Syntax	Description	
dim:EQ:value	True if the dimension and value are equal.	
	Example: "filter": "StageName:EQ:Closed Won	

Filter Expression Syntax	Description	
dim:R:val0:val1	True if the left dimension falls within the specified range between val0 and val1.	
	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').	
	<pre>Example: "filter": "LastName:R:Li"</pre>	
dim:R::val	True if the dimension is less than or equal to the value based on binary sort order.	
	<pre>Example: "filter": "LastName:R::Levy"</pre>	
dim:N:val	True if the dimension and value are not equal.	
	<pre>Example: "filter": "RoleName:N:Manager"</pre>	
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 two values. For example, to compare against three values, use the following syntax: dim1:EQ:val1 val2 val3.	
	<pre>Example: "filter": "Lead Status:EQ:Open Contacted"</pre>	
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 two dimensions. For example, to compare three dimensions, use the following syntax: dim1:EQ:val1,dim2:EQ:val2,dim3:EQ:val3.	
	<pre>Example: "filter": "Lead Status:EQ:Qualified,Rating:EQ:Hot"</pre>	

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
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_field":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.

ld

Id identifies each node in the Salesforce role hierarchy.

ParentRoleId

ParentRoleld 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.

IN THIS SECTION:

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.
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, set the IsSystemField metadata attribute to false for each field in the flatten transformation. For more information about metadata attributes, see Overriding Metadata Generated by a Transformation.

SEE ALSO:

flatten Transformation

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.

When you upload the dataflow definition file, Wave Analytics validates access to Salesforce objects and fields based on the user profile of the user who uploads the file. If the user profile does not have read access to a field or object, the upload fails.

At run time, Wave Analytics runs the dataflow as the Integration User. Wave Analytics validates access to Salesforce objects and fields based on the user 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 Wave Analytics 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 Wave Analytics, see the Wave Analytics 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 Wave Analytics 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 20,000 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.

IN THIS SECTION:

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, Wave Analytics can add a default value to records that have missing values for a field.

Unsupported Salesforce Objects and Fields in Wave

The sfdcDigest transformation can't extract data from all Salesforce objects and fields. Consider these limitations before configuring the extraction of Salesforce objects.

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.

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—Wave Analytics 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.

IN THIS SECTION:

Structured Filter in sfdcDigest Transformation

You define a structured filter using JSON syntax.

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.

SEE ALSO:

sfdcDigest Transformation

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. Wave Analytics 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:
	<pre>"filterConditions": [{ "field": "OwnerId", "operator": "=", "value": "a07B00000012HYu" }]</pre>
!=	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):
	<pre>"filterConditions": [{ "field": "Nicknamec", "operator": "!=", "value": "\"Sammy\"" }]</pre>

Operator	Comment
>	Filter condition is true if the value in the field is greater than the specified value.
	Example:
	<pre>"filterConditions": [{ "field": "Amount", "operator": ">", "value": "100000" }]</pre>
<	Filter condition is true if the value in the field is less than the specified value.
	Example (using a date literal):
	<pre>"filterConditions": [{ "field": "CloseDate", "operator": "<", "value": "THIS_MONTH", "isQuoted": false }]</pre>
>=	Filter condition is true if the value in the field is greater than or equal to the specified value. Example:
	<pre>"filterConditions": [</pre>
<=	Filter condition is true if the value in the field is less than or equal to the specified value.
	<pre>Example (using a SOQL function): "filterConditions": [</pre>

Operator	Comment
]
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 one character.
	 The LIKE operator is supported for string fields only.
	• 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:
	<pre>"filterConditions": [{ "field": "FirstName", "operator": "LIKE", "value": "Chris%" }]</pre>
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:
	<pre>"filterConditions": [</pre>
NOT IN	Filter condition is true if the value in the field does not equal any of the values in the specified list.
	Example:
	<pre>"filterConditions": [{ "field": "BillingState", "operator": "NOT IN", "value": ["California", "New York"]</pre>

Operator	Comment	
	1	
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:	
	<pre>"filterConditions": [{ "field": "BillingState", "operator": "INCLUDES", "value": ["California"] }]</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.

```
"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—Wave Analytics 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": "00540000000HfUz"
                                 },
                                     "field": "OwnerId",
                                     "operator": "=",
                                     "value": "00540000000HfV4"
                            ]
                        }
                    ]
                }
            ]
   },
    "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.

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.

```
"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 = '00540000000HfUz' AND (Amount > 100000
OR Probability > 75)"
```

```
}
},

"Register_Opportunities_Dataset": {
    "action": "sfdcRegister",
    "parameters": {
        "alias": "FilteredOpportunities",
        "name": "FilteredOpportunities",
        "source": "Extract_Filtered_Opportunities"
}
}
```

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, Wave Analytics 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 to override the field metadata:

- defaultValue
- type
- fiscalMonthOffset
- isYearEndFiscalYear
- firstDayOfWeek
- isMultiValue
- multiValueSeparator
- precision
- scale

For a description of each of these field parameters, see Field Parameters. For information about using metadata attributes to configure dates, see Date Handling in Datasets.



Example: Let's look at an example. You would like to override metadata extracted from the Opportunity object.

You add the bold text in the following dataflow definition to override field metadata from the Opportunity object.

```
"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"
   }
```

SEE ALSO:

sfdcDigest Transformation

Unsupported Salesforce Objects and Fields in Wave

The sfdcDigest transformation can't extract data from all Salesforce objects and fields. Consider these limitations before configuring the extraction of Salesforce objects.

For information about all Salesforce objects and fields, see the Object Reference for Salesforce and Force.com.

Unsupported Objects

The sfdcDigest transformation can't extract data from these Salesforce objects.

- ActivityMetric
- ApexEmailNotification
- AuthProvider
- BrandTemplate
- ChatterConversation
- ChatterConversationMember
- ChatterMessage
- ConnectedApplication
- ContentFolderLink

- ContentWorkspace
- ContentWorkspaceDoc
- ContentWorkspaceMember
- ContentWorkspacePermission
- CopyExport
- CorsWhitelistEntry
- DataAssessmentFieldMetric
- DataAssessmentMetric
- DataAssessmentValueMetric
- DataHubSetupData
- DataHubSetupDefinition
- DirectMessage
- DirectMessageFeed
- DirectMessageMember
- EmailCapture
- EmailDomainKey
- EmailServicesAddress
- EmailServicesFunction
- EmailTemplate
- EnvironmentHub
- EnvironmentHubInvitation
- EnvironmentHubMemberRel
- EventType
- EventTypeParameter
- FeedPollChoice
- FeedPollVote
- KnowledgeArticleVersion
- LeadChangeEvent
- LoginGeo
- LoginHistory
- ModelFactor
- NetworkActivityAudit
- NetworkModeration
- OrganizationProperty
- OrgWideEmailAddress
- PackageLicense
- PartnerNetworkSyncLog
- ReputationLevel
- ReputationLevelLocalization
- ReputationPointsRule

- SalesforcelqUser
- 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.

Unsupported Fields

The sfdcDigest transformation can't extract data from these fields.

Object	Unsupported Fields	
Account	CleanStatus	
ActionPlanItem	ltemld	
AuthSession	• LoginGeold	
	 LoginHistoryId 	
CaseArticle	KnowledgeArticleId	
Contact	 CanAllowPortalSelfReg 	
	• CleanStatus	
ContentDocument	Parentld	
CustomPersonp	Title	
DocumentAttachmentMap	ParentId	
EmailMessage	ActivityId	
EmailRoutingAddress	EmailServicesAddressId	
EnvironmentHubMember	EnvironmentHubld	
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 	

Object	Unsupported Fields
PermissionSetLicense	MaximumPermissionsEditEventMaximumPermissionsEditTask
Profile	PermissionsEditEventPermissionsEditTask
ThirdPartyAccountLink	SsoProviderId
User	LastPasswordChangeDateUserPreferencesEnableVoicePilot
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 extraction, which extracts only changes to the Salesforce object since the last dataflow run. Valid values: true or false.
		Note: Incremental extraction is available if you have enabled replication.
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 extraction is available if you have enabled replication.
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.

Parameter	Required?	Value
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, Wave Analytics can add a default value to records that have missing values for a field. If you don't override the values, Wave Analytics 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.
type	No	Wave Analytics 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 Wave Analytics. Example:
		"type": "Text"
fiscal Month Offset	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 Wave Analytics generates fiscal date fields. To generate fiscal

Attribute	Required?	Value
		date fields, set fiscalMonthOffset to a value other than 0.
		For more information, see Date Handling in Datasets.
is Year End Fiscal Year	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.
		For more information, see Date Handling in Datasets.
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
		For more information, see Date Handling in Datasets.
isMultiValue	No	For text fields only. Indicates whether the specified field has multiple values.
		Example:
		"isMultiValue": false

Attribute	Required?	Value
multiValueSeparator	No	For text fields only. 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 between 1 and 18, inclusive. For text values: Value defaults to 255 characters, and must be between 1 and 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 between 0 and 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	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:
		<pre>{ "field": "<field name="">",</field></pre>

Parameter	Required?	Value
		<pre>"operator": "<operator>", "value": "<value>", "isQuoted": true false}</value></operator></pre>
operator	No	 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.
		<pre>"filterConditions": [</pre>
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": [

Parameter	Required?	Value
		<pre>"field": "StageName", "operator": "IN", "value": "('Closed Won', 'Closed Lost')", "isQuoted": true }]</pre>
		Example with non-quoted values:
		<pre>"filterConditions": [{ "field": "StageName", "operator": "IN", "value": ["Closed Won", "Closed Lost"], "isQuoted": false }]</pre>
		If you don't include isQuoted for a filter on a string value, Wave Analytics 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:

sfdcDigest Transformation

Filtering Records Extracted from a Salesforce Object

sfdcRegister Transformation

The sfdcRegister transformation registers a dataset to make it available for queries. Users cannot view or run queries against unregistered datasets.

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 Wave Analytics automatically registers these datasets for you.

Carefully choose which datasets to register because:

- The total number of rows in all registered datasets cannot exceed 250 million per platform license.
- 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\""
   }
}
```

IN THIS SECTION:

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.

Parameter	Required?	Value
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 is the input source for this transformation.
rowLevelSecurityFilter	No	The predicate used to apply row-level security on the dataset when the dataset is first created.
		Example: "rowLevelSecurityFilter": "'Ownerld' == "\$User.ld""
		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, Wave Analytics ignores it's 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.

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.

IN THIS SECTION:

sliceDataset Parameters

When you define a sliceDataset transformation, you set the action attribute to sliceDataset and specify the parameters.

sliceDataset Parameters

When you define a sliceDataset transformation, you set the action attribute to sliceDataset and specify the parameters. 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 <i>remove</i> the fields. Set it to select to <i>keep</i> the fields.
fields	Yes	An array of names of the fields that you want to drop or select from the specified source node.

update Transformation

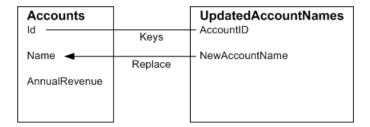
The update transformation updates the specified field values in an existing dataset based on data from another dataset, which we'll call the lookup dataset. The transformation looks up the new values from corresponding fields in the lookup dataset. The transformation stores the results in a new dataset.

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—ld, 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"
     }
  }
}
```

IN THIS SECTION:

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 me records in the other dataset. If you use composite key, the left and right keys have the same number of columns in same order. For an example, see updated a Transformation on page 116.	
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"; "RightColumnN": "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	

Parameter	Required?	Value		
		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:

update Transformation

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:

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available for an additional cost in: **Enterprise**, **Performance**, and **Unlimited** Editions

```
"Extract Opportunities": {
   "action": "sfdcDigest",
   "parameters": {
      "object": "Opportunity",
      "fields": [
         { "name": "Name" },
         { "name": "Amount" }
   },
   "schema": {
      "objects": [
            "label": "Sales Opportunities",
            "fields": [
               {
                  "name": "Amount",
                  "label": "Opportunity Amount"
               }
            1
         }
      1
  }
```

Object Attributes

You can override the following object attributes.

Object Attribute	Туре	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 Wave Analytics 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:
		"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 Wave Analytics 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

Field Attribute	Type	Description		
format	String	The display format of the numeric value.		
		Examples:		
		"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
#	Zero or 1 digit
	This symbol is the default decimal separator. Use the decimal Separator field to set the decimal separator to a different symbol.
-	Minus sign
1	Grouping separator
\$	Currency sign

- Note: The format for numeric values isn't used in data ingestion. It is used only to specify how numeric values are formatted when displayed in the UI. 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
ld 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"
   }
```

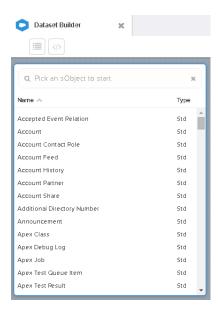
Create a Dataset with the Dataset Builder

You can use the dataset builder to create a single dataset based on data from one or more Salesforce objects. The dataset builder generates and appends the associated JSON to the dataflow definition file. The dataset is created the next time the dataflow runs. The data in the dataset refreshes each time the dataflow runs. You can also edit the dataflow definition file to add transformations that manipulate the dataset.

1. On the home page or on an app page, click **Create** > **Dataset**.

2. Click Salesforce.

The dataset builder opens on the Dataset Builder tab.



EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available for an additional cost in: **Enterprise**, **Performance**, and **Unlimited** Editions

USER PERMISSIONS

To access the dataset builder:

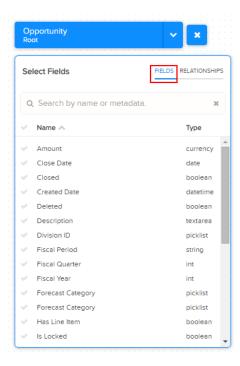
"Edit Wave Analytics Dataflows"

3. 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 it's children objects. To change the root object, refresh the page and start over.

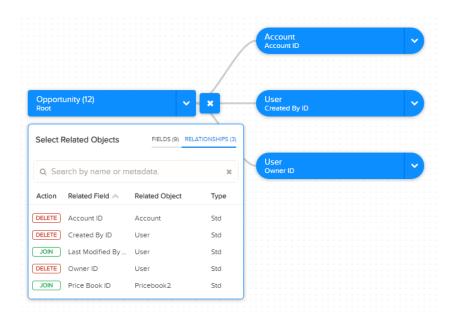
4. Hover over the root object, and then click .

The Select Fields dialog box appears. By default, the Fields tab appears and shows all available object fields from which you can extract data.

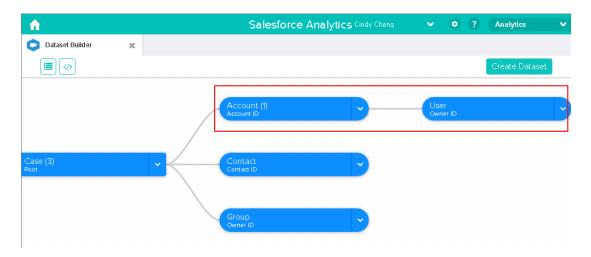


- Note: You can view this dialog box for any object included in the canvas.
- **5.** In the Fields tab, select the fields from which you want to extract data.

 To locate fields more quickly, you can search for them or sort them by name or type.
 - (1) Important: You must 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 it's fields, the dataflow fails at run time.
- **6.** In the Relationships tab, click **Join** to add the related objects to the canvas. When you add a related object, the related object appears in the canvas.

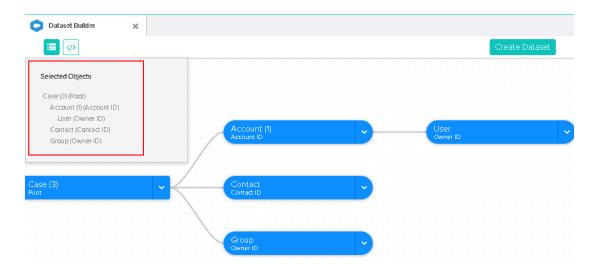


- 7. To remove a related object, click **Delete**.
 - Warning: When you delete a related object, you also delete all objects that descend from the related object in the diagram. For example, if you delete Account shown below, you delete the branch that contains Account and User.



- **8.** For each related object, select the fields from which you want to extract data.
- **9.** To move the entire diagram, select a white space in the canvas and drag it. You might need to move the diagram to view a different section of the diagram.
- **10.** To view all objects included in the canvas, click .

 The Selected Objects dialog box shows a tree structure of all objects included in the canvas. The root object appears at the top of the tree.



If you select one of the objects, the dataset builder focuses on the object by placing the object in the center of the canvas.

When you create the dataset, the dataset builder appends the JSON to the dataflow definition file.

12. Click Create Dataset.

13. Enter the name of the dataset, and select the app that will contain the dataset if it's not already selected.



🕜 Note: If you enter a dataset name that is already used, when you create the dataset, the dataset builder appends a number to the dataset name. For example, if you entered MyOpportunities, the dataset builder creates MyOpportunities1. The dataset name cannot exceed 80 characters.

14. Click Create.

The dataset builder appends the underlying JSON to the dataflow definition file. The dataset is created the next time the dataflow runs. You can manually run the dataflow to immediately create the dataset.

Optimize Your Dataflows with Replication

Replication decouples the extract of Salesforce data from your dataflows, letting you run it on a separate schedule. By scheduling these extracts ahead of time, your dataflow has less to do and runs faster. To lighten the load even more, replication extracts Salesforce data incrementally by default, meaning that only data that's changed gets extracted.

Without replication, a dataflow performs a separate extract each time it needs data from a Salesforce object. Let's take a 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		E	*				8
Sales Wave App Dataflow		E	"		$\dot{\star}$		8
Service Wave App Dataflow		E					8

Every time these dataflows run, they have to extract all this Salesforce data. And the more data there is, the longer the dataflow takes to run. What's more, the dataflows are duplicating effort by performing separate extracts from the same object. For example, Accounts data is extracted by all three dataflows.

With replication, 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
Replication		E	<u> </u>		*	â	2

This replicated data is then available to all your dataflows, which run faster because they no longer have to extract any data—just load and transform.

IN THIS SECTION:

Understand What Happens When You Enable Replication

When you enable replication, Wave looks at the sfdcDigest nodes in your existing **scheduled** dataflows to see which Salesforce objects and fields you're currently extracting. Using this information, Wave enables replication for each of these objects to extract the data separately. Understanding exactly what happens when you enable replication helps you identify any additional actions that you might need to take to ensure replication runs successfully.

Enable Replication

Enable replication to decouple the extract of Salesforce data from your dataflows.

Schedule, Run, and Monitor Replication

You can schedule replication to run automatically, or run it manually, and monitor its progress, all in the data manager.

Extract Data Incrementally in Replication

By default, replication extracts data from each Salesforce object incrementally. Records are inserted, updated, or deleted to match changes in the object since the previous replication run. Because only changes are extracted, replication runs faster.

Configure Replication Settings

When a Salesforce object is enabled for replication, you can configure its replication settings to disable fields that you don't want to replicate, add a filter to only replicate specific data, or disable replication for an object. You do all this through the data manager.

Disable Replication for an Object

Disable replication for an object in the data manager to exclude it from replication.

Add Objects and Fields to Replication

It is not possible to add objects or fields to replication in the data manager. To add an object, add an sfdcDigest node for it to one of your dataflows. To add a field for an 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 replication when you upload the updated dataflow definition file.

Considerations When Creating or Updating Dataflows with Replication Enabled

If you have enabled replication, here are some things to keep in mind the next time you create or update one of your dataflows.

Considerations When Using Wizard-Based Wave Apps with Replication

If you use wizard-based Wave Analytics apps, such as Sales or Service Wave, here are some things to consider before and after you enable replication.

Replication Limits and Limitations

Here are some things to consider when you're working with replication.

Understand What Happens When You Enable Replication

When you enable replication, Wave looks at the sfdcDigest nodes in your existing **scheduled** dataflows to see which Salesforce objects and fields you're currently extracting. Using this information, Wave enables replication for each of these objects to extract the data separately. Understanding exactly what happens when you enable replication helps you identify any additional actions that you might need to take to ensure replication runs successfully.

Here's what happens, and the action we recommend that you take.

Your Existing Dataflows Are Validated

Because replication is based on your existing dataflows, Wave validates your scheduled dataflows to ensure that they are error free. Although dataflows with errors are not considered for replication, Wave alerts you to any errors in sfdcDigest nodes when you enable replication.

Recommended Action

Before you enable replication, review you scheduled dataflows for errors. In particular, look out for incorrect object or field names in sfdcDigest nodes. If you see errors displayed when you enable replication, fix them in your dataflows before you continue. When you fix errors in a dataflow, Wave automatically considers it for replication when you upload the dataflow definition file.



Important: Wave does not consider unscheduled dataflows when you enable replication. Make sure that any dataflow that you want to be considered is scheduled. See Reschedule and Unschedule a Dataflow.

Your Dataflows Are Processed in Order of Dataflow Creation Date

Wave considers the sfdcDigest nodes in your dataflows in the order in which they were created. If there are multiple sfdcDigest nodes for the same object in your dataflows, they are merged. If there are conflicts with object-level parameters, the values from the most recently created dataflow are used for replication. 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)
{
  "Extract OpportunitiesA": {
                                                   "Extract OpportunitiesB": {
     "action": "sfdcDigest",
                                                     "action": "sfdcDigest",
     "parameters": {
                                                      "parameters": {
         "object": "Opportunity",
                                                         "object": "Opportunity",
         "incremental": false,
                                                         "incremental": true,
         "fields": [
                                                         "fields": [
            { "name": "Id" },
                                                            { "name": "Id" },
            { "name": "Name" },
                                                            { "name": "Name" },
            { "name": "StageName" },
                                                            { "name": "StageName" },
            { "name": "CloseDate" },
                                                            { "name": "CloseDate" },
            { "name": "AccountId" },
                                                            { "name": "AccountId" },
            { "name": "OwnerId" }
                                                            { "name": "OwnerId" }
         1
                                                         1
     }
                                                      }
  }
                                                   }
```

In the resulting replication for this object, the incremental parameter is set to true, from the most recently created dataset. This also applies to other object-level attributes, such as fullRefreshToken and enablePKChunking.

Recommended Action

Review your dataflows to ensure that object-level parameters are consistent across all sfdcDigest nodes for the same object.

All Fields in Multiple sfdcDigest Nodes for an Object Are Included in Replication

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 replication. 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 replication. 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) "Extract OpportunitiesA": { "Extract OpportunitiesA": { "action": "sfdcDigest", "action": "sfdcDigest", "parameters": { "parameters": { "object": "Opportunity", "object": "Opportunity", "fields": ["fields": [{ "name": "Id"}, { "name": "Id"}, { "name": "Name" }, { "name": "Name" }, { "name": "StageName" }, { "name": "StageName" }, { "name": "CloseDate", { "name": "CloseDate", "fiscalMonthOffset": 9 }, "fiscalMonthOffset": 3 } { "name": "AccountId" }, 1 { "name": "OwnerId" } } }] } }

In the resulting replication 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 replication includes fields that you don't require, you can disable them in the replication settings for that object (see Configure Replication Settings). If you have field attribute conflicts, ensure that the attributes you want to use for replication are in the most recently created dataflow.

Advanced Filters on sfdcDigest Nodes Are Included in Replication

If an sfdcDigest node includes a complexFilterConditions parameter to apply an advanced filter, Wave adds this filter to the replication 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 replication settings. Wave ignores structured filters applied through filterConditions parameters.

Recommended Action

After you enable replication, review each object's replication settings to verify that filters have been applied correctly (see Configure Replication Settings). To use multiple filters for an object, we recommend that you remove any filter from replication settings, and add the filters to the corresponding dataflow using the filter transformation (see filter Transformation).



Note: If you have participated in the replication pilot program and have sfdcDigest node filters in your replication dataflow, these filters are not added to replication. Add these filters manually through the object's replication settings. See Configure Replication Settings.

SEE ALSO:

Configure the Dataflow sfdcDigest Transformation

Filtering Records Extracted from a Salesforce Object

Enable Replication

Enable replication to decouple the extract of Salesforce data from your dataflows.

(1) Important: We recommend that you read Understand What Happens When You Enable Replication before you enable replication. This will help you identify any additional actions that you might need to take to ensure replication runs successfully.

To enable replication, perform the following steps.

- 1. From Setup, enter Wave Analytics in the Quick Find box, then select **Settings**.
- 2. Select Enable Replication, and then click Save.
- (1) Important: After you enable replication, make sure you run replication before your dataflows next run. Dataflows with sfdcDigest nodes fail until replication has run and completed for the first time. See Schedule, Run, and Monitor Replication.

Schedule, Run, and Monitor Replication

You can schedule replication to run automatically, or run it manually, and monitor its progress, all in the data manager.

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 enable replication:

"Customize Application"

(1) Important: After you enable replication, make sure you run replication and it completes before your dataflows next run. Dataflows with sfdcDigest nodes fail until replication has run for the first time.

IN THIS SECTION:

Schedule Replication

Schedule replication to run automatically.

Run Replication

Run replication manually at any time in the data manager.

Monitor Replication

Monitor the progress of replication in the data manager.

Schedule Replication

Schedule replication to run automatically.

When you schedule replication, you schedule it to run for all your enabled objects at the same time. You cannot schedule separate replications for each object.



Note: A replication run counts towards your limit of 24 dataflow runs per day. For example, in an organization using the Sales Wave and Service Wave apps, a user dataflow, and replication, there are four separate dataflows. If you schedule each one to run every six hours, you will reach the daily limit. Consider this limit when you schedule replication.

To schedule replication, perform the following steps.

- 1. In Wave Analytics, click the gear icon () and then click **Data Manager** to open the data manager.
- **2.** In the data manager, click the Ingest tab.

 The Ingest tab displays a list of Salesforce objects enabled for replication.
- 3. Click Edit Replication Schedule.
 - The schedule dialog appears.
- **4.** Enter the frequency and then click **Save**.
- 5. Click Done.

Run Replication

Run replication manually at any time in the data manager.



Note: When you run replication manually, consider that it counts towards your limit of 24 dataflow runs per day.

To run replication, perform the following steps.

- 1. In Wave Analytics, click the gear icon (and then click **Data Manager** to open the data manager.
- **2.** In the data manager, click the Ingest tab.

 The Ingest tab displays a list of Salesforce objects enabled for replication.
- 3. Click Run Replication Now.

The Monitor page of the data manager opens to display the replication of each object as a separate job.

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 schedule replication:

 "Edit Wave Analytics Dataflows"

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 run replication:

 "Edit Wave Analytics Dataflows"

Monitor Replication

Monitor the progress of replication in the data manager.

To monitor replication, perform the following steps.

- 1. In Wave Analytics, click the gear icon () and then click **Data Manager** to open the data manager.
- In the data manager, click Monitor.
 The Jobs View of the monitor displays information about your replication jobs.

Extract Data Incrementally in Replication

By default, replication extracts data from each Salesforce object incrementally. Records are inserted, updated, or deleted to match changes in the object since the previous replication run. Because only changes are extracted, replication runs faster.



Note: A full extract is performed the first time replication runs for an object.

To turn off incremental mode for a replicated object, add the incremental parameter to the object's sfdcDigest node in your dataflow, and set its value to false. For example:

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 monitor replication:

 "Edit Wave Analytics Dataflows"

To force a full extract for a replicated object, add the fullRefreshToken parameter to the object's sfdcDigest node, and give it a random value. For example:

```
"Extract_Opportunities": {
    "action": "sfdcDigest",
    "parameters": {
        "object": "Opportunity",
        "fullRefreshToken": "bb",
        "fields": [
            { "name": "Id" },
            { "name": "Name" },
            { "name": "StageName" },
            { "name": "CloseDate" },
            { "name": "AccountId" },
            { "name": "OwnerId" }
```

```
]
}
}
```

The next time you need to force a full extract for this object, change the value in the fullRefreshToken parameter.

Considerations When Using Incremental Extract with Replication

- If incremental mode is explicitly turned off in an sfdcDigest node, and not explicitly turned on in later nodes for the same object, it is not turned on when you enable replication. You have to manually turn it on by setting the incremental parameter to true in the dataflow.
- When you enable replication, incremental mode is not turned on for Salesforce objects that perform hard deletes on records, unless it's explicitly turned on in the dataflow.
- Changes made less than one hour before replication runs are not picked up. Bear this in mind if you have records that are frequently updated.
- As a best practice, force a full refresh regularly, using the fullRefreshToken parameter, to ensure that all changes are picked up.
- Formula fields can become out of sync with your replicated object in incremental mode. We recommend that you turn off incremental extracts for objects containing formula fields.
- If you make field changes in an sfdcDigest dataflow node, such as adding or removing fields, or changing attributes, Wave triggers a full extract of data the next time replication runs.

SEE ALSO:

Configure the Dataflow sfdcDigest Transformation

Configure Replication Settings

When a Salesforce object is enabled for replication, you can configure its replication settings to disable fields that you don't want to replicate, add a filter to only replicate specific data, or disable replication for an object. You do all this through the data manager.

To configure replication settings, perform the following steps.

- 1. In Wave Analytics, click the gear icon () and then click **Data Manager** to open the data manager.
- In the data manager, click the Ingest tab.The Ingest tab displays a list of Salesforce objects enabled for replication.
- **3.** Click an object to open its replication settings. A preview of the replicated data appears.
- **4.** To disable replication for a field, hover over the field's column header and click the cross that appears.

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 configure replication settings:

 "Edit Wave Analytics Dataflows"



Note: You can't disable replication for a field that's used in a dataflow, or disable replication for an object's ld field.

5.

To filter the data that's replicated, click



a. Enter a filter using a Salesforce Object Query Language (SOQL) WHERE clause expression. For example:
RecordType = 'B2B' AND Status = 'Open'

b. Click Done.



Note: Your preview data does not reflect any filter you add. Filters are only applied when you save the replication settings.

- **6.** Click **Save** to save the replication settings.
- Note: If you disable replication for a field, you cannot enable it again through the replication settings. You can re-enable a field by adding it to the sfdcDigest node in your dataflow. See Add Objects and Fields to Replication.

Disable Replication for an Object

Disable replication for an object in the data manager to exclude it from replication.

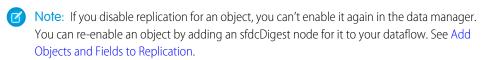


Note: You can't disable replication for an object that's used in a dataflow.

To disable replication for an object, perform the following steps.

- 1. In Wave Analytics, click the gear icon () and then click **Data Manager** to open the data manager.
- **2.** In the data manager, click the Ingest tab.

 The Ingest tab displays a list of Salesforce objects enabled for replication.
- Click the actions button () to the right of the object and then select Disable Replication.
 A warning message appears.
- **4.** Read the warning message carefully and then click **Disable**.



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 disable replication for an object:

 "Edit Wave Analytics Dataflows"

Add Objects and Fields to Replication

It is not possible to add objects or fields to replication in the data manager. To add an object, add an sfdcDigest node for it to one of your dataflows. To add a field for an 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 replication when you upload the updated dataflow definition file.

SEE ALSO:

Configure the Dataflow sfdcDigest Transformation

Considerations When Creating or Updating Dataflows with Replication Enabled

If you have enabled replication, here are some things to keep in mind the next time you create or update one of your dataflows.

- Wave validates the dataflow definition file when you upload it. If you see errors displayed, correct them and upload the file again.
- Dataflow definition file uploads might take longer. This is because Wave is validating the file and using the sfdcDigest nodes to define replication settings.
- If you remove an object's sfdcDigest node from the definition file, replication is not disabled for that object. If necessary, disable replication for an object on the Replication tab of the data manager. See Disable Replication for an Object.
- If you remove a field from an sfdcDigest node in the definition file, replication is not disabled for that field. If necessary, disable replication for the field in replication settings. See Configure Replication Settings.
- If you add a field in the definition file, that field is enabled in the replication settings.
- If you unschedule the dataflow, Wave does not disable replication for the objects and fields.

Considerations When Using Wizard-Based Wave Apps with Replication

If you use wizard-based Wave Analytics apps, such as Sales or Service Wave, here are some things to consider before and after you enable replication.

Enabling Replication When You Are Already Using Wave Apps

If you already have Wave apps when you enable replication, it's important that replication includes all the Salesforce objects and fields used in these apps. If it doesn't, the affected Wave app dataflows fail. Complete these steps to ensure all the required objects and fields are included.

- 1. Before you enable replication, set a recurring schedule on each of your Wave app dataflows. Wave considers objects and fields in only scheduled dataflows when you first enable replication.
- **2.** After you enable replication, check replication settings to verify that all the required objects and fields have been included. See Configure Replication Settings.
- 3. Run replication to ensure that all objects and fields in the Wave app dataflows have been replicated.
- **4.** Run your Wave app dataflows to verify that they run to completion.
- 5. Schedule replication to run before all your dataflows, to keep your Wave app data up to date.

Creating a New Wizard-Based Wave App After Enabling Replication

When you create a wizard-based Wave app, objects and fields from the new dataflow are automatically added to replication, if they're not already included. Complete these steps after you create a Wave app.

1. Check replication settings to verify that all the required objects and fields have been included. See Configure Replication Settings.

- 2. Run replication to ensure that all objects in the new dataflow have been replicated.
- 3. Run the new Wave app dataflow.
- **4.** Schedule the new Wave app dataflow to run after replication.

Replication Limits and Limitations

Here are some things to consider when you're working with replication.

- You can enable up to 50 objects for replication. If you are currently extracting more than 50 objects in your dataflows, contact Salesforce Customer Support before you enable replication
- Replication only performs one extract per object. Keep this in mind if you need to perform multiple extracts from the same object. For example, if you need to extract both filtered and unfiltered data from an object.
- By default, replication extracts data from each object incrementally. However, changes made less than one hour before replication runs are not picked up. Keep this in mind if you have records that are frequently updated.
- Replication counts towards your limit of 24 dataflow runs per day. For example, in an organization using the Sales Wave and Service
 Wave apps, a user dataflow, and replication, there are four separate dataflows. If you schedule each one to run every six hours, you
 will reach the daily limit.
- There is a known issue where replication does not generate notification emails.

Wave Connector for Excel Data

The Salesforce Wave Connector makes it easy to import data from Microsoft Excel 2013 to Wave Analytics.

The Wave Connector is available as an app for Excel 2013 on the desktop and Excel Online in Office 365. The Connector is available as an app from the Microsoft Apps for Office store or your organization's private app catalog. After you install the Connector just point and click to import data from Excel to Salesforce.

IN THIS SECTION:

Install the Wave Connector Excel App

The Wave Connector app gives users a fast, easy way to import data from Excel 2013 into Salesforce Wave Analytics..

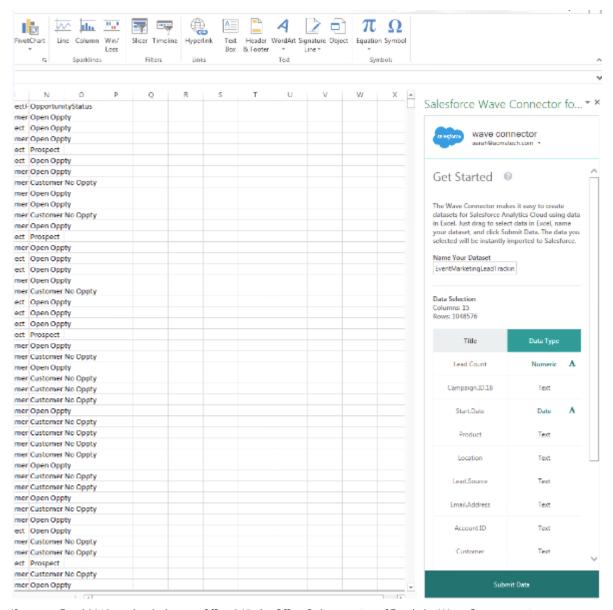
Install the Wave Connector Excel App

The Wave Connector app gives users a fast, easy way to import data from Excel 2013 into Salesforce Wave Analytics..

USER PERMISSIONS

To import data from Excel 2013 to Wave Analytics :

 Upload External Data to Wave Analytics



If you use Excel 2013 on the desktop or Office 365, the Office Online version of Excel, the Wave Connector gives you a great way to get your data into Salesforce Wave Analytics. After installing the Connector, you just select data from Excel, click Submit, and the Connector does the work for you, importing the data to Wave Analytics and creating a dataset.

Here's how to install the Connector:

- 1. Open Excel, either on your desktop or in Office Online.
- 2. Open the Insert tab.
- 3. Click Apps for Office.
- 4. Search for the Wave Connector, and click to install it.
- **5.** Enter your Salesforce credentials to open the Connector.

Once you've installed the Connector, follow the instructions in the Connector window to create datasets based on Excel data. Opening the Connector automatically logs you in to Salesforce Wave Analytics. Click the Connector Help icon for complete information about using the app.

Create a Dataset with External Data

You can either upload external data through the user interface or through the External Data API to create a dataset. When you upload an external data file (in .csv, .gz, or .zip format), you can also provide a metadata file. A metadata file contains metadata attributes that describe the structure of the data in the external data file. If you upload a .csv from the user interface, Wave Analytics automatically generates the metadata file, which you can preview and change. If you do not provide a metadata file, Wave Analytics imports all external data file columns as dimensions.

?

Tip: Wave temporarily stores the uploaded CSV and metadata files for processing only. After the datasets are created, Wave purges the files. If you want to use the files again later, keep a copy.

Before uploading external data files, review the format requirements and examples of the .csv and metadata files in the External Data Format Reference.

Ø

Note: You can also use the the External Data API to upload external data files. Use the API to take advantage of additional features, like performing incremental extracts and performing append, delete, and upsert operations. For more information about the External Data API, see the External Data API Developer's Guide.

1. On the home or app page, click **Create** > **Dataset**.



2. Type the name of your dataset in the Dataset Name field.

The name cannot exceed 80 characters.

- 3. If you want to create the dataset in a different app, change the app in the App drop-down list.
- 4. Click CSV.
- 5. Add the .csv file.

After you add the .csv file, Wave Analytics automatically generates and adds the corresponding metadata file.

EDITIONS

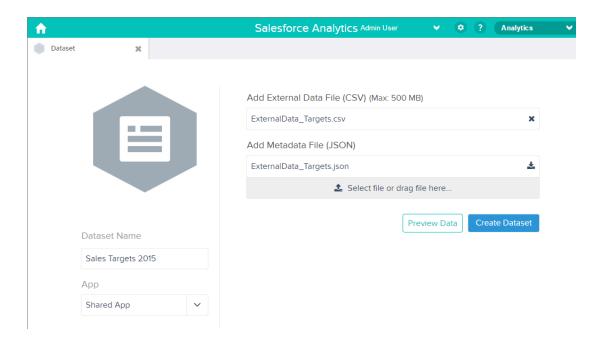
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

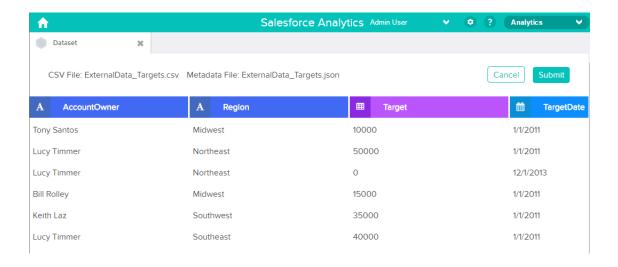
USER PERMISSIONS

To upload external data:

"Upload External Data to Wave Analytics"



- Note: Instead of using the generated metadata file, if you want, you can upload a different metadata file that you created from scratch. If you upload your own metadata file, the **Preview Data** button is disabled.
- **6.** Perform the following tasks to change the metadata attributes in the generated metadata file.
 - a. Click **Preview Data** to view and change the required metadata attributes.



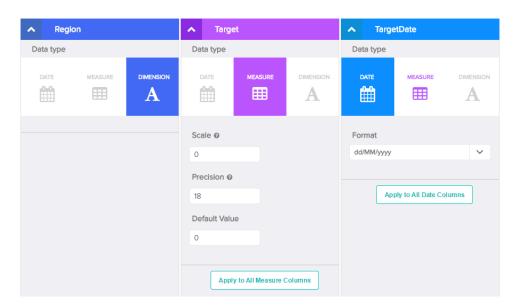
b. Click a column name to change it. The column name is the display name in the dataset. The column name cannot exceed 40 characters.



c. Click the column header to change other attributes for the column.



You can change the attributes for measure and date columns only.



- **d.** To apply the changes to all other columns of the same data type, click **Apply to All <data type> Columns**.
- e. Click **Submit** to save the metadata changes to the metadata file.
 - Note: If there are errors, the **Submit** button is grayed out.
- **f.** Click **OK** to close the confirmation message.
- g. To change optional metadata attributes, click 👪 to download the metadata file, edit the file, and then upload it.

7. Click Create Dataset.

Your data files are scheduled for upload. It might take some time to process the data upload job; you can monitor its status in the data monitor. If upload is successful, the new dataset is available from the home or app page.

8. Click **Continue** to dismiss the confirmation message.

IN THIS SECTION:

Rules for Automatic Generation of a Metadata File

When you upload a CSV file from the user interface, Wave Analytics 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, Wave Analytics kicks off a job that uploads the data into the specified dataset. You can use the data monitor to monitor and troubleshoot the upload job.

Rules for Automatic Generation of a Metadata File

When you upload a CSV file from the user interface, Wave Analytics automatically generates the metadata file as long as the CSV file meets certain requirements.

To enable Wave Analytics 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 Wave Analytics requirements as mentioned in the External Data Format Reference.

Wave Analytics generates the metadata attributes for each CSV column based on the first 100 rows in the CSV file. Wave Analytics 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, Wave Analytics kicks off a job that uploads the data into the specified dataset. You can use the data monitor to monitor and troubleshoot the upload job.

The Jobs view (1) of the data monitor shows the status, start time, and duration of each dataflow job and external data upload job. It shows jobs for the last 7 days and keeps the logs for 30 days.

EDITIONS

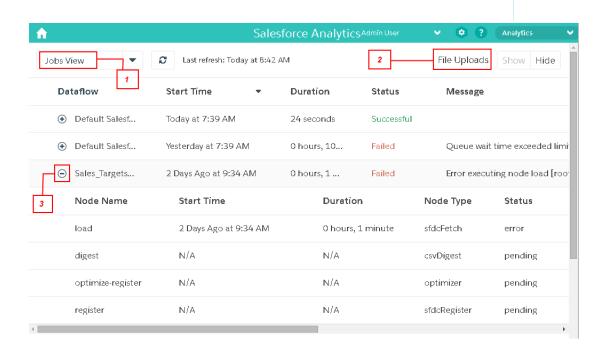
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To access the data monitor:

 "Edit Wave Analytics Dataflows," "Upload External Data to Wave Analytics," or "Manage Wave Analytics"



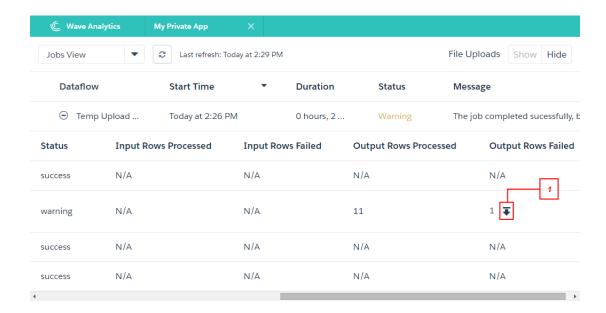
- In Wave Analytics, click the gear icon () and then click **Data Manager**.

 The data manager opens on the Monitor tab, with the Jobs view selected by default. The Jobs view displays dataflow and upload jobs. 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, **Show** in the File Uploads field (2) must be selected. It's selected by default.

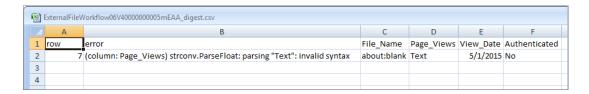
To see the latest status of a job, click the Refresh Jobs button (). Each job can have one of the following statuses.

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.	

- **3.** To view the run-time details for a job, expand the job node (3). 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. Also, click the download button (1) in the run-time details section to download the error log.
 - Note: Only the user who uploaded the external data file can see the download button.



The error log contains a list of failed rows.



5. To troubleshoot a failed job, view the error message and the run-time details.

Edit a Dataset

USER PERMISSIONS

To view a dataset edit page:	"Use Wave Analytics" AND Editor access to the dataset's app
To update a dataset name, app, and extended metadata:	"Use Wave Analytics" AND Editor access to the dataset's app
To delete a dataset:	"Use Wave Analytics" AND Editor access to the dataset's app
To upload and preview data:	"Upload External Data to Wave Analytics" AND Editor access to the dataset's app
To edit a dataset security predicate:	"Edit Wave Analytics Dataflows"

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

You can edit a dataset to change the dataset name, app, security predicate, or extended metadata (XMD) file associated with the dataset. For datasets created from an external data file, you can also upload a new external data file or metadata file to update the data or metadata.

If you add an external data file, Wave Analytics generates and adds the corresponding metadata file. To make further changes to the metadata, you can click **Preview Data** or download and edit the generated metadata file. You can also upload your own metadata file to overwrite the generated file.

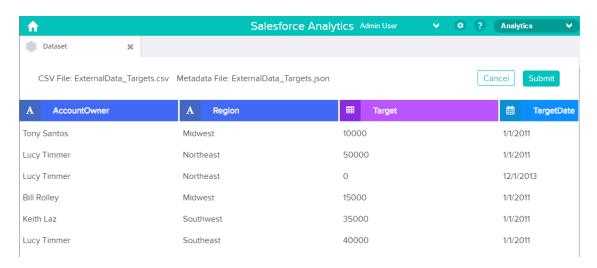
- 1. On the home or app page, click the **Datasets** tab.
- 2. Hover over the dataset that you want to edit, and then click Edit.
- **3.** Configure the following options if applicable.

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.
Арр	Select a new app if you'd like to move the dataset to a different app.
Add Extended Metadata File (JSON)	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.
Add External Data File (CSV)	Specify an external data file if you'd like to replace the existing data in the dataset with data from the external data file. Maximum file size is 500 MB. You can upload a .CSV, .GZ, or .ZIP file.

Option	Description	
	Refer to External Data Format Reference for information about external data files and metadata files.	
Add Metadata File (JSON)	Specify a metadata file if you'd like to redefine the structure of the external data file. If you upload a new metadata file, you must also upload the corresponding external data file.	
	Refer to External Data Format Reference for information about metadata files.	
Security Predicate	Add a security predicate if you'd like to apply row-level security on the dataset.	
	For information about predicates, see Row-Level Security for Datasets.	

- **4.** If you uploaded a new .csv file, click **Preview Data** to view and change the required metadata attributes. You can change the optional metadata later.
 - Note: The **Preview Data** button is disabled if you uploaded your own metadata file.

After you click **Preview Data**, the preview page appears.



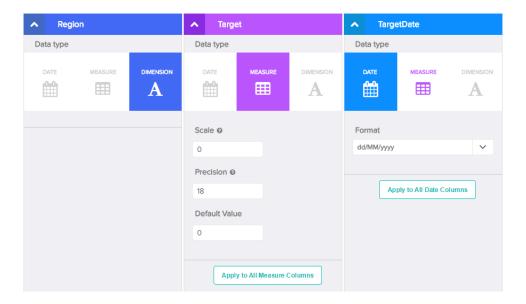
- **5.** For each column:
 - **a.** Click a column name to change it. The column name is the display name in the dataset. The column name cannot exceed 40 characters.



b. Click the column header to change other required attributes for the column.



You can change the attributes for measure and date columns only.



- c. To apply the changes to all other columns of the same data type, click **Apply to All <data type> Columns**.
- **6.** Click **Submit** to save the metadata changes in the preview page to the metadata file.
 - Note: The **Submit** button is grayed out if there are errors.
- 7. Click **OK** to close the confirmation message.
- **8.** To change optional metadata attributes—which are not visible in the preview page—click **L** to download the metadata file, edit the file, and then upload it.
- 9. Click Update Dataset.
- **10.** Click **Continue** to dismiss the confirmation message.

IN THIS SECTION:

Update a Dataset's Extended Metadata

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.

Update a Dataset's Extended Metadata

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.

- (1) 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 Wave Analytics page, click the **Datasets** tab.
- 2. Hover over the dataset that you want to update, click , and then click **Edit**. The dataset edit page opens.
- **3.** Under Add Extended Metadata File (JSON), click the Download button to download the current XMD file. Save this to use as a backup.



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

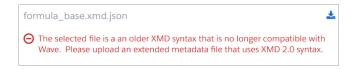
To update a dataset's extended metadata:

 "Use Wave Analytics" AND Editor access to the dataset's app

- 4. Under Add External Metadata File (JSON), click **Select file or drag file here...** to upload the new XMD file.
- 5. Click **Update Dataset**.

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.



Perform step 2 to download the file, then convert it to XMD 2.0. Then perform steps 4–5 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.

Delete a Dataset

Delete unnecessary datasets from your My Private App or in shared apps on which you have at least Editor access. Removing datasets reduces clutter and helps you avoid reaching your org's limit for rows across registered datasets.

When you delete a dataset, Wave Analytics permanently deletes the dataset and doesn't delete the corresponding lenses or dashboards that reference the dataset. Lenses and dashboards that reference a deleted dataset will no longer be available. As a result, Salesforce.com recommends that you remove the associated lenses and dashboards before you delete a dataset.

If a dataflow transformation —like edgemart or sfdcRegister—references the dataset, you must remove the reference before you can delete the dataset. For example, to delete the "Opportunities" dataset, you must remove the sfdcRegister transformation from the dataflow snippet shown below.

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 delete a dataset:

"Use Wave Analytics"
AND Editor access to the dataset's app

```
"Register_Dataset": {
    "action": "sfdcRegister",
    "parameters": {
        "alias": "Opportunities",
        "name": "Opportunities",
        "source": "Extract_Opportunities"
    }
},
```

- Warning: You can't recover a deleted dataset.
- 1. On the home or app page, click the **Datasets** tab.
- 2. Hover over the dataset that you want to delete, and then click Edit.
- 3. Click Delete Dataset.

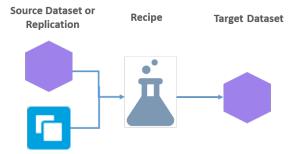
If applicable, Wave Analytics shows a list of all lenses and dashboards that reference the dataset and that you have access to view. After you delete the dataset, any lens or dashboard that reference the dataset will become unusable.

4. Click **Delete Permanently** and confirm.

Prepare Datasets with Recipes

A dataset recipe is a user-interface tool that lets you take data from your existing datasets and replications, prepare it, and output the results to a new dataset. Use a recipe to combine data from multiple datasets or replications, bucket the data, add formula fields, and cleanse the data by transforming field values. You can remove fields and filter rows that you don't need, before you create the target dataset.

A dataset recipe is simply a saved set of transformations, or steps, that you want to perform on a specific source dataset or replication. When you run a recipe, it applies the transformations and outputs the results to a new target dataset. You can schedule a recipe to run on a recurring basis to keep your target dataset up to date.



Use a recipe to combine data from different sources, and modify field values to ensure consistency in the new dataset you create. This new dataset can then be used as a standalone dataset for exploration or dashboard creation, or used in your dataflows or other recipes.

Here's what you can do with your source data in a recipe:

- Add fields from another dataset or replication.
- Remove fields.
- Filter rows.
- Add bucket fields.
- Add formula fields.
- Transform field values.

IN THIS SECTION:

Create a Dataset Recipe

Use a recipe to combine data from multiple datasets or replications, add bucket and formula fields, filter rows, and transform field values. You can then output the results to a new target dataset.

Manage and Monitor Dataset Recipes

View the status of your recipes and target datasets on the Prepare tab of the data manager. You can open recipes to update or run them, or delete them.

Create a Dataset Recipe

Use a recipe to combine data from multiple datasets or replications, add bucket and formula fields, filter rows, and transform field values. You can then output the results to a new target dataset.



Tip: Before you get started, it's a good idea to think about the requirements for the target dataset you want to create. Consider these questions:

- Does the data you want to work with exist in a dataset or as a replicated object?
- If you are combining data, how is the data related?
- What fields do you want in the target dataset?
- Do you need to transform any field values?
- In what app are you going to create the target dataset?
- Do you need to control access to the data in the target dataset?

When you're ready, follow these steps to create a dataset recipe.

IN THIS SECTION:

Start a Dataset Recipe

To start a dataset recipe, perform the following steps.

Set Preview Data in a Dataset Recipe

A dataset recipe displays a preview of your data as you work. You can set the fields 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 Data in a Dataset Recipe

Combine data in a recipe by adding fields from other datasets or replications.

Add a Filter in a Dataset Recipe

Filter data in a dataset recipe to remove rows that you don't need in your target dataset.

Bucket a Measure Field in a Dataset Recipe

Add a field to a dataset recipe to bucket values in a specified measure field.

Bucket a Dimension Field in a Dataset Recipe

Add a field to a dataset recipe to bucket values in a specified dimension field.

Add a Formula Field in a Dataset Recipe

Add a formula field to calculate new values from measures in your recipe, using arithmetic operators and math functions.

Transform Fields in a Dataset Recipe

Data is not always consistent, especially when you combine data from different sources. In a dataset recipe, you can transform fields to ensure that values are consistent in your target dataset. Change case, split or truncate values to get just the parts you need, and replace incorrect values.

Navigate Fields in a Dataset Recipe

As you add data and transform fields, the number of fields in your recipe preview increases. This makes it harder to find the fields that you want to work with. Use the field navigator to quickly find the fields that you need, and hide the ones you don't. The navigator also shows you the attributes of a field, such as its API name, and field type.

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 Dataset Recipe

If you're not ready to create the target dataset, save the recipe and come back to it later. Saving simply saves your steps without you having to create the dataset.

Run a Dataset Recipe

Run a new recipe to create a target dataset, or run an existing recipe to update a target dataset.

Start a Dataset Recipe

To start a dataset recipe, perform the following steps.

1. In Wave Analytics, 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 Prepare tab. The Prepare tab displays a list of your existing recipes.

3. Click Create Recipe.

- **4.** From the list of datasets, click the one that contains the data you want to prepare. Alternatively, if you have enabled replication, you can click the Replications tab to select a replication instead.
 - Note: Make sure the replication you select has run and completed at least once.

You are prompted to enter a name for the recipe.

- **5.** Enter a recipe name and then click **Continue**.
 - Note: The recipe name you enter here is also used for the target dataset name.

The dataset recipe page opens, displaying a preview of the data from your selected dataset or replication.

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 a recipe:

 "Edit Wave Analytics Dataflows"

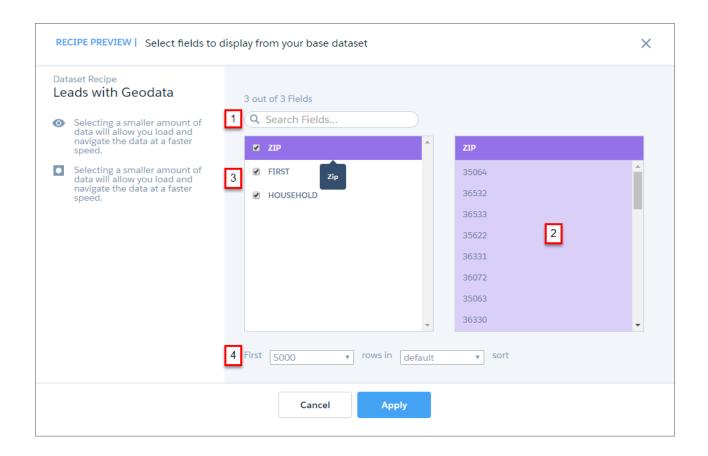
Set Preview Data in a Dataset Recipe

A dataset recipe displays a preview of your data as you work. You can set the fields 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.

As you work in a dataset recipe, the size of your preview is always visible at the top of the page.



Click the
to view or change what you see in the preview.



Here's what you can do to set your recipe preview data:

- 1. Can't see a field in the list? Start typing the field name here to narrow down the list.
- 2. Not sure if you need a field in your preview? Click a field in the list to see a preview of its values here.
- 3. Don't need a field in your preview? Uncheck it here to remove it.
- 4. Want fewer rows in your preview? Choose to see the first 1000, 5000, or 10000 rows and select a field to sort them by.

Click **Apply** when you're done, to return to the recipe.

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 Data in a Dataset Recipe

Combine data in a recipe by adding fields from other datasets or replications.

When you add data to a recipe, you select the lookup dataset or replicated object that contains the fields you want to add. You also select the merge key fields to match the records in the existing and new data. For example, to add User fields to Opportunity records, use the User Id and Opportunity Owner Id fields as your merge keys.

Let's look at another example here, to illustrate how you add data. Imagine you have started a recipe from a US Leads dataset, and you want to add demographic data based on zip code from another dataset. You can use the Zip Code field from each dataset as the merge keys.

- 1. On the dataset recipe page, click the Add Data button (.................................).
- 2. Select either the Datasets or Replication tab, and click the dataset or object you want to add.



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 a recipe:

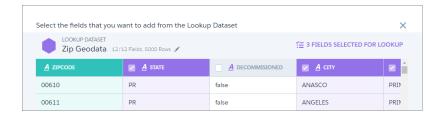
 "Edit Wave Analytics Dataflows"

3. Click a column header to select it as the merge key field in the origin dataset, and do the same in the lookup dataset.



Then click Continue.

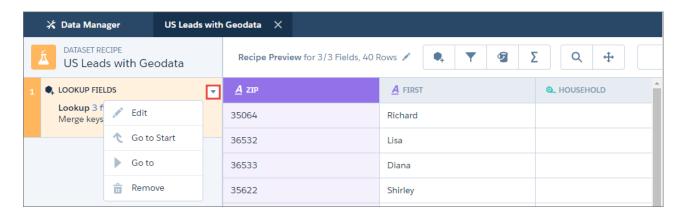
- (1) Important: Wave datasets contain date and time component fields, such as Year, Quarter, and Month, that are derived automatically from existing date and time fields. Use of these component fields as merge keys is not supported.
- **4.** Click the column header or column checkbox to select each field that you want to add.



Then click Continue.



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 edit and remove options.



Add a Filter in a Dataset Recipe

Filter Data Rows

Select Field

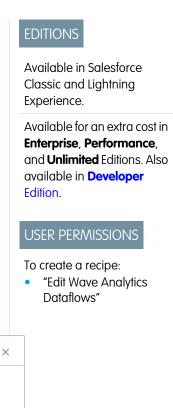
Include rows that meet ALL these conditions.

Filter data in a dataset recipe to remove rows that you don't need in your target dataset.

Select Comparison

Cancel

To add a filter to a dataset recipe, follow these steps.



+

2. From the **Select Field** list, select the field to filter by. You can select from any field in the recipe, even if it has been hidden.

- **3.** From the **Select Comparison** list, select the operator.
- **4.** In the third field, enter the filter value. Then click **Continue**.
 - Note: Filters in dataset recipes are case-sensitive.
- **5.** To enter another condition, click + and repeat steps 2–4.
- **6.** When you have added all your conditions, click **Filter Rows**. The filter appears as a step in the left pane of the recipe.



Note: Filters in dataset recipes only filter rows that meet all conditions.

Bucket a Measure Field in a Dataset Recipe

Add a field to a dataset recipe to bucket values in a specified measure field.

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. Select the field to bucket by either clicking its column header in the recipe preview, or clicking it in the field navigator.
- 2. Click the Add Bucket button (). The bucket dialog appears with two ranges.

EDITIONS

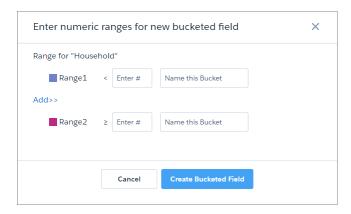
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To create a recipe:

 "Edit Wave Analytics Dataflows"



3. 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.

- **4.** If your bucket field has more than two buckets, click **Add>>** to add another range.
- **5.** 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.



- **6.** Repeat steps 4–5 to add any additional buckets that you need.
- 7. 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.



8. When you have completed all the ranges, click **Create Bucketed field**. 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.

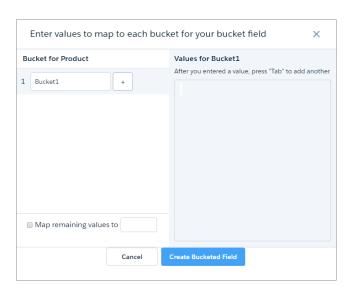
Bucket a Dimension Field in a Dataset Recipe

Add a field to a dataset recipe to bucket values in a specified dimension field.

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. Select the dimension field to bucket by either clicking its column header in the recipe preview, or clicking it in the field navigator.
- Click the Add Bucket button ().
 The bucket dialog appears.



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 a recipe:

"Edit Wave Analytics Dataflows"

- 3. In the Bucket1 field, enter a name for the first bucket.
- **4.** 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.



- **5.** To add more buckets, click + and repeat steps 3–4.
- **6.** To bucket remaining values that you have not bucketed, select **Map remaining values to** 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.
- **7.** When you have completed all the buckets, click **Create Bucketed field**. The new bucket field appears next to the source field in the recipe preview.



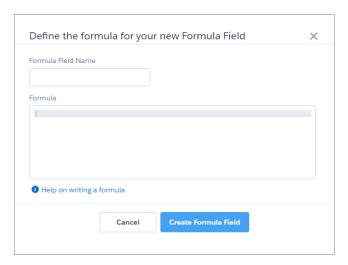
- ? Tip: You can use a bucket field to filter rows in a recipe.
- Note: You can't bucket a date field.

Add a Formula Field in a Dataset Recipe

Add a formula field to calculate new values from measures in your recipe, using arithmetic operators and math functions.

A formula field lets you calculate new values based on existing values in measure fields. To help you, a formula builder suggests fields and functions as you type.

Click the Add a Formula Field button (Σ).
 The formula field dialog appears.



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 a recipe:

"Edit Wave Analytics Dataflows"

- 2. In the Formula Field Name field, enter a name.
- **3.** In the **Formula** box, start typing the formula.

 As you type, the builder suggests matching math functions and measure fields.



- 4. To select a suggestion, either click it, or use your cursor up and down keys to select it and then press Enter or Tab.
- 5. Use your keyboard to enter parentheses, arithmetic operators, and numbers as needed.
- **6.** When you have completed the formula, click **Create Formula Field**. 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.

You can use the following math functions in a formula field. The arguments in each function can be either literal values or measure fields.

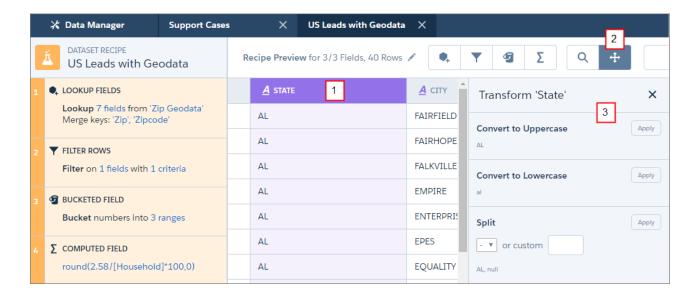
Function	Description	Format
abs	Calculates the absolute value of a number. The absolute value of a number is the number without its positive or negative sign.	abs(number)
ceil	Rounds a number up to the nearest integer.	ceil(number)
ехр	Returns a value for e raised to the power of a number you specify.	exp(number)
floor	Returns a number rounded down to the nearest integer.	floor(number)
log	Returns the base 10 logarithm of a number.	log(number)
power	Raises a number to a power of a specified number.	power(number, power)
round	Returns the nearest number to a number you specify, constraining the new number by a specified number of digits.	round(number, number of digits)
sqrt	Returns the positive square root of a given number.	sqrt(number)
trunc	Truncates a number to an integer.	trunc(number)

Transform Fields in a Dataset Recipe

Data is not always consistent, especially when you combine data from different sources. In a dataset recipe, you can transform fields to ensure that values are consistent in your target dataset. Change case, split or truncate values to get just the parts you need, and replace incorrect values.

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 the field header (1), and then click the Transform Suggestions button (2). Your transform options appear in the side panel (3). Click **Apply** to apply a transform. When you're finished, click another field header to move on to another field.



These are the field transforms you can apply.

Transform	Description	Example
Convert to Uppercase Apply	Convert values in selected field to uppercase.	Correct US state values. ca becomes CA.
Convert to Lowercase Apply	Convert values in selected field to lowercase.	Correct lead email addresses captured at a trade show. JSMITH@FORRESTER.COM becomes jsmith@forrester.com.
Split - ▼ or custom	Divide values into multiple parts at the delimiter and create a new 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.
Truncate Position 1 Length	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.

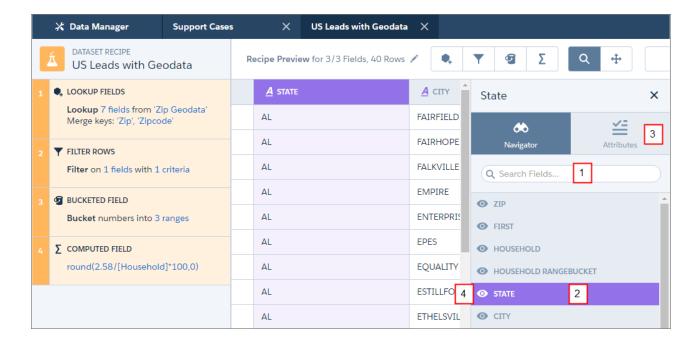


(1) Important: Wave datasets contain date and time component fields, such as Year, Quarter, and Month, that are derived automatically from existing date and time fields. Transformations on these component fields are not supported.

Navigate Fields in a Dataset Recipe

As you add data and transform fields, the number of fields in your recipe preview increases. This makes it harder to find the fields that you want to work with. Use the field navigator to quickly find the fields that you need, and hide the ones you don't. The navigator also shows you the attributes of a field, such as its API name, and field type.

To open the field navigator in a recipe, click the Navigate Fields button (...).



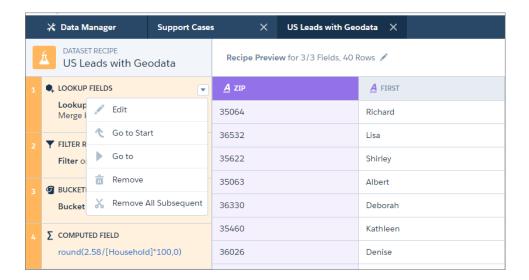
To quickly find a field, start typing its name in the Search Fields box (1) to narrow down the list of fields. When you see the field that you want, click it (2) to see it highlighted in the recipe preview. To learn more about the field's metadata, click **Attributes** (3). If you decide you don't need to see the field, click the eye (4) to hide it. Click the eye again to show a field. You can also hide a field by hovering over its header in the preview, and clicking the eye.

Note: If you hide a field in the navigator, by default it is not included when the recipe runs. However, you can choose to include the field 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. You can edit formula fields and field transformations inline.
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.

Action	Description
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, Wave 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 Dataset Recipe

If you're not ready to create the target dataset, save the recipe and come back to it later. Saving simply 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 the name you provided when you started. The recipe does not run. Use this option to save your progress.

Save As New...

Save a copy of your recipe with the same name you provided when you started. 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.

(1) Important: A recipe you create with the **Save As New** option has the same recipe name and target dataset name as the original recipe. We recommend that you run either the original recipe or the new recipe, but not both, to avoid creating target datasets with the same name.

Open a saved recipe on the Prepare tab of the data manager to update it or run it. See Manage and Monitor Dataset Recipes.

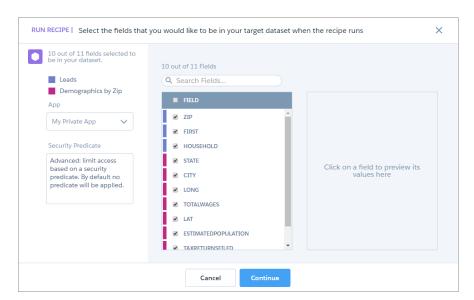
Run a Dataset Recipe

Run a new recipe to create a target dataset, or run an existing recipe to update a target dataset.

When you run a recipe, you select an app for the target dataset, enter a security predicate if necessary, and select the fields to include in your target dataset. You can run a recipe once only, or set it to run on a schedule.

Note: A recipe run counts toward your limit of 24 dataflow runs per day. For example, in an org using the Sales Wave app, a user dataflow, and two recipes, there are four separate dataflows. If you schedule each one to run every six hours, you reach the daily limit. Consider this limit when you run or schedule a recipe.

1. On the dataset recipe page, click **Create Dataset**. The Run Recipe dialog appears.



EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

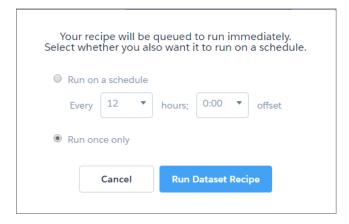
USER PERMISSIONS

To create a recipe:

 "Edit Wave Analytics Dataflows"

- 2. From the App picklist, select where you want to create the target dataset.
- **3.** If required, enter a security predicate for the target dataset. See Row-Level Security for Datasets.
- **4.** In the list of fields, make sure the fields that you require are selected.
 - Note: By default, fields that you have removed or hidden in the recipe preview are not selected. Make sure that you review these fields.
- 5. Click Continue.

The schedule dialog appears.



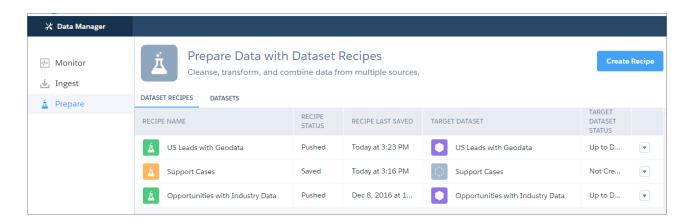
- **6.** Either select to run on a schedule and enter a frequency, or select to run once only.
- **7.** Click **Run Dataset Recipe**. Your recipe is queued to run.

Manage and Monitor Dataset Recipes

View the status of your recipes and target datasets on the Prepare tab of the data manager. You can open recipes to update or run them, or delete them.

To open the data manager in Wave Analytics, click the gear icon (🔯) and then click **Data Manager**.

The Prepare tab in the data manager displays your recipes and their associated target datasets.



A recipe's icon and status indicate its current state.

Icon	Status	Explanation
Á	Pushed	New or updated recipe that has been run.
Á	Saved	New or updated recipe that has been saved but not run.

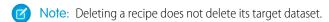
A target dataset's icon and status indicate its current state.

Icon	Status	Explanation
	Up to Date	The dataset has been created and is up to date.
	Not Up to Date	The dataset has been created, but is not currently up to date. This is because its recipe has been updated but not run, or is queued to run.
	Not Created	The dataset has not been created. This is because its recipe has been created but never run, or is queued to be run.

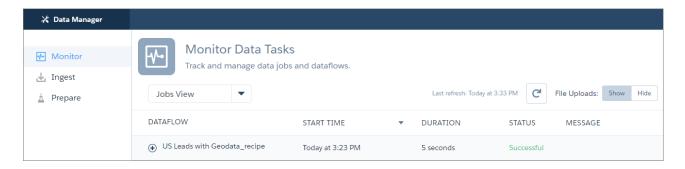
Open a recipe to update it or run it. Either click the recipe or open the actions menu and select **Open Recipe**.



You can also delete a recipe from here.



Click **Monitor** to view the progress of recipes that you have run. Recipes show as separate jobs in the Jobs View of the monitor.



Row-Level Security for Datasets

If a Wave Analytics user has access to a dataset, they have access to all records in the dataset, by default. However, you can implement row-level security on a dataset to restrict access to records. Some records might contain sensitive data that shouldn't be accessible by everyone.

To implement row-level security, define a predicate for each dataset on which you want to restrict access to records. A *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, Wave Analytics checks the predicate to determine which records the user has access to. If the user doesn't have access to a record, Wave Analytics does not return that record.

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.

IN THIS SECTION:

Considerations when Defining a Predicate for a Dataset

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.

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

SEE ALSO:

sfdcRegister Transformation sfdcRegister Parameters

Considerations when Defining a Predicate for a Dataset

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.

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, Wave Analytics 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. Wave Analytics 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.



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.

IN THIS SECTION:

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

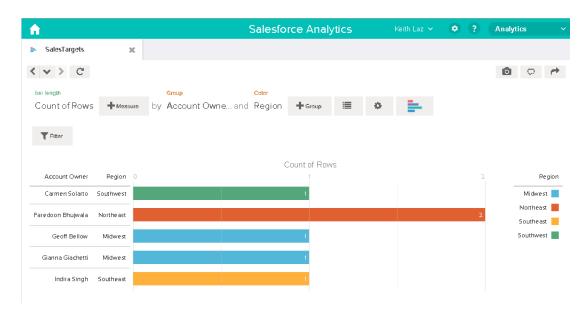
First, determine what data you want to include in the dataset. For this example, you will create a Targets dataset that contains all sales targets.

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

AccountOwner	Region	Target	TargetDate
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, Wave Analytics 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. Wave Analytics 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",
         "fullyOualifiedName": "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
      }
      ]
      }
```

1 }

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 Wave Analytics, go to the home page.
- 2. Click Create > Dataset
- 3. Click CSV.

The following screen appears.

EDITIONS

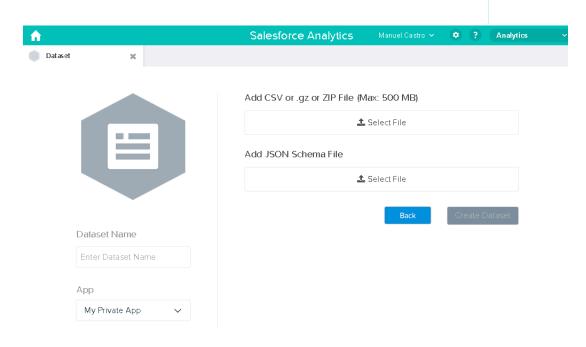
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To upload a CSV and metadata file:

"Upload External Data to Wave Analytics"



- 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.

Wave Analytics 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 (o) 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.

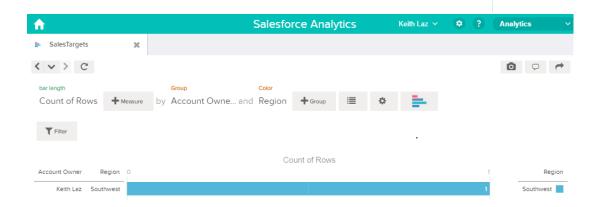
- 1. Log in to Wave Analytics as Keith.
- **2.** Open the SalesTargets dataset.

 As shown in the following lens, notice that Keith can see only his sales target.



Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.



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.

IN THIS SECTION:

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

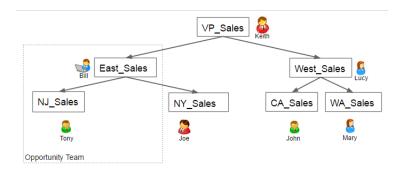
You must verify that the predicate is applied properly and that each user can see the appropriate opportunities.

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.

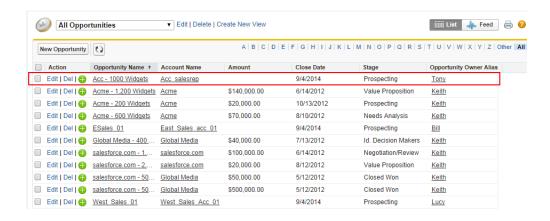


EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

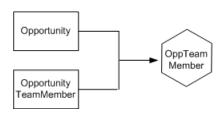
Your organization also contains the following opportunities, most of which are owned by Keith.



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.



EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

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.

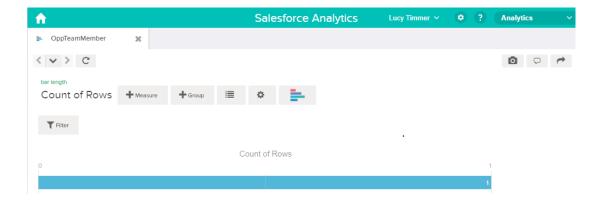
Based on the design, you create the JSON shown below.

```
"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, Wave Analytics 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.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

```
'UserId' == "$User.Id"
```

This predicate compares the Userld column in the dataset against the ID of the user running a query against the dataset. The Userld 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, Wave Analytics looks up the ID of the user making the query in the User object.

For each match, Wave Analytics 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.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

```
"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.

 In Wave Analytics, 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.

EDITIONS

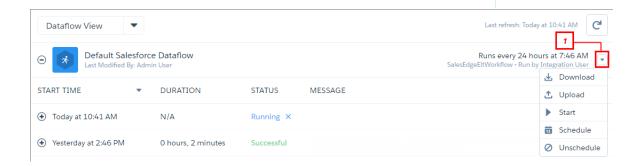
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To download, upload, run, and monitor a dataflow:

 "Edit Wave Analytics Dataflows"



- **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.
- 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

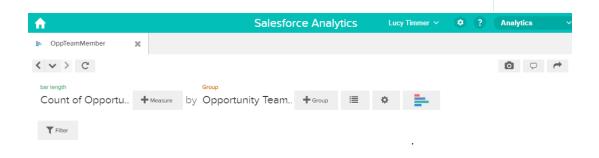
You must verify that the predicate is applied properly and that each user can see the appropriate opportunities.

- 1. Log in to Wave Analytics 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.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

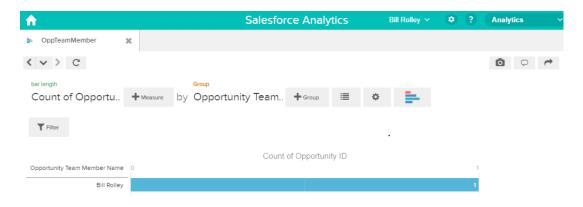
Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.



No results meet those criteria. Try changing your filters.

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.

IN THIS SECTION:

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.

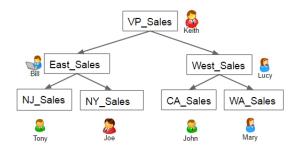
You will obtain opportunities from the Opportunity object and user details from the User object. Both are objects in Salesforce.

In this example, your Salesforce organization has the following role hierarchy and users.

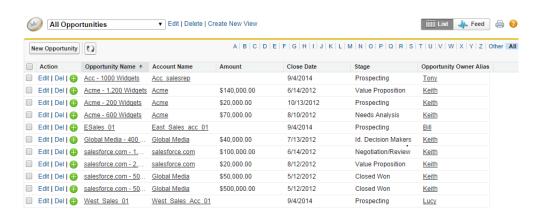
EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

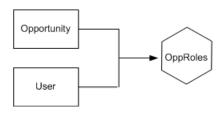


Also, your organization contains the following opportunities, most of which are owned by Keith.



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.

Available for an extra cost in Enterprise, Performance, and Unlimited Editions, Also

Available in Salesforce Classic and Lightning

EDITIONS

Experience.

and **Unlimited** Editions. Also available in **Developer** Edition.

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.

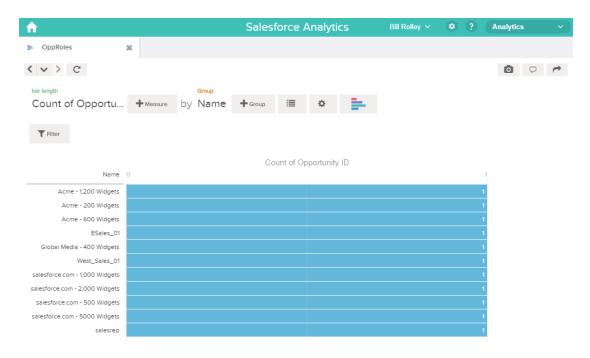
Based on the design, you create the JSON shown below.

```
{
   "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, Wave Analytics 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.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

```
'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, Wave Analytics returns an opportunity record if:

• The user who submits the query is a parent of the opportunity owner based on the Salesforce role hierarchy. Wave Analytics 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. Wave Analytics looks up the user role ID of the user who submits the query from the User object.
'Ownerld' == "\$User.ld"	Ownerld 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. Wave Analytics 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.

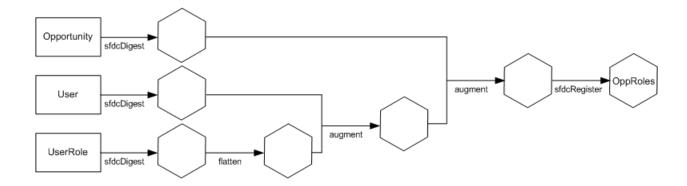
In this scenario, you have to make changes to the dataflow based on the predicate.

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:

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.



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" }
      1
   }
},
"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"
      1
```

```
}
  },
  "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 Ownerld. 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.

 In Wave Analytics, 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.

EDITIONS

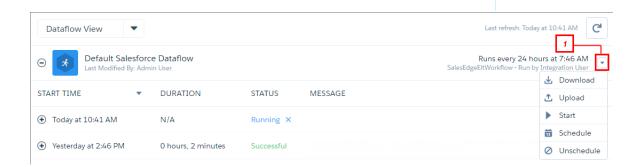
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To download, upload, run, and monitor a dataflow:

 "Edit Wave Analytics Dataflows"



- **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 Wave Analytics 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.

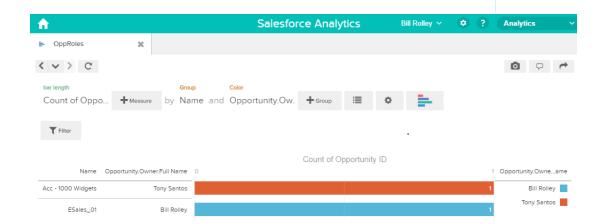


Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

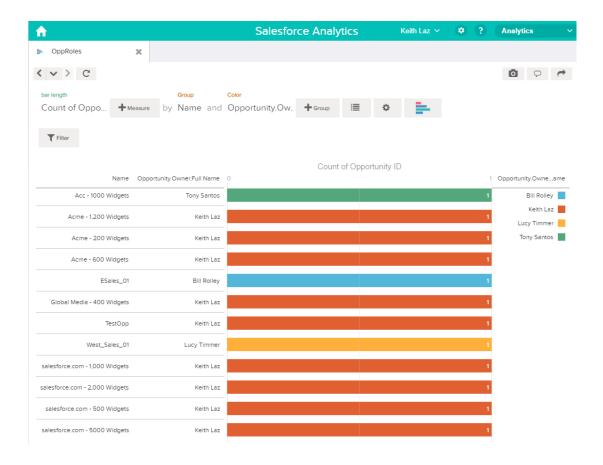


- ""
- ""



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 1,000 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.

IN THIS SECTION:

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

Wave Analytics supports UTF-8 characters in dataset column names and values in a predicate expression. Wave Analytics replaces

non-UTF-8 characters with the UTF-8 symbol (). If Wave Analytics 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 Wave Analytics.

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

Value Type	Requirements	Predicate Expression Examples
field value	When referencing a field from the User object, use the \$User.[field] syntax. Use the API name for the field.	<pre>'Owner.Role' == "\$User.UserRoleId" 'GroupID' ==</pre>
	You can specify standard or custom fields of type string, number, or multi-value picklist.	"\$User.UserGroupIdc" Note: Supported User object field
	When you define a predicate for a dataset, you must have read access on all User object fields used to create the predicate expression.	value types are string, number, and multi-value picklist. Other types (for example, boolean) are not supported.
	However, when a user queries a dataset that has a predicate based on the User object, Wave Analytics 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.	

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

Sequence	Meaning
\\	One backslash character
\0	One ASCII null character

Character Set Support

Wave Analytics supports UTF-8 characters in dataset column names and values in a predicate expression. Wave Analytics replaces non-UTF-8 characters with the UTF-8 symbol (). If Wave Analytics 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 Wave Analytics.

Character	Name	Description
,	Single quote	Encloses a dataset column name in a predicate expression.
		Example predicate expression:
		<pre>'Expected_Revenue' >= 2000.00</pre>
и	Double quote	Encloses a string value or field value in a predicate expression.
		<pre>Example predicate expression: 'OpportunityOwner' == "Michael Vesti"</pre>
()	Parentheses	Enforces the order in which to evaluate a predicate expression.
		Example predicate expression:
		<pre>('Expected_Revenue' > 4000 'Stage Name' == "Closed Won") && 'isDeleted' != "False"</pre>
\$	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"

Character	Name	Description
	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.

IN THIS SECTION:

Comparison Operators

Comparison operators return true or false.

Logical Operators

Logical operators return true or false.

Comparison Operators

Comparison operators return true or false.

Wave Analytics 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
<=	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).

Operator Name	Description
	Example predicate expression:
	'Demog' in ["\$User.Demographicc"]
	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.

Wave Analytics 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
ОррА	2000.00	Bill	20	Prospecting	True
ОррВ	3000.00	Joe	22	Closed Won	False
ОррС	1000.00	可爱的花	36	Closed Won	False
OppD	5000.00	O'Fallon	18	Prospecting	True
ОррЕ		Joe	22	Closed Won	True

Analytics Datasets

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.
<pre>'Expected_Rev' > 1000 && 'Expected_Rev' <= 3000</pre>	
'Owner' = "Joe" 'Owner' = "Bill"	
<pre>('Expected_Rev' > 4000 'Stage Name' == "Closed Won") && 'isDeleted' != "False"</pre>	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.

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 do not reach the limit (250 million) for the maximum number of rows in all registered datasets, deleted unused datasets.

SEE ALSO:

Create a Dataset with External Data Edit a Dataset Delete a Dataset

Data Monitor

The data monitor provides options for you to manage jobs.

SEE ALSO:

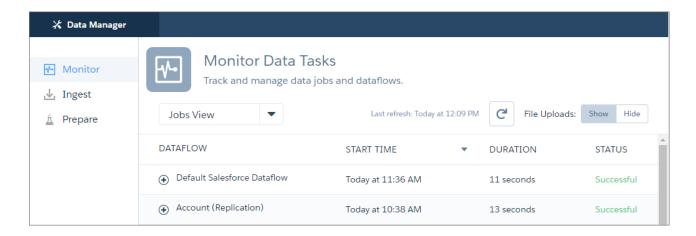
Start and Stop a Dataflow Reschedule and Unschedule a Dataflow Monitor a Dataflow Job Monitor an External Data Upload

Data Manager

The data manager is where you monitor all your jobs and dataflows, view and edit your replications, and prepare your data with recipes.

Analytics External Data

To open the data manager in Wave Analytics, click the gear icon (and then click **Data Manager**.



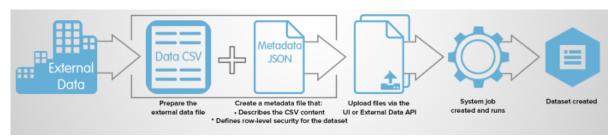
You land on the Monitor tab, where you can see the progress of your recipe, replication, and other dataflow jobs. If you have replication enabled, the Ingest tab is where you can view and update settings for your replication-enabled Salesforce objects. Go to the Prepare tab to create and manage your dataset recipes.

SEE ALSO:

Optimize Your Dataflows with Replication Prepare Datasets with Recipes Data Monitor

External Data

You can integrate external data into Wave Analytics to make the data available for queries from explorer and designer.



The External Data API enables you to upload external data files to Wave Analytics. 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.

Explore and Visualize Your Data

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 Wave Analytics 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 Wave Analytics do the work of producing dynamic charts that look sharp and communicate volumes.

IN THIS SECTION:

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 Wave Analytics

Wave collects and organizes your data in datasets, lenses, dashboards, and apps. These collections, also known as "Wave 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.

Explore Your Data

See what unexpected insights you can surface by interactively exploring and visualizing your data, using explorer tools.

Clone a Visualization

Build upon a visualization by cloning it to a new tab where you can continue exploring while saving 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 Wave Analytics. 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.

Export a Visualization as an Image

Export your current visualization so that you can save it as an image file.

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 Wave Analytics

Wave collects and organizes your data in datasets, lenses, dashboards, and apps. These collections, also known as "Wave 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 Wave Analytics retrieves information from the source data.

A *measure* is a quantitative value, 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 Wave Analytics, 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.
- 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 Wave Analytics returns an incorrect result.

A dimension is a qualitative value, like region, product name, and model number. 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.

Explore Your Data

See what unexpected insights you can surface by interactively exploring and visualizing your data, using explorer tools.

IN THIS SECTION:

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. For example, to look for seasonal patterns in your sales cycle, group opportunity data by the month when deals close.

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 unit, project, or management team. For example, limit your visualization to certain fiscal years.

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 two visualizations side by side.

Change the Chart Scale

Change the scale of a chart, such as applying a logarithmic scale. The options that are available depend on the chart type.

Navigate Compare Table Columns

Create new columns and navigate between existing ones using clicks or keystrokes.

Create Calculated Columns Using the Compare Table

Use the Compare Table to define custom formulas with columns of data. View measures side by side, and perform math across the table's columns and rows.

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 view visualizations:

"Use Wave Analytics"

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.

Return to the Initial View

Exploring your data can lead you down many paths. Not all of them are fruitful, and that's OK. To go back to the dashboard's original state, or just see those nifty animations again, you can use the reset option to quickly return to the initial view.

Apply Groups and Filters by Copying and Pasting Using Your Browser's Clipboard

Sometimes you want to go multiple directions when exploring a dataset, such as trying different groupings or focusing on different angles. Instead of exploring linearly in a lens and using the history tool to backtrack and then branch off in a new direction, you can explore multiple directions at the same time by opening the same dataset in multiple Wave Analytics tabs. Then use explorer's copy and paste tools to replicate specific groupings and filters from one tab to another.

View the Query Behind Your Lens

Wave Analytics uses Salesforce Analytics Query Language (SAQL) behind the scenes in lenses and dashboards to gather data for visualizations. Open the SAQL viewer to watch the underlying SAQL query being written as you explore or to 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, Wave adds the query as a step to the most recently used dashboard. If a dashboard isn't open, Wave adds the step to a new dashboard in Wave dashboard designer. After you add the step 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. For example, this lens opens as a bar chart that shows the number of rows in an opportunity dataset.

EDITIONS

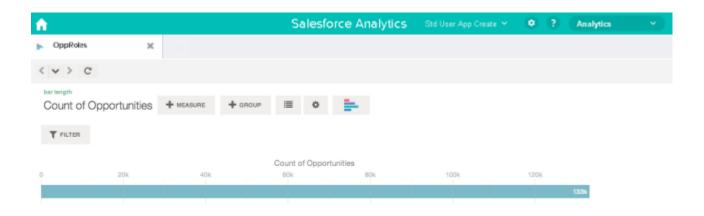
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To view visualizations:

"Use Wave Analytics"



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. Click the **Group** button.



2. Using the search box, type the first few letters of the dimension that you want to group by, such as Close Date.

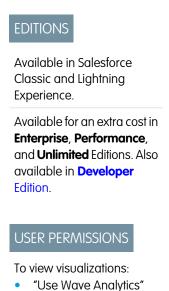
The dimensions in your data determine which categories you can group by.



3. Hover over the Close Date line. Click the **Scale** button that appears at the right.



4. In the Calendar Date Scale list, click Month.





The bar chart changes from a single bar to multiple bars that show the data grouped separately by the closed year and month.



- 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.
- **5.** To add a secondary grouping, also called a *stacked grouping*, click the **Group** button. Using the search box, find the StageName dimension, and then select it.

At this point, we're just adding a group. If you had clicked the **Filter** button, a group and a filter would have been added.



The visualization changes to show the data grouped by both dimensions. The entries in the last subgroup are assigned a color for each category, and the color legend is displayed on the right. You can have a maximum of four groupings in a lens.



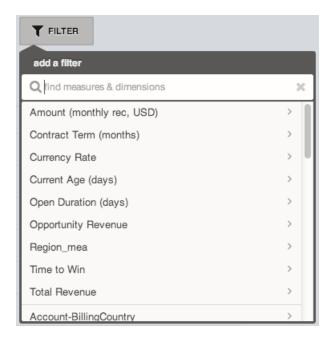
Filter Your Data

Apply filters in a lens to unclutter your chart and focus on the subset of data that's most relevant for your business unit, project, or management team. For example, limit your visualization to certain fiscal years.

1. Click Filter.



In the Add a Filter dialog box, measures are listed at the top, followed by dimensions.



2. Select the FiscalYear dimension. The values for FiscalYear appear.

EDITIONS

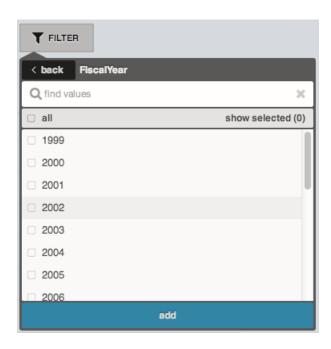
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To view visualizations:

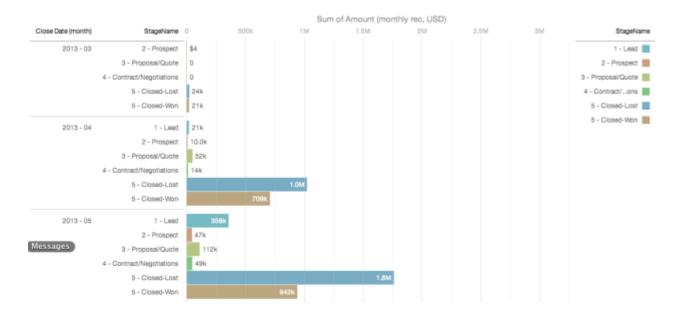
"Use Wave Analytics"



3. Select 2012 and 2013, and then click **Add**. If you use the search box to find values to filter by, be aware that this search is case-sensitive. The filter that you've applied is displayed next to the **Filter** button.



The chart visualization changes to show data for fiscal years 2012 and 2013.



Change the Sort Order

Choose whether data in your chart visualization is sorted ascending, sorted descending, or unsorted.

1. Click the **Options** button.



2. In the dialog box, choose **Asc** (sorted ascending), **Dsc** (sorted descending), or **Off** (unsorted).

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 view visualizations:

"Use Wave Analytics"

Change the Chart Type

Switch between chart types to see which visualization options tell the most compelling story for your data.

1. Click the **Change Chart Type** button.



A gallery of chart types that are available for this lens is displayed.



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

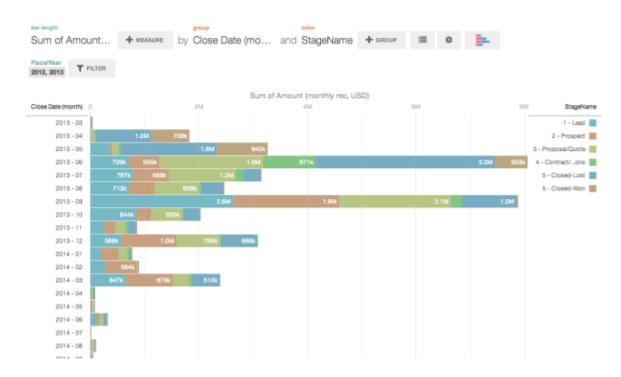
To view visualizations:

"Use Wave Analytics"

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 two visualizations side by side.

1. Click the aggregation **Sum**, and then click the Amount (monthly rec, USD) measure. The visualization changes to show the Sum of Amount.

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 view visualizations:

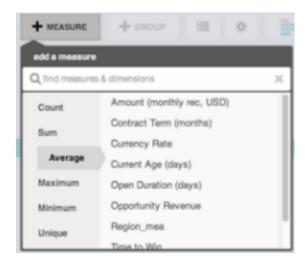
"Use Wave Analytics"



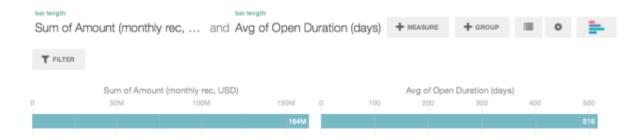
2. To add a second measure, click the **Measure** button.



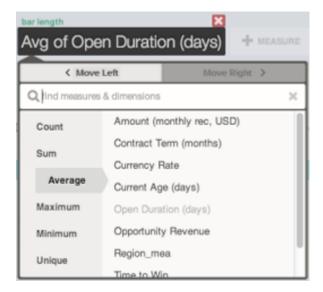
3. Click the aggregation **Average**, and then click the Open Duration (days) measure.



The new measure is displayed on the right.



4. To move the new measure to the left, click Avg of Open Duration (days), and then click **< Move Left**.



The two visualizations swap positions.



5. To add additional measures, click the **Measure** button again.

You can have a maximum of four measures in a lens.

Change the Chart Scale

Change the scale of a chart, such as applying a logarithmic scale. The options that are available depend on the chart type.

1. Click the **Options** button.



2. In the dialog box, choose a scale.

Navigate Compare Table Columns

Create new 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.

EDITIONS

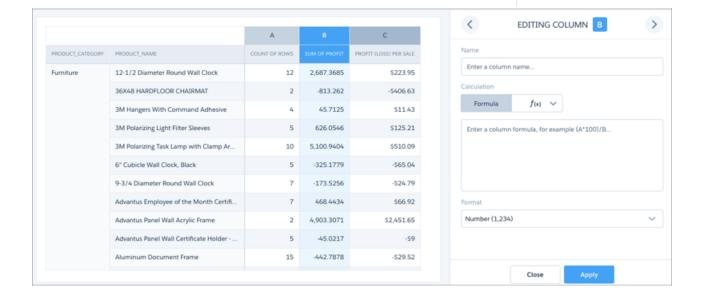
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

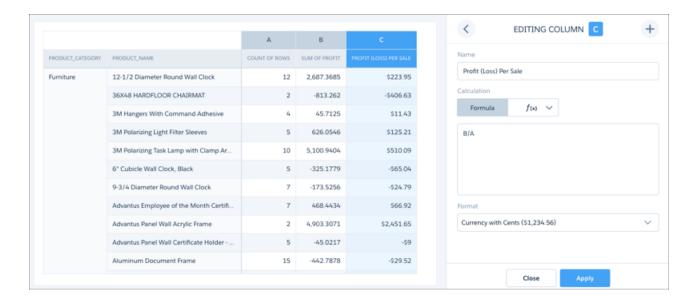
USER PERMISSIONS

To view visualizations:

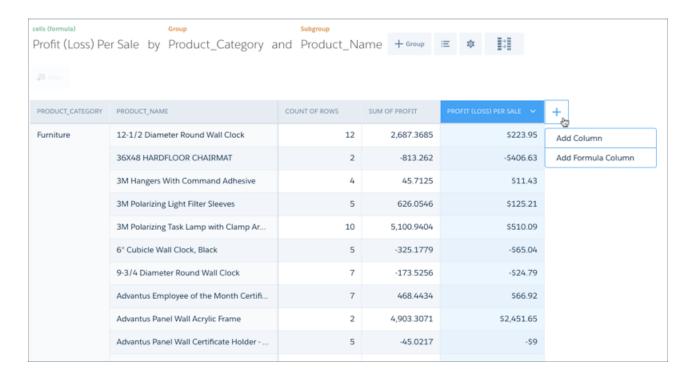
"Use Wave Analytics"



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 new formula column.



You can also add columns directly from the compare table using the new add-column menu.



Choose Add Formula Column to add a column and open the column editor.

Use the new **Close** button to exit the column editor without applying changes to the column.

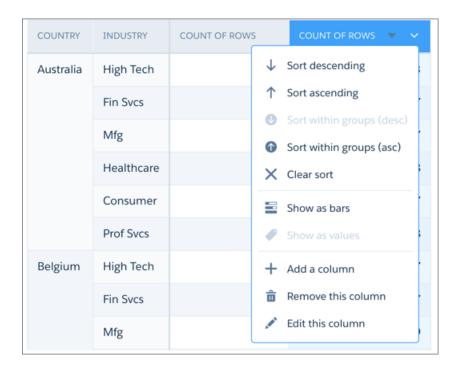
Create Calculated Columns Using the Compare Table

Use the Compare Table to define custom formulas with columns of data. View measures side by side, and perform math across the table's columns and rows.

1. Select **Compare Table** from the chart type gallery.



- 2. Add dimensions by clicking **Group**.
 - Note: Null values that occur in the data are displayed as minus signs ("-").
- **3.** Change the first measure by clicking the **cells** selection ("Count of Rows"). Add more measures by clicking the arrow in the measure column header and then selecting **Add a column**.



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 view visualizations:

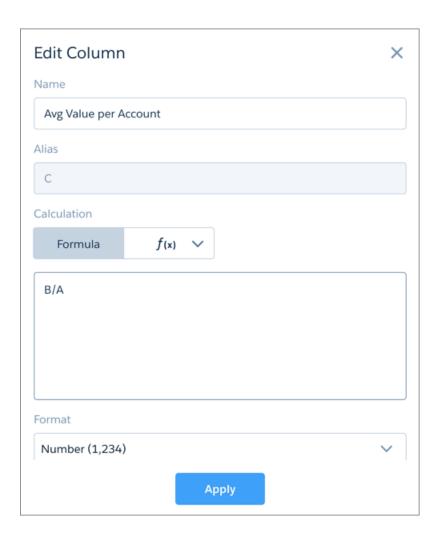
"Use Wave Analytics"

With the new column selected, click the **cells** selection to change the measure. You can also apply a filter by clicking the header of a measure column to select it and then clicking **Filter**.

INDUSTRY	CLOSE DATE (YEAR)	CLOSE DATE (MONTH)	COUNT OF ROWS A	CHANGE FROM PREVIOUS	RANK V
Consumer	2015	11	1,737	9.66%	6
		10	1,584	0%	4
		12	1,566	-9.84%	3
	2016	01	1,725	10.15%	5
		02	1,440	-16.52%	2
		03	579	-59.79%	1
Fin Svcs	2015	12	6,423	4.08%	6
		11	6,171	20.29%	4
		10	5,130	0%	2
	2016	01	6,306	-1.82%	5

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.

4. Add a measure column to hold a custom formula, and then select **Edit this column** from the drop-down menu.



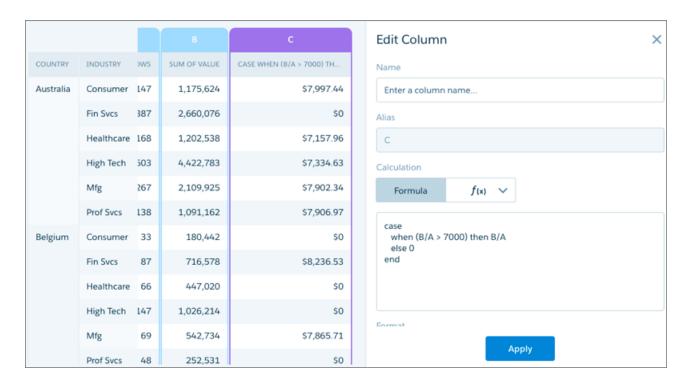
In the **Formula** field, create formulas using operators from this table.

Note: Measure columns are identified by uppercase letters from left to right.

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.

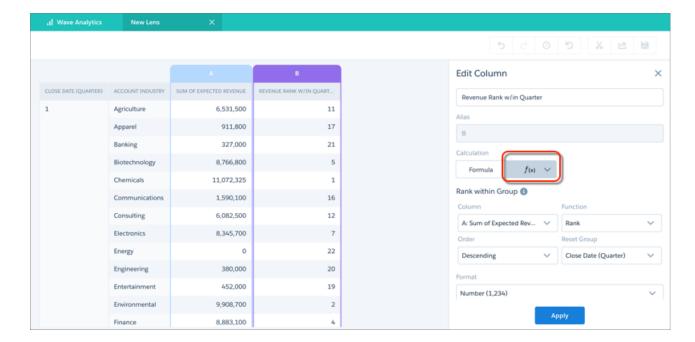
In the column editor, you can enter SAQL formulas, including math functions, or choose a windowing function to calculate the values in the column. You can also name the column and choose a number format. If any of the formulas contain division by zero, the result is displayed as a minus sign ("-").

5. To use SAQL, enter it directly into the formula editor, then click Apply to see the result.

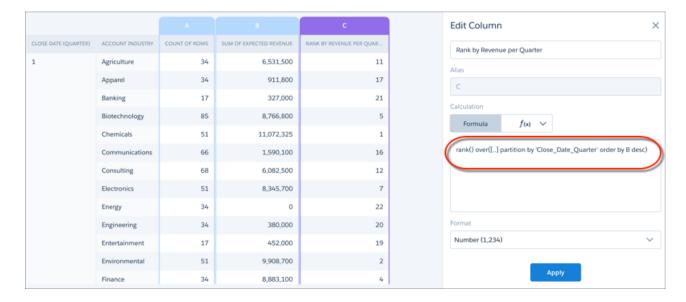


See the Wave Analytics SAQL Reference for more information on writing SAQL.

- Note: The percentile_desc and percentile_cont functions can't be used in windowing functions entered in the formula editor.
- **6.** 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.



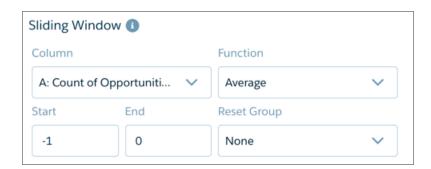
Switch to the formula editor to see the SAQL generated by the chosen function. You can edit the SAQL in the formula editor after generating it with a function just as if you'd written it.



Window Functions Available in the Compare Table

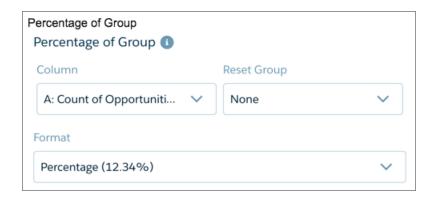
The compare table's function menu provides UI-based access to useful window functions, saving you the time of having to write them in SAQL. Once you choose a function, you can edit the generated SAQL in the formula editor.

Sliding Window



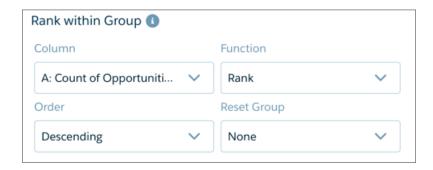
Applies an aggregate function to the current row with respect to a configurable range of rows. The function can be reset on a grouping defined in the table. Use the sliding window for calculating values such as running totals and moving average.

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.

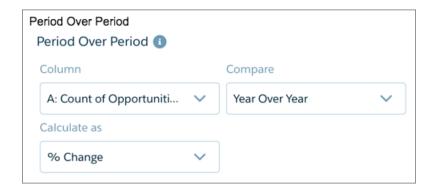
Rank Within Group



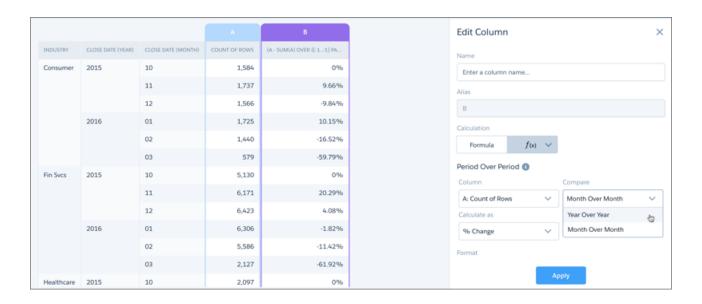
Provides rank, dense rank, cumulative distribution, and row-number functions. Ranking can be reset on a grouping defined in the table.

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



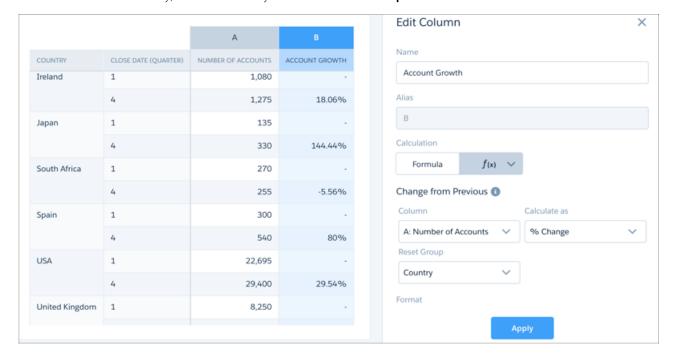
Compares periods of time to calculate changes in values. For example, year-over-year sales, or quarter-over-quarter closed opportunities, can be calculated by choosing the column and setting the granularity of the time period.



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.

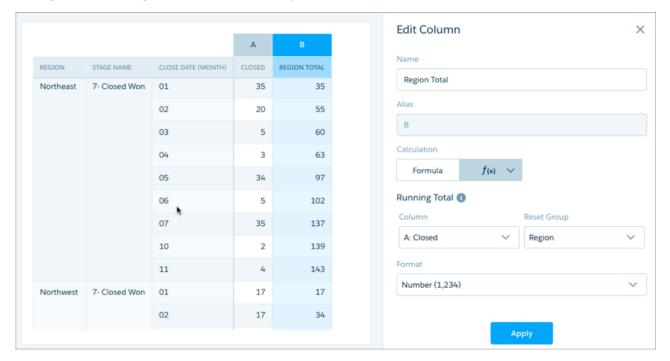
Change from Previous

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**.



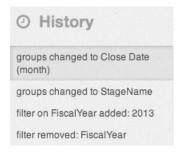
Return to the Initial View

Exploring your data can lead you down many paths. Not all of them are fruitful, and that's OK. To go back to the dashboard's original state, or just see those nifty animations again, you can use the reset option to quickly return to the initial view.

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 go to that point. Or use the buttons to **Step Backward** and **Step Forward** through the list.



To view visualizations:

"Use Wave Analytics"

Apply Groups and Filters by Copying and Pasting Using Your Browser's Clipboard

Sometimes you want to go multiple directions when exploring a dataset, such as trying different groupings or focusing on different angles. Instead of exploring linearly in a lens and using the history tool to backtrack and then branch off in a new direction, you can explore multiple directions at the same time by opening the same dataset in multiple Wave Analytics tabs. Then use explorer's copy and paste tools to replicate specific groupings and filters from one tab to another.

From the home page, each time you click a dataset, it opens in a new tab. The copy and paste tools in explorer are designed to be used when you have the same dataset in multiple tabs.

- 1. If the chart type that you're viewing allows you to select a subset of data, make your selection by clicking or by clicking and dragging within the chart.

 Everything except your selected data is unavailable.
- 2. Click the Actions button.



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 view visualizations:

- "Use Wave Analytics"
- **3.** Copy the selection to your browser's clipboard. Click **Copy** to capture the selected items in the clipboard. You can decide later whether to paste your selection as a group or as a filter. Click **Focus** to both capture the selected items in the clipboard and apply them as a filter to the current chart.
- **4.** Switch to a different Wave Analytics tab that has the same dataset.
- **5.** Click the **Actions** button.



6. Apply the selection from the clipboard to the current chart. Click **Paste as Group** to make the data display grouped by the selected dimensions. Click **Paste as Filter** to make the data display with your selection applied as a filter.

To restore a chart to the full dataset, remove your selection by clicking



and then Clear

View the Query Behind Your Lens

Wave Analytics uses Salesforce Analytics Query Language (SAQL) behind the scenes in lenses and dashboards to gather data for visualizations. Open the SAQL viewer to watch the underlying SAQL query being written as you explore or to copy it for use elsewhere.

1. Open the Options menu () to access the **Show SAQL** control.



2. Select On.

The SAQL Query pane opens. Continue exploring and see your work expressed in SAQL.

EDITIONS

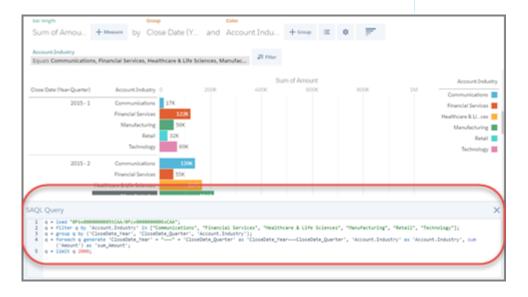
Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

USER PERMISSIONS

To view visualizations:

"Use Wave Analytics"



For more information about SAQL, see the Wave Analytics SAQL Reference.

Clip a Lens to a Dashboard

To include the lens in a dashboard, clip it. When you clip the lens, Wave adds the query as a step to the most recently used dashboard. If a dashboard isn't open, Wave adds the step to a new dashboard in Wave dashboard designer. After you add the step 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 🔏 .

Wave creates a step based on query of the lens, and adds it to the step panel of the dashboard designer.

3. To save the step, save the dashboard.



Note: Changes to the step properties don't affect the lens.

Clone a Visualization

Build upon a visualization by cloning it to a new tab where you can continue exploring while saving the original in its own tab.

1. Click the Clone in New Tab icon.



2. Continue exploring in the new tab, building upon the visualization. When you reach the next view that you want to keep, click **Save** and then **Clone in New Tab** again.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

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 view visualizations:

"Use Wave Analytics"

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.

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 view visualizations:

"Use Wave Analytics"

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 Wave Analytics. 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
 - b. Get URL provides a unique URL to the asset. Copy the link and then paste it wherever you want to share it.

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 view visualizations:

"Use Wave Analytics"

- 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 Wave Analytics 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.

Export a Visualization as an Image

Export your current visualization so that you can save it as an image file.

1. Click the Actions button.



2. Click Export as Image.

A new browser window opens with the visualization in a .png image file.

- **3.** Right-click the image, and then choose **Save Image As...**.
- 4. Type a file name, and then click Save.

Visualizing 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. Wave provides a chart for every need, each a means for illustrating key aspects of your business in just the right way.

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 view visualizations:

"Use Wave Analytics"

Wave charts are integrated with the Wave tools you already use to explore data. Start with a dataset, explore with a lens, clip it to Wave dashboard designer, and create a stunning dashboard.

IN THIS SECTION:

Bubble Map Charts

Use a bubble map chart to visually indicate measure values in specified geographical areas. The bubble-map chart is found in the Wave dashboard designer.

Combo Charts

Use a combo chart to dispaly two related series of data, such as discrete grouped values as vertical bars overlayed with a line chart representing an average value. The combo chart is found in the Wave dashboard designer and classic designer.

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. The donut chart is found in the Wave dashboard designer, classic designer, and explorer.

Funnel Charts

Use a funnel chart to visualize sequential data that can be broken up into stages, such as a sales cycle. The funnel chart is found in the Wave dashboard designer and classic designer.

Gauge Charts

Use gauge charts to track progress along a single measure, such as how much revenue has been realized versus the target. The gauge chart is found in the Wave dashboard designer.

Horizontal Bar Charts

Use a horizontal bar chart to provide a quick visual comparison of related values. The horizontal bar chart is found in the Wave dashboard designer, classic designer, and explorer.

Heat Maps

Use a heatmap to visually enhance high and low-value data when there's a single measure and multiple dimensions. The heatmap chart is found in the Wave dashboard designer, classic designer, and explorer.

Map Charts

Use a map if you have data with a geographical component that can shade areas in proportion to mapped values, allowing for visual pop-out of high-value areas. The map chart is found in the Wave dashboard designer and classic designer.

Custom Map Charts

Using map charts that can shade areas in proportion to mapped values allows for visual pop-out of high-value areas. You can use custom maps with geographical units or to illustrate any type of area that can be represented by polygons. Custom map charts can be used in the Wave dashboard designer and classic designer.

Matrix Charts

Use matrix charts (new in Wave dashboard designer) to create a table that represents measures visually, allowing you to quickly spot extreme values. The matrix chart is found in the Wave dashboard designer, classic designer, and explorer.

Pyramid Charts

Use a pyramid chart to visually highlight relative sizes of stacked dimension values. The pyramid chart is found in the Wave dashboard designer.

Radar Charts

Use radar charts to display a small dataset with one dimension and at least three measure columns. The radar chart is found in the Wave dashboard designer.

Rating Charts

Use a rating chart to get a sense of how well a measured quality, such as customer satisfaction, is meeting expectations. The rating chart is found in the Wave dashboard designer.

Scatter Charts

Use a scatter chart to visualize correlation between two groups of data. The scatter chart is found in the Wave dashboard designer, classic designer, and explorer.

Timeline Charts

Use a timeline chart to show how a value changes over time. The timeline chart is found in the Wave dashboard designer, classic designer, and explorer.

Treemap Charts

Use treemaps to visualize hierarchical quantitative data, where containing rectangles designate relationship using color, and "leaf" rectangles represent quantity using area. The treemap chart is found in the Wave dashboard designer.

Vertical Bar Charts

Use a vertical bar chart (also called column chart) to show relative counts of things, such as leads or dollars. The vertical bar chart is found in the Wave dashboard designer, classic designer, and explorer.

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. The waterfall chart is found in the Wave dashboard designer and classic designer.

Bubble Map Charts

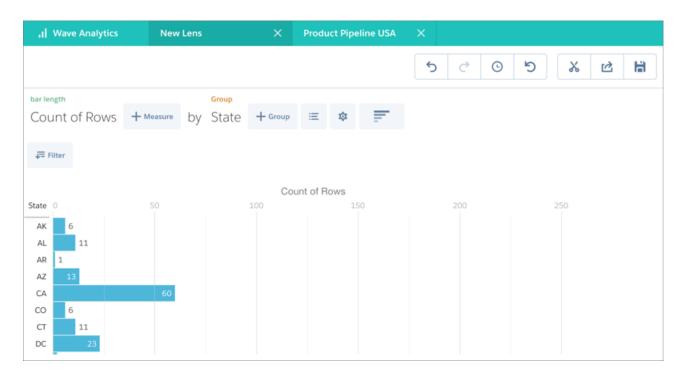
Use a bubble map chart to visually indicate measure values in specified geographical areas. The bubble-map chart is found in the Wave dashboard designer.

Create a Bubble Map Chart

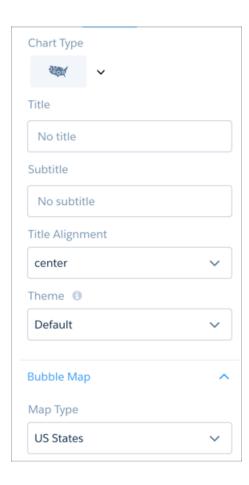
Wave bubble-map charts use the same maps available to map charts, giving you another choice representing measure values on the map.

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.

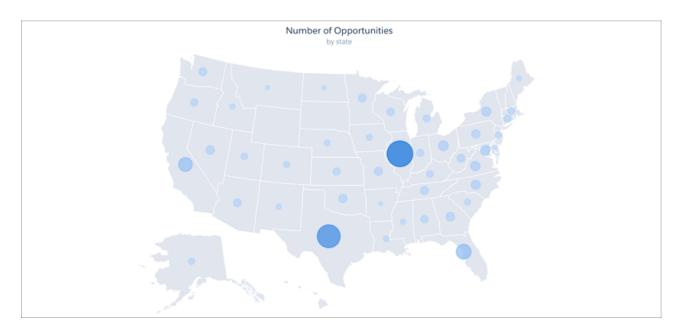
1. In Wave, group the data by state, then change the measure.



- 2. Clip the lens to Wave dashboard designer.
- 3. Choose Bubble Map from Chart Type, then choose US States from Map Type.



4. Enter a title for the chart and make other adjustments as necessary.



Combo Charts

Use a combo chart to dispaly two related series of data, such as discrete grouped values as vertical bars overlayed with a line chart representing an average value. The combo chart is found in the Wave dashboard designer and classic designer.

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. The donut chart is found in the Wave dashboard designer, classic designer, and explorer.

Funnel Charts

Use a funnel chart to visualize sequential data that can be broken up into stages, such as a sales cycle. The funnel chart is found in the Wave dashboard designer and classic designer.

Gauge Charts

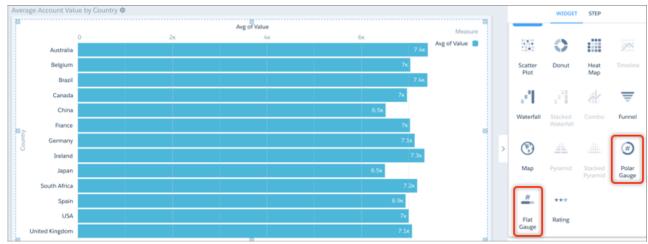
Use gauge charts to track progress along a single measure, such as how much revenue has been realized versus the target. The gauge chart is found in the Wave dashboard designer.

Create a Gauge Chart

If you've set goals for, as an example, average account value in each of the countries where your company operates, 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 the company is in reaching the associated goal. Each gauge chart can display 1 value per country, and highlight value ranges.

1. In Wave Analytics, create a lens using the accounts dataset, then clip it to Wave dashboard designer.



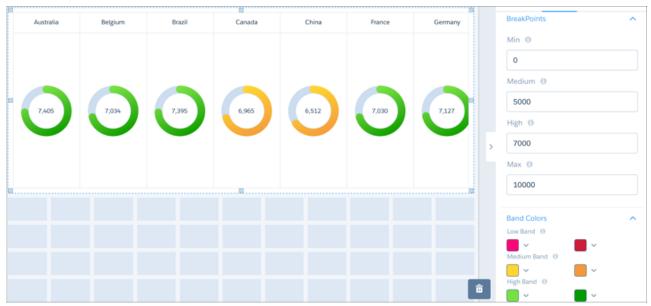
2. Change the chart type to Polar Gauge (round).

A Flat Gauge (linear) is also available.

3. Adjust the size of the center and set the number style.



4. Set the breakpoint values and associated band colors.



The breakpoints set the range values where the color bands transition. The maximum is 10,000, and you'd like to the average account to exceed 7,000 in each country.

5. Set the **Trellis** type to **Wrap** so the charts are easier to view all at once without having to scroll.



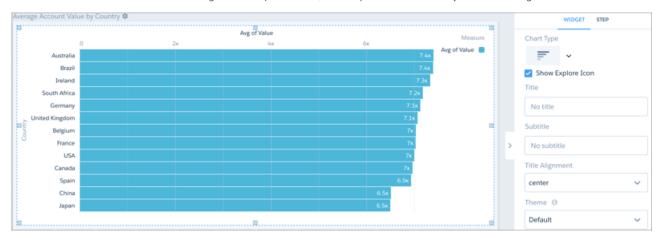
Values above 7,000 are easy to spot due to the color-band configuration of the gauge chart. It's also easy to see which countries are close to the desired value.

Horizontal Bar Charts

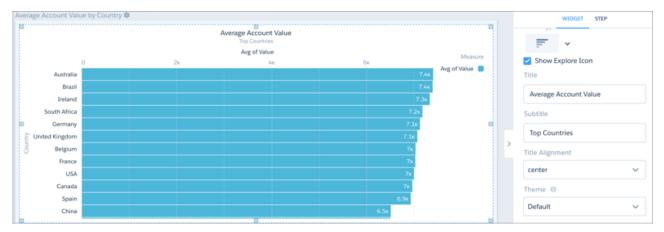
Use a horizontal bar chart to provide a quick visual comparison of related values. The horizontal bar chart is found in the Wave dashboard designer, classic designer, and explorer.

Optimize the Fit of a Horizontal Bar Chart

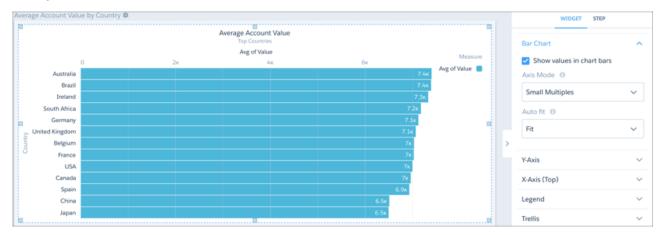
Using the auto-fit feature, you can fit a bar chart into a crowded dashboard without the necessity of long scroll bars or loss of information. You create a lens with a bar chart showing a list of top countries, then clip it to a dashboard you are building.



But after you add a title and subtitle, the lower bars drop out of view.



To bring all the bars back into view, choose Fit from Auto fit.



If you resize the bar chart while arranging the dashboard, you could lose the labels.



To restore the labels, choose Preserve labels from Auto fit.



Heat Maps

Use a heatmap to visually enhance high and low-value data when there's a single measure and multiple dimensions. The heatmap chart is found in the Wave dashboard designer, classic designer, and explorer.

Map Charts

Use a map if you have data with a geographical component that can shade areas in proportion to mapped values, allowing for visual pop-out of high-value areas. The map chart is found in the Wave dashboard designer and classic designer.

Data for map charts must contain geographical names corresponding to the geographical entities represented on the map. Wave 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 ISO 3166-1 (alpha 3). 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".

Properties of Maps

The map chart is configured with the properties in the Wave dashboard designer. These properties have equivalent JSON properties. 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", "South East Asia", "South East Asia", "South East Asia", "South East Asia", "Australasia", "Africa", "East Africa", "Middle Africa", "North Africa", "Southern Africa", "Western Africa", "EMEA", "APAC".

Custom Map Charts

Using map charts that can shade areas in proportion to mapped values allows for visual pop-out of high-value areas. You can use custom maps with geographical units or to illustrate any type of area that can be represented by polygons. Custom map charts can be used in the Wave dashboard designer and classic designer.

Use a custom map if you have data with a geographical component that doesn't match any of the standard maps provided by Wave charts, such as custom regions. Other areas that can be represented with GeoJSON, such as stadium seat sections, city blocks, or floor plans, can also be represented as a map in Wave. Data is overlaid on a custom map using either an id or name property as a key.

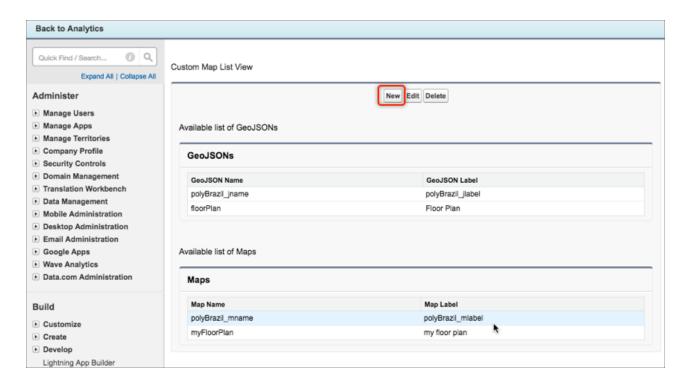
Custom map charts in Wave can be created from GeoJSON with a geometry type of Polygon or MultiPolygon. See the reference section on page 236 for more information.

Create a Custom Map Chart

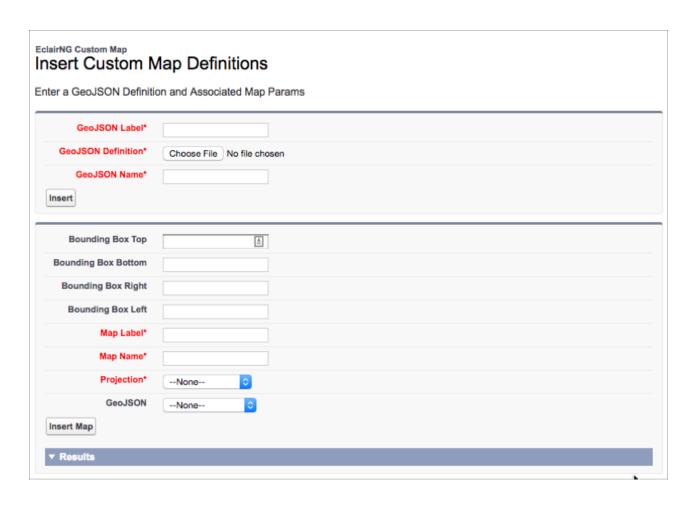
Create a custom map by uploading a GeoJSON file containing the data that defines the map.

While logged in to your Salesforce org, navigate to the URL formed by appending
 /analytics/wave/web/custommap.apexp to your org's base URL.
 For example, if your org's base URL is "https://na30.salesforce.com", go to
 https://na30.salesforce.com/analytics/wave/web/custommap.apexp.

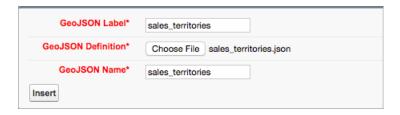
2. Click New.



The insertion window for a new custom map appears. It has 2 panels, one for the GeoJSON and one for the map itself.



3. Fill in the required fields (shown in red) in the GeoJSON section and choose the GeoJSON file to load.

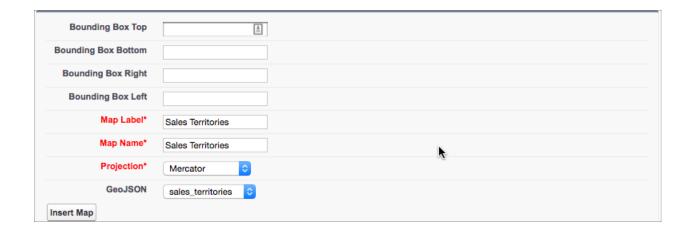


The GeoJSON name and label are used in the custom-map administration screens.

- **4.** Click **Insert** and look for a confirmation message in the Results section.
- **5.** Fill in the required fields (shown in red) in the map section.

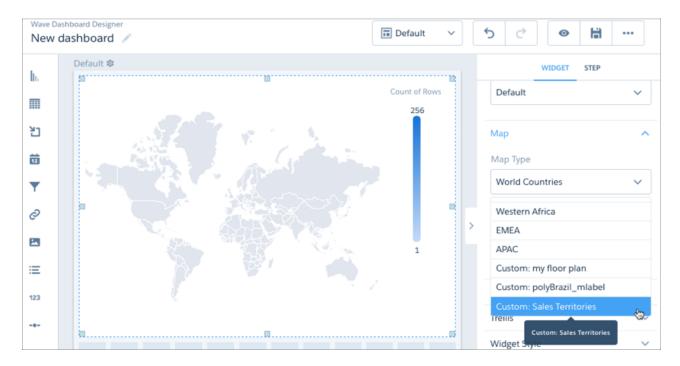
The map name and label are used in Wave.

6. Choose a projection based on the type of geo data you are uploading, and choose the GeoJSON definition you created.



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.

- 7. Click **Insert Map** and look for a confirmation message in the Results section.
- **8.** To see the map, navigate to Wave, choose the map chart, and click **Map Type**.



You must have appropriate step data clipped to the Wave dashboard designer to enable map charts. With trellis off, 1 measure and 1 grouping are required.

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",
        "propertyl": "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 each feature block in the GeoJSON has a property called "id" with values that exactly match the values in one of the columns in the data, then that column must also be named "id". That way "id" can be used as key to match rows of data to specific shapes in the GeoJSON. In the example, "id": "SI" would match a row in the id column that had the value "SI".

Wave has an important requirement for the GeoJSON structure regarding the property that's used as a key. It has to be at the same level as "properties". Often, however, it's a child of "properties":

The original GeoJSON must be edited to move the key property up one level. 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.

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:

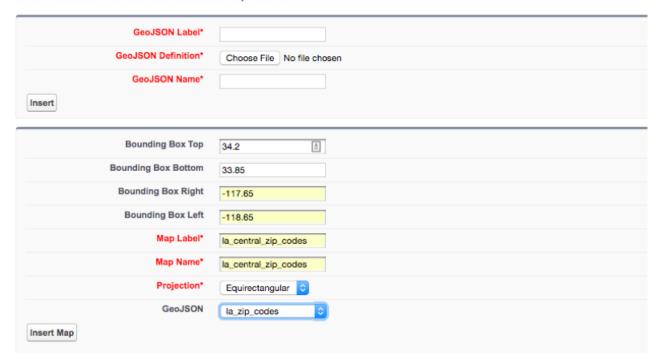


The areas of interest, where data has been overlaid on the map, are indicated by the red rectangle. To make the map more readable, the view should zoom in on the zip-code areas that have data. The coordinates (in this case, latitude and longitude) bordering the area of interest can be found using a GeoJSON 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 that were found:

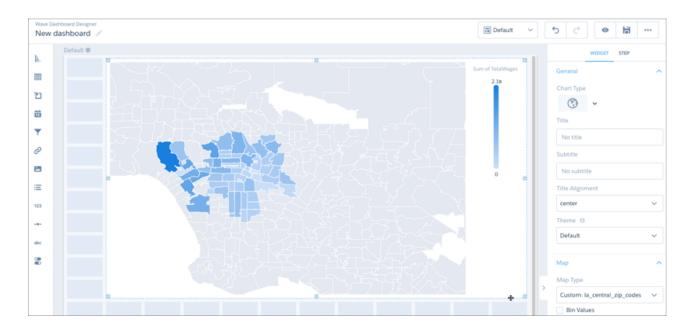
EclairNG Custom Map

Insert Custom Map Definitions

Enter a GeoJSON Definition and Associated Map Params



When viewed in Wave, this new map is zoomed in on the area of interest:



Matrix Charts

Use matrix charts (new in Wave dashboard designer) to create a table that represents measures visually, allowing you to quickly spot extreme values. The matrix chart is found in the Wave dashboard designer, classic designer, and explorer.

For example, you can show which product families are generating the most opportunities.



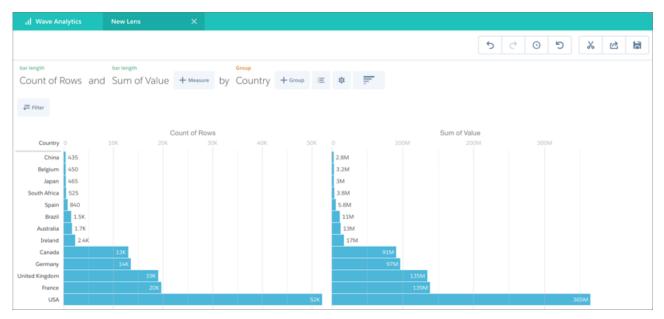
Pyramid Charts

Use a pyramid chart to visually highlight relative sizes of stacked dimension values. The pyramid chart is found in the Wave dashboard designer.

Create a Pyramid Chart

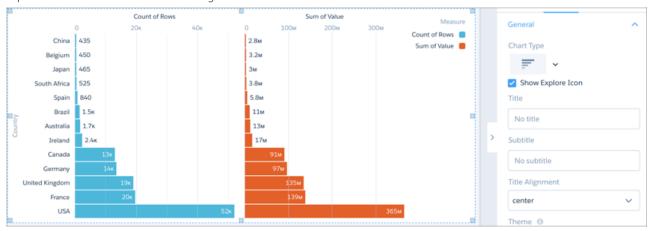
A pyramid chart is a good choice for presenting related measures for one particular dimension, such as accounts and account values in a particular category.

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.



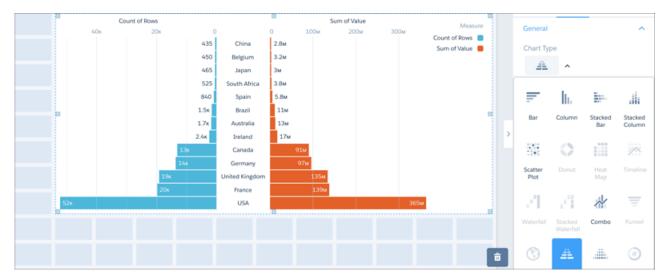
Changing it to a pyramid chart improves both overall appearance and readability.

1. Clip the bar chart to Wave dashboard designer.



A different color is automatically applied to the bars for the second measure.

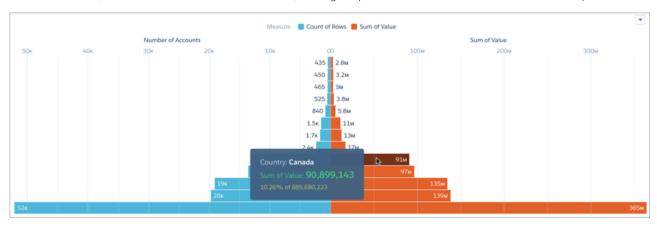
2. Change the chart type to pyramid.



3. Edit the title of the left axis, remove the center axis, and position the legend at top-center.



These adjustments improve the pyramid's appearance, but the loss of the country names for the center axis could make it a little less clear. However, when the chart is saved to a dashboard, floating the pointer over a bar reveals an informative tooltip.



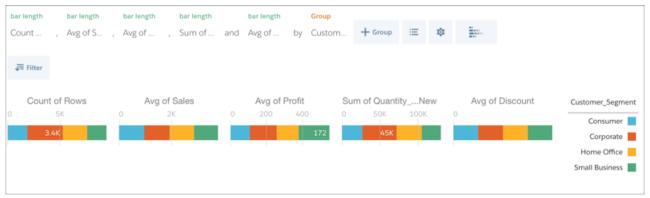
Radar Charts

Use radar charts to display a small dataset with one dimension and at least three measure columns. The radar chart is found in the Wave dashboard designer.

Create a Radar Chart

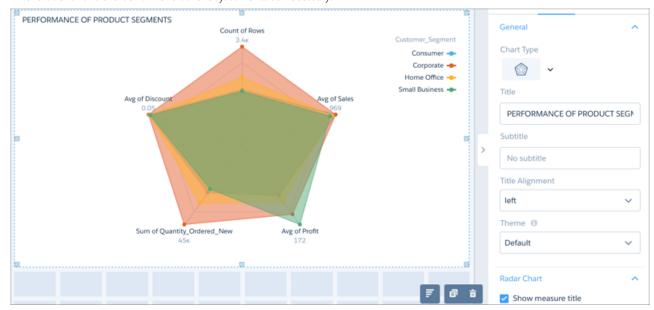
A radar chart is useful when you're trying to visually represent a dimension along three or more measure axes which have different scales. For example, if your business tracks key metrics for specific product segments, you want to visualize the product segments along with those metrics.

1. In Wave, group the data by product segment, then add the measures that reflect the key metrics.



Using a stacked bar chart to compare the product segments along the chosen measures, it's difficult to evaluate the performance of the different segments relative to each other.

- 2. Clip the lens to Wave dashboard designer, then choose Metrics Radar from Chart Type.
- 3. Enter a title for the chart and make other adjustments as necessary.



Rating Charts

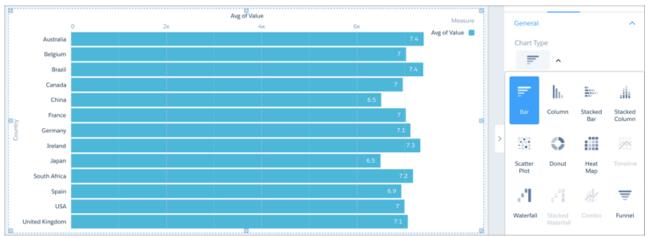
Use a rating chart to get a sense of how well a measured quality, such as customer satisfaction, is meeting expectations. The rating chart is found in the Wave dashboard designer.

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 Wave Analytics, create a lens using your customer-satisfaction dataset, then clip it to Wave dashboard designer.



- 2. Change the chart type to Rating.
- 3. Set the breakpoint values and associated band colors.



The breakpoints set the range values where the color bands transition. The maximum is 10, and you'd like to exceed 7.2 in each country.

4. Set the **Trellis** type to **Wrap** so the charts are easier to view all at once without having to scroll.



Values above 7.2 are easy to spot due to the two-color configuration of the rating chart. It's also easy to see which countries are close to the desired value.

Scatter Charts

Use a scatter chart to visualize correlation between two groups of data. The scatter chart is found in the Wave dashboard designer, classic designer, and explorer.

Timeline Charts

Use a timeline chart to show how a value changes over time. The timeline chart is found in the Wave dashboard designer, classic designer, and explorer.

Treemap Charts

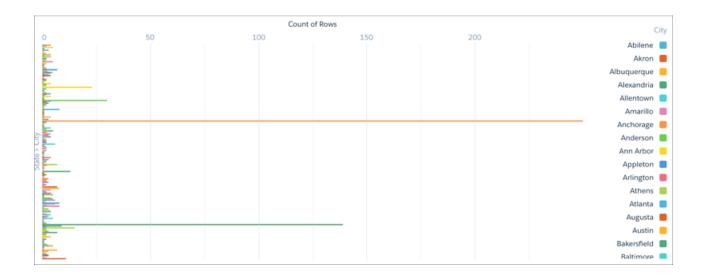
Use treemaps to visualize hierarchical quantitative data, where containing rectangles designate relationship using color, and "leaf" rectangles represent quantity using area. The treemap chart is found in the Wave dashboard designer.

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 city in each state in a way that reveals top markets.

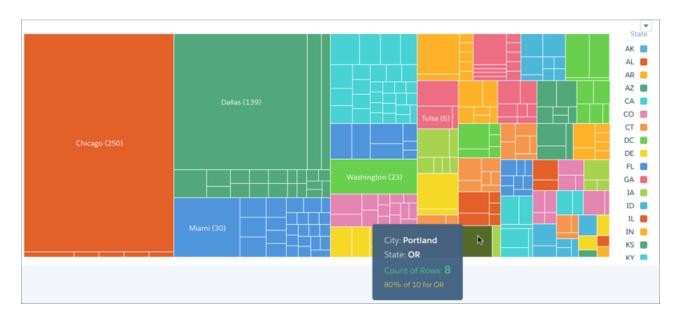
- 1. In Wave, group opportunities by state, then group by city.
- 2. Clip the lens to Wave dashboard designer, then view the resulting chart in the dashboard.



With the groupings in a bar chart, it's both difficult to take in the entire bar chart in a glance or quickly locate top markets.

3. Choose **Treemap** from **Chart Type**.

In the treemap chart, high-value cities are easy to spot. Using tooltips, you can view details for any rectangle representing a city.



Vertical Bar Charts

Use a vertical bar chart (also called column chart) to show relative counts of things, such as leads or dollars. The vertical bar chart is found in the Wave dashboard designer, classic designer, and explorer.

Analytics Build Wave Dashboards

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. The waterfall chart is found in the Wave dashboard designer and classic designer.

Build Wave Dashboards

Build a Wave dashboard to continuously monitor key metrics of your business, analyzing the results by key dimensions, like region, products, and time period. You can add interactive charts that synthesize information into an easy-to-read format. You can add tables to show low-level record details and create columns with calculated values. You can also add filters to allow the dashboard viewer to focus the results.

Wave provides the following designers to build dashboards.

Classic designer

This legacy designer enables you to build basic dashboards.

Wave dashboard designer

This newer designer has significant usability enhancements compared to the classic designer, which help you build dashboards more efficiently.

After the admin grants you access, you can use either designer to create and edit dashboards. You can also convert dashboards built in the classic designer to the Wave dashboard designer.

IN THIS SECTION:

What Is a Dashboard?

A Wave 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 Differences Between the Classic Designer and Wave Dashboard Designer

Consider the following differences between the classic designer and the Wave dashboard designer.

Build a Dashboard with the Wave Dashboard Designer

The Wave 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 layout. Create customized layouts to optimize the display of a dashboard on different types of devices, like mobile phones, tablets, and desktops. 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.

Build a Dashboard with the Classic Designer

The classic designer is the legacy designer that enables you to build basic Wave dashboards.

Keyboard Shortcuts For Wave Dashboards and Lenses

You can do some basic actions from your keyboard. The dashboard designer shortcuts vary based on whether you are using the classic designer or Wave dashboard designer.

Best Practices for Building Wave Dashboards

Follow best practices to design and build useful, effective Wave dashboards, while minimizing rework and addressing potential gaps.

Convert Your Dashboard to the Wave Dashboard Designer

To take advantage of its new features, convert dashboards created in the classic designer to the Wave dashboard designer.

Analytics Build Wave Dashboards

What Is a Dashboard?

A Wave 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 Wave Analytics 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.

IN THIS SECTION:

Key Elements of a Wave Dashboard

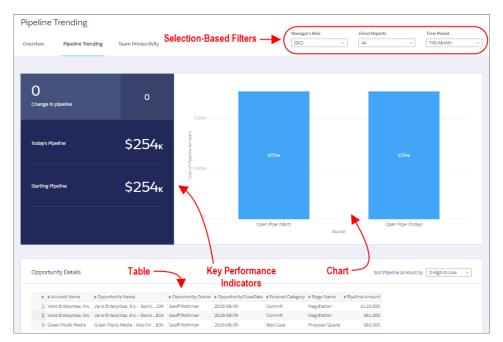
A dashboard isn't just for reading. It's for carrying on a conversation based on one or more datasets that matter to your company. Don't just look at the pictures—use the tools on the dashboard to dive deeper into the data that you care about. Go ahead, click around, and explore!

SEE ALSO:

Build a Dashboard with the Wave Dashboard Designer Build a Dashboard with the Classic Designer

Key Elements of a Wave 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 Wave 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. There are different types of selection-based filter widgets: date, list, range, and toggle.

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.

Key Differences Between the Classic Designer and Wave Dashboard Designer

Consider the following differences between the classic designer and the Wave dashboard designer.

• The Wave dashboard designer provides a user interface for creating layouts for different types of devices. With the classic designer, you have to set up mobile layouts using the dashboard JSON.

• The Wave dashboard designer provides wizards that walk you through creating widgets, with their underlying steps created behind the scenes. With the classic designer, you have to manually create the lenses in explorer, clip them, and then apply them to widgets. (A "step" in the Wave dashboard designer is the same as a "lens" in the classic designer.)

- The Wave dashboard designer uses snap-to-grid, letting you quickly position, align, and resize widgets. In the classic designer, you specify pixels to define each widget's position and size.
- You can create steps, including static steps, from within the Wave dashboard designer. In classic designer, you have to clip the lens from explorer to the designer or build the step using JSON.
- The Wave dashboard designer allows you to hide steps and properties to give you more workspace when designing a dashboard. This isn't possible in the classic designer.
- The Wave dashboard designer has a single table widget that can be used to create a values table or compare table. The classic designer has a separate widget for each type of table.
- Unlike the box widget in the classic designer, the container widget in the Wave dashboard designer groups widgets, which allows you to move them all at once.
- Only dasboards built with the Wave dashboard designer support annotations and notifications.
- Only the classic designer and explorer support the following chart types: parallel coordinates, horizontal dot plot, vertical dot plot, and matrix. When clipped to a Wave dashboard designer, these chart types convert to a bar chart.

SEE ALSO:

Build a Dashboard with the Wave Dashboard Designer Build a Dashboard with the Classic Designer

Build a Dashboard with the Wave Dashboard Designer

The Wave 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 layout. Create customized layouts to optimize the display of a dashboard on different types of devices, like mobile phones, tablets, and desktops. 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.

The Wave dashboard designer is enabled by default. If you can't access it, your admin might have disabled it.

IN THIS SECTION:

1. Create a Wave Dashboard with a Template

Each template has a unique layout with empty widgets (widgets without steps), sections, and a color scheme. To populate the widgets with your data, use the widget wizards. To create your own layout, choose the blank template.

2. Add Widgets to the Dashboard

Widgets are the basic building blocks of a dashboard. In the Wave 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.

3. Make the Dashboard Widgets Interactive

Wave 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.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

4. Set Initial Filter Selections in the Dashboard

If the Wave dashboard contains filter selection widgets or charts, you can set initial selections that are applied when the dashboard first opens. For example, you can use a selection to filter the dashboard to show only opportunities owned by the logged in user. While viewing the dashboard, the user can change the selections to filter the results differently or remove the filters altogether.

5. Generate Unique Wave 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.

SEE ALSO:

What Is a Dashboard?

Best Practices for Building Wave Dashboards

Key Differences Between the Classic Designer and Wave Dashboard Designer

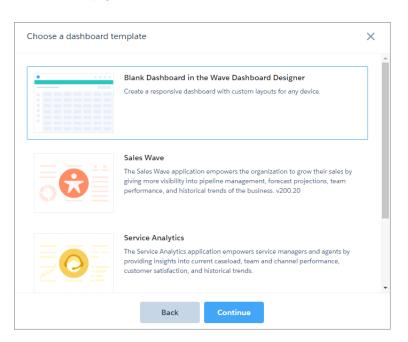
Grant Users Access to the Wave Dashboard Designer and Classic Designer

Build a Dashboard with the Classic Designer

Create a Wave Dashboard with a Template

Each template has a unique layout with empty widgets (widgets without steps), sections, and a color scheme. To populate the widgets with your data, use the widget wizards. To create your own layout, choose the blank template.

1. From the home page, select **Create** > **Dashboard**.



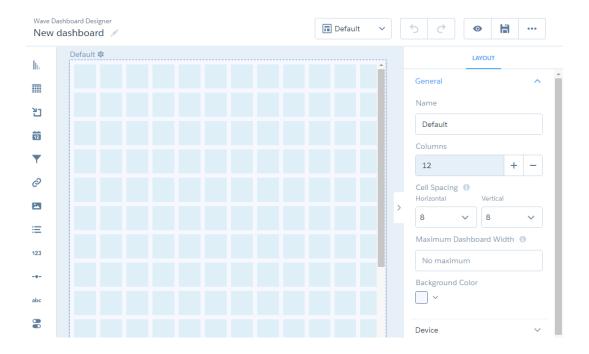
2. Select a template, and then click **Continue**.

For example, select **Blank Dashboard in the Wave Dashboard Designer**.

USER PERMISSIONS

To create a dashboard:

 "Create and Edit Wave Analytics Dashboards"



In addition to the blank template, you can choose a template with a prebuilt dashboard design. These templates are built for Wave dashboard designer only.

- **3.** To edit the dashboard, click ...
- **4.** If you selected a prebuilt template, you can add a step to each widget by clicking the icon inside an empty widget. A wizard guides you through adding the step.

When you complete the wizard, Wave populates the widget based on the results of the step. Wave also adds the step to the step panel.

5. To save the dashboard to an app, click 🗎 .

the dashboard, review the app permissions.

- **6.** Enter a title and description, and then select the app in which you want to save the dashboard. The permissions on the app determine each user's access to the dashboard. To ensure that the right users have the right access to
- 7. Click Save.
- **8.** Click and then click **Dashboard Properties** to set the default widget properties.

When you create a widget, you can specify whether the widget uses the default properties. Use defaults to ensure that the background and borders of all widgets are consistent.

IN THIS SECTION:

Dashboard Properties

These properties are default widget properties that you can apply to each widget to ensure a consistent appearance. The properties apply only to dashboards created in the Wave dashboard designer.

Dashboard Properties

These properties are default widget properties that you can apply to each widget to ensure a consistent appearance. The properties apply only to dashboards created in the Wave dashboard designer.

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

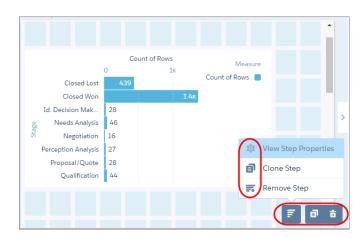
Add Widgets to the Dashboard

Widgets are the basic building blocks of a dashboard. In the Wave 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.

Some widgets require a step. A step returns the data. A widget determines how to use or display the data. For example, a step 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 step 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 step. You can also build your own step and apply it to a widget. Multiple widgets can use the same step, unless it's a date or range widget. In that case, no other widgets can use the step.

Clone widgets to create similar looking components in a Wave dashboard. For example, you can clone chart widgets and change the chart type to get a different view of the same results. You can also quickly create three number widgets and swap out their steps to show total open, total won, and total lost opportunity amounts. To swap out a step, remove the step from the widget and then add a new step to it. To clone widgets, remove steps, clone steps, and view step properties, select the widget while editing the dashboard, and then click the appropriate button.



To configure how a widget displays, 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 step properties in the Step tab. For example, you can choose which measure to display in the widget when its step contains multiple measures. Changes to step properties impact all widgets that are built on the step.

IN THIS SECTION:

Add Number Widgets to Include Key Metrics

Key metrics help you track critical measurements of your business. You can add other widgets to the Wave dashboard to filter the key metric results to focus on a particular area. 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 Wave dashboard allow you to visually summarize and trend your data. For example, a donut chart can show the distribution of opportunities across industry type. Because charts are interactive, dashboard viewers can select a segment of a chart to filter the results of other faceted widgets.

Add a Global Filter Widget to Exclude Dataset Records from the Dashboard

Global filters restrict the content available to all users who view the Wave dashboard. Use a global filter to apply the same filter to multiple steps in the dashboard. For example, if the dashboard has two charts comparing two sales reps' performance, both charts should have the same filters so you are comparing apples to apples. You can add multiple global filter widgets to a dashboard. You can also specify which steps to apply the global filter to. Unlike selection-based filter widgets, a global filter can't be changed while previewing or presenting the dashboard.

Add Selection-Based Filter Widgets to Enable Users to Filter the Results

Users viewing the dashboard can make selections in these widgets to apply whatever filters they wish. These widgets give them 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 global filter or query filter, the viewer can change the selection to filter the results differently or remove the filter altogether.

Add a Table Widget to Show Record Details and Create Calculated Columns

You can add different types of tables in a Wave dashboard. To view record-level details, add a values table. For example, this type of table can show case details, like how long the 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 step contains at least one grouping, you can create a pivot table.

Add a Container Widget to Create Sections in the Dashboard

Use sections in a Wave 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 filter selection 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 a container, 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 Wave dashboard, icons to categorize results, or graphics to animate the dashboard.

Add Text Widgets to Label Parts of the Dashboard

Label your Wave 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 Link Widget to Link to a Wave Lens or Dashboard

Instead of packing all possible key performance metrics, charts, and tables into a single dashboard, break up the content into multiple dashboards. Then use link widgets to tie the dashboards together. Dashboard viewers appreciate not having to figure out what to focus on in a busy dashboard. You can also use links to create analysis paths for users to follow or to link related assets.

Create Steps in the Wave Dashboard Designer

Steps return results that are displayed in widgets. For example, a number widget displays the result of a calculation—that calculation is defined in a step. Steps can query a data source, like a dataset or a Salesforce object. They can also be a static list of user-defined values—a static step. You can reuse each step in multiple widgets. In addition to clipping a lens from an exploration, you can use the following ways to create steps in the Wave dashboard designer: use the wizard or create it manually.

Widget Properties for Wave Dashboards

Widget properties define how widgets appear in the Wave 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 Wave dashboard designer.

Step Properties for Wave Dashboards

Step 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 Wave dashboard. The properties apply to dashboards created in the Wave dashboard designer only.

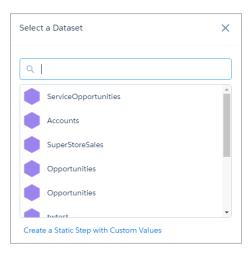
Add Number Widgets to Include Key Metrics

Key metrics help you track critical measurements of your business. You can add other widgets to the Wave dashboard to filter the key metric results to focus on a particular area. 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 step that has at least one measure. If the step has multiple measures, you can choose which measure to display—by default, the first measure shows. If you include groupings, the widget displays the measure for the first group in the first defined grouping.

- 1. Drag the number widget to the dashboard canvas.
- **2.** To use an existing step for the widget, drag the step from the step panel and place on top of the widget. If you don't see the step panel, click the dashboard header to show it.
- **3.** To create a step:
 - **a.** Click the button inside the widget to open the wizard.

 The Select a Dataset screen appears, showing datasets in the most recently used order.
 - Note: If you see a bar chart in the explorer instead, Wave selected the dataset that you used to create your previous widget step. To use a different dataset, click **Back**.



b. Select the dataset that contains the measure. Type in the search box if your dataset isn't easy to find. The explorer shows a bar chart with the Count of Rows measure.

c. Add measures that you want to show in the widget.

To add a measure, click + Measure . To change a measure, select the measure in the explorer and then choose a new one.

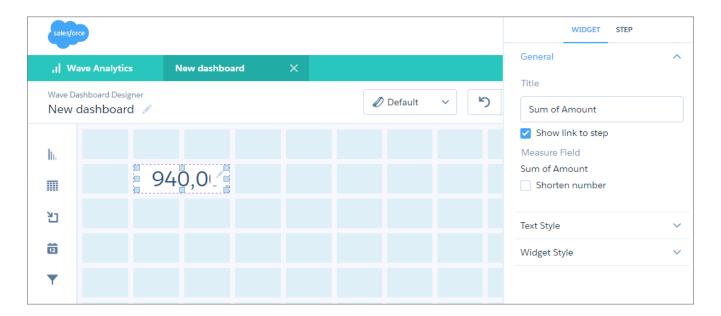
d. If needed, click **Filter** to filter the results. You can add multiple filters.

e. Click Save.

Wave adds the step to the steps panel, making it available for use by other widgets in the dashboard. In addition, the widget shows the calculated measure.

4. To change the widget and step properties, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.



- **5.** Set the widget properties to customize the appearance of the widget. When you set a property, the widget immediately shows the effects of your change.
- **6.** To show and set the step properties, click **Step** in the panel.
- **7.** To preview your changes to the dashboard, click .
- **8.** To save your changes to the dashboard, click 🔒 .



(1) Important: If you close the dashboard without saving it, you lose your changes.

SEE ALSO:

Widget Properties for Wave Dashboards Step Properties for Wave Dashboards Create Steps in the Wave Dashboard Designer

Add Chart Widgets to Visualize Data

Charts in a Wave dashboard allow you to visually summarize and trend your data. For example, a donut chart can show the distribution of opportunities across industry type. Because charts are interactive, dashboard viewers can select a segment of a chart to filter the results of other faceted widgets.

Wave supports multiple chart types. Each type has unique widget properties and step requirements. For details about each chart type, see Visualizing Data With Charts.



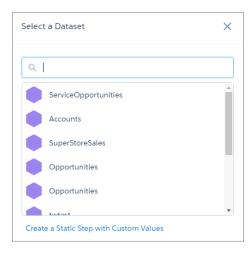
Note: Unlike the explorer and classic designer, the Wave dashboard designer doesn't support the following chart types: parallel coordinates, horizontal dot plot, vertical dot plot, and matrix. Although you can create a step with these chart types, the Wave dashboard designer converts them to a bar chart when added to a chart widget.

The selection in this widget filters the results of other widgets with faceted steps. You can configure this widget's step to accept single or multiple selections, and specify whether the selections are optional or required. You can also set an initial selection that applies when the dashboard opens. While viewing the dashboard, you can undo the selection or change the selection to filter the results differently.

- Note: To enable this widget to filter other widgets, you must enable faceting on its step.
- 1. Drag the chart widget to the dashboard canvas.
- 2. To use an existing step for the widget, drag the step from the step panel and place on top of the widget. If you don't see the step panel, click the dashboard header to show it.
- **3.** To create a step:
 - **a.** Click the button inside the widget to open the wizard. The Select a Dataset screen appears, showing datasets in the most recently used order.



Note: If you see a bar chart in the explorer instead, Wave selected the dataset that you used to create your previous widget step. To use a different dataset, click **Back**.



- **b.** Select the dataset on which you want to build the chart. The explorer shows a bar chart with the Count of Rows measure.
- c. Add measures that you want to show in the widget.

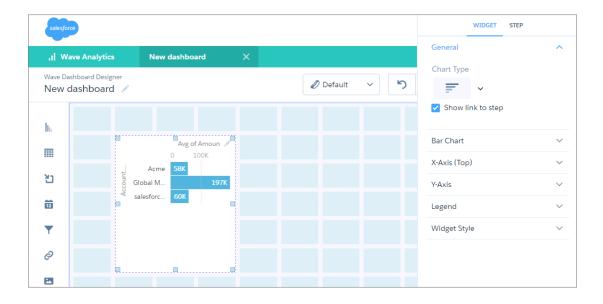
To add a measure, click ** Measure* . To change a measure, select the measure in the explorer and then choose a new one.

- **d.** If needed, add groupings.
- e. If needed, click Filter to filter the results.You can add multiple filters.
- f. Click Save.

Wave adds the step to the steps panel, making it available for use by other widgets in the dashboard. In addition, the widget shows the results in a bar chart.

4. To change the widget and step properties, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.



5. Set the widget properties to customize the appearance of the widget.

For example, select a different chart type in the Chart Type widget property. If the step requirements aren't met for a chart type, you can't select the chart type in a widget. For instance, some chart types require steps with only one measure.

- **6.** To show and set the step properties, click **Step** in the panel.

 You can configure this step to accept single or multiple selections, and specify whether the selections are optional or required.
- 7. To set the initial selections that are applied when a user opens the dashboard, click ____, and then click **Pick Initial Filters**. You can make initial selections on any chart, date, list, toggle, or range widget. For example, you can select a bar in a bar chart and set that as the initial filter. If the underlying step for the bar chart is faceted, other widgets with faceted steps are also filtered based on this initial selection. While viewing the dashboard, the viewer can change the selection in the chart to filter the results differently or remove the filter altogether.
 - Note: No selections are made in this widget when the user opens the dashboard unless you pick the initial filter. If the step is configured to require a selection, select the initial filter. Otherwise, the first value in the filter list is used.
- **9.** To save your changes to the dashboard, click 🔒 .
 - Important: If you close the dashboard without saving it, you lose your changes.

SEE ALSO:

Widget Properties for Wave Dashboards
Step Properties for Wave Dashboards
Create Steps in the Wave Dashboard Designer

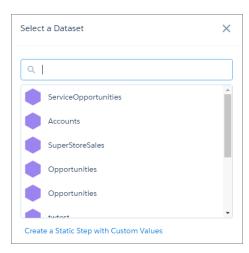
Add a Global Filter Widget to Exclude Dataset Records from the Dashboard

Global filters restrict the content available to all users who view the Wave dashboard. Use a global filter to apply the same filter to multiple steps in the dashboard. For example, if the dashboard has two charts comparing two sales reps' performance, both charts should have the same filters so you are comparing apples to apples. You can add multiple global filter widgets to a dashboard. You can also specify which steps to apply the global filter to. Unlike selection-based filter widgets, a global filter can't be changed while previewing or presenting the dashboard.

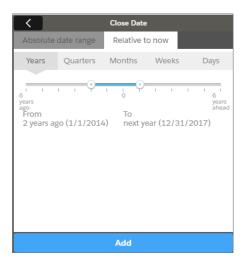
The global filter widget requires a step that has at least one filter and no groupings. Each filter can be based on a measure, dimension, or date field. All filters defined in the step are applied to the global filter.

- 1. Drag the global filter widget to the dashboard canvas.
- **2.** To use an existing step for the widget, drag the step from the step panel and place on top of the widget. If you don't see the step panel, click the dashboard header to show it.
- **3.** To create a step:
 - **a.** Click the button inside the widget to open the wizard.

 The Select a Dataset screen appears, showing datasets in the most recently used order.
 - Note: If you see a bar chart in the explorer instead, Wave selected the dataset that you used to create your previous widget step. To use a different dataset, click **Back**.



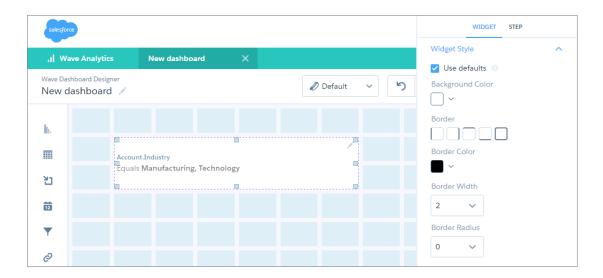
- **b.** Select the dataset that you want to apply the filter on.
- c. Click = Filter
- **d.** Select a measure, date, or dimension field that you want to filter on.
- **e.** To create the filter, select particular values of the field. For example, to filter based on dates, select the start and end dates in the Absolute date range tab or Relative to now tab.



- **f.** To apply the filter, click **Add**.

 If needed, you can add multiple filters.
- g. Click Save.
 Wave adds the step to the steps panel, making it available for use by other widgets in the dashboard. In addition, the widget displays the filters.
- **4.** To change the widget and step properties, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.



- **5.** Set the widget properties to customize the appearance of the widget. When you set a property, the widget immediately shows the effects of your change.
- **6.** To show and set the step properties, click **Step** in the panel.
- **8.** To save your changes to the dashboard, click 🛗 .
 - Important: If you close the dashboard without saving it, you lose your changes.

To apply the global filter on a widget, select **Apply global filters** in the step properties of the widget. By default, the global filter is applied to all widgets that have steps that: 1) are built in the explorer; 2) are based on the same dataset; and 3) have groupings based on the same field used in the global filter. To apply the global filter on widgets based on steps built from SAQL queries, you must also set autoFilter equal to true in the step definition in the dashboard JSON. For more information about setting autoFilter, see Query.

SEE ALSO:

Widget Properties for Wave Dashboards
Step Properties for Wave Dashboards
Create Steps in the Wave Dashboard Designer

Add Selection-Based Filter Widgets to Enable Users to Filter the Results

Users viewing the dashboard can make selections in these widgets to apply whatever filters they wish. These widgets give them 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 global filter or query filter, the viewer can change the selection to filter the results differently or remove the filter altogether.

A selection-based filter is different from a query filter. When a selection is made, all values in the step's original query still appear in the widget, just with particular values highlighted. However, a query filter modifies the step's query, reducing the number of results that appear in the widget. A global filter is a query filter that can be applied to multiple steps in a dashboard.

The following illustration shows the difference between a selection and a query filter on the same step.



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.



Note: You must facet or bind the steps of widgets to filter them based on a selection in another widget. For more information, see Filter the Dashboard Results with Faceting.

IN THIS SECTION:

Add a Date Widget to Filter Dashboard Results Based on a Date

A date filter widget allows dashboard viewers to filter the Wave dashboard results based on a selected date range. The range can use relative dates based on UTC (like last month) or absolute dates (like 1/15/2016). For example, you can add a date widget to filter support cases, showing only cases that closed this month or last month.

Add a List Widget to Filter Dashboard Results Based on a Dimension

A list filter widget allows dashboard viewers to filter the Wave dashboard results based on a dimension. For example, a list widget can filter results based on a particular case status or region.

Add a Range Widget to Filter Dashboard Results Based on a Measure

A range widget allows dashboard viewers to filter the Wave dashboard results based on a measure. For example, a range widget can show only opportunities with amounts between \$100,000 and \$900,000.

Add a Toggle Widget to Filter Dashboard Results Based on a Dimension

A toggle widget allows dashboard viewers to filter the Wave dashboard results based on a date or dimension. For example, a toggle widget can show results based on a specific record type.

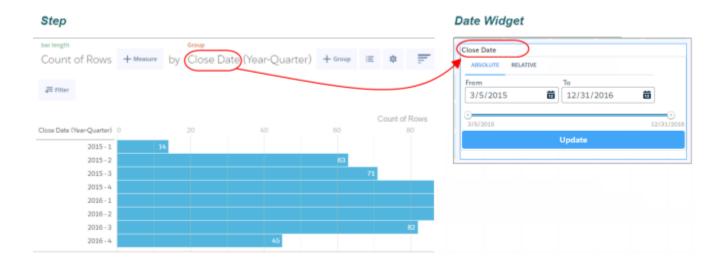
SEE ALSO:

Set Initial Filter 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 Wave dashboard results based on a selected date range. The range can use relative dates based on UTC (like last month) or absolute dates (like 1/15/2016). For example, you can add a date widget to filter support cases, showing only cases that closed this month or last month.

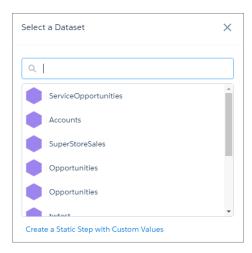
The date widget requires a step that has a single grouping based on a date field.



The selection in this widget filters the results of other widgets with faceted steps. You can configure this widget's step to accept single or multiple selections, and specify whether the selections are optional or required. You can also set an initial selection that applies when the dashboard opens. While viewing the dashboard, you can undo the selection or change the selection to filter the results differently.

- Note: To enable this widget to filter other widgets, you must enable faceting on its step.
- 1. Drag the date widget to the dashboard canvas.
- **2.** To use an existing step for the widget, drag the step from the step panel and place on top of the widget. If you don't see the step panel, click the dashboard header to show it.
- **3.** To create a step:
 - **a.** Click the button inside the widget to open the wizard.

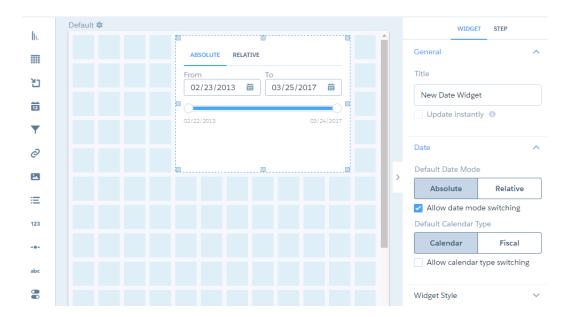
 The Select a Dataset screen appears, showing datasets in the most recently used order.
 - Note: If you see a bar chart in the explorer instead, Wave selected the dataset that you used to create your previous widget step. To use a different dataset, click **Back**.



- **b.** Select the dataset that you want to apply the filter on.
- **c.** Select the date field that you want to filter on.
- **d.** If needed, click **Filter** to filter the results. You can add multiple filters.
- e. To add the step to the widget, click **Create**.

 Wave adds the step to the steps panel. The widget shows the absolute date range, which spans from the minimum to the maximum values of the date field. Wave doesn't apply any selections to the widget.
- **4.** To change the widget and step properties, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.



5. Set the widget properties to customize the appearance of the widget.

When you set a property, the widget immediately shows the effects of your change.

6. To show and set the step properties, click **Step** in the panel.

You can configure this step to accept single or multiple selections, and specify whether the selections are optional or required.

- 7. To set the initial selections that are applied when a user opens the dashboard, click ____, and then click **Pick Initial Filters**. You can make initial selections on any chart, date, list, toggle, or range widget. For example, you can select a bar in a bar chart and set that as the initial filter. If the underlying step for the bar chart is faceted, other widgets with faceted steps are also filtered based on this initial selection. While viewing the dashboard, the viewer can change the selection in the chart to filter the results differently or remove the filter altogether.
 - Note: No selections are made in this widget when the user opens the dashboard unless you pick the initial filter. If the step is configured to require a selection, select the initial filter. Otherwise, the first value in the filter list is used.
- **8.** To preview your changes to the dashboard, click . . .
- **9.** To save your changes to the dashboard, click 🔒 .
 - Important: If you close the dashboard without saving it, you lose your changes.

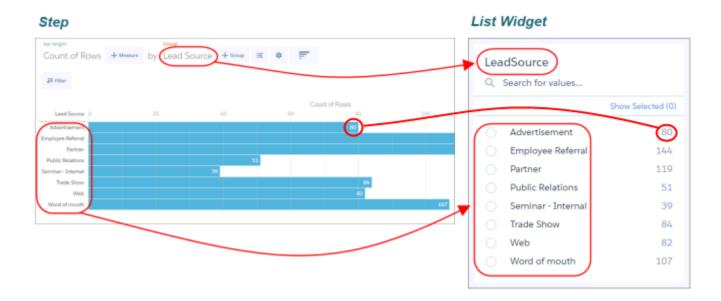
SEE ALSO:

Widget Properties for Wave Dashboards
Step Properties for Wave Dashboards
Create Steps in the Wave Dashboard Designer

Add a List Widget to Filter Dashboard Results Based on a Dimension

A list filter widget allows dashboard viewers to filter the Wave dashboard results based on a dimension. For example, a list widget can filter results based on a particular case status or region.

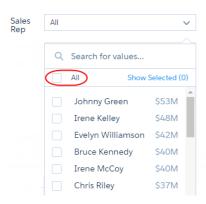
The list widget requires a step that has a single grouping based on a dimension. The values of the grouping appear as the values in the list.





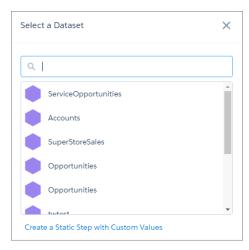
Note: You can also create a static step to specify a custom list of values for the list widget. For more information, see Create a Static Step with Custom Values.

The selection in this widget filters the results of other widgets with faceted steps. You can configure this widget's step to accept single or multiple selections, and specify whether the selections are optional or required. You can also set an initial selection that applies when the dashboard opens. While viewing the dashboard, you can undo the selection or change the selection to filter the results differently.



- Note: To enable this widget to filter other widgets, you must enable faceting on its step.
- 1. Drag the list widget to the dashboard canvas.
- **2.** To use an existing step for the widget, drag the step from the step panel and place on top of the widget. If you don't see the step panel, click the dashboard header to show it.
- **3.** To create a step:
 - **a.** Click the button inside the widget to open the wizard.

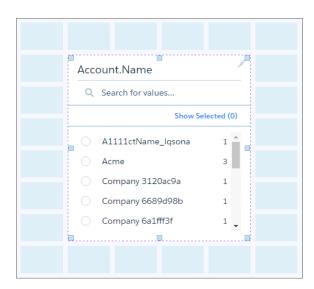
 The Select a Dataset screen appears, showing datasets in the most recently used order.
 - Note: If you see a bar chart in the explorer instead, Wave selected the dataset that you used to create your previous widget step. To use a different dataset, click **Back**.

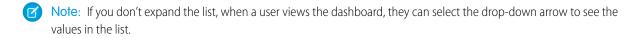


- **b.** Select the dataset that you want to apply the filter on.
- **c.** Select the dimension field that you want to filter on.
- d. To add the step to the widget, click Create.
 Wave adds the step to the steps panel. In addition, by default, the list widget is collapsed and Wave doesn't apply any selections to the widget.



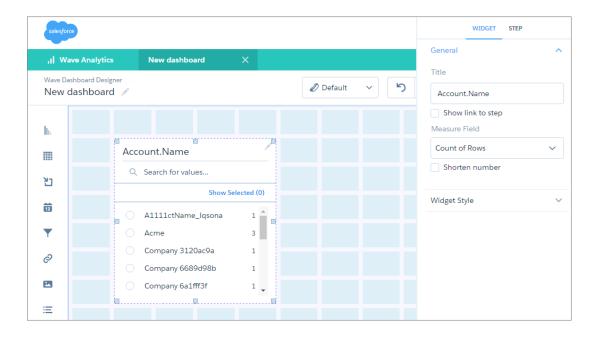
e. To expand the list and show more values, increase the size of the widget.





4. To change the widget and step properties, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.



- **5.** Set the widget properties to customize the appearance of the widget. When you set a property, the widget immediately shows the effects of your change.
- **6.** To show and set the step properties, click **Step** in the panel.

 You can configure this step to accept single or multiple selections, and specify whether the selections are optional or required. You can also specify whether to show the measure next to each value in the list.
- 7. To set the initial selections that are applied when a user opens the dashboard, click _____, and then click **Pick Initial Filters**. You can make initial selections on any chart, date, list, toggle, or range widget. For example, you can select a bar in a bar chart and set that as the initial filter. If the underlying step for the bar chart is faceted, other widgets with faceted steps are also filtered based on this initial selection. While viewing the dashboard, the viewer can change the selection in the chart to filter the results differently or remove the filter altogether.
 - Note: No selections are made in this widget when the user opens the dashboard unless you pick the initial filter. If the step is configured to require a selection, select the initial filter. Otherwise, the first value in the filter list is used.
- **8.** To preview your changes to the dashboard, click .
- **9.** To save your changes to the dashboard, click 📋 .
 - Important: If you close the dashboard without saving it, you lose your changes.

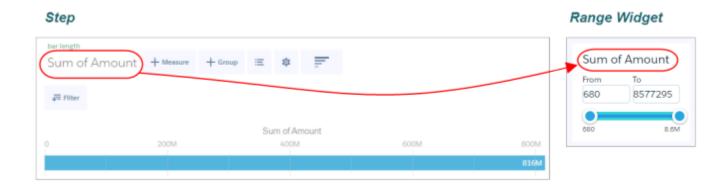
SEE ALSO:

Widget Properties for Wave Dashboards
Step Properties for Wave Dashboards
Create Steps in the Wave Dashboard Designer

Add a Range Widget to Filter Dashboard Results Based on a Measure

A range widget allows dashboard viewers to filter the Wave dashboard results based on a measure. For example, a range widget can show only opportunities with amounts between \$100,000 and \$900,000.

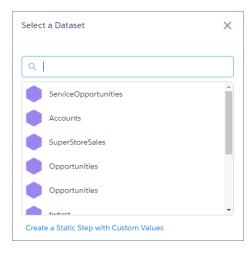
The range widget requires a step that has a single measure and no groupings. The selected range applies to the values of the measure. The range shown in the widget is from the minimum to the maximum values of the measure.



The selection in this widget filters the results of other widgets with faceted steps. You can configure this widget's step to accept single or multiple selections, and specify whether the selections are optional or required. You can also set an initial selection that applies when the dashboard opens. While viewing the dashboard, you can undo the selection or change the selection to filter the results differently.

- Note: To enable this widget to filter other widgets, you must enable faceting on its step.
- 1. Drag the range widget to the dashboard canvas.
- **2.** To use an existing step for the widget, drag the step from the step panel and place on top of the widget. If you don't see the step panel, click the dashboard header to show it.
- **3.** To create a step:
 - **a.** Click the button inside the widget to open the wizard.

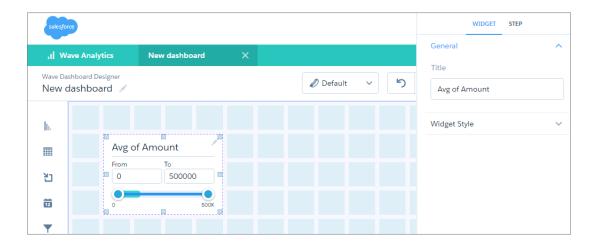
 The Select a Dataset screen appears, showing datasets in the most recently used order.
 - Note: If you see a bar chart in the explorer instead, Wave selected the dataset that you used to create your previous widget step. To use a different dataset, click **Back**.



- **b.** Select the dataset that you want to apply the filter on.
- **c.** Select the measure field that you want to filter on.
- **d.** If needed, click **Filter** to filter the results. You can add multiple filters.
- e. To add the step to the widget, click **Create**.

 Wave adds the step to the steps panel. The range of the widget spans from the minimum to the maximum values of the measure field. Wave doesn't apply any selections to the widget.
- **4.** To change the widget and step properties, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.



- **5.** Set the widget properties to customize the appearance of the widget. When you set a property, the widget immediately shows the effects of your change.
- **6.** To show and set the step properties, click **Step** in the panel.

 You can configure this step to accept single or multiple selections, and specify whether the selections are optional or required.
- 7. To set the initial selections that are applied when a user opens the dashboard, click _____, and then click **Pick Initial Filters**. You can make initial selections on any chart, date, list, toggle, or range widget. For example, you can select a bar in a bar chart and set that as the initial filter. If the underlying step for the bar chart is faceted, other widgets with faceted steps are also filtered based on this initial selection. While viewing the dashboard, the viewer can change the selection in the chart to filter the results differently or remove the filter altogether.
 - Note: No selections are made in this widget when the user opens the dashboard unless you pick the initial filter. If the step is configured to require a selection, select the initial filter. Otherwise, the first value in the filter list is used.
- **9.** To save your changes to the dashboard, click 🔒 .

(1) Important: If you close the dashboard without saving it, you lose your changes.

SEE ALSO:

Widget Properties for Wave Dashboards

Step Properties for Wave Dashboards

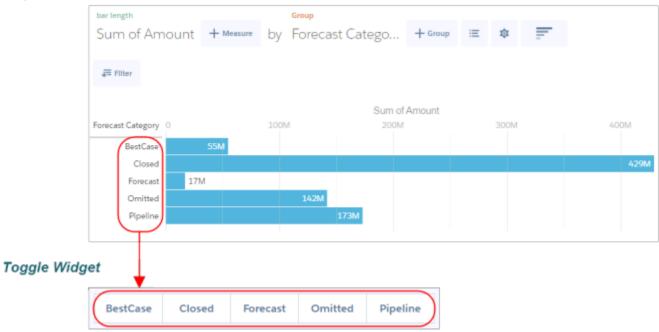
Create Steps in the Wave Dashboard Designer

Add a Toggle Widget to Filter Dashboard Results Based on a Dimension

A toggle widget allows dashboard viewers to filter the Wave 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 step that has a single grouping based on a dimension. The values of the grouping appear as the toggle options.

Step



Note: You can also create a static step to specify a custom list of values for the toggle widget. For more information, see Create a Static Step with Custom Values.

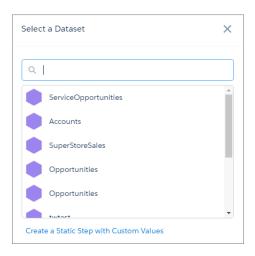
The selection in this widget filters the results of other widgets with faceted steps. You can configure this widget's step to accept single or multiple selections, and specify whether the selections are optional or required. You can also set an initial selection that applies when the dashboard opens. While viewing the dashboard, you can undo the selection or change the selection to filter the results differently.

- Note: To enable this widget to filter other widgets, you must enable faceting on its step.
- 1. Drag the toggle widget to the dashboard canvas.
- 2. To use an existing step for the widget, drag the step from the step panel and place on top of the widget.

If you don't see the step panel, click the dashboard header to show it.

- **3.** To create a step:
 - **a.** Click the button inside the widget to open the wizard.

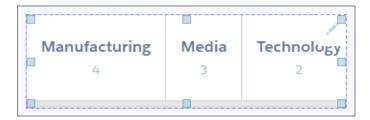
 The Select a Dataset screen appears, showing datasets in the most recently used order.
 - Note: If you see a bar chart in the explorer instead, Wave selected the dataset that you used to create your previous widget step. To use a different dataset, click **Back**.



- **b.** Select the dataset that you want to apply the filter on.
- **c.** Select the date or dimension field that you want to filter on.
- **d.** If needed, click **Filter** to filter the results. You can add multiple filters.
- e. To add the step to the widget, click Create.Wave adds the step to the steps panel. Wave doesn't apply any selections to the widget.

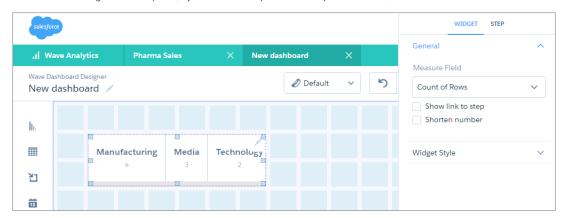


f. If needed, increase the size of the widget to show the display names of all toggle options.



4. To change the widget and step properties, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.



- **5.** Set the widget properties to customize the appearance of the widget. When you set a property, the widget immediately shows the effects of your change.
- **6.** To show and set the step properties, click **Step** in the panel.

 You can configure this step to accept single or multiple selections, and specify whether the selections are optional or required.
- 7. To show and set the step properties, click **Step** in the panel.

 You can configure this step to accept single or multiple selections, and specify whether the selections are optional or required. You can also specify whether to show the measure in the toggle options.
- 8. To set the initial selections that are applied when a user opens the dashboard, click ____, and then click **Pick Initial Filters**. You can make initial selections on any chart, date, list, toggle, or range widget. For example, you can select a bar in a bar chart and set that as the initial filter. If the underlying step for the bar chart is faceted, other widgets with faceted steps are also filtered based on this initial selection. While viewing the dashboard, the viewer can change the selection in the chart to filter the results differently or remove the filter altogether.
 - Note: No selections are made in this widget when the user opens the dashboard unless you pick the initial filter. If the step is configured to require a selection, select the initial filter. Otherwise, the first value in the filter list is used.
- **9.** To preview your changes to the dashboard, click <u>•</u> .
- **10.** To save your changes to the dashboard, click 🔒 .
 - (1) Important: If you close the dashboard without saving it, you lose your changes.

SEE ALSO:

Widget Properties for Wave Dashboards
Step Properties for Wave Dashboards
Create Steps in the Wave Dashboard Designer

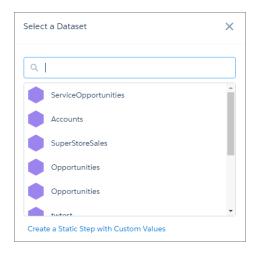
Add a Table Widget to Show Record Details and Create Calculated Columns

You can add different types of tables in a Wave dashboard. To view record-level details, add a values table. For example, this type of table can show case details, like how long the 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 step contains at least one grouping, you can create a pivot table.

The table widget requires a step that specifies which columns to include in the table. The step varies depending the type of table. A values table shows the dataset fields as columns. You can choose which fields to show. A compare table shows calculated fields based on the columns available in the table. A pivot table shows the values of the last grouping as columns.

- Note: By default, the table widget shows up to 100 records. You can show up to 10,000 records. To increase the number of records, change the limit parameter in the SAQL guery. For more information about SAQL gueries, see Wave Analytics SAQL Reference.
- 1. Drag the table widget to the dashboard canvas.
- **2.** To use an existing step for the widget, drag the step from the step panel and place on top of the widget. If you don't see the step panel, click the dashboard header to show it.
- **3.** To create a step:
 - **a.** Click the button inside the widget to open the wizard.

 The Select a Dataset screen appears, showing datasets in the most recently used order.
 - Note: If you see a bar chart in the explorer instead, Wave selected the dataset that you used to create your previous widget step. To use a different dataset, click **Back**.



- **b.** Select the dataset that contains the columns that you want to include in the table. The explorer shows a bar chart with the Count of Rows measure.
- To make it easier to see which columns you are adding to the table, click



and change the chart type to:

Values table

c.

Use to show dataset fields as table columns. To choose which columns to include in the table, click the columns at the top of the explorer. You can't create calculated columns in a values table.

Compare table

Use to add calculated columns to the table. To add dimensions from the dataset as columns in the table, click **+Group**. To add other columns, including calculated columns, click **+**. For more information about creating a compare table, see Creating Calculated Columns Using a Compare Table.

Pivot table

Use to show the values of the last grouping as columns. To add dimensions as columns in the table, click **+Group**. The table is pivoted based on the last defined grouping.

d. If needed, click **Filter** to filter the records included in the table.

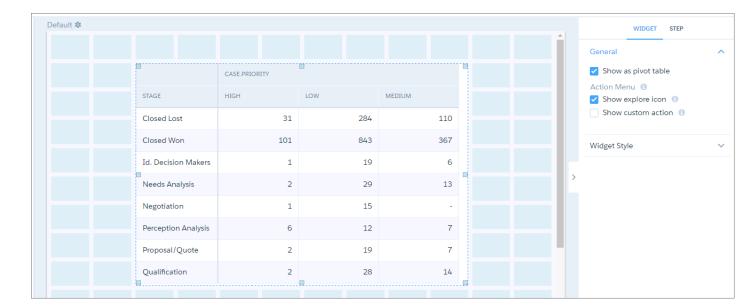
You can add multiple filters.

e. Click Save.

The widget shows the results in the table. Wave adds the step to the steps panel.

4. To change the widget and step properties, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.



- **5.** Set the widget properties to customize the appearance of the widget.
- **6.** To show and set the step properties, click **Step** in the panel.
- 7. To preview your changes to the dashboard, click .
- **8.** To save your changes to the dashboard, click $\frac{1}{100}$.
 - Important: If you close the dashboard without saving it, you lose your changes.

SEE ALSO:

Widget Properties for Wave Dashboards
Step Properties for Wave Dashboards
Create Steps in the Wave Dashboard Designer

Add a Container Widget to Create Sections in the Dashboard

Use sections in a Wave 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 filter selection 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 a container, the contained widgets also move while maintaining their spacing and alignment.

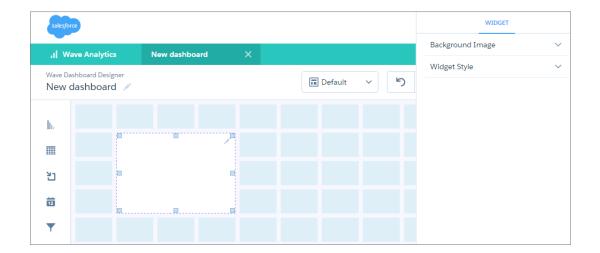
You can place any widget inside a container widget, except another container widget.

- **1.** Drag the container widget to the dashboard canvas. An empty container appears.
- **2.** Drag other widgets inside the container.

You can increase the size of the container to fit more widgets.

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 some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.



- **4.** Set the widget properties to customize the appearance of the widget. For example, apply a background color or image.
- To preview your changes to the dashboard, click .
- **6.** To save your changes to the dashboard, click 📋 .
 - Important: If you close the dashboard without saving it, you lose your changes.

SEE ALSO:

Widget Properties for Wave Dashboards

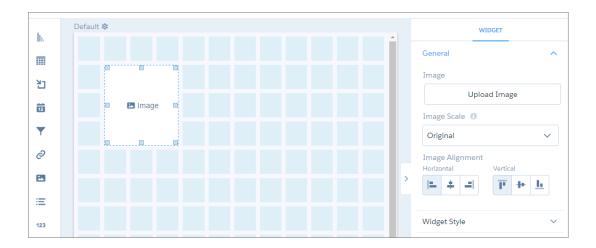
Add an Image Widget to Display Graphics in the Dashboard

You can include company logos to brand the Wave dashboard, icons to categorize results, or graphics to animate the dashboard.

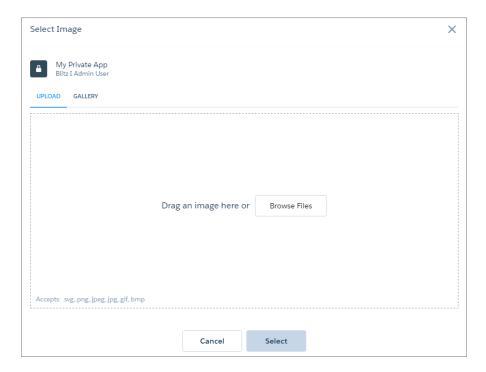
You can also add images as backgrounds for container widgets and layouts. Wave in a web browser supports .svg, .png. jpeg, .jpg, .gif, .bmp. The Wave Analytics mobile app supports all of these file types, except .svg.

- 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.** To change the widget properties, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.



3. Click Image inside the image widget.



- **4.** To upload a new image, click **Browse Files**. Or, select an uploaded image from the Gallery tab.

 Wave 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.
- **5.** Set the widget properties to customize the appearance of the widget.
- **6.** To preview your changes to the dashboard, click .
- **7.** To save your changes to the dashboard, click 📋 .

(1) Important: If you close the dashboard without saving it, you lose your changes.

SEE ALSO:

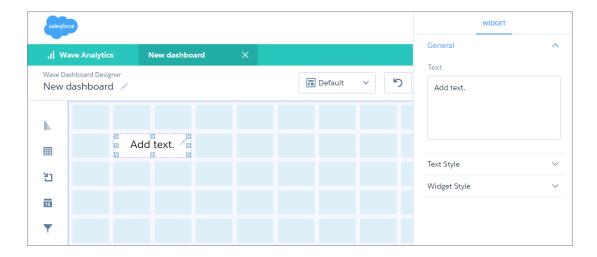
Widget Properties for Wave Dashboards

Add Text Widgets to Label Parts of the Dashboard

Label your Wave 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, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.



- **3.** Set the widget properties to customize the appearance of the widget. For example, enter the text that you want to appear in this widget.
- **4.** To preview your changes to the dashboard, click .
- **5.** To save your changes to the dashboard, click 📋 .
 - (1) Important: If you close the dashboard without saving it, you lose your changes.

SEE ALSO:

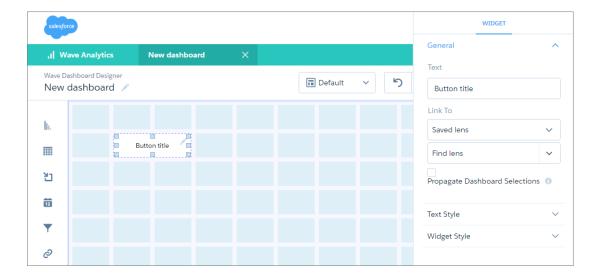
Widget Properties for Wave Dashboards

Add a Link Widget to Link to a Wave Lens or Dashboard

Instead of packing all possible key performance metrics, charts, and tables into a single dashboard, break up the content into multiple dashboards. Then use link widgets to tie the dashboards together. Dashboard viewers appreciate not having to figure out what to focus on in a busy dashboard. You can also use links to create analysis paths for users to follow or to link related assets.

- 1. Drag the link widget to the dashboard canvas.
- 2. To change the widget properties, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.



3. Set the widget properties to customize the appearance of the widget.

For example, specify the lens or dashboard to link to. Specify whether the filters selections in the dashboard are passed to the linked asset. For example, you view a dashboard that shows only open cases. When you click the link, the linked dashboard applies the same selection filter, showing only open cases.

- **4.** To preview your changes to the dashboard, click • .
- **5.** To save your changes to the dashboard, click $\begin{cases} \blacksquare \end{cases}$.
 - [] Important: If you close the dashboard without saving it, you lose your changes.

SEE ALSO:

Widget Properties for Wave Dashboards

Create Steps in the Wave Dashboard Designer

Steps return results that are displayed in widgets. For example, a number widget displays the result of a calculation—that calculation is defined in a step. Steps can query a data source, like a dataset or a Salesforce object. They can also be a static list of user-defined values—a static step. You can reuse each step in multiple widgets. In addition to clipping a lens from an exploration, you can use the following ways to create steps in the Wave dashboard designer: use the wizard or create it manually.

The widget wizard creates a basic step that meets the minimum requirements for the widget. If you need a more robust step, like one that has multiple measures or a customized SAQL query, manually create your own. The method you choose depends on how much flexibility you need.

Because IDs are not always descriptive, you can add a step label. For example, you can add the "Won Opportunity Amount" label to step ID "Amount_1." Specify the label when you create the step because Wave generates the step ID based on the label. After creating the step, you can change the label but not the ID, unless you edit the JSON.



Tip: If you create a query but Wave takes too long to execute it, speed up the query results by performing the calculation in the dataflow. The dataflow stores the results in the dataset, allowing for quick retrieval of the precalculated values.

IN THIS SECTION:

Widget Steps in a Wave Dashboard

Some widgets require steps. A step contains a query that returns results. The widget determines how to display the results. For example, a step 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 Wave dashboard designer, most widgets have a wizard that you can use to build the widget and its step. You can also build your own step and attach it to a widget.

Create a Step with the Explorer

Instead of using a widget wizard to create the step, you can build your own step using the explorer. Unlike with the widget wizard, you can customize your step, like adding multiple measures.

Create a Step with the SAQL Editor

You can use the SAQL Editor to customize the query of a step. SAQL query steps are more robust than steps created by widget wizards and the explorer. For example, you can write a SAQL query that queries two datasets.

Create a Static Step with Custom Values

You can use static steps in apply custom values to filters, groups, measures, sort order, and limits of other step queries. For example, you can create a toggle widget that filters case records based on the following values that you came up with: High Priority Cases, Medium Priority Cases, and Low Priority Cases. You can define the logic used to determine the case priority.

Clone a Step

Clone a step to quickly create a step based on an existing one. Cloning is useful when you can reuse an existing step with just a few tweaks.

Widget Steps in a Wave Dashboard

Some widgets require steps. A step contains a query that returns results. The widget determines how to display the results. For example, a step 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 Wave dashboard designer, most widgets have a wizard that you can use to build the widget and its step. You can also build your own step and attach it to a widget.

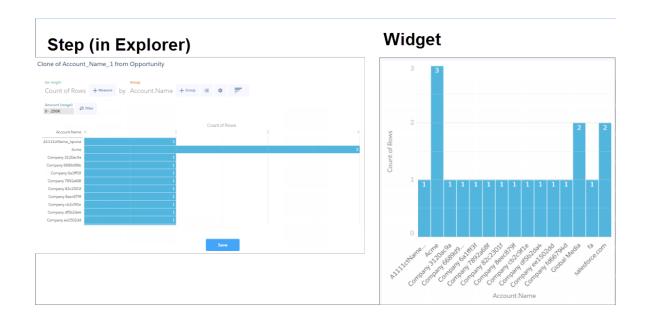
In the Wave dashboard designer, a "lens" and a "step" are similar, but not the same. They both contain queries used to calculate results. However, unlike a step, a lens includes the visualization and can be saved as a standalone asset in an app. For example, a lens can show the results of a query as a timeline chart or a compare table.

You can build a widget using one of the following types of steps.

Query Step

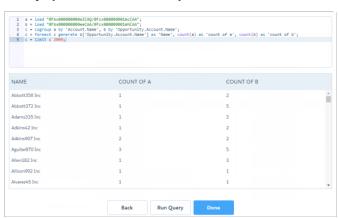
Most steps are query steps. A query step contains a query against a data source. The type of query is based on the data source. For instance, Wave can execute a SAQL query against one or more datasets or a soql query against a Salesforce object. You can create a query steps in different ways.

You can use the explorer to query 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 step that populates a chart widget.

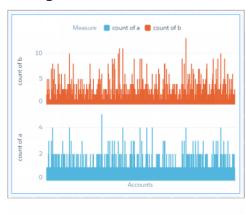


To create more robust SAQL query steps, use the SAQL editor. For example, you can create a SAQL query based on multiple datasets, which isn't possible using the explorer. This step creates a chart widget.

Step (in SAQL Editor)



Widget



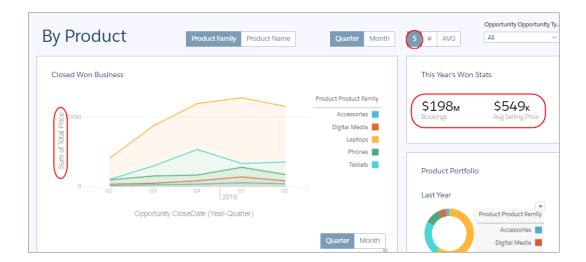
Wave creates different step types based on the widget type. By default, all steps that query a dataset, except steps used for a values table, can return up to 2,000 results. A step for a values table can return up to 100 results. To return more results, set the "limit" attribute in the SAQL query or in step definition of the dashboard JSON up to 10,000.

For more information about building SAQL queries and setting the "limit" query attribute, see the Wave Analytics SAQL Reference. For more information about step types and manually defining a soql query step in the dashboard JSON, see the Wave Analytics Dashboard JSON Reference.

Static Step

A static step is more rare. It doesn't query the data. Instead, the step 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, bind the static values to the steps 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 Static Step with Custom Values.

SEE ALSO:

Create Steps in the Wave Dashboard Designer Make the Dashboard Widgets Interactive

Create a Step with the Explorer

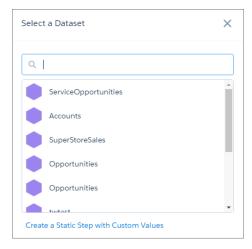
Instead of using a widget wizard to create the step, you can build your own step using the explorer. Unlike with the widget wizard, you can customize your step, like adding multiple measures.

One way to create a step with explorer is to clip an exploration to a dashboard. For information about this method, see Clip a Lens to a Dashboard. You can also create the step from the designer, which is covered here.

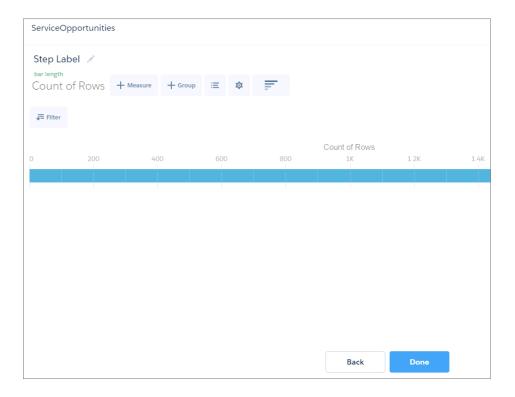
1. To open the steps panel, click an empty space in the dashboard canvas.

2. Click Create Step.

The Select a Dataset screen appears, showing datasets in the most recently used order.



3. Select the dataset that contains the data that you want to include in the step. Type in the search box if your dataset isn't easy to find. The explorer appears, showing the Count of Rows measure, by default.



- **4.** Next to the Step Label, click 🖊 .
- **5.** Enter the step label.
- **6.** Add measures that you want to show in the widget.

To add a measure, click **Measure**. To change a measure, select the measure in the explorer and then choose a new one.

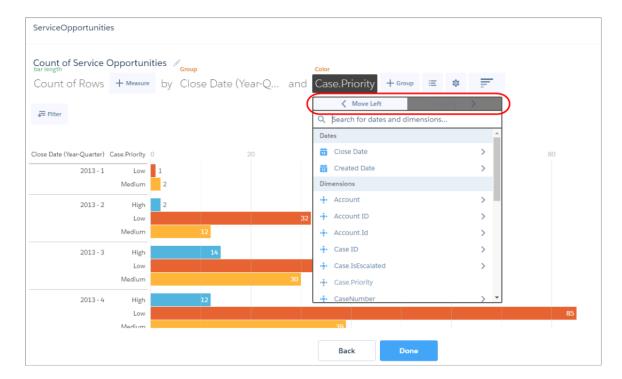
You can add as many measures as needed. If you add multiple measures, you can specify which one to show in a widget by configuring the step properties.

7.

To group the measures by a date or dimension, click



You can add multiple groups. To change a grouping, click the grouping and select a new field. To reorder the groupings, click the grouping and then click **Move Left** or **Move Right**.



8. If needed, click **Filter** to filter the results.

You can add multiple filters.

9.

To change the chart type for the step, click

This setting applies only if you don't set the chart type in the widget properties. The widget properties override this setting.

- 10. To sort the results, show the results using a logarithmic scale, or show the SAQL query, click .

 When viewing the SAQL query, you can edit it to override the query generated by the explorer. For more information about SAQL query steps, see Create a Step with the SAQL Editor.
- 11. Click Done.

Wave adds the step to the steps panel, making it available for use by other widgets in the dashboard.

- 12. Click Save.
- 13. Save the dashboard so you don't lose your changes.

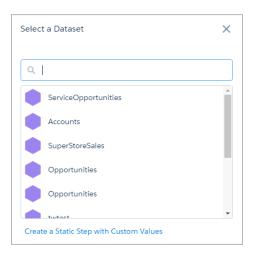
Create a Step with the SAQL Editor

You can use the SAQL Editor to customize the query of a step. SAQL query steps are more robust than steps created by widget wizards and the explorer. For example, you can write a SAQL query that queries two datasets.

1. To open the steps panel, click an empty space in the dashboard canvas.

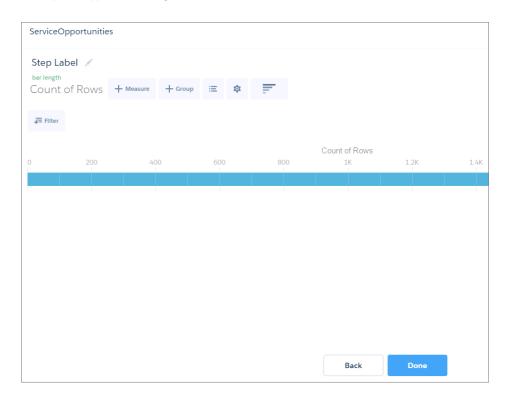
2. Click Create Step.

The Select a Dataset screen appears, showing datasets in the most recently used order.

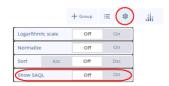


3. Select the dataset that contains the data that you want to include in your query. Type in the search box if your dataset isn't easy to find.

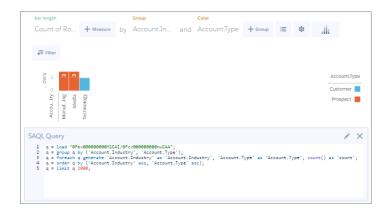
The explorer appears, showing the Count of Rows measure.



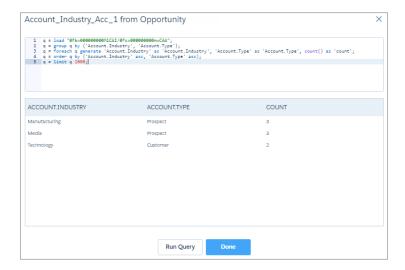
4. Click 🏚 .



5. In the Show SAQL option, click **On**. The SAQL query for the current step shows at the bottom.



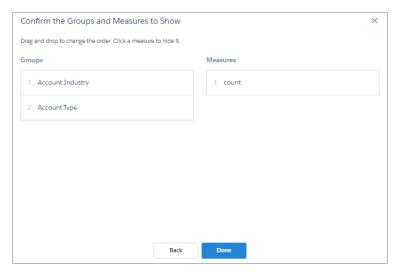
6. To edit the SAQL query, click 🖊 .



7. Edit the SAQL query, as needed.

The SAQL editor provides contextually smart auto-completion suggestions. When you make changes in the SAQL editor, the **Done** button is disabled until you select **Run Query** and the editor verifies that the query is valid. If the query is not valid, an error appears.

8. Click Done.



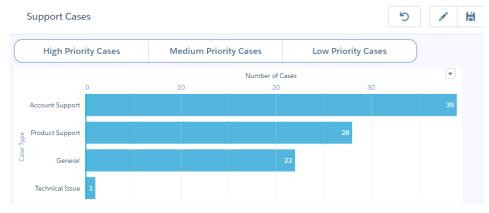
- 9. Confirm or change the order of the groupings and measures.
- **10.** To hide a measure, click it. Click it again to show it.
- 11. Click Done.

Wave adds the step to the steps panel, making it available for use by other widgets in the dashboard.

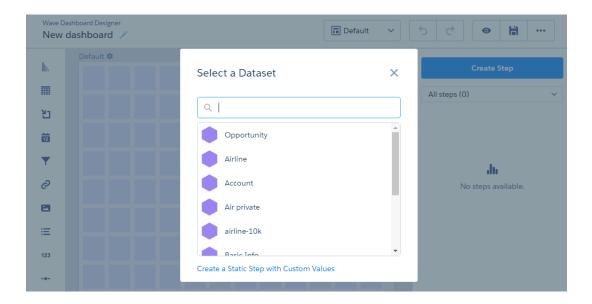
12. Save the dashboard so you don't lose your changes.

Create a Static Step with Custom Values

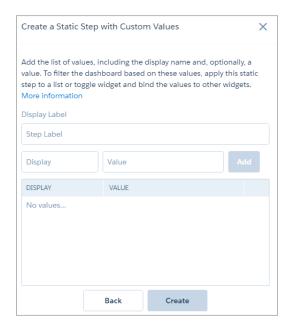
You can use static steps in apply custom values to filters, groups, measures, sort order, and limits of other step queries. For example, you can create a toggle widget that filters case records based on the following values that you came up with: High Priority Cases, Medium Priority Cases, and Low Priority Cases. You can define the logic used to determine the case priority.



1. In the step panel of the Wave dashboard designer, click **Create Step**. If you don't see the step panel, click the dashboard header to show it.



2. Click Create a Static Step with Custom Values.



- 3. Enter a step label.
- **4.** Add the list of values, including the display name and value for each.
 - Note: The display name appears in the associated widget. Values are optional, but if included, you can bind them to other steps to filter the results of other widgets in the dashboard.
- **5.** To create the static step, click **Create**. Wave adds the step to the steps panel, making it available for use by other widgets in the dashboard.
- **6.** To apply this static step to a list or toggle widget:
 - **a.** Drag the step to an empty space in the canvas.

By default, a toggle widget is added to the dashboard, showing the custom values as toggle options.

b. To change the widget and step properties, click the widget.

The Widget panel groups the widget properties into sections. So that you don't accidentally overlook some properties, notice that some sections might be collapsed, by default. To expand or collapse a section, click the arrow next to the section header.

- **c.** Set the widget properties to customize the appearance of the widget.
 - Note: In the Measure widget property, you can select a value only after you add measures for the list options. To add measures, edit the dashboard JSON and add measures to the corresponding step.
- **d.** To preview your changes to the dashboard, click .
- **e.** To save your changes to the dashboard, click 🕌 .
 - **(1) Important**: If you close the dashboard without saving it, you lose your changes.

If you applied the static step to a list or toggle widget, you can see the widget in the dashboard. But it's not tied to any other widgets. To filter the dashboard results based on selections in the list or toggle widget, bind the custom values to the steps of the other widgets. For more information about binding static steps, see the *Wave Analytics Bindings Developer Guide*.

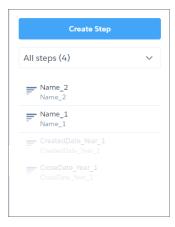
SEE ALSO:

Widget Properties for Wave Dashboards Step Properties for Wave Dashboards

Clone a Step

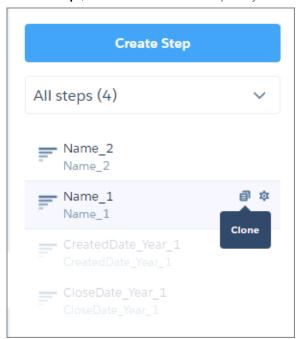
Clone a step to quickly create a step based on an existing one. Cloning is useful when you can reuse an existing step with just a few tweaks.

- Note: You can't clone a SAQL query step for a single compare table column or a static step.
- 1. To open the steps panel, click an empty space in the dashboard canvas.



Note: You can't apply grayed-out steps to widgets. But you can clone them and view their properties.

2. Select **All steps**, and then hover over the step that you want to clone.



3.

Click

It opens in exploration mode, where you can easily modify it. If the step was created using SAQL, the clone opens in the SAQL editor instead.

- **4.** Make your changes to the clone.
- 5. Click Save.

The clone is saved with "Clone of" prepended to the original step's name. Wave adds the step to the steps panel, making it available for use by other widgets in the dashboard.

- **6.** Save the dashboard so you don't lose your changes.
- Note: When you clone a step and edit its SAQL query, the resulting JSON doesn't always reflect the function applied to measures. After you run and save the SAQL, the function is changed to count. For example, if the original step 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.

Widget Properties for Wave Dashboards

Widget properties define how widgets appear in the Wave 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 Wave dashboard designer.

Wave groups widget properties into sections.

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 step to have at least one grouping. Wave pivots the table on the last defined grouping. Disabled, by default.
Show Explore Icon	Allow dashboard viewers to click a link in the widget to open the underlying step in explorer.
	Note: This option only applies widgets based on steps in compact form. You can't explore widgets that are built on SAQL form steps.
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.
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.
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 step. You can also use this option to not show the measure in the widget.
Shorten number	Display shortened numbers, like 60 K instead of 60,499. Shortened numbers are rounded.
Text	Widget label.
Link To	Link to a dashboard, lens, or step.
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 step query. You can insert additional characters directly into the editor box to format the display. For instance, you can add parentheses to enclose the measure value as shown here.

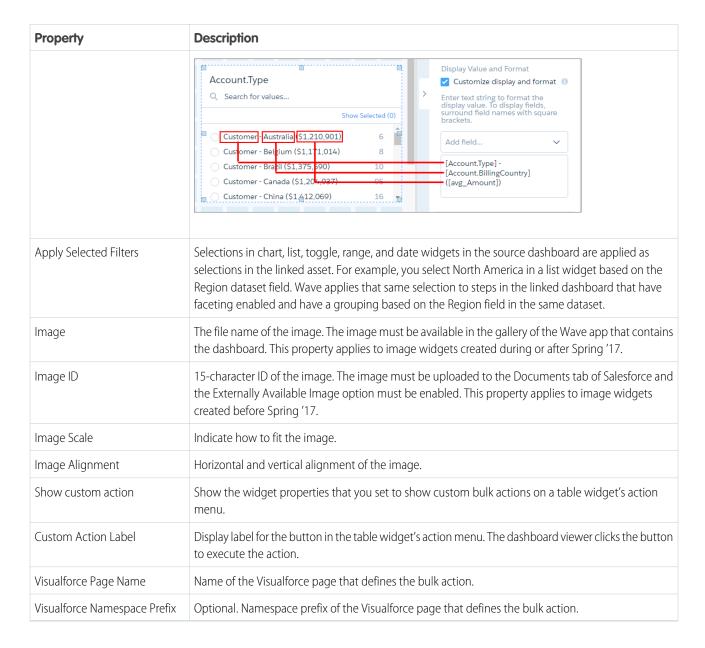


Chart Details

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.
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.

Property	Description
	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. 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.

Property	Description
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.
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.

X-Axis, Y-Axis, Left Axis, Center 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.

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.

BreakPoints

Property	Description
Min	Minimum value of the low color band.
Medium	The value separating the low and medium colors.
High	The value separating the medium and high colors.
Max	Maximum value of the high color band.

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.

Trellis

Property	Description
Show as trellis	Create separate graphs based on groupings.
Туре	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.

Property	Description
	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.

Text Style

Property	Description
Text Color	Color of the text.
Text 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.

Background Image

Property	Description
Image	The file name of the image. The image must be available in the gallery of the Wave 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.

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.

Step Properties for Wave Dashboards

Step 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 Wave dashboard. The properties apply to dashboards created in the Wave dashboard designer only.

Property	Description
Display Label	Step label. Use to provide a descriptive label for the step.
Facet	Enables this step to facet and be faceted by other steps. Faceting is when a selection in a widget filters other steps in the dashboard. By default, steps from the same dataset are faceted. To facet steps from different datasets, connect the data sources.
	Note: If this step's query contains a cogrouping, it facets others steps based on the first specified dataset only. Steps based on other datasets specified in the cogrouping aren't faceted.
Apply global filters	Apply the global filters to this step.
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 Filters step property. If a selection is required but not set, Wave selects the first value in the list.
Initial Filters	Show the initial filters that are applied when the dashboard opens. You must enable selections in the Selection Type step property before you can set an initial filter. To set the initial filters for steps,
	click and then select Pick Initial Filters . Any selections that you make on the widget filter the underlying step and immediately filter other widgets with faceted steps.

Make the Dashboard Widgets Interactive

Wave 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.

IN THIS SECTION:

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 steps. When the underlying steps are faceted, selections made on chart widgets or filter selection widgets (date, list, range, and toggle) automatically filter other widgets in the dashboard. For example, when you select a rep in the Sales Rep list widget, Wave applies the same filter to the chart and table widgets in the dashboard.

Dynamically Modify the Query and Widgets with Bindings

Compared to facets, bindings are more programmatic, and provide more ways to specify interactions. Bindings can dynamically modify the query of a step or change the display of a widget based on what happens in another widget. For example, you can use bindings to change the color of the metric in a number widget based on whether its value is high, medium, or low. You can also use a binding to change the measure in a step query based on a selection in a static step. Each binding is unidirectional, meaning it affects other steps, but other steps don't affect it.

SEE ALSO:

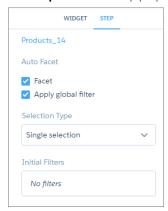
Widget Steps in a Wave Dashboard

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 steps. When the underlying steps are faceted, selections made on chart widgets or filter selection widgets (date, list, range, and toggle) automatically filter other widgets in the dashboard. For example, when you select a rep in the Sales Rep list widget, Wave applies the same filter to the chart and table widgets in the dashboard.

By default, steps from the same dataset are faceted together. To facet steps from different datasets, connect the datasets. Steps with a cogrouping that query multiple datasets only facet steps based on the first dataset specified in the datasets field.

- 1. To enable the faceting option for a step, complete the following tasks.
 - **a.** Click / to edit the dashboard.
 - **b.** Select the widget that uses the step. The panel shows the widget properties.
 - **c.** Click **Step** to view the step properties.



d. Select Facet.



- 2. To enable a step in SAQL form to receive facets, you must also set the autofilter attribute to true in the dashboard JSON. For more information about the autofilter attribute for a step definition, see the Wave Analytics Dashboard JSON Reference.
- 3. To facet steps between different data sources—like different datasets or static steps—you must also connect them.

IN THIS SECTION:

Configure Cross-Dataset Faceting with Connected Data Sources

By default, steps from the same dataset are faceted—automatically filtering each other. To facet steps from different datasets or even static steps in a Wave designer dashboard, connect them using common fields between them.

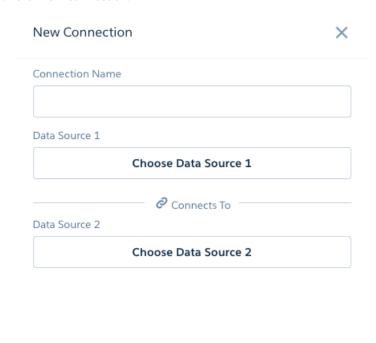
SEE ALSO:

Add Selection-Based Filter Widgets to Enable Users to Filter the Results

Configure Cross-Dataset Faceting with Connected Data Sources

By default, steps from the same dataset are faceted—automatically filtering each other. To facet steps from different datasets or even static steps in a Wave designer dashboard, connect them using common fields between them.

- Note: You can't connect static steps that specify date and 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.

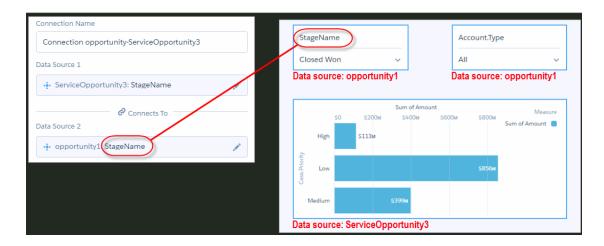
Back

- 5. Click **Choose Data Source 1** to select the first data source. For each data source, select a dataset or a static step used in the dashboard.
- **6.** Choose one field from the data source so that Wave can match records between the specified sources.

Save

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.

Dynamically Modify the Query and Widgets with Bindings

Compared to facets, bindings are more programmatic, and provide more ways to specify interactions. Bindings can dynamically modify the query of a step or change the display of a widget based on what happens in another widget. For example, you can use bindings to change the color of the metric in a number widget based on whether its value is high, medium, or low. You can also use a binding to change the measure in a step query based on a selection in a static step. Each binding is unidirectional, meaning it affects other steps, but other steps don't affect it.

For more information about bindings, see Wave Analytics Bindings Developer Guide

Set Initial Filter Selections in the Dashboard

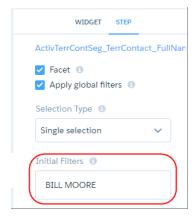
If the Wave dashboard contains filter selection widgets or charts, you can set initial selections that are applied when the dashboard first opens. For example, you can use a selection to filter the dashboard to show only opportunities owned by the logged in user. While viewing the dashboard, the user can change the selections to filter the results differently or remove the filters altogether.

You must set the initial selection for widgets that require it. If you don't, Wave selects the first value when the dashboard opens.

Wave ignores the selection if it can't find the initial selection value in the step results when opening the dashboard. This issue can occur when the underlying step for the filter selection widget has a saql step type and not all results are returned. (This issue doesn't happen for other step types.) For more information about query limits for a saql step type, see saql Step Type Properties.

- 1. While editing the dashboard, click ___ and then select **Pick Initial Filters**. Wave previews the dashboard.
- 2. Make selections in chart, list, toggle, date, and range widgets.
- 3. Click Done.

Wave saves the selections as start values for the widget steps. You can view the initial filter selections in the step properties.



IN THIS SECTION:

Dynamically Set Initial Filter Selections Based on the Logged In User

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.

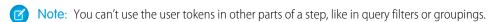
SEE ALSO:

Add Selection-Based Filter Widgets to Enable Users to Filter the Results

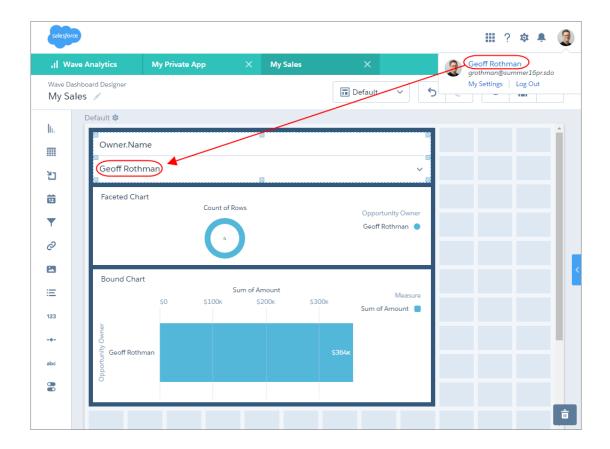
Dynamically Set Initial Filter Selections Based on the Logged In User

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, Wave retrieves the values of these tokens from Salesforce environment variables—these values don't come from datasets.



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 step (Owner_Name_1). This selection filters the following charts:

- Donut chart because its step (Owner Name 3) is faceted with step Owner Name 1.
- Bar chart because its step (Owner Username 3) contains a binding based on step 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 steps.

```
"isGlobal": false,
    "query": {
        "measures": [
                "count",
            ]
        ],
        "groups": [
           "Owner.Name"
    },
    "selectMode": "multi",
    "start": [
        "!{User.Name}"
    ],
    "type": "aggregateflex",
    "useGlobal": false,
    "visualizationParameters": {
       "options": {}
},
"Owner Name 3": {
    "datasets": [
            "id": "0FbB0000000pNNKAY",
           "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": "0FbB0000000pNNKAY",
                        "label": "Opportunities",
                        "name": "opportunity",
                       "url": "/services/data/v38.0/wave/datasets/0FbB0000000pNNKAY"
                    }
                ],
                "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": "0FbB0000000pNNKAY",
        "label": "Opportunities",
        "name": "opportunity",
```

```
"url": "/services/data/v38.0/wave/datasets/0FbB00000000pNNKAY"
}
]
```

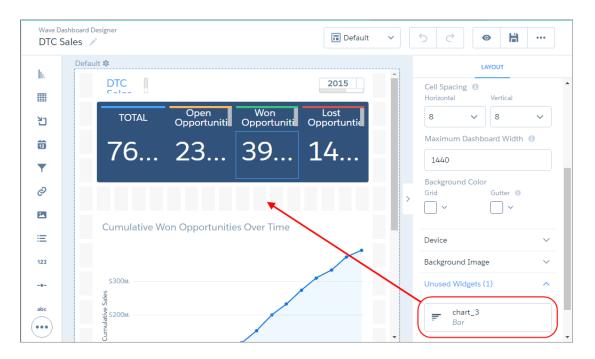
For information about editing the dashboard JSON, see the Wave Analytics Dashboard JSON Reference.

Generate Unique Wave Dashboard Layouts for Different Devices

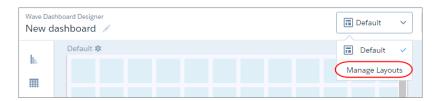
After you add widgets to the dashboard, optimize the layout for each device on which the dashboards can be viewed. For example, you can remove widgets from a mobile phone layout to reduce the dashboard size for the smaller screen. You can also move widgets around in one layout and it doesn't affect the other layouts.

When you generate a new layout, Wave adds the widgets from the currently selected layout to the new layout. This step is important because each layout can contain different widgets. Before you get started, select the layout that contains the widgets that you want to include in your new layout.

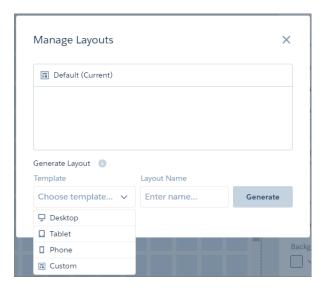
If you add a widget to a layout, Wave doesn't add it to other layouts. To add it, open the other layout and drag the widget from the "Unused Widgets" section in the layout properties.



- Note: Although a mobile layout can show all charts, some charts are not supported on mobile devices. For more information, see Wave Analytics mobile limitations.
- 1. With the dashboard open in the Wave dashboard designer, from the Layouts menu, select Manage Layouts.



2. Select the layout template, and enter a unique layout name.



Each layout template comes with predefined layout properties that you can change.

3. Click Generate.

The layout properties appear for the new layout. If needed, click v to show the Layout panel.



4. 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.

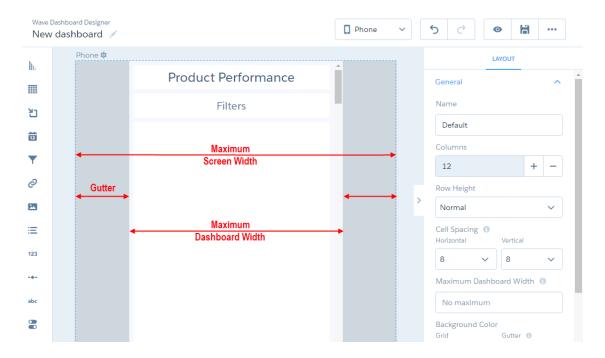
Device

Specify information about the devices that can use this layout. For more information about how Wave uses these properties to choose the right layout, see Wave's Rules for Choosing a Layout for a Device.

Background Image

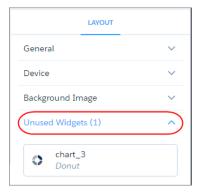
To apply a background image to the entire dashboard when this layout is used, enter the details about the background image. You don't have to include a background image.

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.

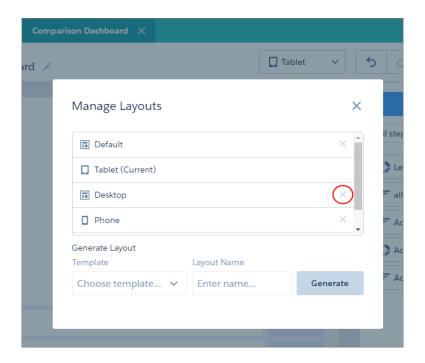


- **5.** Rearrange the widgets, if needed.
- 6. To hide a widget from the layout, select the widget, and click .

 If you hide a widget, it appears in the Unused Widgets section of the layout properties.



- 7. To add an unused widget to the layout, drag the widget from the Unused Widgets section to the canvas.
- **8.** To add a new widget, drag the widget from the widget toolbar to the canvas.
 - Note: Wave adds the widget only to the current layout.
- **9.** To save your layout changes and the dashboard, click 🗎 .
- 10. To delete a layout, in the Layouts menu, click **Manage Layouts**, and then click x next to the layout.



IN THIS SECTION:

Layout Properties for Wave 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 Wave dashboard designer.

Wave's Rules for Choosing a Layout for a Device

If multiple layouts are defined for a dashboard, Wave chooses the optimal layout when displaying the dashboard on a device. To determine the optimal layout, Wave uses the device properties specified for each layout. By understanding Wave's logic, you can set the properties appropriately.

Layout Properties for Wave 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 Wave 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 Wave designer grid. Increase the number of columns to increase the precision in the layout.
Row Height	Height of the rows in the Wave designer grid. Values are Normal and Fine . When using the Fine option, the vertical spacing is set to 0 and can't be changed. Wave determines the default sizes of

Property	Description
	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 Wave designer grid.
Maximum Dashboard Width	Maximum width (in pixels) that the dashboard can use. If needed, Wave 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 theWave designer grid.
	Gutter Color of the gutter that appears when the device's screen is wider than the dashboard's maximum width.

Device Properties

Wave 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.

Background Image Properties

Property	Description
Image	The file name of the image. The image must be available in the gallery of the Wave 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.

Wave's Rules for Choosing a Layout for a Device

If multiple layouts are defined for a dashboard, Wave chooses the optimal layout when displaying the dashboard on a device. To determine the optimal layout, Wave uses the device properties specified for each layout. By understanding Wave's logic, you can set the properties appropriately.

Wave 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.

Build a Dashboard with the Classic Designer

The classic designer is the legacy designer that enables you to build basic Wave dashboards.



Note: Although you can build a dashboard in the classic designer, Salesforce.com recommends using the new Wave dashboard designer, which supports more features and is easier to use. The new designer replaces the classic designer.

IN THIS SECTION:

1. Create the Dashboard in Classic Designer

When you create a Wave dashboard, you can apply a prepackaged layout or start with a blank canvas to create your own layout. To build the dashboard, you can use the classic designer or the new Wave dashboard designer. In this example, we'll use the classic designer.

2. Add Key Performance Indicators to the Wave Dashboard

Sometimes, a single number can provide a quick and critical measurement of the business. For example, the total number of open support cases and number of new support cases open this week are key metrics that a service manager would find useful. Add Number widgets to provide key indicators to a dashboard.

3. Add Visualizations to the Wave Dashboard

Add charts to visualize and interact with the data to tell a story about it. Each chart is based on the underlying query in a lens.

4. Facet Widgets to See Data from Multiple Angles

Widgets that come from the same dataset are faceted by default. Faceting means that when you filter one widget, all the related widgets in the Wave dashboard are filtered in the same way. You can turn faceting on or off for each widget.

5. Add Images to the Wave Dashboard

Add images to your dashboard to help your audience with context and to make the dashboard more fun to read. For example, if your dashboard is designed to help track sales of multiple products, consider distinguishing the products by their logos.

6. Link to Lenses and Other Dashboards

Use links to tie a Wave dashboard to related lenses and dashboards. This widget is useful when you want to break up analysis across multiple assets, rather than cramming it all into a single dashboard. It's also useful for creating pathways to analyze your data.

7. See Details About a Widget's Lens

To ensure that a widget in a Wave dashboard is showing the right data, review details about the widget's lens.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

8. Filter or Remove Clipped Lenses from the Wave Dashboard

Curate the clipped lenses in a dashboard to reduce clutter. You can filter the list of clipped lenses to see which ones aren't used and then remove them from your dashboard.

SEE ALSO:

Convert Your Dashboard to the Wave Dashboard Designer

What Is a Dashboard?

Best Practices for Building Wave Dashboards

Keyboard Shortcuts For Wave Dashboards and Lenses

Key Differences Between the Classic Designer and Wave Dashboard Designer

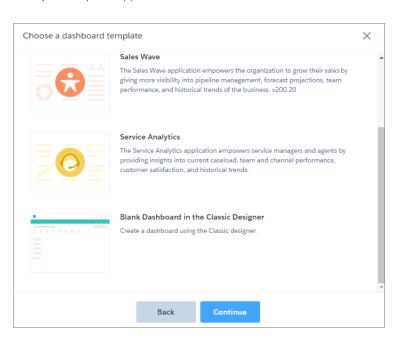
Build a Dashboard with the Wave Dashboard Designer

Create the Dashboard in Classic Designer

When you create a Wave dashboard, you can apply a prepackaged layout or start with a blank canvas to create your own layout. To build the dashboard, you can use the classic designer or the new Wave dashboard designer. In this example, we'll use the classic designer.

This topic is about creating dashboards in Wave Analytics. For information on creating dashboards for reports, review the article Create a Salesforce Classic Dashboard.

From the home page, click Create > Dashboard.
 The layout templates appear.



USER PERMISSIONS

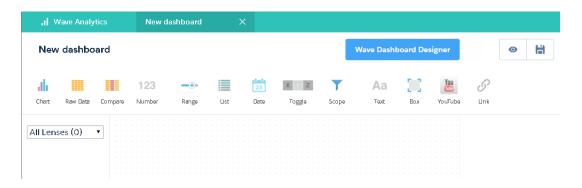
To create a dashboard:

 "Create and Edit Wave Analytics Dashboards"

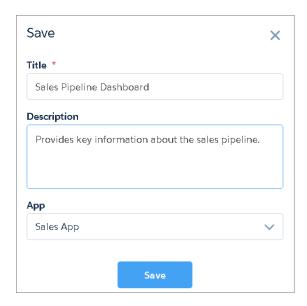
- 2. Choose a layout template, or select an option to create a blank dashboard in the classic designer or Wave dashboard designer. The layout templates create dashboards in the Wave dashboard designer. In this case, we'll create a blank dashboard using the classic designer.
- **3.** Select **Blank Dashboard in the Classic Designer**, and then click **Continue**. If you haven't previously declined to use the Wave dashboard designer, the "Ready for Wave Dashboard Designer?" pop-up appears.

4. Select Not Now.

The classic designer opens.



5. Click 🛗 to save the dashboard to an app.

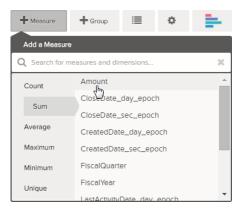


6. Click Save.

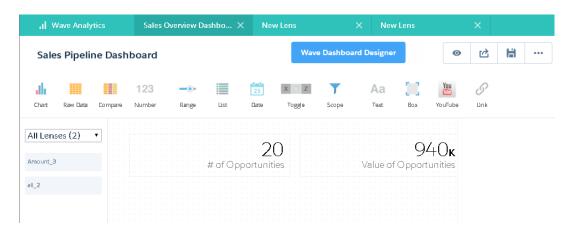
Add Key Performance Indicators to the Wave Dashboard

Sometimes, a single number can provide a quick and critical measurement of the business. For example, the total number of open support cases and number of new support cases open this week are key metrics that a service manager would find useful. Add Number widgets to provide key indicators to a dashboard.

- To add a key performance indicator to a classic designer dashboard, open the Opportunities dataset, and click to calculate the total value of all opportunities in the pipeline.
 - **a.** In the measure picker, select Sum from the left column.
 - **b.** Pick a dataset field from the right column that you would like to sum. In this case, select Amount.



- Note: Don't add any groupings. A number, by definition, consists of just one group. Later, you will add a chart that shows the value for different groupings.
- 2. While we're here, let's add another kind of measure to bring more context to the story that we're telling with this dashboard. For example, you can select Count of iswon to get the number of successfully closed deals.
- **3.** Click **3.** to clip this lens to the classic designer.
 - 1 Tip: If you have multiple dashboards, the lens is clipped to whichever dashboard you had open last.
- 4. In the classic designer, click the number widget (123) to create an empty number widget, and then select the first clipped lens that contains a sum of amount.
 - The number that appears in the number widget is the first measure (on the far left) that you specified in the lens.
- **5.** To choose a different measure from the lens or configure other useful properties, select the Number widget and set the properties in the Widget Properties section.
 - **a.** Give the widget a title. The default title, which is taken from the field name, is sometimes too cryptic for the audience.
 - **b.** For a large number, select **Shorten number** to reduce the length of the display. Doing so turns a number such as 2, 148, 352 into the more readable 2M.
- **6.** Using the same procedure, add another number widget based on the second clipped lens that provides a count of won deals. Your dashboard now contains the two number widgets.



Add Visualizations to the Wave Dashboard

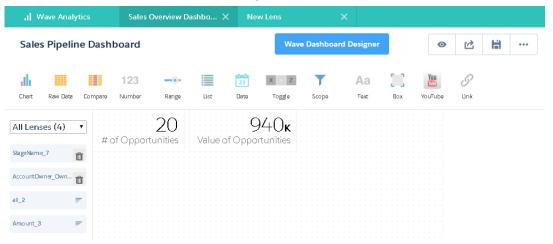
Add charts to visualize and interact with the data to tell a story about it. Each chart is based on the underlying query in a lens.

In the next steps, we'll create two interactive charts in the classic designer dashboard that let the audience analyze the sales pipeline. The first will analyze the pipeline by account owner. The second will analyze it by stage. Both charts are based on an Opportunities lens, which contains a count of the number of opportunities and no groupings.

- Verify that the Sales Pipeline dashboard is open.
 You must have the dashboard open to clip lenses to it.
 - Note: If no dashboard is open, Wave creates a dashboard.
- 2. From the home page, open the Opportunities lens.
- **3.** Group the data by Account Owner.

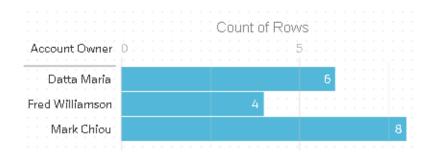


- **4.** Click **X** to clip the lens to the dashboard.
- Change the Account Owner grouping to a Stage Name grouping.To change a grouping, click the grouping and then select the new grouping.
- **6.** Clip the lens to the dashboard. Now that you have both lenses clipped to the dashboard, let's build charts based on them.
- 7. View the Sales Pipeline dashboard in the classic designer.

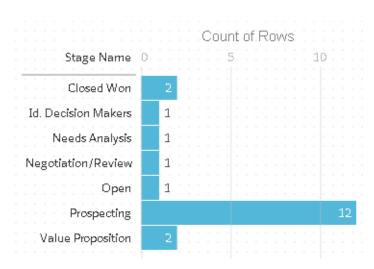


8. In the classic designer, click the chart widget, and then select the first clipped lens that contains number of opportunities grouped by account owner.

The chart shows of the results from the underlying query in the clipped lens.

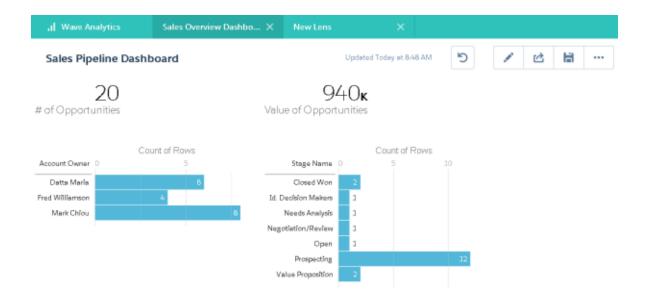


9. Create another chart based on the other clipped lens—the one grouped by stage name. The new chart widget shows how many opportunities are in each stage.



10. Click **o** to view the dashboard in presentation mode.

Now you're looking at the audience's view of your dashboard, and you're ready to ask some questions.

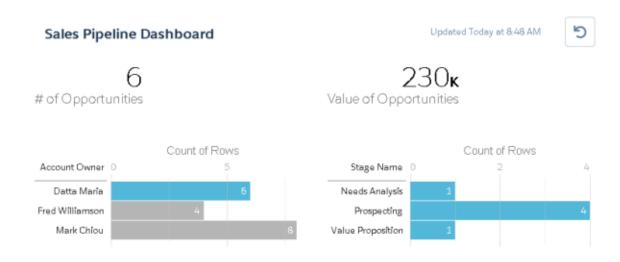


Facet Widgets to See Data from Multiple Angles

Widgets that come from the same dataset are faceted by default. Faceting means that when you filter one widget, all the related widgets in the Wave dashboard are filtered in the same way. You can turn faceting on or off for each widget.

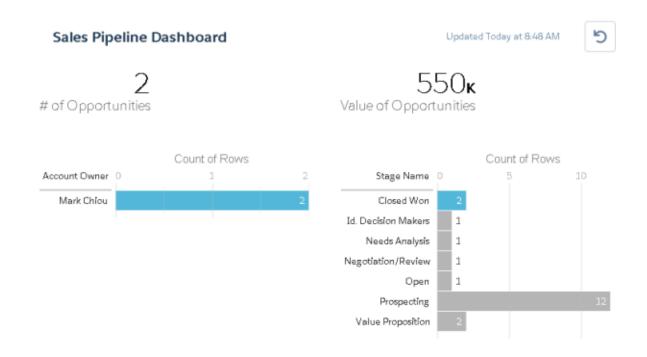
Now we let's look at how faceting works with the number and chart widgets that we added to our classic designer dashboard.

1. In the Sales Pipeline dashboard, click the first bar in the chart that shows the number of opportunities grouped by account owner. Notice that first bar is now highlighted and the results of the other widgets changed. Because of faceting, the number widgets and the other chart now show the results for the selected account owner only. Each of your widgets shows the same set of data from a different angle. Widgets created from the same dataset are faceted, by default.



- **2.** Click the first bar again to remove the filter based on the account owner. Notice that the results are now shown for all account owners.
- 3. In the chart that shows the number of opportunities grouped by stage name, click the first bar.

The Account Owners chart changes to show the values of the deals that each salesperson has in the selected stage. Because Mark is the only one that has deals in this stage, only he appears in the Account Owners chart.

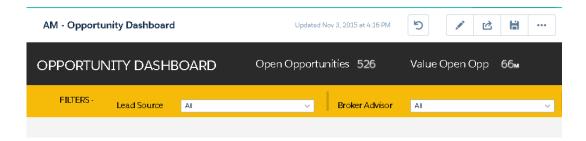


- **4.** To disable faceting for a particular widget, clear the **Facet this lens** option in the lens properties.
- Tip: You can use bindings to facet widgets created from different datasets. For more information, see the Wave Analytics Dashboard ISON Reference

Add Images to the Wave Dashboard

Add images to your dashboard to help your audience with context and to make the dashboard more fun to read. For example, if your dashboard is designed to help track sales of multiple products, consider distinguishing the products by their logos.

- 1. Upload the image that you want to use to Documents and select the "Externally Available Image" option. If you don't select the option, the image will not appear in the dashboard.
- 2. Copy the record ID for the document in the URL in the address bar.
- 3. In the dashboard, click the box widget and paste the record ID into the Image Document ID field in widget properties.
- Note: You can also use the box widget to shade a section of the dashboard or add a background color to the dashboard. For example, you can use colored areas to highlight key metrics and filters.



To do this, don't include an image and set the background color in the **Background Color** field of the widget properties.

SEE ALSO:

Uploading and Replacing Documents

Link to Lenses and Other Dashboards

Use links to tie a Wave dashboard to related lenses and dashboards. This widget is useful when you want to break up analysis across multiple assets, rather than cramming it all into a single dashboard. It's also useful for creating pathways to analyze your data.

- 1. In the dashboard, click the link icon.
- 2. In the widget properties, specify the link properties, such as the ones listed here.

Specify whether the link widget leads to a saved lens, saved dashboard, or new lens.

Destination

Specify the lens or dashboard asset to link to.



Note: If you try to create a Link widget in a dashboard that has not been saved in Spring '16 and you don't see the asset in the picklist, open the asset first, and then look for the asset in the picklist.

Include Dashboard State

When enabled, the selections made in widgets with faceted steps are transferred to the linked lens or dashboard. When transferring dashboard states, Wave Analytics verifies that the destination has lenses that are compatible with the lenses from the source dashboard.



Note: Wave doesn't transfer a dashboard state in the following cases.

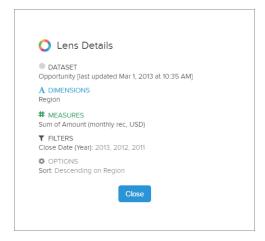
- The source lens isn't faceted.
- The destination lens uses a different dataset than the source lens.
- When linking to another dashboard, the destination lens uses different groupings than the source lens. If the link destination is a lens, then groupings don't prevent dashboard state from being transferred.

Text

The link label that you click to access the linked asset.

See Details About a Widget's Lens

To ensure that a widget in a Wave dashboard is showing the right data, review details about the widget's lens.



Lens details display.

2. Review the dataset and applied dimensions, measures, filters, and other details about the widget's lens.

Filter or Remove Clipped Lenses from the Wave Dashboard

Curate the clipped lenses in a dashboard to reduce clutter. You can filter the list of clipped lenses to see which ones aren't used and then remove them from your dashboard.

1. From the clipped lens menu, filter by All Lenses, Active Lenses, or Unused Lenses.



2. Optionally, if a lens isn't used, then you can remove it from your dashboard by clicking in.

If the lens is saved to Wave Analytics, then it's still available and can be opened from Lenses after being removed from your dashboard.

Keyboard Shortcuts For Wave Dashboards and Lenses

You can do some basic actions from your keyboard. The dashboard designer shortcuts vary based on whether you are using the classic designer or Wave dashboard designer.

Lens Keyboard Shortcut	Description
I	Clip to designer
s	Save
n	Clone in new tab
d	Delete
Ctrl+E (Windows) Cmd+E (Mac OS)	Open Expert Editor Mode (JSON editor)

Classic Designer Keyboard Shortcut	Description
Arrow keys	Move the widget in 10-pixel increments.
Ctrl+Arrow keys (Windows) Cmd+Arrow keys (Mac OS)	Move the widget in 1-pixel increments.
Shift+Arrow keys	Resize the widget by 10 pixels.
Ctrl+Shift+Arrow keys (Windows) Cmd+Shift+Arrow keys (Mac OS)	Resize the widget by 1 pixel.
x	Delete the widget from the canvas.
[Put the widget beneath other widgets.
1	Put the widget on top of other widgets.

Wave Dashboard Designer Keyboard Shortcut	Description
Arrow keys	Move the widget.
C	Share the dashboard.
d	Delete the dashboard.
e	Toggle between previewing or editing the dashboard.
q	Show clipped lenses or create a lens in the dashboard.
r	Reset the dashboard back to the last saved state.
S	Save the dashboard.
X	Redo the change.
Z	Undo the latest change.
Delete	Delete the selected widgets in the dashboard.

Wave Dashboard Designer Keyboard Shortcut	Description
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)	Select multiple widgets.
Cmd+click (Mac OS)	
Ctrl+A (Windows)	Select all widgets in the dashboard canvas.
Cmd+A (Mac OS)	

JSON Editor Keyboard Shortcut	Description
Ctrl+E (Windows) Cmd+E (Mac OS)	View dashboard with changes to JSON.
Ctrl+3 (Windows) Cmd+3 (Mac OS)	Disregard changes and load the original JSON.
Ctrl+F (Windows) Cmd+F (Mac OS)	Search (RegExp, case-sensitive, or whole-word searches available).
Ctrl+F twice (Windows) Cmd+F twice (Mac OS)	Search and replace (RegExp, case-sensitive, or whole-word searches available).

Best Practices for Building Wave Dashboards

Follow best practices to design and build useful, effective Wave dashboards, while minimizing rework and addressing potential gaps.

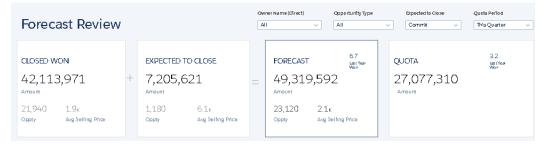
- Before you build the dashboard, take into account the following design best practices:
 - Sketch your dashboard on paper or a whiteboard before you start building.
 - Keep in mind that a chart in Wave Analytics is primarily a way to ask questions, not a way to illustrate a conclusion. A good dashboard invites the audience to drill down and seek ever more focused and useful information.
 - 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, actionable charts near the top left, and place charts with supporting information lower.
 - 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.

Place information that can be read at a glance high and to the left, and place the details lower. For example, place numbers that
display a single measure, such as revenue for the current quarter, high and to the left.

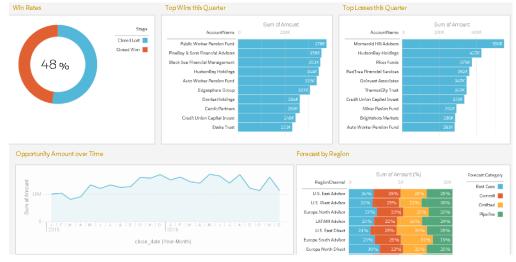
Highlight facts that the audience wants to know at a glance.



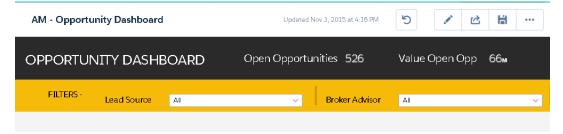
- Choose chart types based on the characteristics of the data, not for look or variety. For example, if most of your charts display value changes over time, it's OK if they're all line graphs.
- If a chart seems to need a lengthy caption or title, reconsider whether the chart is doing its job. Well-chosen 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.

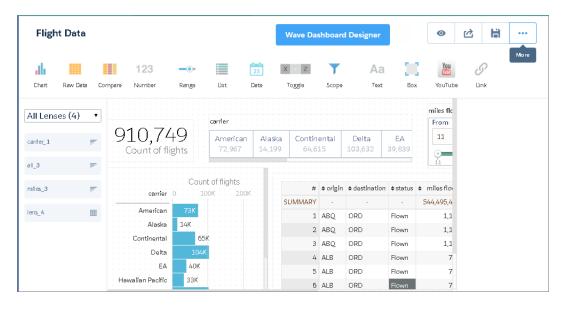
SEE ALSO:

Build a Dashboard with the Wave Dashboard Designer Build a Dashboard with the Classic Designer

Convert Your Dashboard to the Wave Dashboard Designer

To take advantage of its new features, convert dashboards created in the classic designer to the Wave dashboard designer.

When editing the dashboard, click the **Wave Dashboard Designer** button to convert it to the new designer.



When you convert a dashboard to the new designer, Wave makes a copy of the dashboard. Before you delete the original version of the dashboard, know that you can't open the new version in the classic designer. Any changes that you make in the new designer can't be converted back to the classic designer.

IN THIS SECTION:

Considerations Before Converting to the Wave Dashboard Designer

Although dashboards created in the classic designer and Wave designer have a similar look and feel, they use different runtime engines and have different JSON syntax. When you convert a dashboard to the Wave designer, certain elements of your dashboard might not convert correctly. You might need to make some manual changes to preserve the original behavior.

Considerations Before Converting to the Wave Dashboard Designer

Although dashboards created in the classic designer and Wave designer have a similar look and feel, they use different runtime engines and have different JSON syntax. When you convert a dashboard to the Wave designer, certain elements of your dashboard might not convert correctly. You might need to make some manual changes to preserve the original behavior.

Widgets

The following widget issues can arise when you convert a classic dashboard to Wave designer.

- Wave designer resizes and rearranges widgets to fit them into the 12-column grid and to ensure that no widgets (except containers) overlap. After conversion, you can use the snap-to-grid feature in Wave designer to quickly resize and rearrange widgets. You can also add more columns to the grid to get better precision when laying out the widgets.
- Wave designer converts the following chart types to bar charts because it doesn't support them: calendar heatmap, parallel coordinates, horizontal dot plot, vertical dot plot, and matrix.
- Wave designer removes YouTube and Box widgets because it doesn't support them.
- Wave designer drops the setting for the Allow Filtering by step property. After the conversion, you can reapply the setting to the Selection Type widget property. The name and location of the property is different in Wave designer.
 - Note: If the step contains no measures, you still must specify "measures": []

Steps

The following step issues can arise when you convert a classic dashboard to Wave designer.

• If you specify the compare table step query in SAQL form, you can't specify custom labels for compare table columns. To specify custom labels, the step must be in compact form. You can specify the column label in the header attribute, as shown here.

- The step breaks in the following cases:
 - A field in a static step doesn't have a consistent format across all rows in the step. For instance, if a "value" field is defined as an array, then every row in the step must define "value" as an array.

For example, this step is valid:

```
"value": { "Type": "sum('SplitAmount')" }
},
{
   "display": "Percentage",
   "value": { "Type": "avg('SplitAmount')" }
}
```

The following step is not valid.

- A grain step contains SAQL. Wave designer doesn't support SAQL in grain steps. To fix this issue, manually convert the grain step in the dashboard JSON to an aggregateflex step using this process. Wave designer supports SAQL queries in the other types of steps, like aggregateflex.
- A multi step for a compare table contains a column-level SAQL query. While converting the dashboard, Wave designer converts multi steps to aggregate steps because it doesn't support multi steps. However, it can't convert multi steps that have a column-level SAQL query. To fix this issue, re-create the step as shown here.



A multi step contains bindings. During conversion, the Wave designer generates the measures and groupings sections in the dashboard JSON using the metadata in the columns definitions of the original multi step. However, if the metadata contains bindings, Wave designer can't resolve them to determine what values to add to the new step. In this case, Wave designer adds the step query, but not the values for the measures and groupings sections. To fix the step, determine which measures and groupings are being defined in the bindings specified in the query, and add them to the measures and groupings sections.



Note: If the step contains no measures, you still must specify "measures": [].

Mobile Layouts

The following mobile layout issues can arise when you convert a classic dashboard to Wave designer.

- Mobile layouts created in classic designer dashboards are dropped when the dashboard is converted to Wave designer. To fix this
 issue, re-create the layouts again using the layouts user interface in the Wave designer.
- Unlike classic designer dashboards, Wave designer dashboards don't support paging--you can only use one page.

Start Values and Selections

The following issues with start values and selections can occur when you convert a classic dashboard to Wave designer.

- When invoking an embedded dashboard, you can specify criteria to be passed as selections. If you set a selection value for a step and that value doesn't exist in the step's result set, the selection isn't applied. This behavior is different from classic designer, which blindly applies this invalid selection, resulting in no data returned.
- If a step returns more than 2,000 results, a configured start value for the step or a manually coded selection value for an embedded dashboard might not be set. Specifically, if the start or selection value isn't one of the initial 2,000 results returned by the step, the value isn't found and, therefore, not set. To reduce the chance of this happening, increase the query limit up to 10,000 in the step definition. For more information about the limit parameter in a step query, see the *Wave Analytics SAQL Reference*.

Bindings

The following binding issues can arise when you convert a classic dashboard to Wave designer. If a binding doesn't work after the conversion, the dashboard doesn't render. If this issue occurs, contact Salesforce Customer Support.

- Wave designer dashboards run on the new runtime engine, which has a new bindings syntax. When converted to Wave designer,
 the dashboard bindings are not updated with the new syntax. The new runtime engine can process a subset of the original bindings
 syntax. It can process bindings that set measure selections, groupings, filters, sort order, or limits based on string or numeric values
 in steps. Other bindings might not work after conversion. To resolve invalid bindings, rewrite the bindings using the new bindings
 syntax.
- The new runtime engine is stricter when evaluating bindings defined using the old bindings syntax. If a binding isn't well formed, the step returns an error.

For example, an error occurs if the binding contains redundant parentheses. The outermost binding function can't be enclosed in parentheses, as shown here.

```
"measures": "{{ (value(field(selection(Select_Graph),'compact'))) }}"
```

The following error occurs in this case.

```
"(value(field(selecti -^ Expecting 'FUNCTION', got '('."
```

To fix this issue, remove the outermost set of parentheses, as illustrated here.

```
"measures": "{{ value(field(selection(Select_Graph),'compact')) }}"
```

• When defining date ranges in a static step, use a double array, as shown here.

Here's the binding that references the static step.

Because the selection binding is defined inside an array, the following example throws an error.

Here's the binding that references the static step above.

SEE ALSO:

Convert Your Dashboard to the Wave Dashboard Designer

Trending Data in Wave

By creating trending Wave dashboards that track key metrics over time, you can make sense of your growing, ever-changing datasets. Trending data in Wave helps you discover hidden insights and allows you to share them with colleagues.

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.

IN THIS SECTION:

Trend Wave Dashboards

To trend data in Wave, you trend a report that's collecting that data.

Share Trending Dashboards

You can share a trended dashboard just as you can with any Wave dashboard.

Limits and Limitations for Trending Data in Wave

To prevent the performance of your org from degrading, certain limitations are placed on the size of the snapshots and the number of trended datasets each user can create.

Trend Wave Dashboards

To trend data in Wave, you trend a report that's collecting that data.

1. Create a dataset.

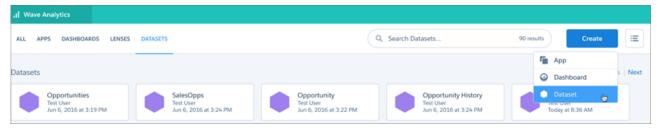
USER PERMISSIONS

To trend reports in Wave Analytics:

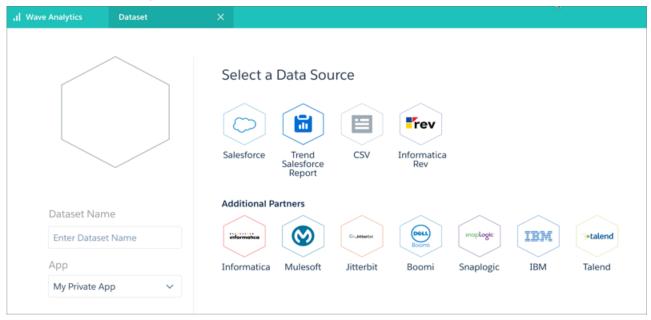
 "Use Wave Analytics" OR "Use Wave Analytics Templated Apps"

AND

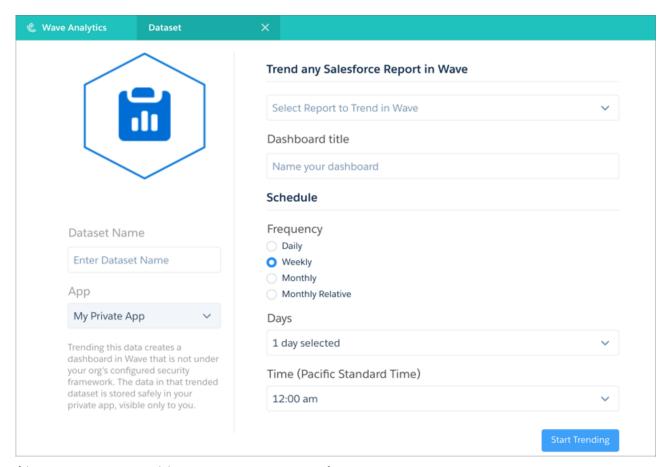
"Trend Report Data in Wave Analytics"



2. Choose **Trend 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.

Mote:

- 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 Wave datasets list.

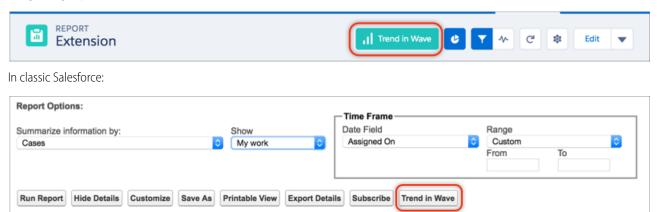
8. Connect to the new trending dashboard using the link in the notification email, or by going to the app in which you saved it (after receiving the notification email).

Ø

Note: At first, the trending dashboard displays only 1 data point. As snapshots are taken and added to the new Wave trending dataset underlying the dashboard, more data points are added.

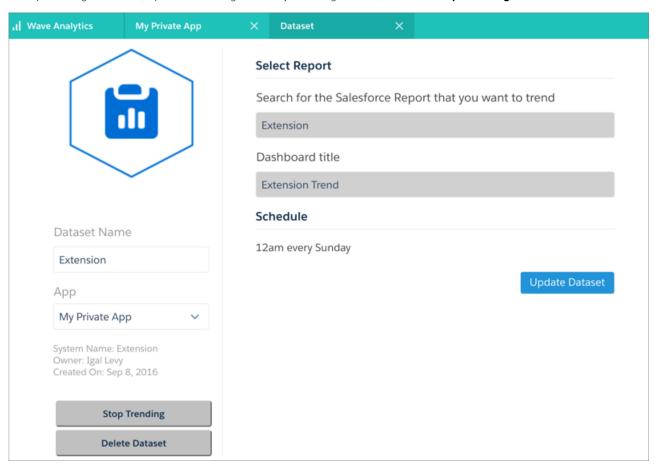
You can also trend any report directly by clicking **Trend in Wave** on that report.

In Lightning Experience:



Once you start trending, the **Trend in Wave** button on the report becomes **View Trend**.

To stop trending from Wave, 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.

For best results with trending, follow these guidelines:

- 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 Wave 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.

Share Trending Dashboards

You can share a trended dashboard just as you can with any Wave 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 Wave 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 Wave app that your target audience can access.

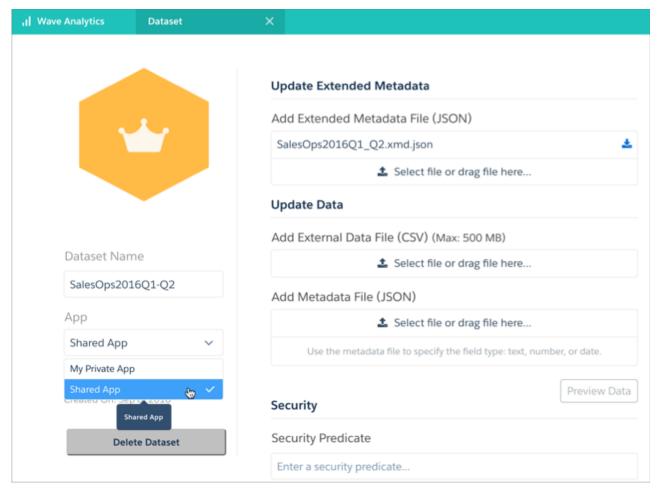
USER PERMISSIONS

To trend reports in Wave Analytics:

 "Use Wave Analytics" OR "Use Wave Analytics Templated Apps"

AND

"Trend Report Data in Wave Analytics"



(1) Important: Sharing the trending Wave 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 Wave 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 Wave dashboard regardless of any sharing restrictions or field-level security that has been set.

Limits and Limitations for Trending Data in Wave

To prevent the performance of your org from degrading, certain limitations are placed on the size of the snapshots and the number of trended datasets each user can create.

- Maximum number of trended datasets per user: 5
- Maximum number of rows per snapshot: 100,000
- Max total number of rows in a trended dataset: 5000000
- Maximum monthly number of rows for all snapshots per org: 40 million
- Trended datasets count towards the overall Wave platform limits, including total number of rows

To control the impact that trending datasets have on general org-wide limits, configure these limits, available on the Wave 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 Wave dataset containing the snapshots can have. If the dataset exceeds the limit, the oldest snapshot is deleted from the dataset.

These limits don't apply to admin users.

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.

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.

The Wave 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 Wave dashboard designer being enabled.

Users whose profiles have View Encrypted Data enabled can't create trended datasets because Wave does not support encrypted data.

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.

Extend Wave Everywhere

Incorporate Wave Analytics throughout your business. The Wave Analytics visualizations you've built are more powerful when you share them across your Salesforce experience by integrating them into custom pages, Visualforce pages, community sites, and more. More ways to share include packaging assets, presentation mode, annotations, and downloading filtered data from Wave. In addition, custom menus in lenses and dashboards allow you to perform common Salesforce actions directly from Wave.

IN THIS SECTION:

Embed Dashboards Everywhere

Embed Wave 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.

Integrate Wave into Salesforce with an Analytics Tab

Access your personal Analytics home and run Wave apps from a tab within Salesforce Classic or Lightning Experience.

Wave Migration, Packaging, and Distribution

You can migrate Wave 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.

Enable Actions and Links with Custom Actions Menus in Lenses and Dashboards

Custom menus in Wave Analytics lenses and dashboards let users take advantage of Salesforce actions and open records from Salesforce or other websites. You create action menus by working with a Wave dataset's extended metadata (XMD) file.

Configure Bulk Actions on Table Records

You can create custom bulk actions that apply to all records shown in a table widget of a Wave designer dashboard. Use bulk actions to update records or perform an action on each of them.

Embed Dashboards Everywhere

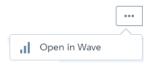
Embed Wave dashboards in every Salesforce experience. From embedded dashboards, users can explore and click to linked assets. If you set them up with a Share icon, embedded dashboards offer Post to Feed and Download sharing options.

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
Salesforce Classic only	Record detail page	Component in the enhanced page layout editor	Wave Analytics Assets
Lightning Experience and Salesforce Classic	Any page	Visualforce page component	wave:dashboard
Salesforce1	Lightning app page or Visualforce tab	Lighning app builder or Visualforce page component	Wave Dashboard or wave:dashboard
Communities	Visualforce page	Visualforce component on a Force.com community page	wave:dashboard

Embedded Wave dashboards can include these features.

- Annotations, notifications, and full-screen presentation
- Sharing and downloading options
- Links to other Wave dashboards
- Record actions menus
- Link to the native app, whether it's Wave on desktop or Wave Mobile (available for Wave users but not Community users)



Embedded Wave dashboards have these limitations.

- The option to explore in Wave isn't included.
- The charts don't include hover-over information (tooltips).
- Some filters, such as date selectors, are difficult to set.

IN THIS SECTION:

Embed Wave Dashboards in Lightning Pages

Add Wave Analytics 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 Wave Analytics window.

Embed Wave Dashboards in Salesforce Classic Pages

Add a Wave Analytics 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 Wave Analytics window.

Add a Wave Analytics Dashboard to a Visualforce Page

By embedding a Wave 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 Wave Analytics.

Whitelist Trusted Sites for Embedded Dashboards

Embed Wave in any web page or web-based app. By specifying your trusted sites in the Wave Analytics whitelist, you can include embedded Wave dashboards in websites and apps outside of Salesforce servers.

View Embedded Dashboards on Mobile Devices

Add Wave dashboard components to Visualforce pages in Salesforce websites accessed by mobile users, including Communities. Or add Wave dashboard components to Lightning app pages. Embedded Wave dashboards can responsively select layouts optimized for any device--including phones and tablets.

Share Wave with Communities

With Wave Analytics for Communities, external users can view apps shared with them via Wave dashboards embedded in Visualforce pages in their community.

Embed Wave Dashboards in Lightning Pages

Add Wave Analytics 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 Wave Analytics window.

To add a dashboard to your Lightning page, follow these steps.

- 1. Select the **Wave Dashboard** component in the Lightning app builder.
- 2. From the Dashboard drop-down list, select the dashboard to display.
- **3.** From the layout drop-down list, select the type of screen where your Lightning page will be viewed.

EDITIONS

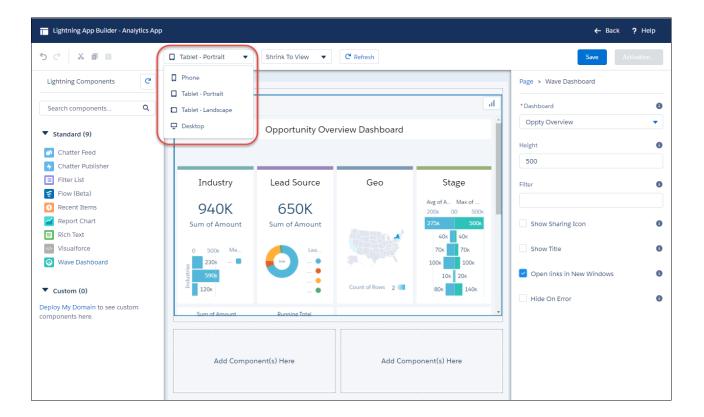
Available in: Lightning Experience

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

USER PERMISSIONS

To view embedded Wave dashboards:

 Analytics Cloud - Wave Analytics Platform permission set license with "Use Wave Analytics" permission



- **4.** Configure the attributes of the embedded Wave Analytics dashboard:
- 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.
- The **Show Title** checkbox gives you control over the visibility of the dashboard title.
- With the **Open Links in New Windows** attribute, you can specify where links from the dashboard to other assets are opened.
- The **Hide On Error** checkbox gives you control over whether the Wave Analytics dashboard appears if there is an error (such as the dashboard can't be found).
- With the **Filter** attribute, you can set up the embedded dashboard to show only the data that's relevant for the record being viewed. The **Filter** attribute links data fields in the dashboard to the object's fields. For example, the filter 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:

```
{'datasets':{'account':[{'fields':['Name'], 'filter':{'operator': 'matches',
'values':['$Name']}}]}}
```

Refer to the table to understand the notation for the **Filter** attribute.

17. The previous syntax, noted here, is still supported.

Notation	Description	Notes
datasets	Wave dataset	Where to find it: On the Wave 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.)

Notation	Description	Notes
fields	Dimension or Measure in the Wave 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.
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: If a filter or selection is specified that does not exist, the field mapping is ignored the dashboard appears with all its data and no selection.

The **Filter** attribute follows this format:

The example above shows a filter that causes the embedded dashboard to appear with only the filtered data showing. Here's an example that applies the account name as a selection within a dashboard that shows all of its data:

```
{'datasets' : {'account': [ {'fields': ['Name'], 'selection': ['$Name']}}}
```

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:

```
{ 'datasetSystemName1': {'field1': ['$value1']}, 'datasetSystemName2': {'field1': ['$value1', '$value2'],'field2': ['$value3']} }
```

For example:

```
{'opportunity': {'AccountId': ['$Id']}}
```

For information about working with the Lightning App Builder, start on the **Lightning App Builder** overview page.

Embed Wave Dashboards in Salesforce Classic Pages

Add a Wave Analytics 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 Wave Analytics window.

EDITIONS

Available in: Salesforce Classic

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

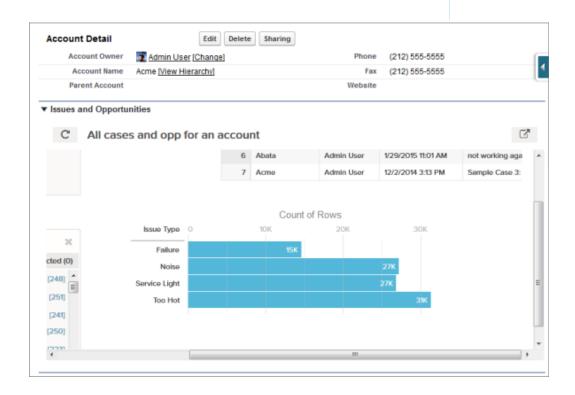
USER PERMISSIONS

To customize page layouts:

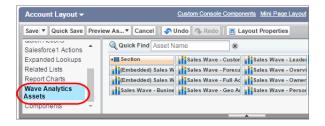
"Customize Application"

To view embedded Wave dashboards:

 Analytics Cloud - Wave Analytics Platform permission set license with "Use Wave Analytics" permission

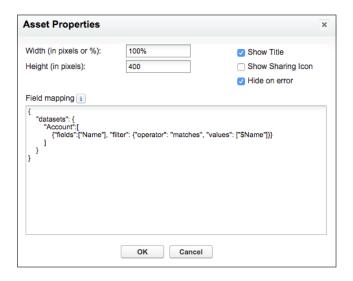


1. In the enhanced page layout editor, select Wave Analytics Assets in the left column of the palette.



- **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 ().

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.

17. The previous syntax, noted here, is still supported.

Notation	Description	Notes
datasets	Wave dataset	Where to find it: On the Wave 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 Wave 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."

Notation	Description	Notes
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.
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: If a filter or selection is specified that does not exist, the field mapping is ignored the dashboard appears with all its data and no selection.

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"]
   }
  }
 ]
 }
}
```

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:

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"]
  }
 }
1
            }
```

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 Retail 1, Wholesale 1, 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"]
}
```

Additional properties of Wave Analytics 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.
- The Hide on error checkbox gives you control over whether the Wave Analytics 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.

Be aware of these limits and limitations:

- You can add one dashboard per page layout.
- Wave Analytics dashboards aren't supported in the original page layout editor. If you open and then save a layout with a Wave 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.

Add a Wave Analytics Dashboard to a Visualforce Page

By embedding a Wave 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 Wave Analytics.

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 Wave Analytics assets.

Here are two examples. Note that 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.

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 view embedded Wave dashboards:

 Analytics Cloud - Wave Analytics Platform permission set license with "Use Wave Analytics" permission OR Analytics Cloud - Wave Community Users permission set license with "View Wave Analytics on Communities Pages" permission

```
</apex:page>
```

- The showTitle attribute gives you control over the visibility of the dashboard title.
- 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.
- With the openLinksInNewWindow attribute, you can specify where links from the dashboard to other assets are opened.
- The hideOnError attribute gives you control over whether a Wave Analytics 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.
- 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, in the following format:

For example:

Refer to the table to understand the notation for the filter attribute.



Notation	Description	Notes
datasets	Wave dataset	Where to find it: On the Wave 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 Wave 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."

Notation	Description	Notes
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.
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.

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: If a filter or selection is specified that does not exist, the field mapping is ignored the dashboard appears with all its data and no selection.
- 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:

```
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 Wave Analytics, see the Visualforce Developer Guide Standard Component Reference.

Whitelist Trusted Sites for Embedded Dashboards

Embed Wave in any web page or web-based app. By specifying your trusted sites in the Wave Analytics whitelist, you can include embedded Wave dashboards in websites and apps outside of Salesforce servers.

First, add a Wave dashboard component to your site, using Visualforce embedding, Lightning App Builder, or Lightning Out. This code sample shows how to add a Wave dashboard component via Lightning Out.

```
$Lightning.use("wave:waveApp", function() {
    $Lightning.createComponent("wave:waveDashboard", {
        dashboardId: "0FKxx00000006bOGAQ"
    }, "app");
}, url, accessToken)
```

Second, set up a whitelist using the Content Security Policy (CSP). See Set Up a Whitelist of Trusted Sites for Embedded Wave Dashboards

The whitelist for embedded Wave includes the following by default:

- *.salesforce.com:*
- *.force.com:*
- Your org's custom domain.

View Embedded Dashboards on Mobile Devices

Add Wave dashboard components to Visualforce pages in Salesforce websites accessed by mobile users, including Communities. Or add Wave dashboard components to Lightning app pages. Embedded Wave dashboards can responsively select layouts optimized for any device--including phones and tablets.

Dashboards built in the Wave dashboard designer are responsive to any screen size. If you've created a tablet, phone, watch, or custom layout in the Wave dashboard designer, the dashboard automatically optimizes to fit your screen. For more information, see Generate Unique Wave Dashboard Layouts for Different Devices .

When your dashboard has the right layouts, embed it via Visualforce or the Lightning app builder. Note that for Salesforce1, only Lightning app pages are supported. See Lightning Pages for more information.

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 view Wave Analytics:

"Use Wave Analytics"

Share Wave with Communities

With Wave Analytics for Communities, external users can view apps shared with them via Wave dashboards embedded in Visualforce pages in their community.



Note: Only users with a Customer Community Plus or Partner Community license can use this feature. This feature is supported in communities but not in portals.

Community users can view dashboards embedded in Visualforce pages in their community, and they can explore lenses, facets, and links in the embedded dashboards. Community users can't access the Wave Analytics home page or create, update, or delete any Wave asset. They also can't upload data. This table summarizes differences between internal Wave users and external Community users.

	Wave Users	Community Users
Access, Explore, and Facet Dashboards and Lenses (embedded in Visualforce pages)	▽	✓
Create, Update, or Delete Dashboards and Lenses	~	
Share Dashboards and Lenses	~	~
Access the Wave Home Page	~	
Access Wave Templated Apps (for example, Sales Wave)	~	
Analyze Large Datasets	~	~
Benefit from Security Predicates	~	~
Use Custom Action Menus	~	
Access Wave from Mobile Devices	~	

To set up Wave Analytics for Communities, complete the following steps.

- **1.** Enable Wave Analytics for Communities and set up community members. See Enable Wave Analytics for Communities.
- 2. Create Wave dashboards and save them in a Wave app that you can share with your community.
- **3.** Embed your Wave dashboards in Visualforce pages in your community. For more information, see Add a Wave Analytics Dashboard to a Visualforce Page. If you embed a dashboard in a new Visualforce page, create a tab for the page and add the tab to your community.
- **4.** From the Wave app containing your dashboards, select **Share**.
- 5. In the Share dialog, select **Enable sharing with Communities**.

EDITIONS

Available in: Salesforce Classic

Available for an extra cost in: **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in: **Developer** Edition

USER PERMISSIONS

To create, customize, or activate a community:

"Create and Set Up Communities"

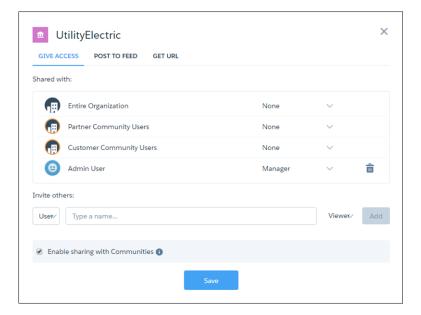
AND is a member of the community they're trying to update

To modify Wave Analytics settings:

 Analytics Cloud - Wave Analytics Platform permission set license with "Manage Wave Analytics" permission

To view, explore, and share embedded Wave dashboards:

 Analytics Cloud - Wave Community Users permission set license with "View Wave Analytics on Communities Pages" permission



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.



6. Invite community partners and customers to share the app by entering their names and clicking **Add**. Community members in the sharing list are highlighted in orange to distinguish them as external users, and Viewer is the only access available for them.

Note that security is enforced at these levels.

- By role, group, or user, via the Sharing dialog.
- At the Wave app level, via the Enable sharing with Communities checkbox.
- At the org setup level, via security predicates.

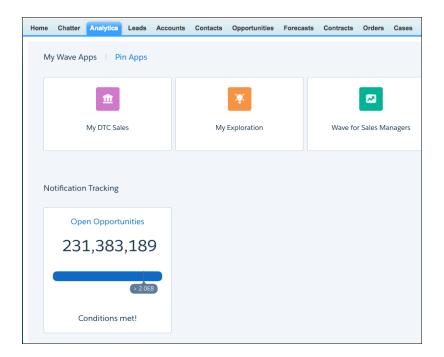
Integrate Wave into Salesforce with an Analytics Tab

Access your personal Analytics home and run Wave apps from a tab within Salesforce Classic or Lightning Experience.

The Analytics tab is available in new Wave-enabled orgs.

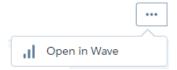


In existing orgs, set up the Analytics tab as you would any new tab.



Wave in the Analytics tab can include these Wave features:

- Annotations, notifications, and full-screen presentation
- Sharing and downloading options
- Links to other Wave assets
- Record actions menu
- Link to native Wave (available for Wave users but not Community users)



Wave Migration, Packaging, and Distribution

You can migrate Wave 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.

IN THIS SECTION:

Migrate Wave Assets with Change Sets

Use change sets to move customized Wave assets between orgs that have a deployment connection. For example, create a Wave app containing dashboards, lenses, datasets, dataflows in your Sandbox org, then migrate the app assets to your production org once testing is complete.

Package Wave Assets in Managed Packages

You can create managed packages of Wave assets, including Wave apps, dashboards, lenses, datasets, and dataflows. Use packages to distribute those assets to other users or organizations, including those outside your company.

Distribute Wave 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 Wave Assets and User XMD

Wave provides full support for the Metadata API, which can be used to retrieve, deploy, create, update, or delete customizations for your organization.

Migrate Wave Assets with Change Sets

Use change sets to move customized Wave assets between orgs that have a deployment connection. For example, create a Wave app containing dashboards, lenses, datasets, dataflows in your Sandbox org, then migrate the app assets to your production org once testing is complete.

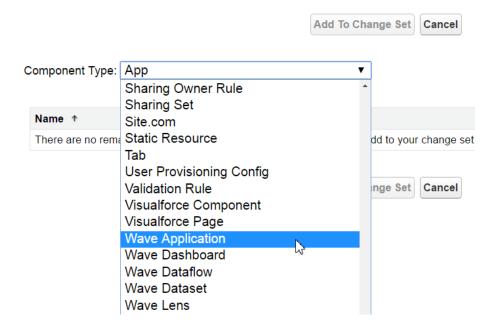


Tip: Remember, this topic describes how to migrate Wave assets—such as lenses, customized dashboards, dataset containers, and so on. It does not discuss migrating Salesforce or Wave data from org to org. Where this topic refers to a Wave dataset, what actually gets migrated is the container; the dataset metadata.

To migrate Wave assets:

- 1. Create and test your Wave 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 Wave assets to the change set. 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.

Choose components to add to your change set.



- (1) Important: Wave Analytics components include Wave apps, dashboards, dataflows, datasets, lenses, and master user XMD. As you package Wave components, keep these tips and best practices in mind.
 - Wave unmanaged packages, as opposed to managed packages, are considered a developer-only feature and are not supported for general-purpose distribution. While Wave unmanaged packages should work as expected within the constraints of Salesforce unmanaged packages, they have not been subject to the same level of 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.
 - Wave Admin permissions are required to create a package, but not for deployment, which requires only Salesforce admin permissions.
 - There is no spidering between datasets and dataflows, meaning there is no dependency following. When packaging both, they must be added manually. If they are not, an error appears during deployment. The same is true for change sets—when packaging both datasets and dataflows, add them manually.
 - Before the Spring '17 release, images would not render when deploying a dashboard that used an image widget that referenced image files not available on the target org. There were two workarounds; to either 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. Consequently, you will be unable to delete a dashboard if it is referenced in a link. In such cases, either re-create the image or link widgets in the dashboard in the source org, and then re-package, 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, it's now Type = text). Comments in JSON are removed. Nodes
 may 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 Wave 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.

Package Wave Assets in Managed Packages

You can create managed packages of Wave assets, including Wave apps, dashboards, lenses, datasets, and dataflows. 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—Wave 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 Wave 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 Wave assets to the package. 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.
 - (1) Important: Wave Analytics components include Wave apps, dashboards, dataflows, datasets, lenses, and master user XMD. As you package Wave components, keep these tips and best practices in mind.
 - Wave unmanaged packages, as opposed to managed packages, are considered a developer-only feature and are not supported for general-purpose distribution. While Wave unmanaged packages should work as expected within the constraints of Salesforce unmanaged packages, they have not been subject to the same level of 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.
 - Wave Admin permissions are required to create a package, but not for deployment, which requires only Salesforce admin permissions.
 - There is no spidering between datasets and dataflows, meaning there is no dependency following. When packaging both, they must be added manually. If they are not, an error appears during deployment. The same is true for change sets—when packaging both datasets and dataflows, add them manually.
 - Before the Spring '17 release, images would not render when deploying a dashboard that used an image widget that referenced image files not available on the target org. There were two workarounds; to either 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. Consequently, you will be unable to delete a dashboard if it is referenced in a link. In such cases, either re-create the image or link widgets in the dashboard in the source org, and then re-package, 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, it's now Type = text). Comments in JSON are removed. Nodes may 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:

Introduction to the License Management App

Distribute Wave 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 Wave 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 Wave 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.

SEE ALSO:

Security Review Overview
Introduction to the License Management App

Use the Metadata API for Wave Assets and User XMD

Wave 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 Wave dataflow are all metadata updates. Metadata is the information that describes the configuration of your organization and of your Wave 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, stackybar, 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 Wave 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 Wave metadata types are supported in the Metadata API:

- WaveApplication
- WaveDashboard
- WaveDataflow
- WaveDataset
- WaveLens
- WaveXmd
- WaveTemplateBundle

SEE ALSO:

Understanding Metadata API

Enable Actions and Links with Custom Actions Menus in Lenses and Dashboards

Custom menus in Wave Analytics lenses and dashboards let users take advantage of Salesforce actions and open records from Salesforce or other websites. You create action menus by working with a Wave dataset's extended metadata (XMD) file.

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 view visualizations:

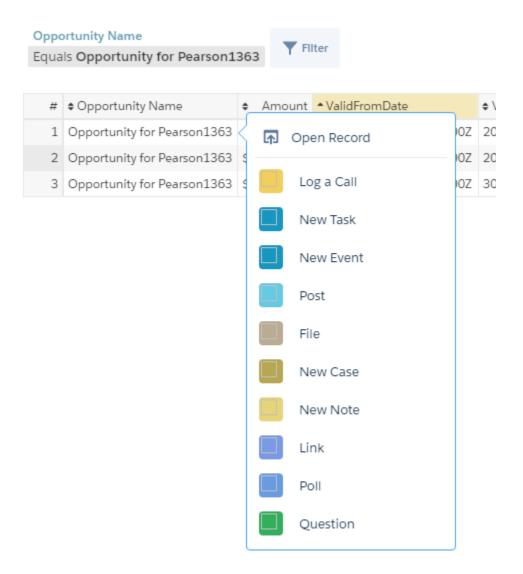
• "Use Wave Analytics"

To edit a dataset's extended metadata (XMD) file:

"Edit Wave Analytics Dataflows"

To create actions:

"Customize Application"



Custom actions menus can be accessed from these Wave chart and table interface elements:

• Value labels on charts (such as "Opportunity for Pearson" in the image above).



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 displays the menu, to see the menu's contents and select from the available actions or open a record.

Actions in the Actions Menu

Actions come from object-specific actions and/or global actions that you've previously added to Salesforce objects. Salesforce actions let users create records that have automatic relationships to other records, make changes and updates to specific records, and interact with records in ways you define. For example, you can add actions that let users create new cases, events, and accounts as well as update record, post to Chatter, and so on.

Open Record Link in a Quick Actions Menu

Links can pass specific bits of data, such as Salesforce account IDs, from lenses or dashboards to records. For example, a link from an account name in a dashboard could be configured to pass the related account ID to Salesforce; when a user clicks an open record link, the appropriate account record opens in a new tab. Custom links can also look up records for employees in Workday® or expense reports in Concur®. Any website can be linked to as long as it uses HTTP or HTTPS protocols.

Actions Menu Support for Objects and Actions

The current actions menu implementation provides limited support for Salesforce objects and actions.

You can currently create Wave quick actions menus for custom objects, sObjects, and the following standard Salesforce objects:

- Account
- Cases
- Lead
- Opportunity

Actions on Contacts, User, or User Profile page layouts are not supported.

The current implementation of action menus supports Chatter actions (Post, File, Thanks, Link, Poll, and Question), object-specific actions, and global actions. It does not support Salesforce1 actions.

Other objects and actions will be supported in future releases.

Create Action Menus

Follow these basic steps to add an action to a Wave custom actions menu:

- 1. Create the action and add it to a Salesforce object that's part of a Wave dataset.
- **2.** Apply the action to the object's page layout.
- 3. Work with the dataset's Extended Metadata (XMD) file to make the action or link available from action menus in Wave charts and tables. Add a dimensions section to the XMD and edit its attributes to create the menu and control the behavior of actions and links to records. Prebuilt Wave apps, such as Sales Wave Analytics, already include menus, and their dataset XMD files already include dimensions sections. You can customize dashboards in prebuilt Wave apps by editing the XMD. For more information, see *Dimensions Section* in the Extended Metadata (XMD) Reference.

Note: To add an open record link to menu for a dataset with records from *multiple* Salesforce org instances, add an organizations section to the XMD file. The organizations section lets you map from org IDs to the URLs for each org instance referred to in the dataset. See *Organizations Section* in the Extended Metadata (XMD) Reference

The example in the following links walks you through adding an action to Wave menus. Follow the steps in the order shown here.

Ø

Note: These steps work for any Salesforce object and Wave dataset in your org. The example uses the Opportunities object and dataset from the Sales Wave Analytics app created in the Wave Apps Basics Trailhead module. If you want to follow along, complete the first two units of the module first.

IN THIS SECTION:

- Create an Action and Add It to an Object
 Select the action you'd like to make available to users in your org, and add it to an object.
- 2. Apply an Action to an Object's Page Layout

 Open the page layout assigned to the profile for the users you want to access the action, and apply the action to it.
- 3. Edit the Dataset Extended Metadata File to Make the Action Available to Users
 Specify the field in a dataset to indicate where the action appears in Wave charts and tables. Also, specify the ID field for the Salesforce object that's impacted by the action.

Create an Action and Add It to an Object

Select the action you'd like to make available to users in your org, and add it to an object.

(1) Important: Create Sales Wave Analytics in the Wave Apps Basics Trailhead module to follow the example that starts here.

You can add actions to fields in any Salesforce object. The example shows how to add an Update a Record action to the Opportunities object in the app you create in Wave Apps Basics. Once it's added, users can update fields in the Opportunities objects directly from a Wave chart or table without having to flip back and forth from Wave to the Sales Cloud. As part of the example, you also add fields that can be updated using the action.

- 1. Go to Setup in the Developer Edition org you used for the Wave Apps Basics Trailhead module.
- In the Quick Find box at the top of Setup, enter the name of the object you want to add actions
 to. In the case of our example, type Opportunities in the Quick Find box. Then select
 Buttons, Links, and Actions.
- 3. Click **New Action** to see what's available.
- **4.** Open the Action Type drop-down list and select the action you want to add. In this case, select **Update a Record**.

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 view visualizations:

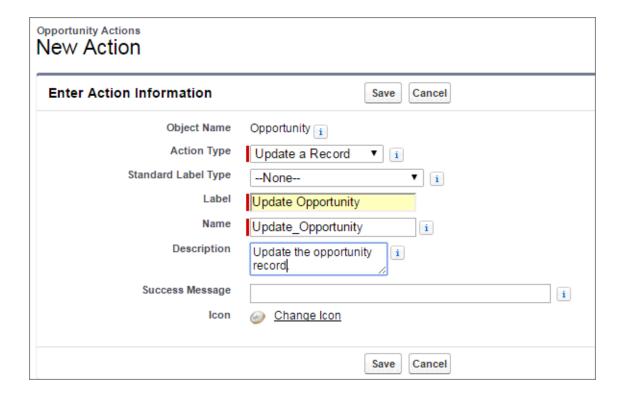
"Use Wave Analytics"

To edit a dataset's extended metadata (XMD) file:

 "Edit Wave Analytics Dataflows"

To create actions:

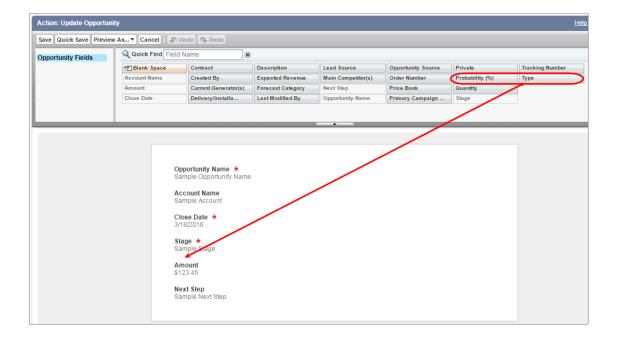
"Customize Application"



5. Click Save.

The next page shows the default fields that can be updated. You can choose any you'd like. To make the new action even more powerful, select fields in the object that users can update from Wave. For the example, you 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 Mac).



7. Click Save

Apply an Action to an Object's Page Layout

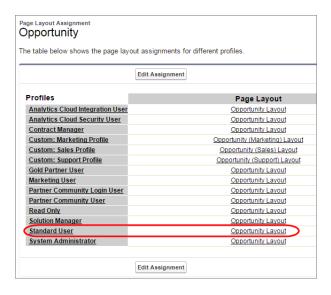
Open the page layout assigned to the profile for the users you want to access the action, and apply the action to it.

The example here shows how to apply the action you just added to the opportunity page layout assigned to the Standard User profile. It uses the example Developer Edition org containing the Sales Wave Analytics app created in the Wave Apps Basics Trailhead module.

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 the example, you'll enable a user to see the Update Opportunity custom action in opportunity records in Wave. You do that by adding the action to the opportunity page layout assigned to the user's Standard User profile.

You can use the same techniques with any Salesforce page layout and user profile.

- 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.
 Notice that there are different page layouts for opportunity data.
- **2.** To see which page layout is assigned to a user's profile, in the case Standard User, click **Page Layout Assignment**.



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 view visualizations:

"Use Wave Analytics"

To edit a dataset's extended metadata (XMD) file:

"Edit Wave Analytics Dataflows"

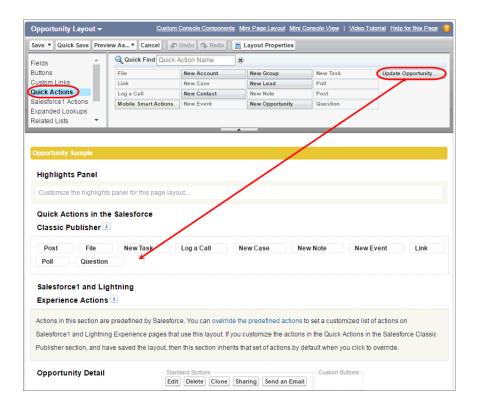
To create actions:

"Customize Application"

Edit the page layout assigned to the profile for the users you want to access the action. In our 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 as well.)

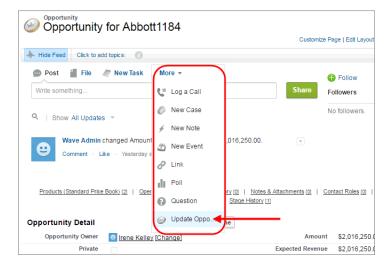
- 3. Open the layout. Click the **Opportunity Layout** link next to a profile—it doesn't matter which profile. Next, select the actions to add to the layout. For the example, you add the Update Opportunity layout.
- **4.** In the palette, click **Quick Actions**. The list of all available actions appear to the right.

5. Drag the **Update Opportunity** custom action and drop it after the existing actions in the Quick Actions in the Salesforce Classic Publisher section.



6. Click Save.

This action is now available on the Opportunity details page for all users assigned the Opportunities page layout. You can see it in the Developer Edition org you sign up for in the Wave Apps Basics Trailhead module used for the example. If you've done that and created the app, go the org, find an opportunity, and check it out. You can access it from the More option.



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.

Edit the Dataset Extended Metadata File to Make the Action Available to Users

Specify the field in a dataset to indicate where the action appears in Wave charts and tables. Also, specify the ID field for the Salesforce object that's impacted by the action.

The object's ID field ties the action in Wave to a record in Sales Cloud or another part of Salesforce. When Salesforce loads records into Wave, it also loads record IDs into the dataset. When a user performs an action on a record in Wave, it uses the ID to identify the corresponding record (the one users impact with the action) in Salesforce. Each Salesforce record ID contained in a Wave dataset record points back to a record in Salesforce.

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 view visualizations:

"Use Wave Analytics"

To edit a dataset's extended metadata (XMD) file:

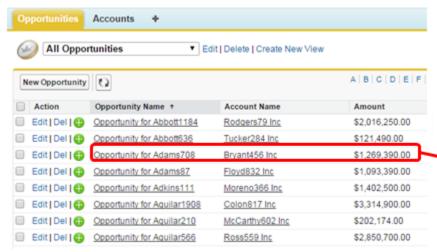
"Edit Wave Analytics Dataflows"

To create actions:

"Customize Application"

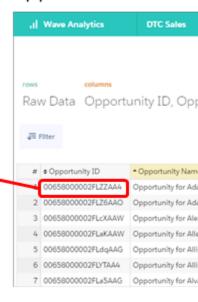
Salesforce

Opportunity Object



Wave

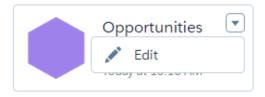
Opportunities Data



You specify these fields in *dimensions* in the extended metadata (XMD) file for the dataset containing the object you want the user to take action on.

For our example, that's the XMD file for the Opportunities dataset, which contains the two fields to specify: Opportunity Name and Opportunity ID. Often you'd manually add *dimensions* to the file and specify all its attributes. Our example uses the Sales Wave Analytics app created in the Wave Apps Basics Trailhead module. Like all prebuilt Wave apps, Sales Wave Analytics already includes actions, so you edit the *dimensions* section that's already in the XMD file. Read complete details about adding actions in the XMD file in the Dimensions topic in the Extended Metadata (XMD) Reference.

- Note: Dataset XMD files can also include a *dimension* (singular) section, which is used for other purposes. Be sure to refer to the Dimensions topic in the XMD Reference, *not* the Dimension topic.
- 1. Add fields to the XMD file for the Wave dataset—in this case Opportunities. In Wave, click the **Browse** tab and open the Sales Wave app. Click the icon in the upper left and select **Browse All** to see all the apps dashboards and datasets.
- 2. Position the cursor over the Opportunities dataset. Click the down arrow and then Edit.



The edit dataset page appears.

- 3. In the Add Extended Metadata File (JSON) field, click to download the XMD file for the Opportunities dataset.
- **4.** Open the XMD file in a text editor.
 - Tip: If the JSON isn't formatted, copy it into a JSON formatter tool, like JSON Editor Online, and then paste the results back into the XMD file. In JSON Editor Online, copy the unformatted JSON into the left pane and click the right arrow to show the formatted version in the right pane.
 - Warning: If you modify JSON in your production org, check with your security department to see which external JSON tools are recommended.
- 5. Because the JSON contains lots of lines, search for the second dimensions attribute, which is shown below.

```
"values_query_columns": ...,
"hide measures": ...,
"formats": {...},
"colors": {...},
"labels": {
    "measures": {...},
    "keys": {...},
    "dimensions": {...}
                             //Not this one.
},
"dimensions": [
                             //This one.
        "recordIdField": "AccountId",
        "field": "Account.Name",
        "linkTemplate": "/{{row.AccountId}}",
        "linkTooltip": "Open Account"
    },
```

```
"recordIdField": "Id",
    "field": "Name",
    "linkTemplate": "/{{row.Id}}",
    "linkTooltip": "Open Opportunity"
},
{
    "recordIdField": "OwnerId",
    "field": "Owner.Name",
    "linkTemplate": "/{{row.OwnerId}}",
    "linkTooltip": "Open User"
}
```

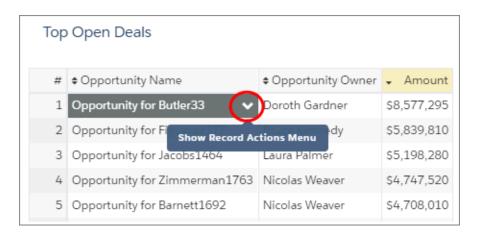
Note: There are two fields called "dimensions" in the example code. Ignore the one under "labels"—it's part of the Labels XMD section.

If you're adding actions to charts and tables that aren't part of a Wave app, you'd have to manually add a dimensions attribute. You'd also have to manually enter the field (Name) that contains the custom action along with the ID field (Id) that identifies the Salesforce opportunity records. Because Sales Wave Analytics comes with actions, the fields are already added to the XMD dimensions section.

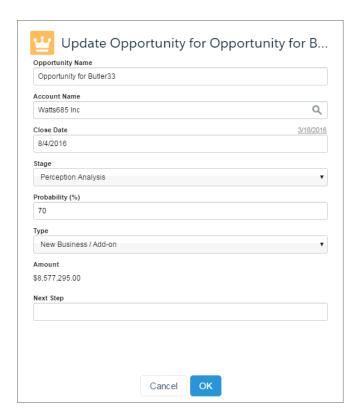
Tip: If you added fields to the XMD file, to apply your changes, you'd have to save your changes and then upload the file in the page where you downloaded it.

If you've followed along with the example using the Sales Wave app created in Wave Apps Basics, have a look at the results of your work in a Wave dashboard.

- 1. Go to the Sales Wave app and open the Sales Leader Pipeline Changes dashboard.
- 2. Scroll down to the Top Open Deals widget.
 - (1) Important: If this DE org contains old data, widgets in a Sales Wave app dashboard might not show results. If a widget doesn't show any results, refresh the Salesforce data in this DE org using the Trailhead Data Manager, and then run the dataflow to refresh the data in the Sales Wave app datasets. For more information, see "Say Hello to the Trailhead Data Manager" in Wave Apps Basics.
- **3.** Hover over an opportunity name to see the Show Record Actions Menu arrow.



- Note: You can see the action because you are assigned the same page layout as Kelly.
- **4.** Click the arrow and select the new **Update Opportunity** action.
- **5.** See that you can update the opportunity record—including the Type and Probability fields you added—directly from Wave. The record will be updated in the Sales Cloud and synced back to Wave the next time the Wave dataflow runs.



Configure Bulk Actions on Table Records

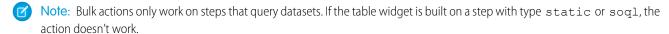
You can create custom bulk actions that apply to all records shown in a table widget of a Wave designer dashboard. Use bulk actions to update records or perform an action on each of them.

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. Wave passes the SAQL query used to populate the table widget to the Visualforce page.
 - Note: The SAQL query that Wave 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 guery.
 - Tip: When issuing a Wave 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 Wave Query API.
- **4.** The Wave 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 3rd party.



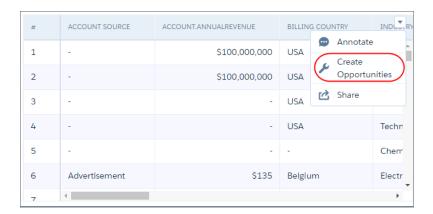
IN THIS SECTION:

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.

Creating Opportunities		
Account	Opportunity	
Open Source Inc.	Creating	
Missoula & Sons Inc.	Creating	
Open Source Inc.	Creating	
Missoula & Sons Inc.	Creating	

3. The Visualforce page shows the opportunity created for each account.

Opportunities Created		
Account	Opportunity	
Open Source Inc.	From Open Source Inc.	
Missoula & Sons Inc.	From Missoula & Sons Inc.	
Open Source Inc.	From Open Source Inc.	
Missoula & Sons Inc.	From Missoula & Sons Inc.	

To implement this bulk action, perform the following tasks.

IN THIS SECTION:

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.

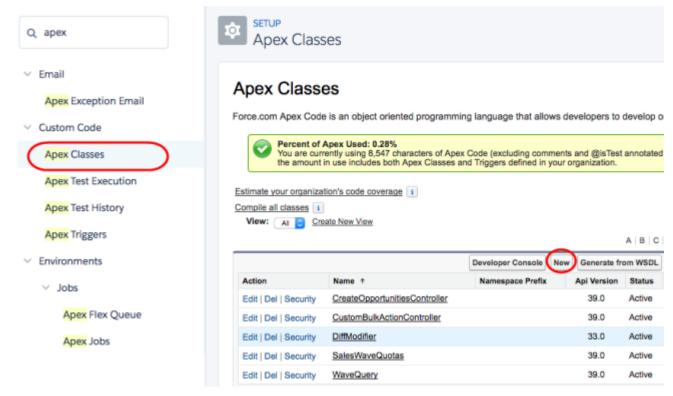
3. Assign Bulk Action to Table Widget

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.

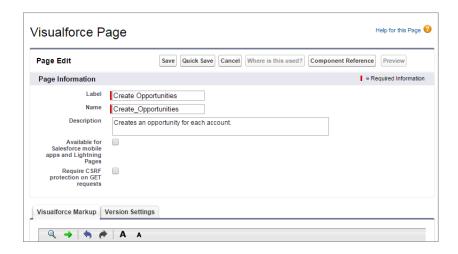
```
public class CreateOpportunitiesController {
    public string query{get; set;}
^{\prime\star} To determine the records to perform the bulk action on, extract the SAQL query ^{\star\prime}
    public PageReference init() {
        query = ApexPages.currentPage().getParameters().get('query');
        return null;
    }
^{\prime} 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,
```

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.

```
}
       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>AccountOpportunity</thad>
           </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++) {</pre>
                       record = data.results.records[i];
                       row = $('');
                       row.append($('').html(record['Account.Name']));
                       row.append($('
'">').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++) {</pre>
                        $('td.' +
accountRecords[i].AccountId).html(result[accountRecords[i].AccountId]);
                  $('#message').html(accountRecords.length + ' Opportunities Created');
                else {
                    $('#message').html('Error: ' + event.message);
            });
    </script>
</apex:page>
```

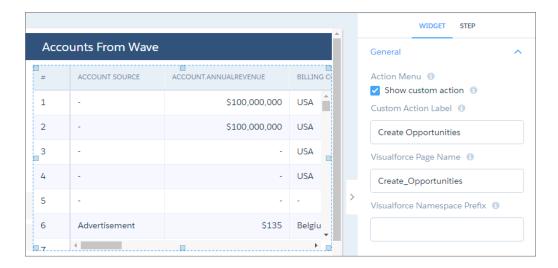
6. Click Save

Assign Bulk Action to Table Widget

Configure the table widget properties to expose the bulk action defined in the Visualforce page.

- 1. Open the Wave 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.

Put It All Together with Wave Apps

Create Wave apps to run presentations, organize your data projects, and control asset sharing.

Wave Analytics apps can contain datasets, lenses, and dashboards. Create them to organize and run your analytics presentations, and share them with other people in your organization.

Examples of Wave apps are the prebuilt Salesforce apps such as the Sales Wave Analytics app and Service Wave Analytics app. Like the apps you create yourself, these are collections of Wave assets—in both cases, the apps include dashboards and datasets. The difference is that Salesforce has created the assets to provide a ready-made path through Salesforce data to help users get started fast with Wave. Your own apps contain assets that you create yourself.

IN THIS SECTION:

App-Level Sharing

Wave Analytics 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

Target your app for customers with custom navigation that shows them the right dashboards.

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.

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 Wave Analytics.

App-Level Sharing

Wave Analytics 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 Wave Analytics 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	Χ	Χ	Χ
See who has access to the app	Χ	Χ	Χ
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)		Χ	X
Change the app's sharing settings			Χ
Rename the app			Χ
Delete the app			Χ

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. Click App.
- **3.** Type the name of your app.
- 4. Click Create.

Your app is created and displayed in a new tab. You can now put lenses, dashboards, and datasets in it.

- **5.** Add a description of the app's purpose and contents so that your colleagues understand how to use the data.
 - **a.** Click in the description box under the app's name.
 - **b.** Type a description.
 - **c.** Click the check mark **v** to save.
- **6.** Optionally, change the app's default icon to make it easier to distinguish from other apps.
 - a. Hover over the default icon.
 - **b.** Click **Change Icon**.
 - **c.** Click an icon from the gallery.

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 an app:

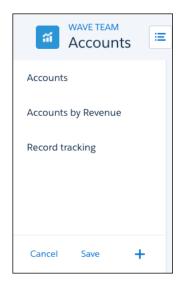
 "Create Wave Analytics Apps"

Set Up Navigation Through Your App

Target your app for customers with custom navigation that shows them the right dashboards.

Follow these steps to set up an app's navigation.

- 1. Click to open the navigation panel.
- **2.** If you're the manager of an app, you see the **Organize** button. Click it to set up the app.



- **3.** Change the presentation order by sliding dashboards up or down, and click dashboard names to hide or show them.
- **4.** To add more dashboards from the app, click the plus sign.
- **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 dashboard 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.

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 share an app:

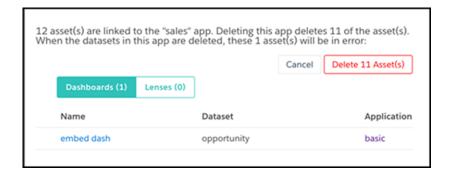
"Use Wave Analytics" and Manager access to the app

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 Wave Analytics.

- 1. Open the app, and then click **Delete**.
 - Warning: You can't recover a deleted app.

If applicable, Wave Analytics 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.



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 delete an app:

 "Use Wave Analytics" and Manager access to the app

- (1) 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).

Prebuilt Wave Apps

Prebuilt Wave Analytics apps guide you along a ready-made path through your Salesforce data. You get best-practice dashboards and KPIs in a single place your team can use to drive business success.

Prebuilt Wave apps from Salesforce help speed your organization's time-to-value with Wave Analytics. Instead of having to build visualizations yourself, with Wave apps you answer questions in a handy configuration wizard that's part of each app. Wave takes care of the rest, creating dashboards designed and built by Salesforce based on our years of experience helping companies manage their customer interactions.

Prebuilt apps are designed for desktop and mobile devices. They come with dashboards that you can embed in Salesforce pages to give users direct access to business intelligence right from the cloud where they do their everyday work. And you can drill deeper into key aspects of your business by customizing Wave apps to meet your specific needs.

To get started, see step-by-step instructions for each app by following these links.

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, Performance, and **Unlimited** Editions. Also available in **Developer** Edition.

IN THIS SECTION:

The Sales Wave Analytics App

The Sales Wave Analytics app brings the power of Wave Analytics to Sales Cloud on any device that supports Wave. With intuitive visualizations based on your Salesforce data, Sales Wave Analytics lets you move from insight to action quickly and helps you turn data into smarter sales.

The Service Wave Analytics App

The Service Wave Analytics app gets you started fast with Wave 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.

The Field Service Wave Analytics App

The Field Service Wave Analytics app brings the power of Wave Analytics to Field Service Lightning on any device that supports Wave.

Event Monitoring Wave App

The Event Monitoring Wave App integrates with event monitoring and setup audit trail data to give you insights into your user and org behavior.

The B2B Marketing Wave Analytics App

B2B Marketing Wave 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.

The Sales Wave Analytics App

The Sales Wave Analytics app brings the power of Wave Analytics to Sales Cloud on any device that supports Wave. With intuitive visualizations based on your Salesforce data, Sales Wave 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 Wave Analytics. If you haven't used Wave before, learn more about it from the Wave Analytics Library.

IN THIS SECTION:

1. About the Sales Wave Analytics App

Learn the benefits of Sales Wave Analytics before you create and use the app to explore your Sales Cloud data.

2. Sales Wave Analytics Prebuilt Dashboards and Datasets

The Sales Wave app includes prebuilt dashboards and datasets to accelerate your data exploration.

3. Set Up Permissions for the Sales Wave Analytics App

Set up your organization to use the Sales Wave app by enabling Wave and creating and assigning permission sets.

4. Create and Share the Sales Wave Analytics App

Follow these steps to create Sales Wave Analytics and start uncovering the value of your Salesforce data—fast.

5. Personalize Sales Wave Analytics with the Configuration Wizard

The Sales Wave Analytics configuration wizard lets you create the app to reflect how you prefer to view Salesforce data.

6. Upgrade the Sales Wave Analytics App

Take advantage of the latest Sales Wave Analytics features by upgrading your app every time we release a new version.

7. Update the Sales Wave Analytics Quota (Target) Dataset

If you answer "yes" to the configuration wizard question about whether your business tracks quotas at the user level, you need to update the Sales Wave Analytics Quota dataset so the app includes quotas data for your team.

8. Schedule the Sales Wave Analytics Daily Dataflow

Schedule a daily dataflow at a time outside standard business hours before you start exploring with Sales Wave Analytics.

9. Integrate Sales Wave Analytics With Salesforce

Make Sales Wave Analytics more usable by performing a variety of optional integrations and customizations.

10. Understand Sales Wave Analytics App Limitations

Sales Wave Analytics requires that Sales Cloud include specific data and supports a limited set of Salesforce objects.

11. Get to Know Sales Wave Analytics Data Terminology

To make the best use of Sales Wave Analytics, it's helpful to understand the metrics and terms used in the app.

About the Sales Wave Analytics App

Learn the benefits of Sales Wave Analytics before you create and use the app to explore your Sales Cloud data.

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 use Wave templated apps:

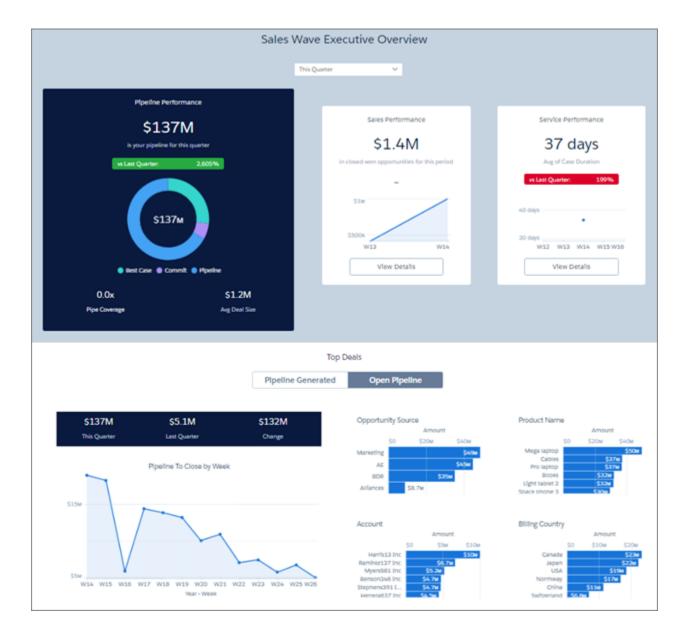
 Use Wave Analytics Templated Apps

To use Sales Wave Analytics:

 Access Sales Cloud Analytics Templates and Apps

To create and manage Wave apps:

- Manage Wave Analytics Templated Apps
- Edit Wave Analytics Dataflows



Salesforce created the Sales Wave Analytics app to make it easy to analyze Sales Cloud data. Whether you're a sales manager, a rep, or on the operations side, Sales Wave Analytics gives you a new level of insight into your Salesforce data.

- Managers start with an overview of team key performance indicators (KPIs) and can dive right in to specifics about quota, changes
 in the pipeline, 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: using a built-in configurator, answer a few questions about how you prefer to view data, and Wave takes care of the rest. Once you've created the app, use its prebuilt datasets and dashboards to explore Sales Cloud data from any device that supports Wave.

You get actionable insights fast from your sales data using the intuitive Wave interface. And you can drill deeper into key aspects of your business by customizing Sales Wave Analytics around your business needs.



Note: Your organization can use the Sales Wave app with or without the Salesforce Wave Analytics platform by purchasing a Sales Wave Analytics license. Each Salesforce Wave Analytics license includes a Sales Wave Analytics App license.

Sales Wave Analytics Prebuilt Dashboards and Datasets

The Sales Wave app includes prebuilt dashboards and datasets to accelerate your data exploration.



Note: These dashboards and datasets are part of Sales Wave Analytics only if you created your app after July 2016. If you use an older version of Sales Wave Analytics, create the app again to see these dashboards and datasets.

Sales Wave Analytics Dashboards

The prebuilt dashboards in the Sales Wave app contain key performance indicators that help you get value from your Salesforce data—fast. The dashboards let you manage pipeline and forecast and understand key business performance drivers, visualize trends, assign actions. They also help you get fast answers to questions you have about business results.

USER PERMISSIONS

To use Wave apps:

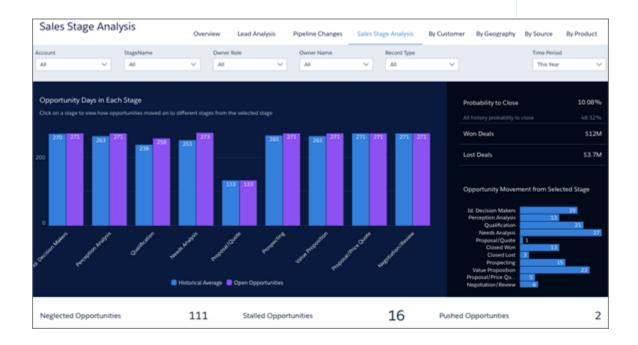
 Use Wave Analytics Templated Apps

To use Sales Wave Analytics:

 Access Sales Cloud Analytics Templates and Apps

To create and manage Wave apps:

- Manage Wave Analytics Templated Apps
- Edit Wave Analytics Dataflows



The following table guides you through Sales Wave Analytics dashboards, which give you an immediate picture of how your business is doing. You can also explore further on your own at any point. To learn more about exploring data in Wave, see Explore and Visualize Your Data.



Note: The dashboards and datasets included in Sales Wave Analytics differ depending on how you answer configuration wizard questions when you create the app.

Table 2: Sales Wave Dashboards

Dashboard Name	Contents	Target User Role	Special Requirements
Executive Overview	Sales execs start here with summary and detailed views of pipeline, sales, and service performance as well as top deal closings and lead conversion rates.	Executive	
Sales Rep Overview	Sales reps start here for a summary of KPIs. Review quota attainment, bookings, and pipeline for a particular time period. Also see key opportunities for accelerating deal closing and number of activities completed in a week.	Sales representative	Must import quotas data to Sales Wave Analytics to see quota attainment.
Opportunity Discovery	Focuses on products sold or not sold to accounts and identifies whitespace or cross-sell and up-sell opportunities.	Sales representative	Must import quotas data to Sales Wave Analytics for this dashboard to appear.
Manager Overview	Sales managers start here for an overview of team KPIs. Review team quota attainment, pipe coverage, bookings, pipe generation, and average sales cycle time. You can select individual reps to review their performance.	Sales manager	 Must import quotas data to Sales Wave Analytics to see quota attainment. Must answer "yes" to configuration wizard question about whether you use the Products object to see pipe coverage.
Pipeline Changes	Highlights pipeline changes, such as closed won/lost, moved in/out, and amount reduced/expanded.	Sales manager	Must import quotas data to Sales Wave Analytics to see quota attainment.
Team Productivity	Shows how reps on your team compare across key performance categories such as close/won, quota attainment, and sales cycle time.	Sales manager	Must import quotas data to Sales Wave Analytics to see quota attainment.
Quota Progress	Helps your team hit its quota by letting you drill into key opportunities.	Sales manager	Must import quotas data to Sales Wave Analytics to see quota attainment.

Dashboard Name	Contents	Target User Role	Special Requirements
Leader Overview	Learn pipe coverage, quota attainment, open pipeline, and other results for each role on the team.	Sales leader	Must import quotas data to Sales Wave Analytics to see quota attainment.
Leader Pipeline Changes	Get an up-to-date view of pipeline changes: the size of the pipe at the beginning and end of a selected time period and what's moved in and out.	Sales leader	None.
Lead Analysis	Analyze lead conversion, including open and converted leads and conversion rate.	Sales leader	None.
Sales Stage Analysis	Analyze efficiency of each phase of the sales process. Discover if deals are moving smoothly, where there are bottlenecks, what's happening to speed up or slow down closing, and so on.	Sales leader	None.
Performance by Geography	Analyze closed business and discover trends across geographic locations.	Sales leader	None.
Performance by Source	Analyze closed business and discover trends across sources.	Sales leader	None.
Performance by Customer	Analyze closed business and discover trends across customers.	Sales leader	None.
Performance by Product	Analyze closed business and discover trends across products.	Sales I	None.

Sales Wave Analytics Datasets

The table describes Sales Wave Analytics datasets become part of the app when you create it.

Table 3: Sales Wave Datasets

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.

Dataset Name	Contents	Special Requirements
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 400 with quotas data to update this dataset if you use quotas in Sales Wave Analytics.
Quota with Users	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 Wave Analytics through the configuration wizard.
Events	Data about events.	None.
Tasks	Data about tasks.	None.
Leads	Salesforce leads data, including converted leads connected to accounts and opportunities.	Only included if you choose to import leads data to Sales Wave Analytics through the configuration wizard.
Campaigns	Salesforce campaign data, including opportunities connected to campaigns.	Only included if you choose to import campaigns data to Sales Wave 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 Wave Analytics through the configuration wizard.

Set Up Permissions for the Sales Wave Analytics App

Set up your organization to use the Sales Wave app by enabling Wave and creating and assigning permission sets.

Each Sales Wave Analytics App license is a single-user license that provides access to Sales Wave Analytics. It includes a single Sales Wave 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.

License	Limit
Analytics Cloud - Sales Wave Analytics App	25 million rows when used without Analytics Cloud - Wave Analytics Platform license. Use of Sales Wave Analytics app license does not increase data limit for platform license.
Analytics Cloud - Additional Data Rows	100 million rows.

USER PERMISSIONS

To create and manage Wave apps:

- Manage Wave Analytics
 Templated Apps
- "Edit Wave Analytics Dataflows"

(1) Important: Sales Wave Analytics App 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 Wave Analytics app with or without the Salesforce Wave Analytics platform.

Each Sales Wave app user needs a Sales Analytics Apps permission set license. The Sales Analytics Apps permission set license enables the following permissions.

Table 5: Sales Wave Analytics User Permissions

Permission	Function
"Access Sales Cloud Analytics Templates and Apps"	Provides specific access to the Sales Wave Analytics App as well as the data and prebuilt dashboards included with it.
"Download Wave Analytics Data"	Download screenshots and data in tabular format through the Wave Analytics user interface.
"Edit Wave Analytics Dataflows"	Enables admins to upload and download JSON for existing data flows.
"Manage Wave Analytics Templated Apps"	Gives admins the ability to create Wave apps for users in their orgs, in this case the Sales Wave Analytics app.
"Use Wave Analytics Templated Apps"	Enables access to all Wave templated apps.

You assign the Sales Analytics Apps permission set license when you create and assign permission sets to users in your organization.

Sales Wave Analytics App Setup Process

To set up your organization to use Sales Wave Analytics, follow these steps:

- 1. Enable Wave for your organization.
- 2. Define the main Sales Wave Analytics user types in your organization.
- 3. Create permission sets to group related user permissions according to the needs of each user type.

When you create permission sets, you also select the Sales Analytics Apps permission set license.

4. Assign a permission set to an individual user, which also *auto-assigns* the Sales Analytics Apps permission set license to that user. For efficiency, you can assign a permission set to groups of users. You can assign multiple permission sets to a user.

After you've completed these steps, create and share Sales Wave Analytics with the users in your organization. Users can only explore Sales Wave Analytics dashboards and datasets after you've shared the app with them.



For best results, follow the steps for setting up the Sales Wave Analytics app in the order shown here.

IN THIS SECTION:

1. Identify Sales Wave Analytics App User Types

Identify Sales Wave Analytics app user types and the tasks they perform to assure that the setup process meets your team's analytics needs.

2. Enable Wave Analytics and Create Sales Wave Analytics Permission Sets

After Salesforce provisions you with the Sales Wave Analytics App license, enable Wave and create permission sets for Sales Wave Analytics users.

3. Assign Sales Wave Analytics Permission Sets to Users

Assign permission sets to one or more Sales Wave Analytics users—either one at a time, or in bulk—to give them access to the app.

Identify Sales Wave Analytics App User Types

Identify Sales Wave Analytics app user types and the tasks they perform to assure that the setup process meets your team's analytics needs.

The Sales Wave Analytics App license includes the permissions required by users in your organization to administer Sales Wave Analytics and to import, work with, and explore data. Set up the Sales Wave Analytics in the most useful way by thinking through the different ways people can access Sales Wave Analytics features and categorizing users into types. Here are a few examples.

In most cases, Sales Wave Analytics addresses the needs of two basic types of people on your team:

- Users, who view Sales Wave Analytics dashboards and datasets.
- Administrators or managers, who access all Sales Wave Analytics functionality to create apps, dashboards, and datasets, import data, and otherwise customize the Sales Wave Analytics experience.

Accordingly, when you set up Sales Wave 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 Sales Wave Analytics.
- A "manage" permission set that's reserved for a select few administrators and managers and contains permissions to use all Sales Wave Analytics features.

In the detailed Sales Wave Analytics app setup instructions that follow, we show you how to create and assign these two permission sets.

Enable Wave Analytics and Create Sales Wave Analytics Permission Sets

After Salesforce provisions you with the Sales Wave Analytics App license, enable Wave and create permission sets for Sales Wave Analytics users.

Enable the Wave Platform

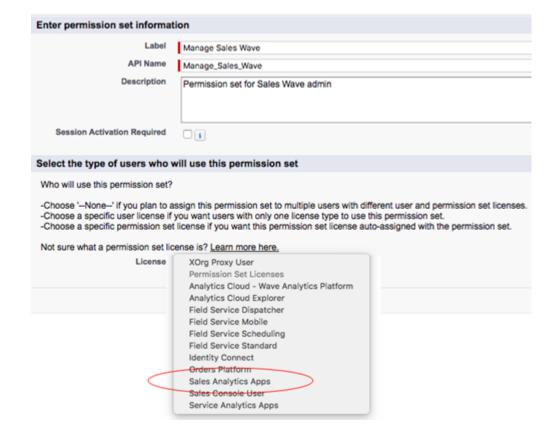
To use Sales Wave Analytics, first enable Wave Analytics for your organization.

- 1. In the Salesforce Setup menu, under administer, click **Analytics Cloud | Getting Started**
- 2. Click Enable Analytics

Create Permission Sets

The next step in the setup process is to create permission sets made up of Sales Wave Analytics user permissions. We show you how to create two permission sets:

- Manage Sales Wave Analytics permission set, for users who create and administer Sales Wave Analytics.
- View Sales Wave Analytics permission set, for users who explore app dashboards and datasets.
- 1. Create a Manage Sales Wave Analytics permission set, which gives users access to all app features.
 - **a.** In the Setup menu, under administer, click **Manage Users | Permission Sets** and then click **New**.
 - **b.** Enter Manage Sales Wave Analytics in the Label field, which automatically creates the API name, as well. You can change the API name, but it can't include spaces, end with an underscore, or have two consecutive underscores. You can also enter a description, but it's optional.
 - **c.** Select the permission set license to associate with this permission set by using the License drop-down option. In this case, select Sales Analytics Apps in the Permission Set License section.



USER PERMISSIONS

To create permission sets:

"Manage Profiles and Permission Sets"

If you select a specific permission set license, any user assigned to the permission set is *auto-assigned* the permission set license. If you leave the default of --None--, you must *manually* assign the permission set license to users before you can add them to the new permission set.

- **d.** Click **Save**. The Permission Set Overview page shows the new permission set. Now add user permissions to the set.
- e. Click **System Permissions** to open the list of user permissions enabled by the Sales Wave Analytics license. Then click **Edit**.
- **f.** Select the following user permissions: "Use Wave Analytics Templated Apps," "Access Sales Cloud Analytics Templates and Apps," "Manage Wave Analytics Templated Apps," and "Edit Wave Analytics Dataflows." Make sure to select all four. Click **Save**. You've successfully created a permission set that enables users to create and administer Sales Wave Analytics and access all app features. You can now assign it to users, which is covered in Assign Sales Wave Permission Sets.
- **2.** To create a View Sales Wave Analytics permission set, which enables users to view app dashboards and datasets, repeat the steps to create a Manage Sales Wave Analytics permission set, with these exceptions:
 - a. Name the permission set View Sales Wave Analytics.
 - **b.** Select only the "Use Wave Analytics Templated Apps" and "Access Sales Cloud Analytics Templates and Apps" user permissions and click **Save**. You've successfully created a permission set that enables users to view the Sales Wave Analytics dashboards and datasets. You can now assign it to users, which is covered in Assign Sales Wave Permission Sets on page 390.

You can build other permission sets out of the individual permissions enabled by the Sales Wave Analytics license, depending on the needs of users in your organization.

Assign Sales Wave Analytics Permission Sets to Users

Assign permission sets to one or more Sales Wave Analytics users—either one at a time, or in bulk—to give them access to the app.

- 1. In the Setup menu, under administer, click **Manage Users | Permission Sets**.
- Warning: Assign the Manage Sales Wave Analytics permission set sparingly because it lets users administer Sales Wave Analytics, which lets them create, edit, and delete the app. Assign it only to users who administer or manage the app.

USER PERMISSIONS

To assign a permission set to users:

"Assign Permission Sets"

Click the permission set you want to assign to users. If you've followed the instructions in Enable Wave Analytics and Create Sales Wave Analytics Permission Sets on page 389, select either the Manage Sales Wave Analytics or View Sales Wave Analytics permission sets.

- 3. Click Manage Assignments and click Add Assignments.
- **4.** Select the users who need to manage or use Sales Wave Analytics, and click **Done**. When you assign the permission set to users, Salesforce *auto-assigns* the Sales Analytics Apps *permission set license* to those users.

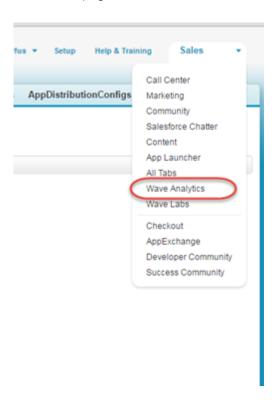
After you've completed these steps, create and share Sales Wave Analytics with the users in your organization. Users can only explore Sales Wave Analytics dashboards and datasets after you've shared the app with them.

Create and Share the Sales Wave Analytics App

Follow these steps to create Sales Wave Analytics and start uncovering the value of your Salesforce data—fast.

Using the built-in configuration wizard lets you create the Sales Wave Analytics app to reflect the way you like to view your sales data. If you're not happy with the result, you can delete the app and create it again quickly. You can also watch the video Create the Sales Wave App, Part 1: Using the Configuration Wizard (Lightning Experience), which covers the steps described here.

- 1. Log in to Salesforce.
- 2. Set Salesforce field-level security to enable the Wave Analytics Cloud 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.** Select **Wave Analytics** from the Force.com menu (top right of the Salesforce window).



4. Click **Create**, select **App**, select **Sales Wave Analytics**, and then click **Continue** to open the configuration wizard. The wizard presents a series of pages with questions and possible answers about your Salesforce data. Sales Wave Analytics personalizes the

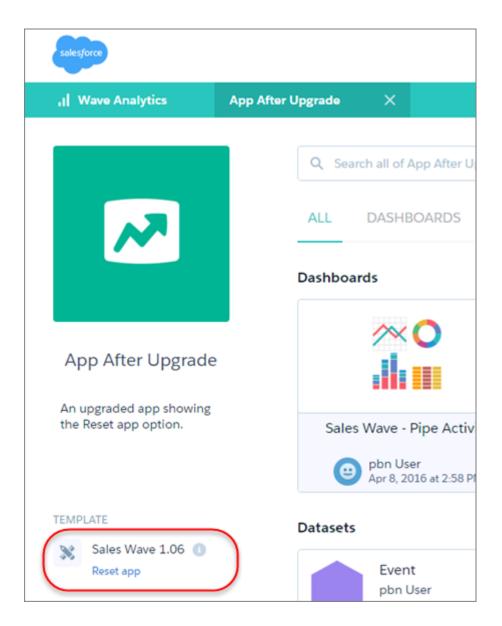
USER PERMISSIONS

To create and manage Wave apps:

- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"

wizard questions based on a check of your org's data. It also preselects default answers to the questions and recommends answers to some of them.

- Note: The original version of Sales Wave Analytics is also known as Sales Wave Analytics (Classic). Select that template to create an app with the previous set of dashboards based on the original dashboard designer.
- **5.** Answer the wizard questions according to how your company prefers to view data. Go to Personalize Sales Wave Analytics with the Configuration Wizard for complete instructions on using the wizard.
- **6.** Go to the last wizard question, which gives you the option of having Wave check to see if data in your org supports the answers you select. We recommend that you answer Yes and run the check, because app creation fails if your org doesn't contain the right data. Wave displays an error message that tells you what to do if required data is missing. Make any required changes in Salesforce, refresh your browser cache, and come back to the wizard and finish creating your app.
- **7.** Give your app a name that's easily recognizable to others in your company.
- **8.** Click **Create**. That starts a dataflow that creates the app and its assets (which include a dataflow definition file, datasets, and dashboards).
- **9.** The app creation process can take a few minutes. You can check the status of the dataflow: Open the Wave Home page, click the gear menu at the upper right of the page, and select Data Monitor. Then select Dataflow View in the pulldown at the upper left and look for your app.
- **10.** If you'd like to change your answers after looking at the dashboards, click the Reset app link on the left side of the app home page. The reset feature is available only after you've upgraded to the latest version of an app. See Upgrade the Sales Wave Analytics App on page 397.



Clicking **Reset app** opens the configuration wizard showing the answers you chose when you created the app. Choose new answers, and click Reset on the last page of the wizard. Sales Wave creates your app again using your new settings. You can also use the reset feature if you deleted dashboards and want to restore them.

- (1) Important: Resetting your app gets rid of dashboard customizations, including fields or objects added to the dataflow, 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. That way, you can copy them into your app if you reset it.
- **11.** You can also delete your app and start the app creation process all over again. To delete, position your cursor over the icon for the app and click Delete. Then click Yes to confirm deletion of all the assets associated with the app.
 - Note: If you delete your app, Wave doesn't retain the answers you selected when you created that version of the app, and you have to start the configuration wizard process all over again. If you wish to reuse the answers you selected, keep the app until you've completed recreating the app and then delete it.

12. 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 Wave Analytics Templated Apps and Access Sales Cloud Analytics Templates and Apps permissions enabled. (For more about Sales Wave app permissions, see Set Up Salesforce Permissions for the Sales Wave App). To share the app, roll your cursor over the icon for the app and click **Share**. Next, type in the names of team members, click **Add**, and click **Save**.

If your business tracks quotas at the user level, update the Quota (Target) dataset that's created along with the app before you start exploring your data. See Update the Sales Wave Analytics Quota (Target) Dataset.

When you first create the app, Sales Wave Analytics dashboards show the number associated with role name in the hierarchy. 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. See Integrate Sales Wave Analytics With Salesforce for details.

SEE ALSO:

Personalize Sales Wave Analytics with the Configuration Wizard Upgrade the Sales Wave Analytics App

Personalize Sales Wave Analytics with the Configuration Wizard

The Sales Wave Analytics configuration wizard lets you create the app to reflect how you prefer to view Salesforce data.

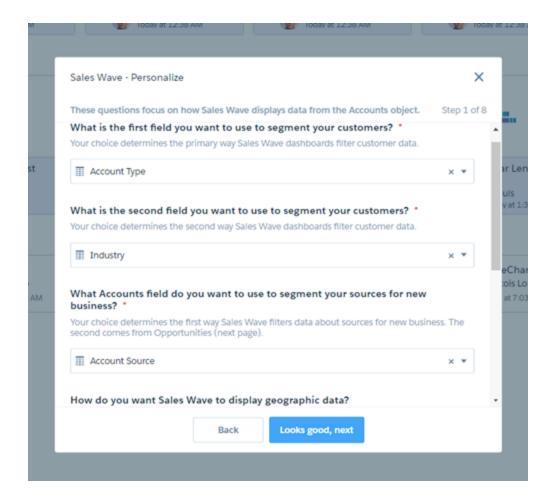
When you create the Sales Wave app, the built-in configuration wizard checks the data in your org and builds a series of questions and default and recommended answers based on that data. Depending on your data, you may not see all the questions described here and some may contain different default answers. Your answers determine how users view your sales data in Sales Wave Analytics's dashboards. Answer these questions according to the way your organization uses Sales Cloud data so the dashboards are as meaningful and useful as possible.

It starts with two pages of basic questions about how your organization uses the Salesforce Accounts

and Opportunities objects. In one of the questions about Opportunities, you're asked if you use the Products object. If you answer yes, you see a page with questions about how you use the Products object. The next several pages help you enhance Sales Wave Analytics with additional data. The last page lets you choose to have Sales Wave Analytics check your org to be sure your data supports the answers you select.

USER PERMISSIONS

- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"



The wizard includes instructions to help you choose appropriate answers, and many questions are self-explanatory. Here are tips for how to use the wizard, including details about how to answer more complex questions.

General Guidelines

Follow these guidelines to assure a great result when you use the wizard.

- 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, Sales Wave Analytics 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 analytics 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.

Page 1: Set How Sales Wave Analytics Displays Accounts Data

Questions on this page let you determine how Sales Wave Analytics filters customer data, sources for new business, and geographic data. All are required.

- Question 3: Asks about the field from Accounts you use to track if an account is new business. This is one way Sales Wave Analytics filters new business data. The second way comes from a field in the Opportunities object, which you choose on Page 2.
- Question 4: Asks about how you want to view geographic data. You can select either a bar chart or a standard world or country map.
- Question 5: Asks about the field from Accounts you want Sales Wave Analytics to use as the primary filter for geographic data. If you selected a world map in Question 4, be sure to select the field that contains country information. If you selected a country map, select the field that contains state information.

Page 2: Set How Sales Wave Analytics Displays Opportunities Data

Questions on this page let you determine how Sales Wave Analytics filters data about opportunities. It also asks about how your organization uses the Opportunities object. All are required.

- Question 2: Lets you choose the second way Sales Wave Analytics filters new business data. You determine the first in Page 1, Ouestion 3.
- Question 5: Asks if you track products at the opportunity level. If you do, answer yes and the Sales Wave Analytics wizard displays a separate page asking you about how you use the Products object.

Page 3: Set How Sales Wave Analytics Displays Products Data

This page displays only if you answer yes to the question on the previous page asking if you track products. If you answer no, you go directly to Page 4 and this page won't appear. The questions here let you determine how Sales Wave Analytics filters data about products.

Page 4: Enhance Datasets and Dashboards

This page asks you to about miscellaneous—but important—aspects of how you want Sales Wave Analytics to use and display data. It includes questions about your fiscal year start date, security settings, account and opportunity teams, and record types. All are required.

- Question 2: Many organizations keep data about their sales team's quotas in a tool outside Salesforce. If you want to import that data so it appears in Sales Wave Analytics dashboards, select Yes. You then have to import a previously created CSV file with the quotas data. See Update the Sales Wave Analytics Quota (Target) Dataset.
- Question 3: Lets you control Sales Wave Analytics user access to data. 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 as in Option 2. 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 4: Lets you control how Sales Wave Analytics handles account team information. Here are the options:
 - Option 1 lets you add account teams data to datasets.
 - Option 2 lets you add account teams data to datasets and also lets account team members see all members' records, not just their own.
 - Option 3 lets you exclude account teams data from datasets.
- Question 5: Lets you control how Sales Wave Analytics handles opportunity team information. Here are the options:
 - Option 1 lets you add opportunity teams data to datasets.

Option 2 lets you add opportunity teams data and also lets opportunity team members see all members' records, not just their
own.

- Option 3 lets you exclude opportunity teams data from datasets.
- Questions 6 and 7 let you choose to add record types from the Opportunities object to Sales Wave Analytics and indicate which record types to include.
 - Question 6: Answer Yes to add an Opportunity record type filter to dashboards. If you answer Yes, also answer Question 7.
 - Question 7: If you answered Yes to question 6, choose the Opportunity record types you want to include. To include all record types, leave the answer blank. If you select one or more record types, Sales Wave Analytics includes only those types and excludes any you don't select.

Page 5: Import Additional Accounts and Opportunities Data

The questions on this page let you add data from Accounts and Opportunities fields to datasets. Once the data is imported, you can view it in dashboards after first modifying them through the user interface or by changing the dashboard JSON. The questions ask you separately about fields including measures, which are numerical values, and dimensions, which are values expressed in words. All questions are optional.

Page 6: Add Objects

Select additional Salesforce objects you want to analyze in Sales Wave Analytics.

Pages 7 and 8: Add Fields From the Objects Selected on Page 6

Add fields to the datasets Sales Wave Analytics creates according to the way you answer the question on Page 6. You can select measures (numerical fields) and dimensions (text fields) from the objects you selected on Page 6.

Page 9: Check Your Org's Data

Go to the last wizard question, which gives you the option of having Wave check to see if data in your org supports the answers you select. We recommend that you answer Yes and run the check, because app creation fails if your org doesn't contain the right data. Wave displays an error message that tells you what to do if required data is missing. Make any required changes in Salesforce, refresh your browser cache, and come back to the wizard and finish creating your app.

SEE ALSO:

Create and Share the Sales Wave Analytics App

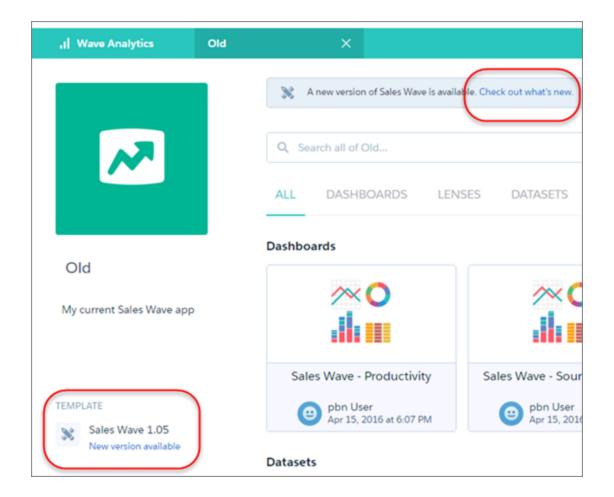
Upgrade the Sales Wave Analytics App

Take advantage of the latest Sales Wave 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.

USER PERMISSIONS

- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"



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.

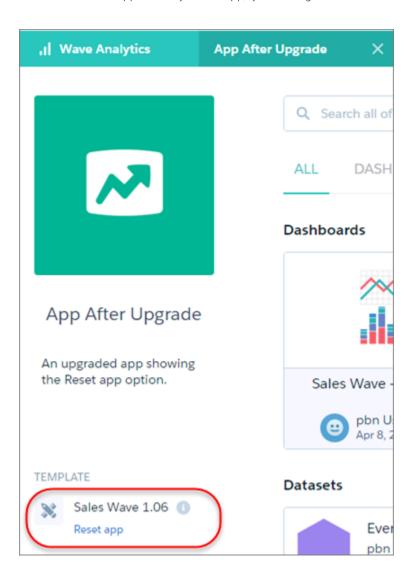
If you click **Upgrade current app**, you see a screen warning you that the upgrade option overwrites the current app and gets rid of any customizations you've made to it. If you're OK with that, check the box and click **Continue**. If you're not, click **Back** to return to the previous screen. Clicking **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, Wave 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.



Update the Sales Wave Analytics Quota (Target) Dataset

If you answer "yes" to the configuration wizard question about whether your business tracks quotas at the user level, you need to update the Sales Wave Analytics Quota dataset so the app includes quotas data for your team.

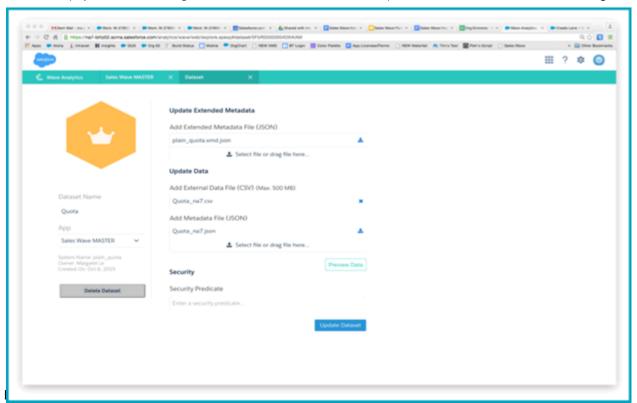
When you create Sales Wave Analytics, a Quota (Target) dataset is created along with the app. To explore how team members are tracking to their monthly, quarterly, or annual quotas in Sales Wave Analytics, you need to update the Quota dataset by completing the following steps. You can also watch the video • Create the Sales Wave App, Part 2: Upload the Quota CSV File and Schedule a Dataflow (Lightning Experience), which covers the steps described here.



USER PERMISSIONS

- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"
- Note: If your business doesn't track quotas at the user level, you can skip this step. Complete the step only if you answer "yes" to the configuration wizard question about whether your business tracks quotas at the user level.
- (1) 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 Wave Analytics Example .CSV File.
- 2. Save the file to a location that you can easily remember.
- 3. In Salesforce, go to the Wave Analytics home page and find the Quota (Target) dataset.
- 4. Hover over the dataset, then click **Edit**.

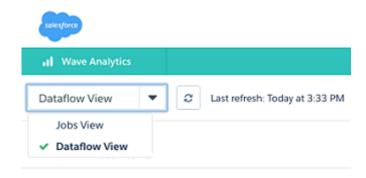
5. Salesforce displays the dataset editing screen for the Quota dataset. From the Update Data section, select Select file or drag file



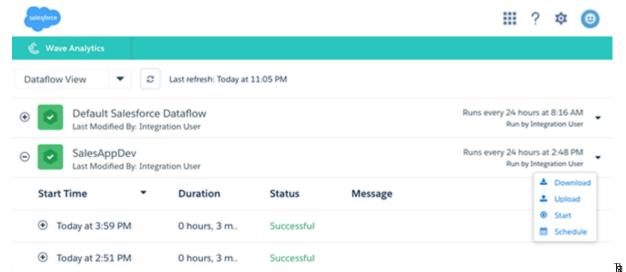
- **6.** In the dialog box that opens, navigate to the .CSV file you created in Step 1, and double-click it.
- 7. Click **Update Dataset**.
- **8.** If your fiscal period is different than calendar period, that is if it starts another date besides January 1, you also need to update the Quota Metadata file.
 - a. Copy the JSON from Sales Wave Analytics App 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 Wave metadata, the numeral "0" stands for January, "1" stands for February, and so on up to "11," which stands for December.
 - **c.** Save the file to your desktop and upload it to Sales Wave Analytics by dragging it to **Select file or drag file here...** in the Add Metadata File (JSON) section of the dataset editing screen.
 - d. Click Update Dataset.
- **9.** After uploading your quota data, re-run the dataflow to update the dashboards.
 - **a.** Click the Gear menu at the upper right of the Sales Wave screen and select Data Monitor.



b. Select Dataflow view from the menu at the top left of the Data Monitor screen.



c. Find your app; you may have to scroll down the page. Open the menu located on the far right of the screen next to the app icon and name, and click



it. The dataflow assures that Sales Wave has your company's latest sales data. You can learn more about dataflows from Schedule the Sales Wave App Daily Dataflow.

IN THIS SECTION:

Sales Wave Analytics Example .CSV File

Here's an example of the .CSV file you create to update the Sales Wave Analytics Quota (Target) dataset.

Sales Wave Analytics App Quota Dataset JSON File

Use this file to update the Sales Wave Analytics Quota dataset with your fiscal year start date if it's not January 1.

Sales Wave Analytics Example .CSV File

Here's an example of the .CSV file you create to update the Sales Wave 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

Create the file using a UTF-8 compliant text editor, and do not open the file with Microsoft Excel or another spreadsheet application. Field names are case-sensitive and must appear in your file exactly as shown here.

See Update the Sales Wave Analytics Quota (Target) Dataset on page 400.



Example:

```
QuotaAmount, StartDate, OwnerName, Username
5000000,2014-01-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-02-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-03-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-04-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-05-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-06-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-07-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-08-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-09-01, Priva Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-10-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-11-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-12-01, Priya Anand, user9.kthjbnrxifyo@yourDomain.com
5000000,2014-01-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-02-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-03-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-04-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-05-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-06-01, Nathan Sinha, user18.esm4nmdicvqc@yourDomain.com
5000000,2014-07-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-08-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-09-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-10-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
5000000,2014-11-01, Nathan Sinha, user18.esm4nmdicvqc@yourDomain.com
5000000,2014-12-01, Nathan Sinha, user18.esm4nmdicvgc@yourDomain.com
```

Sales Wave Analytics App Quota Dataset JSON File

Use this file to update the Sales Wave 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 Wave Analytics following the instructions in Update the Sales Wave Analytics Quota (Target) Dataset.

```
{
  "objects": [
    {
```

```
"connector": "CSV",
        "fullyQualifiedName": "Quota csv",
        "label": "Quota.csv",
        "name": "Quota csv",
        "fields": [
                "fullyOualifiedName": "OuotaAmount",
                 "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 Wave Analytics Daily Dataflow

Schedule a daily dataflow at a time outside standard business hours before you start exploring with Sales Wave Analytics.

When you create Sales Wave Analytics, the creation process includes a dataflow that imports the latest Sales Cloud data to Wave. You can schedule the dataflow to be rerun every day to assure that your app uses up-to-date Salesforce data. Schedule the dataflow to take place sometime outside normal business hours so the dataflow doesn't interrupt your use of the app.



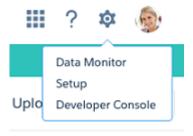
Note: The Sales Wave Analytics dataflow runs only once when you create the app. Schedule it to run daily so the app uses the latest sales data.

USER PERMISSIONS

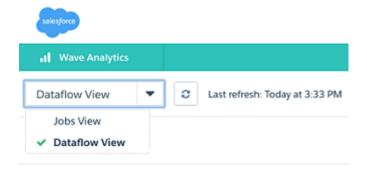
- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"

You can also watch the video • Create the Sales Wave App, Part 2: Upload the Quota CSV File and Schedule a Dataflow (Lightning Experience), which covers the steps described here.

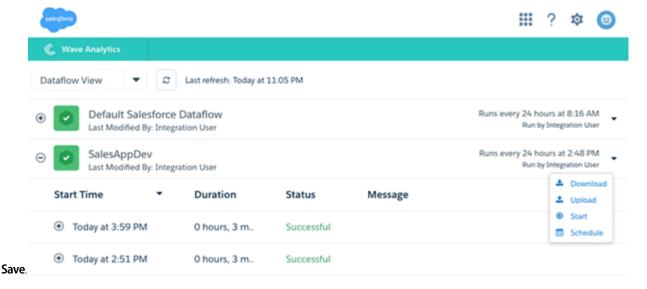
1. Go to **Wave Analytics** in the Force.com menu (top right of the Salesforce window) if you're not already there. Open the Data Monitor by clicking the gear icon at the upper right of the screen.



2. Select Dataflow view from the menu at the top left of the Data Monitor 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



Integrate Sales Wave Analytics With Salesforce

USER PERMISSIONS	
To use Wave templated apps:	"Use Wave Analytics Templated Apps"
To use Sales Wave Analytics:	"Access Sales Cloud Analytics Templates and Apps"
To create and manage Wave apps:	"Manage Wave Analytics Templated Apps"
	"Edit Wave Analytics Dataflows"
To edit a dataset's extended metadata (XMD) file:	"Edit Wave Analytics Dataflows"

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

Make Sales Wave Analytics more usable by performing a variety of optional integrations and customizations.

Overriding role names in datasets, embedding the "Sales Wave Analytics - Account Overview" dashboard in your Salesforce Accounts page, and expose Sales Wave Analytics as a tab in Salesforce Classic can make it easier for your team to use the app.

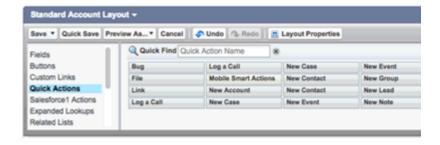
Add Actions to Sales Wave Analytics Dashboards

Sales Wave 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 Wave Analytics dashboards.

Override Role Name in Datasets

Sales Wave Analytics uses Salesforce Role Hierarchy to help you to understand your company's sales performance. When you first create the app, Sales Wave 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 Wave Analytics dashboards, update the Opportunity dataset XMD file with the following:

For more information, see Wave Analytics Extended Metadata (XMD) Reference.

Embed the Account Overview Dashboard in Account Page Layout

We've optimized the Sales Wave 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 Wave Analytics Available in a Salesforce Classic Tab

You can access Sales Wave Analytics by going to Wave Analytics in the Force.com menu and selecting the app from the Wave 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 Wave - Overview" dashboard. Complete instructions are available in Add a Wave Analytics Dashboard to a Visualforce Page.

Understand Sales Wave Analytics App Limitations

Sales Wave Analytics requires that Sales Cloud include specific data and supports a limited set of Salesforce objects.

Sales Wave Analytics Limitations

These limitations apply to Sales Wave Analytics.

Sales Wave AnalyticsData Requirements

Sales Cloud data must meet the following requirements for Sales Wave 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 Wave Analytics using the configuration wizard.
- There are two requirements if you choose to import Leads data to Sales Wave 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 Wave 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 Wave 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 Wave Analytics Support for Salesforce Objects and Fields

Sales Wave Analytics supports all Salesforce standard and custom objects and data. However, when you first create the app by default it includes only a predefined set of objects and fields.

Sales Wave creates a dataflow that exposes selected fields from the following standard Salesforce objects:

- Accounts
- Users
- Roles
- Opportunities
- Projects (Opportunities line item)
- Tasks
- Events

Sales Wave 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.

Sales Wave limits custom objects support to no more than 10 custom objects for each org that implements Sales Wave. 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 Sales Wave dataflow. For details, see Create Datasets with a Dataflow

Sales Wave Analytics Support for Non-Salesforce Data

Sales Wave 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 Wave Analytics, you must upload a CSV file with the details. For more information, see Update the Sales Wave App Quota (Target) Dataset.

Get to Know Sales Wave Analytics Data Terminology

To make the best use of Sales Wave Analytics, it's helpful to understand the metrics and terms used in the app.

Table 6: Sales Wave 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.
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 (\$).

Metric/Term	Formula	Description/Notes
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.

The Service Wave Analytics App

The Service Wave Analytics app gets you started fast with Wave 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 Sales Wave Analytics. If you haven't used Wave before, learn more about it from the Wave Analytics Library.

IN THIS SECTION:

1. About the Service Wave Analytics App

Salesforce created the Service Wave Analytics app 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 Wave Analytics Prebuilt Dashboards and Datasets

The Service Wave Analytics app includes prebuilt dashboards and datasets to accelerate data exploration for both service managers and agents.

- 3. Set Up Permissions for the Service Wave Analytics App
 - Set up your organization to use the Service Wave app by enabling Wave and creating and assigning permission sets.
- 4. Create and Share the Service Wave Analytics App
 - Follow these steps to create the Service Wave Analytics and start uncovering the value of your Salesforce data—fast.
- 5. Personalize Service Wave Analytics with the Configuration Wizard
 - Create Service Wave Analytics with the configuration wizard so the app reflects how your company prefers to view Service Cloud data.
- 6. Setup Service Wave Analytics CSAT Metrics Example
 - This scenario provides detail to help you answer the guestions on Page 4 of the Service Wave Analytics configuration wizard.
- 7. Upgrade the Service Wave AnalyticsApp
 - Take advantage of the latest Service Wave Analytics features by upgrading your app every time we release a new version.
- 8. Schedule the Service Wave Analytics Daily Dataflow
 - Schedule a daily dataflow at a time outside standard business hours before you start exploring with Service Wave Analytics.
- 9. Embed Service Wave Analytics Sidebar Dashboards in a Salesforce Page
 - Add Service Wave Analytics dashboards to Salesforce pages so your service team can get a complete view of their cases and take appropriate action.

10. Understand Service Wave Analytics App Limitations

Service Wave Analytics requires that Service Cloud include specific data, initially supports a limited set of Salesforce objects, and has other miscellaneous limitations.

About the Service Wave Analytics App

Salesforce created the Service Wave Analytics app 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.

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

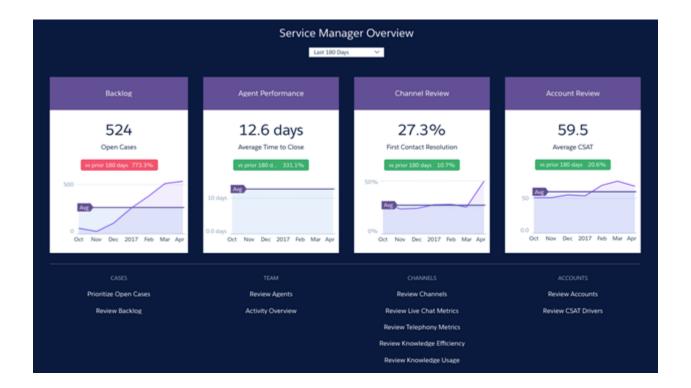
To use Wave templated apps:

 "Use Wave Analytics Templated Apps"

To use Service Wave:

 "Access Service Cloud Analytics Templates and Apps"

- "Manage Wave Analytics Templated Apps"
- "Edit Wave Analytics Dataflows"



Service Wave Analytics gives 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 Wave 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 Wave.

You get actionable insights fast from your data using the intuitive Wave interface. And you can drill deeper into key aspects of your business by customizing Service Wave around your needs.



Note: Your organization can use Service Wave without Salesforce Wave Analytics platform by purchasing a Service Wave license. A Service Wave license is included with the Wave platform license.

Service Wave Analytics Prebuilt Dashboards and Datasets

The Service Wave 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 Wave Analytics can differ. The ones you see depend on how you answer configuration wizard questions when you create the app.

Service Wave Analytics Dashboards

The prebuilt dashboards in the Service Wave app contain best practices that help you get value from your Salesforce data—fast. The dashboards let you manage pipeline and forecast and understand key business performance drivers, visualize trends, assign actions. They also help you get fast answers to questions you have about business results.

USER PERMISSIONS

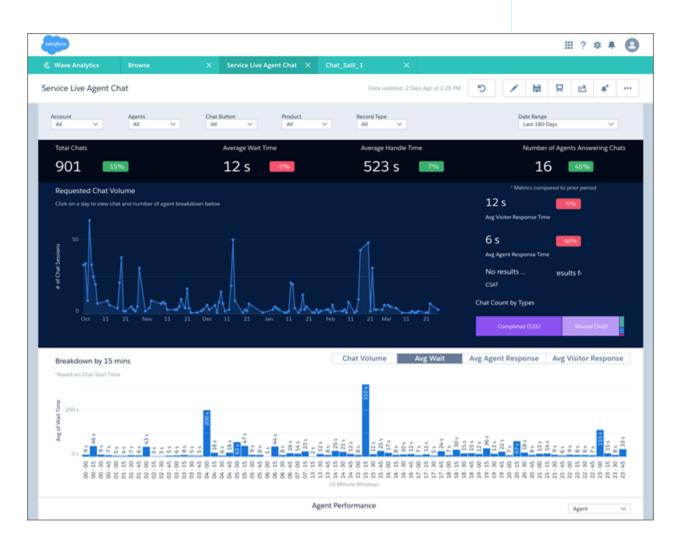
To use Wave apps:

 Use Wave Analytics Templated Apps

To use Service Wave:

 "Access Service Cloud Analytics Templates and Apps"

- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"



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 Wave dashboards. You can also explore further on your own at any point. To learn more about exploring data in Wave, see Explore and Visualize Your Data on page 199.

Table 7: Service Wave Dashboards

Dashboard Name	Contents	Target User
Manager Overview (About Service Wave)	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
Live Agent 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 Wave Analytics Datasets

The table lists all the possible standard datasets that become part of Service Wave when you create the app.

Table 8: Service Wave 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 Wave 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 Wave datasets with data about your team's use of LiveAgent for chat. Includes data from Live Agent transcripts, such as response times, numbers of messages per chat, and chat duration.

Dataset Name	Contents
Service Chat Transcript Event	One of two Service Wave datasets with data about your team's use of LiveAgent for chat. Includes data about Live Agent 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 Wave 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 Wave 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 Wave 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 Wave 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 Wave 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 Wave dashboards to function correctly.)

Set Up Permissions for the Service Wave Analytics App

Set up your organization to use the Service Wave app by enabling Wave and creating and assigning permission sets.

Each Service Wave Analytics App license is a single-user license that provides access to Service Wave. It includes a single Service Wave 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.

USER PERMISSIONS

- Manage Wave Analytics
 Templated Apps
- "Edit Wave Analytics Dataflows"

Table 9: Service Wave Data Storage Limits

License	Limit
Analytics Cloud - Service Wave Analytics App	25 million rows when used without Analytics Cloud - Wave Analytics Platform license. Use of Service Wave app license does not increase data limit for platform license.
Analytics Cloud - Additional Data Rows	100 million rows.

(1) Important: Service Wave Analytics App 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 Wave Analytics app with or without the Salesforce Wave Analytics platform.

Each Service Wave user needs a Service Analytics Apps permission set license. The Service Analytics Apps permission set license enables the following permissions.

Table 10: Service Wave User Permissions

Permission	Function
"Use Wave Analytics Templated Apps"	Enables access to all Wave templated apps.
"Access Service Cloud Analytics Templates and Apps"	Provides specific access to the Service Wave Analytics app as well as the data and prebuilt dashboards included with it.
"Manage Wave Analytics Templated Apps"	Gives admins the ability to create Wave apps for users in their orgs, in this case the Service Wave Analytics app.
"Edit Wave Analytics Dataflows"	Enables admins to upload and download JSON for existing data flows.

You assign the Service Analytics Apps permission set license when you create and assign permission sets to users in your organization.

Service Wave App Setup Process

To set up your organization to use Service Wave, follow these steps:

- 1. Enable Wave for your organization.
- **2.** Define the main Service Wave user types in your organization.
- **3.** Create permission sets to group related user permissions according to the needs of each user type. When you create permission sets, you also select the Service Analytics Apps permission set license.
- **4.** Assign a permission set to an individual user, which also *auto-assigns* the Service Analytics Apps permission set license to that user. For efficiency, you can assign a permission set to groups of users. You can assign multiple permission sets to a user.

After you've completed these steps, create and share Service Wave with the users in your organization. Users can only explore Service Wave dashboards and datasets after you've shared the app with them.

1 Tip: For best results, follow the steps for setting up the Service Wave app in the order shown here.

IN THIS SECTION:

1. Identify Service Wave Analytics App User Types

Identify Service Wave Analytics user types and the tasks they perform to assure that the setup process meets your team's analytics needs.

2. Enable Wave Analytics and Create Service Wave Analytics Permission Sets

After Salesforce provisions you with the Service Wave Analytics App license, enable Wave and create permission sets for Service Wave Analytics users.

3. Assign Service Wave Analytics Permission Sets to Users

Assign permission sets to one or more Service Wave Analytics users—either one at a time, or in bulk—to give them access to the app.

Identify Service Wave Analytics App User Types

Identify Service Wave Analytics user types and the tasks they perform to assure that the setup process meets your team's analytics needs.

The Service Wave Analytics App license includes the permissions required by users in your organization to administer Service Wave and to import, work with, and explore data. Set up the Service Wave in the most useful way by thinking through the ways people access features and categorizing users into types. Here are a few examples.

In most cases, Service Wave Analytics addresses the needs of two basic types of people on your team:

- Users, who view Service Wave dashboards and datasets.
- Administrators or managers, who access all Service Wave functionality to create apps, dashboards, and datasets, import data, and otherwise customize the Service Wave experience.

Accordingly, when you set up Service Wave, 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 Service Wave.
- A "manage" permission set that's reserved for a select few administrators and managers and contains permissions to use all Service Wave features.

In the detailed Service Wave app setup instructions that follow, we show you how to create and assign these two permission sets.

Enable Wave Analytics and Create Service Wave Analytics Permission Sets

After Salesforce provisions you with the Service Wave Analytics App license, enable Wave and create permission sets for Service Wave Analytics users.

Enable the Wave Platform

To use Service Wave, first enable Wave Analytics for your organization.

- 1. In the Salesforce Setup menu, under administer, click **Analytics Cloud | Getting Started**
- 2. Click Enable Analytics

Create Permission Sets

The next step in the setup process is to create permission sets made up of Service Wave user permissions. We show you how to create two permission sets:

- Manage Service Wave permission set, for users who create and administer Service Wave.
- View Service Wave permission set, for users who explore app dashboards and datasets.
- 1. Create a Manage Service Wave permission set, which gives users access to all app features.

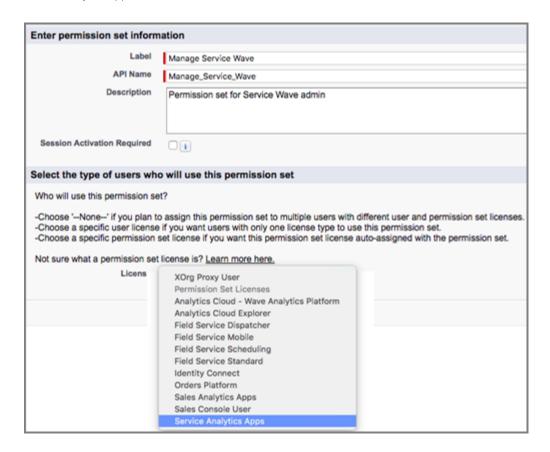
USER PERMISSIONS

To create permission sets:

 "Manage Profiles and Permission Sets"

- **a.** In the Setup menu, under administer, click **Manage Users | Permission Sets** and then click **New**.
- **b.** Enter Manage Service Wave in the Label field, which automatically creates the API name, as well. You can change the API name, but it can't include spaces, end with an underscore, or have two consecutive underscores. You can also enter a description, but it's optional.

c. Select the permission set license to associate with this permission set by using the License drop-down option. In this case, select Service Analytics Apps in the Permission Set License section.



If you select a specific permission set license, any user assigned to the permission set is *auto-assigned* the permission set license. If you leave the default of --None--, you must *manually* assign the permission set license to users before you can add them to the new permission set.

- **d.** Click **Save**. The Permission Set Overview page shows the new permission set. Now add user permissions to the set.
- e. Click System Permissions to open the list of user permissions enabled by the Service Wave license. Then click Edit.
- **f.** Select the following user permissions: "Use Wave Analytics Templated Apps," "Access Service Cloud Analytics Templates and Apps," "Manage Wave Analytics Templated Apps," and "Edit Wave Analytics Dataflows." Make sure to select all four. Click **Save**. You've successfully created a permission set that enables users to create and administer Service Wave and access all app features. You can now assign it to users, which is covered in Assign Service Wave Analytics Permission Sets to Users.
- **2.** To create a View Service Wave permission set, which enables users to view app dashboards and datasets, repeat the steps to create a Manage Service Wave permission set, with these exceptions:
 - a. Name the permission set View Service Wave.

b. Select only the "Use Wave Analytics Templated Apps" and "Access Service Cloud Analytics Templates and Apps" user permissions and click **Save**. You've successfully created a permission set that enables users to view the Service Wave dashboards and datasets. You can now assign it to users, which is covered in Assign Service Wave Analytics Permission Sets to Users.

You can build other permission sets out of the individual permissions enabled by the Service Wave Analytics App license, depending on the needs of users in your organization.

Assign Service Wave Analytics Permission Sets to Users

Assign permission sets to one or more Service Wave Analytics users—either one at a time, or in bulk—to give them access to the app.

- 1. In the Setup menu, under administer, click **Manage Users | Permission Sets**.
- **Warning:** Assign the Manage Service Wave permission set sparingly because it lets users administer Service Wave, which lets them create, edit, and delete the app. Assign it only to users who administer or manage the app.

USER PERMISSIONS

To assign a permission set to users:

"Assign Permission Sets"

Click the permission set you want to assign to users. If you've followed the instructions in Enable Wave Analytics and Create Service Wave Permission Sets on page 419, select either the Manage Service Wave or View Service Wave permission sets.

- 3. Click Manage Assignments and click Add Assignments.
- **4.** Select the users who need to manage or use Service Wave, and click **Done**. When you assign the permission set to users, Salesforce *auto-assigns* the Service Analytics Apps *permission set license* to those users.

After you've completed these steps, create and share Service Wave with the users in your organization. Users can only explore Service Wavedashboards and datasets after you've shared the app with them.

Create and Share the Service Wave Analytics App

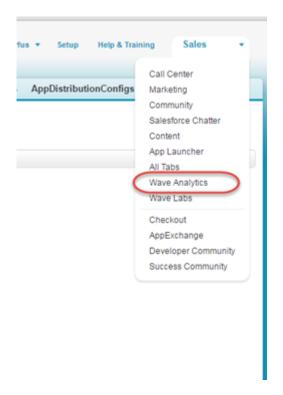
Follow these steps to create the Service Wave Analytics and start uncovering the value of your Salesforce data—fast.

Use the built-in configuration wizard to create the Service Wave app to reflect the way you like to view your Service Cloud data.

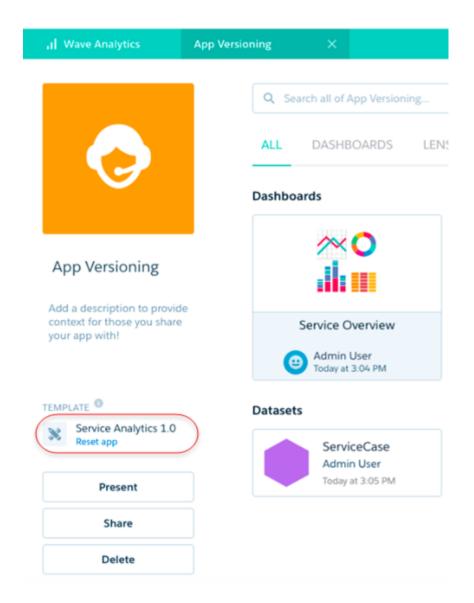
- 1. Log in to Salesforce.
- **2.** Set Salesforce field-level security to enable the Wave Analytics Cloud Integration User to see all fields you'd like your app to use. Integration users run the dataflow, and if they don't have proper field-level security permissions, the dataflow can fail.
 - **a.** 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.** Select **Wave Analytics** from the Force.com menu (top right of the Salesforce window).

USER PERMISSIONS

- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"



- **4.** Click **Create**, select **App**, select **Service Wave Analytics**, and then click **Continue**. Salesforce opens the Service Wave configuration wizard. The wizard presents a series of screens, each with a set of questions and their possible answers in pull-down menus. Service Wave checks the data in your orq and builds the questions and default answers in the configuration wizard.
 - (1) Important: Service Wave requires that your org has at least one closed case and one task. If you see an error message, add the data specified in the message to Salesforce, then repeat steps 1–3, and refresh your browser cache.
 - Tip: The original version of Service Wave is also known as Service Wave Analytics (Classic). Select that template to create an app with the previous set of dashboards based on the original dashboard designer.
- **5.** Answer the wizard questions according to how your company prefers to view data. For more information about how to answer these questions, see Personalize Service Wave Analytics with the Configuration Wizard.
- **6.** Go to the last wizard question, which gives you the option of having Wave check to see if data in your org supports the answers you select. We recommend that you answer Yes and run the check, because app creation fails if your org doesn't contain the right data. Wave displays an error message that tells you what to do if required data is missing. Make any required changes in Salesforce, refresh your browser cache, and come back to the wizard and finish creating your app.
- **7.** Give your app a name that's easily recognizable to others in your company.
- **8.** Click **Create**. That starts a dataflow that creates the app and its assets (which include a dataflow definition file, datasets, and dashboards).
- **9.** The app creation process can take a few minutes. You can check the status of the dataflow: Open the Wave Home page, click the gear menu at the upper right of the page, and select Data Monitor. Then select Dataflow View in the pulldown at the upper left and look for your app.
- **10.** If you'd like to change your answers after looking at the dashboards, click the Reset app link on the left side of the app home page. The reset feature is available only after you've upgraded to the latest version of an app. See Upgrade the Service Wave AnalyticsApp.



Clicking **Reset app** opens the configuration wizard showing the answers you chose when you created the app. Choose new answers, and click Reset on the last page of the wizard. Your app is created again using your new settings. If you deleted dashboards and want to restore them, you can also use the reset feature.

- (1) Important: Resetting your app gets rid of any dashboard customizations. That includes fields or objects added to the dataflow, actions you enable, or changes to security settings and dashboard colors and labels. If you've made any customizations, we recommend that you save copies of your dataflow definition file or dashboards. That way, you can copy them into your app if you reset it.
- **11.** You can also delete your app and start the app creation process all over again. To delete, position your cursor over the icon for the app and click Delete. Then click Yes to confirm deletion of all the assets associated with the app.
 - Note: If you delete your app, Wave doesn't retain the answers you selected when you created that version of the app, and you have to start the configuration wizard process again. If you wish to reuse the answers you selected, keep the app until you've completed recreating the app and then delete it.

12. 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 Wave Analytics Templated Apps and "Access Service Cloud Analytics Templates and Apps" permissions enabled. (For more about Service Wave app permissions, see Set Up Permissions for the Service Wave Analytics App). To share the app, roll your cursor over the icon for the app and click **Share**. Next, type in the names of team members, click **Add**, and click **Save**.

Taking a few extra steps can make Service Wave better serve the needs of your organization:

- Embed the sidebar dashboards on page 430 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 Wave dataset builder on page 123. Service Wave 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 Wave does not support data sources external to Salesforce or registering new datasets from the app-generated dataflow.
- Add objects, fields, and filters to Service Wave. Edit the app's dataflow on page 52 to include standard and custom objects and fields from Salesforce. For complete instructions about adding data to Wave, see Wave Analytics Data Integration.
- Note: To create new datasets with just the Service Wave app license, use the Wave dataset builder. Wave associates new datasets created using the builder with the Wave default dataflow. If you edit the dataflow for the Service Wave app to register new datasets, you can't access the resulting datasets.

Personalize Service Wave Analytics with the Configuration Wizard

Create Service Wave Analytics with the configuration wizard so the app reflects how your company prefers to view Service Cloud data.

When you create Service Wave Analytics, the built-in configuration wizard checks the data in your org and builds a series of questions and default answers based on that data. Depending on your data, you may not see all the questions described here and some may contain different default answers

Your answers let Service Wave know how you want users to view your Service Cloud data in the app's dashboards. Answer these questions according to the way your organization uses Service Cloud data so the dashboards are as meaningful and useful as possible.

USER PERMISSIONS

To create and manage Wave apps:

- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"

The wizard includes instructions to help you choose appropriate answers, and many questions are self-explanatory. Here are tips for how to use the wizard, including details about how to answer more complex questions.

General Guidelines

Following these guidelines can help you get a great result when you use the wizard.

- 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 Wave 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 analytics 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.

Page 1: Determine How Service Wave Displays Case Metrics

Choose the metrics that are important to your organization about the duration of cases, such as how quickly they're resolved and if they're resolved within the bounds of any service level agreement (SLA). All are mandatory and allow you to make only a single selection, except the following:

- 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 Wave 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 use the Business Hours field in the Cases object, Service Wave can include a business hours case duration calculation. See the question about business hours on Page 3 of the wizard
- Question 2, optional: Asks if you use other (secondary) fields related to case duration. You can select multiple fields, for example, *Duration with customer* or *Duration with agent*.
- Question 3: 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). Then, find and replace the field name in Service Wave dashboard JSON (see Wave Analytics Dashboard
 JSON Guide).
- Question 4: 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.

Page 2: Control How the App Drills Into Your Data

Tell Service Wave how you prefer to drill down into data about the status, severity, owner, reason, and type of cases. Also choose how Service Wave displays data about service cases' channels of origin and if they're resolved at first contact. All questions are mandatory and let you select only a single answer, except the following:

- Question 11: Asks if you use queues to assign case ownership. Answer Yes only if you have a queue in your Service Cloud org and have at least one case attached to the queue.
 - (1) Important: If you answer Yes and you do not have queues and a case attached to the queue, Service Wave creation fails. The optional data check you can run at the end of the wizard returns an error message it you answer Yes and don't use queues.
- Question 12, optional: Select measure fields from the Cases object not addressed in previous questions so your dashboards drill into measures that are important to your organization. Measures are quantitative values like case resolution time, number of open and closed cases, and revenue.
- Question 13, optional: Select dimension fields from the Cases object not addressed in previous questions so your dashboards drill into dimensions that are important to your organization. Dimensions are qualitative values, such as date, region, and product name.

Page 3: Select Report Areas

Choose the areas of service data you want Service Wave dashboards to report on. These might include customer satisfaction (CSAT), use of knowledge resources and telephony, and whether cases are attached to sales opportunities. All questions are mandatory and let you make only a single selection.

• Question 2: Asks if you track CSAT score. If you answer Yes, the wizard presents a page of questions about how you track CSAT. See Page Four: Set Up CSAT Metrics.

- Question 3: Asks if you use knowledge resources in the support process. If you answer Yes, the wizard presents a page of questions about how you use knowledge. See Page Five: Set Up Knowledge Data.
 - (1) Important: If your Service Cloud org does not use knowledge and you answer Yes, Service Wave creation fails. The optional data check you can run at the end of the wizard returns an error message it you answer Yes and don't use knowledge.
- Question 4: Asks about use of telephony in the support process. If you answer Yes, Service Wave creates a dashboard related to telephony.
 - (1) Important: You must track telephone usage data on the Task object for Service Wave to successfully create a telephony dashboard. Your org must have telephony data in standard activity fields on the Task object, such as call duration, call object identifier, call result, and call type. Answer Yes if you store call data in these fields. Answer No if you store call data in custom objects or outside Salesforce. If your Service Cloud org does not track telephone usage data and you answer Yes, Service Wave creation fails. The optional data check you can run at the end of the wizard returns an error message it you answer Yes and don't use telephony.
- Question 7: Asks if you use the Event object to track case activity. Select Yes if your org uses the Event object, and Service Wave adds data from the object to datasets. Defaults to No.
 - (1) Important: If your Service Cloud org does not use events and you answer Yes, Service Wave creation fails. The optional data check you can run at the end of the wizard returns an error message it you answer Yes and don't use events.
- Question 8: Asks if you use business hours to calculate case durations. Select Yes if your org uses the Business Hours field, and Service
 Wave adds a business hours case duration calculation to datasets. Defaults to "No, need business hours visibility" if the Analytics
 Cloud Integration User profile does not enable visibility to Business Hours. If that's the case and you use Business Hours, edit field-level
 security to provide the correct visibility for the Business Hours field on the Cases object, as described in Create and Share the Service
 Wave Analytics App on page 421. After doing that, the next time you run the wizard you'll be able to select Yes to add the business
 hours calculation to Service Wave datasets.
 - (1) Important: If your Service Cloud org does not use business hours and you answer Yes, Service Wave creation fails. The optional data check you can run at the end of the wizard returns an error message it you answer Yes and don't use business hours.
- Question 9: Asks if you use Live Agent for chat. Select Yes if your company uses LiveAgent, and Service Wave adds a chat dashboard.
 - Important: If you do not use Live Agent and you answer Yes, Service Wave creation fails.
- Question 10: Asks if you use Omni-Channel to create and route work items. Select Yes if your company uses Omni-Channel, and Service Wave adds datsets with Omni-Channel data.
 - **(1) Important**: If you do not use Omni-Channel and you answer Yes, Service Wave creation fails.

Page 4: Set Up CSAT Metrics (Optional)

Tell Service Wave how you track CSAT so that information is surfaced correctly in dashboards. You see this page—and must answer its questions—only if you answer Yes to Question 1 on Page 3. Follow these instructions carefully to assure success. See Setup Service Wave Analytics CSAT Metrics Example for detail.

- Question 1: Asks which object you use to track CSAT. Service Wave defaults to the Cases object. If you use a different object to track CSAT, select it from the pick list.
- Question 2: Asks you to select the field from the object selected in Question 1 you use to track CSAT.

• Question 3, optional: Answer only if you select an object other than Cases in Question 1. Service Wave has to create a join between that object and Cases for you to see CSAT data in dashboards. (For an example join, see Setup Service Wave Analytics CSAT Metrics Example.) Enter the API name the field where you store case identification information. Usually, that's the 'ld' 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, the API name is shown in the field name column. 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 4, optional: Answer only if you select an object other than Cases in Question 1. Determine which field from the object selected in Question to use for the other side of the join with the Cases object. Use the field that contains the case ID. Enter the API name for the field here.

To find the API name for the field, go to Setup-->AnyObject-->Fields. In the top chart showing standard fields, the API name is shown in the Field Name column. The chart further down for custom fields includes an API Name column.

Page 5: Set Up Knowledge Data (Optional)

You see this page, and must answer its questions, only if you answer Yes to Question 2 on Page 3. Asks how you'd like Service Wave to surface data about knowledge articles in dashboards.

• Question 1, optional: Asks about the article type you want to see data about in Service Wave dashboards. Choose only a single article type.

Page 6: Check Your Org's Data

Go to the last wizard question, which gives you the option of having Wave check to see if data in your org supports the answers you select. We recommend that you answer Yes and run the check, because app creation fails if your org doesn't contain the right data. Wave displays an error message that tells you what to do if required data is missing. Make any required changes in Salesforce, refresh your browser cache, and come back to the wizard and finish creating your app.

Setup Service Wave Analytics CSAT Metrics Example

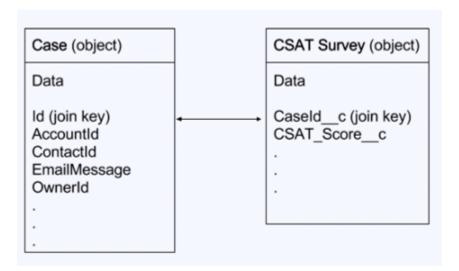
This scenario provides detail to help you answer the questions on Page 4 of the Service Wave Analytics configuration wizard.

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 Wave 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.



Upgrade the Service Wave AnalyticsApp

Take advantage of the latest Service Wave 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, Wave 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

USER PERMISSIONS

- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"

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.

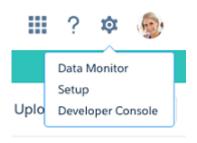
If the link below the app name says "Reset app," you're using the latest version and don't need to upgrade.

Schedule the Service Wave Analytics Daily Dataflow

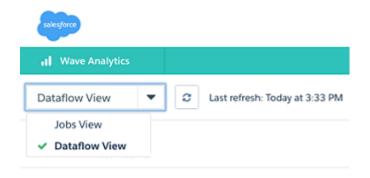
Schedule a daily dataflow at a time outside standard business hours before you start exploring with Service Wave Analytics.

When you create Service Wave Analytics, the creation process includes a dataflow that imports the latest Service Cloud data to Wave. You can schedule the dataflow to be rerun every day to assure that your app uses up-to-date Salesforce data. Schedule the dataflow to take place sometime outside normal business hours so the dataflow doesn't interrupt your use of the app.

- Note: The Service Wave dataflow runs only once when you create the app. Schedule it to run daily so the app uses the latest services data.
- **1.** Go to **Wave Analytics** in the Force.com menu (top right of the Salesforce window) if you're not already there. Open the Data Monitor 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 Monitor screen.

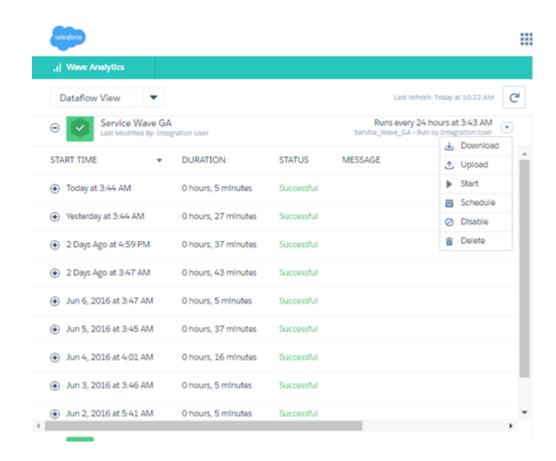


USER PERMISSIONS

To create and manage Wave apps:

- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"

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**.



Embed Service Wave Analytics Sidebar Dashboards in a Salesforce Page

To create a Visualforce page	"Customize Application"
To edit a dataset's extended metadata (XMD) file:	"Edit Wave Analytics Dataflows"
	"Edit Wave Analytics Dataflows"
To create and manage Wave apps:	"Manage Wave Analytics Templated Apps"
To use Service Wave:	"Access Service Cloud Analytics Templates and Apps"
To use Wave templated apps:	"Use Wave Analytics Templated Apps"
USER PERMISSIONS	

EDITIONS

Available in Salesforce Classic and Lightning Experience.

Available for an extra cost in **Enterprise**, **Performance**, and **Unlimited** Editions. Also available in **Developer** Edition.

Add Service Wave Analytics dashboards to Salesforce pages so your service team can get a complete view of their cases and take appropriate action.

Service Wave 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 Wave 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. Here's how.

- 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.
- - **a.** Replace Your_dashboard_ID with the dashboard ID for the sidebar. You can find the dashboard ID by opening the sidebar in Service Wave; 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 Wave Analytics (Classic) app built with the old dashboard designer, use this syntax: filter="{'ServiceCase1': {'AccountId': ['{!case.AccountId}']}}".
 - b. If you're using the latest version of Service Wave Analytics based on the new dashboard designer, use this syntax:
 :filter="{'datasets' : {'ServiceCase1': [{'fields':['AccountId'], 'filter': {
 'operator':'in', 'values': '{!case.AccountId}'} }]}}".

In both 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 Wave Analytics (Classic) with previous dashboards built with the old dashboard designer, use this syntax: filter="{'ServiceCase': {'Reason': ['{!case.Reason}'],'Product_c': ['{!case.Product_c}']}}".
 - b. If you use the latest version of Service Wave Analytics based on the new dashboard designer use this syntax: filter="{'datasets' : {'ServiceCasel': [{'fields':['Reason'], 'filter': { 'operator':'in', 'values': '{!case.Reason}'}}, {'fields':['Product__c'], 'filter': { 'operator':'in', 'values': '{!case.Product__c}' }}]}}".

In both 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 Wave Analytics (Classic) with previous dashboards built with the old dashboard designer, use this syntax: 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 Wave Analytics based on the new dashboard designer use this syntax: filter="{'datasets' : {'ServiceCase1': [{'fields':['Id'], 'filter': { 'operator':'in', 'values': '{!case.Id}'} }]}}".

In both 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 Wave Analytics App Limitations

Service Wave Analytics requires that Service Cloud include specific data, initially supports a limited set of Salesforce objects, and has other miscellaneous limitations.

Service Wave Limitations

The following limitations apply to Service Wave.

Service Wave Data Requirements

Service Cloud data must include at least one each of the following for Service Wave dashboards to function correctly: Events, tasks, closed cases, or published articles attached to a case or an opportunity.

Custom formula fields on the Cases object must exist or be created for the SLA missed (text) and FCR (Boolean) fields.

Service Wave Support for Salesforce Objects and Fields

Service Wave supports all Salesforce standard and custom objects and data. However, 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
- Events
- User
- UserRole
- Task

Service Wave also uses fields from the following options depending on the answers you select in the configuration wizard:

- CaseArticle
- CaseHistory
- CSAT (customer object)
- Group
- Single Knowledge article (custom object)
- RecordType
- Opportunity

Service Wave 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 Wave limits custom objects support to no more than 10 custom objects for each org that implements Service Wave. 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 Wave dataflow. For details, see Create Datasets with a Dataflow

More Service Wave Limitations

Service Wave 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.

The Field Service Wave Analytics App

The Field Service Wave Analytics app brings the power of Wave Analytics to Field Service Lightning on any device that supports Wave.

With intuitive visualizations based on your field service data, Field Service Wave 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 Wave 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 Wave Analytics. If you haven't used Wave before, learn more about it from the *Wave Analytics Library*.

IN THIS SECTION:

1. About the Field Service Wave Analytics App
Learn the benefits of Field Service Wave Analytics before you create and use the app to help manage field service operations.

Field Service Wave Analytics Prebuilt Dashboards
 The Fields Service Wave app includes prebuilt dashboards to accelerate Field Service Lightning data exploration.

3. Set Up and Create the Field Service Wave Analytics App
Set up Field Service Wave Analytics app permissions, make sure your org is enabled to use Field Service Lightning, and assign field level security correctly so you can create and share the app.

About the Field Service Wave Analytics App

Learn the benefits of Field Service Wave Analytics before you create and use the app to help manage field service operations.

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 use Wave templated apps:

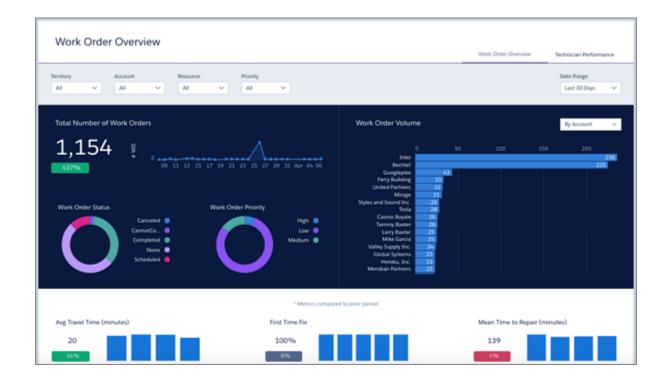
 "Use Wave Analytics Templated Apps"

To use Field Service Wave Analytics:

 "Access Service Cloud Analytics Templates and Apps"

To create and manage Wave apps:

- "Manage Wave Analytics Templated Apps"
- "Edit Wave Analytics Dataflows"



Salesforce created the Field Service Wave Analytics app to help service managers analyze customer service data their teams generate when they use Field Service Lightning. The app is bundled with the Service Wave Analytics app.

(1) Important: Your org must be enabled to use Field Service Lightning. Wave datasets are based on Field Service Lightning's new object model.

We've designed Field Service Wave 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.

Wave automatically creates the app for you based on your Field Service Lightning data. Select the Field Service Wave Analytics template, click Create, and Wave does the rest. You get actionable insights fast from your field service data using the intuitive Wave interface. And you can drill deeper into key aspects of service engagements and technician performance by customizing Field Service Wave around your business needs.

7

Note: Your organization can use Field Service Wave with or without the Salesforce Wave Analytics platform by purchasing a Service Wave Analytics license. Each Salesforce Wave Analytics license includes a Service Wave Analytics license.

Field Service Wave Analytics Prebuilt Dashboards

The Fields Service Wave app includes prebuilt dashboards to accelerate Field Service Lightning data exploration.

USER PERMISSIONS

To use Wave apps:

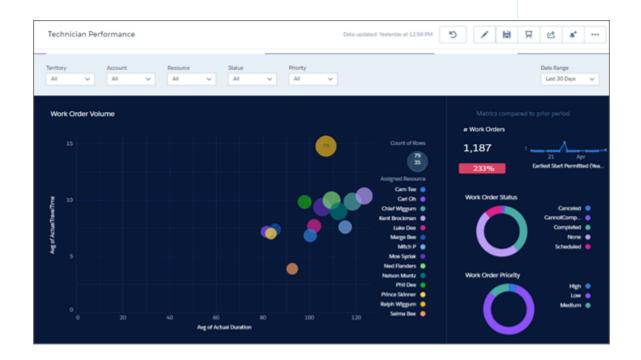
 Use Wave Analytics Templated Apps

To use Field Service Wave Analytics:

 "Access Service Cloud Analytics Templates and Apps"

To create and manage Wave apps:

- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"



The prebuilt dashboards in the Fields Service Wave 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 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 Wave dashboards. You can also explore further on your own at any point. To learn more about exploring data in Wave, see Explore and Visualize Your Data.

Table 11: Field Service Wave 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

Set Up and Create the Field Service Wave Analytics App

Set up Field Service Wave Analytics app permissions, make sure your org is enabled to use Field Service Lightning, and assign field level security correctly so you can create and share the app.

Set Up Field Service Wave Analytics App Permissions

The Service Wave Analytics license provides access to the Field Service Wave Analytics app. To set up permissions for Field Service Wave, follow the detailed instructions starting at Set Up Permissions for the Service Wave Analytics App.

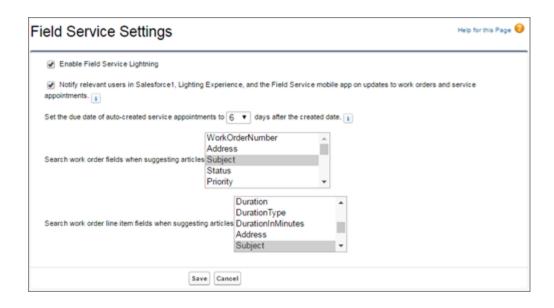
USER PERMISSIONS

To create and manage Wave apps:

- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"

Make Sure That Field Service Lightning Is Enabled in Your Org

- (1) Important: Important: Your org must be enabled to use Field Service Lightning and its new standard object model to use Field Service Wave.
- 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 Lightning** is checked.



If you're unsure that your org is set up to use Field Service Lightning, contact your Salesforce representative.

Assign Field Level Security

Before you create Field Service Wave, set Salesforce field-level security to enable the Wave 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 Wave Analytics App, Step 2.

Create Field Service Wave

Once you've completed the previous setup processes, create Field Service Wave.

- 1. Go to Wave Analytics.
- 2. Click Create in the upper right corner.
- **3.** Select the Field Service Wave Analytics template.
- **4.** Name your app and click Create.

Wave creates your app. This may take a few minutes while Wave runs a new dataflow and creates the app's dashboards. Refresh your browser, and click the app you just created to see the dashboards.

Share Field Service Wave

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 Wave Analytics Templated Apps and "Access Service Cloud Templates and Apps" permissions enabled. To share the app, roll your cursor over the icon for the app and click **Share**. Next, type in the names of team members, click **Add**, and click **Save**.

Event Monitoring Wave App

The Event Monitoring Wave App integrates with event monitoring and setup audit trail data to give you insights into your user and org behavior.



Note: As part of Event Monitoring, you also get the Event Monitoring Wave 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 Wave App is available in English only. If you have an Analytics Cloud - Wave Analytics Platform license, you can access dashboards in the Event Monitoring Wave app.

The Event Monitoring Wave App is the first 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.

IN THIS SECTION:

1. Set Up Salesforce Permissions for the Event Monitoring Wave App

Set up your org to use the Event Monitoring Wave App by enabling Wave, assigning permission set licenses, and then creating and assigning permission sets.

2. Create and Share the Event Monitoring Wave App

After you've enabled the correct permissions, the setup for the Event Monitoring Wave App is simple.

3. Upgrade the Event Monitoring Wave App

Take advantage of the latest Event Monitoring Wave features by upgrading your app every time we release a new version.

4. Schedule the Event Monitoring Wave Daily Dataflow

Schedule a daily dataflow before you start exploring with Event Monitoring Wave.

5. Learn About Event Monitoring Wave App Prebuilt Dashboards

The Event Monitoring Wave App comes with a set of prebuilt dashboards so you can start exploring your data right away. Data from your org is automatically loaded into your Event Monitoring Wave App datasets when it's generated.

6. Get to Know Event Monitoring Wave App Terminology

To make the best use of the Event Monitoring Wave App, it's helpful to understand the metrics and terms used in the app.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To use the Event Monitoring Wave App:

"Use Wave Analytics Templated Apps"

To create and manage Event Monitoring Wave Apps:

- "Access Event Monitoring Wave Templates and Apps"
- "Manage Wave Analytics Templated Apps"
- "Edit Wave Analytics Dataflows"

Set Up Salesforce Permissions for the Event Monitoring Wave App

Set up your org to use the Event Monitoring Wave App by enabling Wave, assigning permission set licenses, and then creating and assigning permission sets.



Note: As part of Event Monitoring, you also get the Event Monitoring Wave 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 Wave App is available in English only. If you have an Analytics Cloud - Wave Analytics Platform license, you can access dashboards in the Event Monitoring Wave app.

Each Event Monitoring Wave license provides access to Event Monitoring Wave 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.

USER PERMISSIONS

To create and manage Event Monitoring Wave Apps:

- "Access Event Monitoring Wave Templates and Apps"
- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"

Table 12: Event Monitoring Wave Data Storage Limits

License	Limit
Analytics Cloud - Event Monitoring Wave App	50 million rows when used without Analytics Cloud - Wave Analytics Platform license. Provides access to up to 10 users.
Analytics Cloud - Additional Data Rows	100 million rows.



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 Wave App with or without the Salesforce Wave Analytics platform. The Event Monitoring Wave app is available for free to any org with the paid Event Monitoring license. The Event Monitoring Wave App license enables the following permissions.

Table 13: Event Monitoring Wave User Permissions

Permission	Function
"Use Wave Analytics Templated Apps"	Enables access to all Wave templated apps, including the Event Monitoring Wave App.
	Warning: This permission allows users to access event monitoring data without requiring the "View Event Log Files" and "API Enabled" user permissions. Use caution when assigning it.
"Access Event Monitoring Wave Templates and Apps"	Gives admins the ability to create Event Monitoring Wave apps.
"Manage Wave Analytics Templated Apps"	Gives admins the ability to create templated Wave apps for users in their orgs.
"Edit Wave 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 Wave app in the order shown.

IN THIS SECTION:

1. Event Monitoring Wave App Permission Set License and User Permissions

The Event Monitoring Wave App permission set license enables the user permissions required to explore data with and manage the Event Monitoring Wave app.

2. Identify Event Monitoring Wave App User Types

Identify Event Monitoring Wave app user types and the tasks they perform to assure that the setup process meets your team's event monitoring analytics needs.

3. Enable Wave Analytics and Event Monitoring and Assign Permission Set Licenses

After Salesforce provisions you with the Event Monitoring Wave license, enable Wave and Event Monitoring and assign Event Monitoring Wave App permission set licenses.

4. Create Event Monitoring Wave App Permission Sets and Assign Users to Them

Give users in your organization access to Event Monitoring Wave features by creating and assigning one or more permission sets based on the capabilities they need.

Event Monitoring Wave App Permission Set License and User Permissions

The Event Monitoring Wave App permission set license enables the user permissions required to explore data with and manage the Event Monitoring Wave app.

Each Event Monitoring app user needs an Event Monitoring Wave App permission set license. The Event Monitoring Wave App permission set license enables the following permissions.

Table 14: Event Monitoring Wave App Permissions

User Permission	What It Enables
"Use Wave Analytics Templated Apps"	General access to all Wave templated apps, including the Event Monitoring Wave app.
"Access Event Monitoring Wave Templates and Apps"	The ability to create Event Monitoring Wave apps.
"Manage Wave Analytics Templated Apps"	The ability to create templated Wave apps for users in their orgs.
"Edit Wave Analytics Dataflows"	Editing, starting, stopping, and rescheduling the dataflow. Creating, editing, and running dataset recipes. Monitoring dataflow and system jobs in the data monitor. 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 Wave Analytics.

Identify Event Monitoring Wave App User Types

Identify Event Monitoring Wave app user types and the tasks they perform to assure that the setup process meets your team's event monitoring analytics needs.

The Event Monitoring Wave license includes the permissions required by users in your organization to administer Event Monitoring Wave and explore event data. Set up the Event Monitoring Wave in the most useful way by thinking through the different ways people can access Event Monitoring Wave features and categorizing users into types. Here are a few examples.

In most cases, Event Monitoring Wave addresses the needs of two basic types of people on your team:

- Users, who view Event Monitoring Wave dashboards and datasets.
- Administrators or managers, who access all Event Monitoring Wave functionality to create apps, dashboards, and datasets, and otherwise customize the Event Monitoring app experience.

Accordingly, when you set up Event Monitoring Wave, 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 Wave.
- A "manage" permission set that's reserved for a select few administrators and managers and contains permissions to use all Event Monitoring Wave features.

In the detailed Event Monitoring Wave app setup instructions that follow, we show you how to create and assign these two permission sets.

Enable Wave Analytics and Event Monitoring and Assign Permission Set Licenses

After Salesforce provisions you with the Event Monitoring Wave license, enable Wave and Event Monitoring and assign Event Monitoring Wave App permission set licenses.

Enable the Wave Platform

- 1. In the Salesforce Setup menu, under administer, click **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 Setup**.
- 2. Enable Enable Event Log File Integration with Event Monitoring Wave App.

Assign Permission Set Licenses

Next, assign a permission set license to each user in your org who needs to access the Event Monitoring Wave App.

- 1. In the Setup menu, click Manage Users | Users.
- 2. Click the name of a user you want to access Wave. You might want to start with User, Admin, or whatever name has been given to the system administrator.
- 3. Position the cursor over Permission Set License Assignments and click Edit Assignments.



- **4.** Select the Event Monitoring Wave App Permission Set Licenses and click **Save**.
- **(1)** Important: Repeat these steps for every user in your org who requires access to the Event Monitoring app.

Create Event Monitoring Wave App Permission Sets and Assign Users to Them

Give users in your organization access to Event Monitoring Wave features by creating and assigning one or more permission sets based on the capabilities they need.

After assigning Event Monitoring Wave App permission set licenses to users, create permission sets made up of Event Monitoring Wave user permissions and assign them to users. You can create any permission sets you need to meet the needs of your users. We show you how to create two:

- *Manage Event Monitoring Wave* permission set, for users who create and administer Event Monitoring Wave.
- View Event Monitoring Wave permission set, for users who explore app dashboards and datasets.

Let's begin with the Manage Event Monitoring Wave permission set.

- 1. From Setup enter *Permission Sets* in the Quick Find box, then select **Permission Sets**.
- 2. Click New.
- **3.** Enter Manage Event Monitoring Wave in the Label field. This automatically creates the API name, which you can change if you'd like.
- 4. 🕒 Warning: Follow this step precisely or some users in your organization may not be able to access Event Monitoring Wave.

Select the permission set license to associate with this permission set by using the License drop-down option. For Event Monitoring Wave, keep the default value --None--.

- 5. Select the feature permissions to enable for your permission set. Use Find Settings... to search for them quickly. Select "Use Wave Analytics Templated Apps," "Access Event Monitoring Wave Templates and Apps," "Manage Wave Analytics Templated Apps," and "Edit Wave Analytics Dataflows." Make sure to select all four.
- **6.** Click **Save**. You've successfully created a permission set that enables users to create and administer Event Monitoring Wave and access all app features. You can now assign it to users. See Assign Permission Sets to a Single User.
- Warning: Assign the Manage Event Monitoring Wave permission set sparingly because it lets users administer Event Monitoring Wave, which lets them create, edit, and delete the app. Assign it only to users who administer or manage the app.

To create *View Event Monitoring Wave* repeat the steps but use View Event Monitoring Wave in the Label field. Select only the "Use Wave Analytics Templated Apps" and "Access Event Monitoring Wave Templates and Apps" user permissions.

USER PERMISSIONS

To create permission sets:

"Manage Profiles and Permission Sets"

Create and Share the Event Monitoring Wave App

After you've enabled the correct permissions, the setup for the Event Monitoring Wave App is simple.

If you haven't enabled permissions and event log file integration, complete the steps in Enable Wave Analytics and Event Monitoring and Assign Permission Set Licenses on page 442.



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. 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 Wave Analytics from the App Picker.
- 3. Click App, select Event Monitoring Wave 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 Monitor** from the gear menu at the top of the page. If any part of your app creation fails, you'll need to recreate the app.

USER PERMISSIONS

To use the Event Monitoring Wave App:

"Use Wave Analytics Templated Apps"

To create and manage Event Monitoring Wave Apps:

- "Access Event Monitoring Wave Templates and Apps"
- "Manage Wave Analytics Templated Apps"
- "Edit Wave Analytics Dataflows"

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 Wave Analytics Templated Apps" and "Access Event Monitoring Wave Templates and Apps" permissions can see the app. To share your app, hover your mouse over the app tile on the Wave Analytics home screen and click Share. You can also share from the app itself.

Upgrade the Event Monitoring Wave App

Take advantage of the latest Event Monitoring Wave 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.
- 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.

USER PERMISSIONS

To create and manage Event Monitoring Wave Apps:

- "Access Event Monitoring Wave Templates and Apps"
- Manage Wave Analytics **Templated Apps**
- "Edit Wave Analytics Dataflows"

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, Wave 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 Wave Daily Dataflow

Schedule a daily dataflow before you start exploring with Event Monitoring Wave.

When you create Event Monitoring Wave, the creation process includes a dataflow that imports the latest Event Monitoring data to Wave. 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 6:00 AM. Schedule the dataflow for a couple of hours after so that the app has the latest data and that the dataflow doesn't interrupt your use of the app. We recommend running the dataflow between 8:00 and 9:00 AM.



1. Go to **Wave Analytics** in the Force.com menu (top right of the Salesforce window) if you're not already there. Open the Data Monitor by clicking the gear icon at the upper right of the screen.

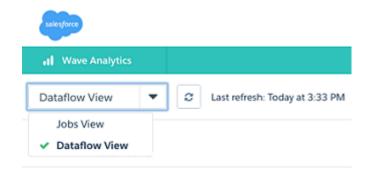


2. Select Dataflow view from the menu at the top left of the Data Monitor screen.

USER PERMISSIONS

To create and manage Event Monitoring Wave Apps:

- "Access Event Monitoring Wave Templates and Apps"
- Manage Wave Analytics Templated Apps
- "Edit Wave Analytics Dataflows"



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**.

Learn About Event Monitoring Wave App Prebuilt Dashboards

The Event Monitoring Wave App comes with a set of prebuilt dashboards so you can start exploring your data right away. Data from your org is automatically loaded into your Event Monitoring Wave 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. Two of the dashboards, My Trust and Setup Audit Trail, don't correspond to event types. Let's get a quick idea about what each dashboard does.

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. The API 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.

Dashboard

Corresponds to the Dashboard event type. The Dashboard dashboard helps you track dashboard adoption and performance.

Visualforce Requests

Corresponds to the Visualforce Request event type. Here you can see trends in Visualforce adoption and page 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

Corresponds to the UI Tracking event type. This dashboard shows you which pages users are accessing in Lightning Experience or Salesforce 1.

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.

Page Views (URIs)

Corresponds to the URI event type. The Page Views dashboard lets you see which pages users are accessing in the Salesforce Classic 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.

Report

Corresponds to the Report event type. The Reports 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.

Rest API

Corresponds to the 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. See what kinds of changes your users are making in the Setup area.

User Logins

Corresponds to the Login event type. The User Logins dashboard shows login trends by user and information about where and how users are accessing your org.

Get to Know Event Monitoring Wave App Terminology

To make the best use of the Event Monitoring Wave 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.

The B2B Marketing Wave Analytics App

B2B Marketing Wave 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 Wave Analytics app currently resides within the Salesforce Pardot Knowledge Base. Go to the B2B Marketing Wave Analytics overview page to get started.

Develop on the Wave Platform

Find resources for developing with Wave.

IN THIS SECTION:

The Wave REST API

You can access Wave Analytics datasets and lenses programmatically using the Wave REST API.

Salesforce Analytics Query Language (SAQL)

Use SAQL to access data in Wave Analytics datasets. Wave Analytics uses SAQL behind the scenes in lenses and dashboards to gather data for visualizations.

Wave Migration, Packaging, and Distribution

You can migrate Wave 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 Wave REST API

You can access Wave Analytics datasets and lenses programmatically using the Wave REST API.

Using the Wave REST API, you can:

- Send gueries directly to the Wave Platform.
- Access datasets that have been imported into the Wave Platform.
- Create and retrieve Wave Analytics lenses.
- Access XMD information.
- Retrieve a list of dataset versions.
- Create and retrieve Wave Analytics applications.
- Create, update, and retrieve Wave Analytics dashboards.
- Retrieve a list of dependencies for an application.
- Determine what features are available to the user.
- Work with and schedule Trend in Wave report snapshots.
- Manipulate replicated datasets.
- Get, add, update, and delete 'eclair' geo map charts.

For complete information about the API, see the Wave REST API Developer Guide.

SEE ALSO:

Enable the Wave REST API

Salesforce Analytics Query Language (SAQL)

Use SAQL to access data in Wave Analytics datasets. Wave Analytics uses SAQL behind the scenes in lenses and dashboards to gather data for visualizations.

Developers can write SAQL to directly access Wave Analytics data via:

Wave REST API

Build your own app to access and analyze Wave Analytics data or integrate data with existing apps.

Dashboard JSON

Create advanced dashboards. A dashboard is a curated set of charts, metrics, and tables.

Analytics Wave Analytics Glossary

For complete information about SAQL, see the Wave Analytics SAQL Reference.

SEE ALSO:

View the Query Behind Your Lens

Wave Migration, Packaging, and Distribution

You can migrate Wave 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.

IN THIS SECTION:

Migrate Wave Assets with Change Sets

Use change sets to move customized Wave assets between orgs that have a deployment connection. For example, create a Wave app containing dashboards, lenses, datasets, dataflows in your Sandbox org, then migrate the app assets to your production org once testing is complete.

Package Wave Assets in Managed Packages

You can create managed packages of Wave assets, including Wave apps, dashboards, lenses, datasets, and dataflows. Use packages to distribute those assets to other users or organizations, including those outside your company.

Distribute Wave 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 Wave Assets and User XMD

Wave provides full support for the Metadata API, which can be used to retrieve, deploy, create, update, or delete customizations for your organization.

Wave Analytics Glossary

Familiarize yourself with common Wave Analytics 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.

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.

Analytics Wave Analytics Glossary

Dataflow job

A dataflow job processes the logic in a dataflow.

Dataset

A dataset contains a set of source data, specially formatted and 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.

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. There are two versions: Wave dashboard designer (the new one) and classic designer (the legacy one).

Dimension

A dimension is a qualitative value, like region, product name, and model number. Dimensions are handy for grouping and filtering your data. Unlike measures, you can't perform math on dimensions.

Explorer

Explorer is the interface where you explore datasets and lenses.

Extended metadata (XMD)

Extended metadata (XMD) allows you to customize the formatting of many dashboard elements in Wave Analytics.

External data

External data is data that resides outside of Salesforce, such as data from outside applications and spreadsheets.

External Data API

Use External Data API to upload external data files to Wave Analytics to create datasets.

Lens

A lens is a particular view into a dataset's data. It's where you do exploratory analysis and visualization.

Measure

A *measure* is a quantitative value, 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

A predicate is a filter condition that defines row-level access to records in a dataset.

Salesforce Analytics Query Language (SAQL)

Use SAQL to access and analyze data in Wave Analytics datasets.

Step

A *step* determines the results shown in a widget. Steps either get data dynamically or statically. For example, a step can query a dataset to get the average opportunity amount for each account, or it can retrieve an unchanging, user-defined value.

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.

Analytics Wave Analytics Limits

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 Wave Analytics retrieves information from the source data.

Wave Analytics Limits

This section describes Wave Analytics limits.

API Call Limits

These limits apply to all supported editions.

Limit	Value
Maximum concurrent Wave Analytics API calls per org	100
Maximum Wave Analytics API calls per user per hour	10,000

Dataset Storage Limits

These limits apply to all supported editions and are contractual, not technical. The licensee agrees to strictly monitor its total number of data rows. Use of the Sales Wave Analytics app license does not increase the data limit for the platform license.

Limit	Value
Maximum number of rows for all registered datasets combined	250 million per platform license (purchased before October 20, 2015)
	100 million per Analytics Cloud - Wave Analytics Platform license
	25 million per Analytics Cloud - Sales Wave App license
	+ 100 million per Analytics Cloud - Additional Data Rows license
Maximum number of fields in a dataset	5,000 (including up to 1,000 date fields)
Maximum value for each numeric field in a dataset	36,028,797,018,963,967
Minimum value for each numeric field in a dataset	-36,028,797,018,963,968

Dataflow Limits

Wave uses the bulk API, which counts 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 100K to 250K, 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.

Limit	Value
Maximum number of dataflow runs in a rolling 24-hour period	24

ELT Job Limits

ELT jobs include user dataflows, CSV uploads, dataset recipes, and replication.

Analytics Wave Analytics Limits

Limit	Value
Timeout for ELT jobs that have been scheduled but not executed	5 minutes

Replication Limits

If you are currently extracting more than 50 objects in your dataflows, contact Salesforce Customer Support before you enable replication.

Limit	Value
Maximum number of objects that can be enabled for replication	50

Trending Data Limits

Trended datasets count toward the overall Wave 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 for all monthly snapshots per org	40 million

Lens and Dashboard Creation Limits

These limits apply to the creation of lenses and dashboards.

Limit	Value
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 number of notifications per user	5

Sales Wave Analytics App Limits

The remaining limits in this section apply to Sales Wave Analytics. Sales Wave Analytics limits custom object support to no more than 10 custom objects and one dataflow for each org that implements the app. These are contractual, not technical, limits.

External Data Limits

Limit	Value
Maximum file size per external data upload	If you use the External Data API: 40 GB. If you use the user interface: 512 MB.
Maximum file size for all external data uploads in a rolling 24-hour period	50 GB

Limit	Value
Maximum number of external data files that can be uploaded 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

Query Limits

Trended datasets count toward the overall Wave platform limits, including total number of rows.

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
10,000 is the default value. To set a different value, use the SAQL limit statement	
Query timeout	2 minutes

Wave Developer Addition Limits

Wave is built on the force.com platform and is subject to the force.com platform limits. For example, the SOQL query limit (20,000 characters) controls the number of fields from which the sfdcDigest transformation can extract data. If you exceed a force.com limit, an error occurs. See force.com platform limits.

Limit	Value
Maximum number of data rows	250,000
Maximum number of concurrent queries	5

Wave Analytics Limitations

Wave Analytics differs from other Salesforce features in some ways.

General Limitations

This section describes Wave Analytics limitations.

Update Schedule

All customers receive Wave Analytics updates on the same schedule, regardless of instance.

Analytics Wave Analytics Limitations

Localization and Internationalization

Wave Analytics 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.

Wave Analytics 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 case-sensitive and accent-sensitive. For example, using the filter "Andre" does not return "Andre" in the results.
- Multicurrency is not supported. When Wave Analytics extracts your organization's default currency, it uses the currency for monetary values and doesn't convert to another currency.
- Sorting data within datasets might not work as expected.

Sales Wave Analytics Limitations

These limitations apply to Sales Wave Analytics.

Sales Wave AnalyticsData Requirements

Sales Cloud data must meet the following requirements for Sales Wave 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 Wave Analytics using the configuration wizard.
- There are two requirements if you choose to import Leads data to Sales Wave 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 Wave 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 Wave 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 Wave Analytics Support for Salesforce Objects and Fields

Sales Wave Analytics supports all Salesforce standard and custom objects and data. However, when you first create the app by default it includes only a predefined set of objects and fields.

Sales Wave creates a dataflow that exposes selected fields from the following standard Salesforce objects:

Accounts

Analytics Wave Analytics Limitations

- Users
- Roles
- Opportunities
- Projects (Opportunities line item)
- Tasks
- Events

Sales Wave 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.

Sales Wave limits custom objects support to no more than 10 custom objects for each org that implements Sales Wave. 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 Sales Wave dataflow. For details, see Create Datasets with a Dataflow

Sales Wave Analytics Support for Non-Salesforce Data

Sales Wave 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 Wave Analytics, you must upload a CSV file with the details. For more information, see Update the Sales Wave App Quota (Target) Dataset.

Service Wave Limitations

The following limitations apply to Service Wave.

Service Wave Data Requirements

Service Cloud data must include at least one each of the following for Service Wave dashboards to function correctly: Events, tasks, closed cases, or published articles attached to a case or an opportunity.

Custom formula fields on the Cases object must exist or be created for the SLA missed (text) and FCR (Boolean) fields.

Service Wave Support for Salesforce Objects and Fields

Service Wave supports all Salesforce standard and custom objects and data. However, 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
- Events
- User
- UserRole
- Task

Service Wave also uses fields from the following options depending on the answers you select in the configuration wizard:

- CaseArticle
- CaseHistory
- CSAT (customer object)
- Group
- Single Knowledge article (custom object)

Analytics Reports and Dashboards

- RecordType
- Opportunity

Service Wave 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 Wave limits custom objects support to no more than 10 custom objects for each org that implements Service Wave. 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 Wave dataflow. For details, see Create Datasets with a Dataflow

More Service Wave Limitations

Service Wave 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.

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 Wave Analytics dataset. For more information, see the Wave Analytics Security Implementation Guide.

Actions Menu Support for Objects and Actions

The current actions menu implementation provides limited support for Salesforce objects and actions.

You can currently create Wave quick actions menus for custom objects, sObjects, and the following standard Salesforce objects:

- Account
- Cases
- Lead
- Opportunity

Actions on Contacts, User, or User Profile page layouts are not supported.

The current implementation of action menus supports Chatter actions (Post, File, Thanks, Link, Poll, and Question), object-specific actions, and global actions. It does not support Salesforce1 actions.

Other objects and actions will be supported in future releases.

Dashboards on Mobile Devices

To view a Wave dashboard on a mobile device, use the Wave Analytics mobile app.

Reports and Dashboards

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

IN THIS SECTION:

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.

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.

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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.

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. Possible formats are tabular, summary, matrix, or joined. Joined reports aren't available in Lightning Experience.

IN THIS SECTION:

1. Build a Report

When you have guestions about your Salesforce data, build a report to get the answers.

2. 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.

7. 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.

8. 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.

9. Export a Report

To work with report data in a dedicated tool, such as a spreadsheet, export report data as a Microsoft Excel $^{\circ}$ (.xls) or comma-separated values (.csv) file.

10. Share a Report

Share your reports with colleagues so everyone's working with the same info.

11. 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.

12. Report Limits

As you report on your data, be aware of these limits and restrictions.

13. Troubleshoot Reports

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

14. Get the Right Data into Your Report

A well-designed report helps you turn raw data into information people can act on. Summarize values and group them into meaningful categories, filter out irrelevant facts, and gather data about multiple objects at once.

15. Communicate Your Report Data Effectively

A report produces information in the form of a table. You can set up that table to help people scan quickly for the information they need. You can also add a graph to help users understand the data visually, and you can display the report's data on a dashboard.

16. The Report Run Page

The Report Run page lets you control what happens to your report when you run it.

17. Report Run Page Options

Customize the report run page to support the specific information you need for the way you work.

18. Get Started with Reports

Set up your reporting environment, use the report builder to create a basic report, and organize your reports to make it easy to find information when you need it. In most cases, you can start with a standard report and customize it to your needs.

19. Reports Give You Access to Your Data

Use reports to leverage the data your organization has accumulated over time. You can examine your organization's data in almost infinite combinations, display it in easy-to-understand formats, and share the resulting insights with others via dashboards.

Build a Report

When you have questions about your Salesforce data, build a report to get the answers.

IN THIS SECTION:

1. Build a New Report

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.

2. 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.

3. 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.

4. Categorize Data Quickly with Buckets

Bucketing lets you quickly categorize report records without creating a formula or a custom field. When you create a bucket field, you define multiple categories (buckets) used to group report values.

5. 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.

6. 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.

7. 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.

8. 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.

9. Upgrade the Report Wizard

Report builder, a powerful drag-and-drop editor, is the standard tool for creating and editing reports. If your organization is still using the old report wizard, you should upgrade to report builder.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

Build a New Report

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)

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, simply close the builder or click the Salesforce logo.

IN THIS SECTION:

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. 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.

4. 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.

5. Create a Custom Report in Accessibility Mode

6. 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.

7. 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.

8. 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.

9. 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.

10. Summarize Your Report Data

A summary field contains numeric values for which you want to know the sum, the average, or the highest or lowest. Summary fields show at all grouping levels. In summary and matrix reports, they also appear at the grand total level.

11. 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

"Create and Customize Reports"

AND

"Report Builder"

12. 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.

13. Work with Formulas in Reports

Formulas let you create custom summaries based on calculated values using report data. These formulas can then be used as columns in your report.

14. 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.

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

IN THIS SECTION:

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.

SEE ALSO:

Why doesn't my report return the data I expect?

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 new custom report to access exactly the right information.

IN THIS SECTION:

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. Forecast Reports

Forecast reports give you information about your customizable forecast data.

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:

Find a Report

The Report Run Page

Creating a Custom Report

Reports Give You Access to Your Data

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** Editions except **Database.com** (The edition determines which reports you see.)

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®.

Last Activity

The last activity date is either the due date of the most recently logged event against the record or the most recently closed task associated with the record. The Last Activity date can show up in activities related lists and reports, as well as on record details pages (such as accounts, contacts, and leads).

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:

The Report Run Page 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 reports on 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** Editions except **Database.com** (The edition determines which reports you see.)

- 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 that 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 that 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.

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 will not be visible depending on your edition.
 - My Activities—Shows activities that you own.
 - My Delegated Activities—Shows activities that you created but are owned by someone else in the same role as you, and below your role, 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:

The Report Run Page Limit Report Results

Administrative Reports

USER PERMISSIONS

To view the Administrative Reports folder:	"View Setup and Configuration"
To run user reports:	"Run Reports"
	AND
	"View Setup and Configuration"
To create, edit, save, and delete user reports:	"Create and Customize Reports"
	AND
	"View Setup and Configuration"
To run document reports:	"Run Reports"
To run document reports: To create, edit, save, and delete document reports:	"Run Reports"
To create, edit, save, and delete document	"Run Reports"
To create, edit, save, and delete document reports:	"Run Reports" "Create and Customize Reports"

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: Contact Manager, Group, Professional, Enterprise, Performance, Unlimited, and Developer Editions

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:

The Report Run Page 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.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

• 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:

The Report Run Page 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.	

EDITIONS

Available in: Salesforce Classic

Available in: Contact Manager, Group, Professional, Enterprise, Performance, Unlimited, and Developer Editions

Field	Description
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.
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.

Field	Description	
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 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.
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.

Field	Description
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:

The Report Run Page Limit Report Results

Forecast Reports

Forecast reports give you information about your customizable forecast data.



Note: This information applies to Customizable Forecasting and not Collaborative Forecasts.



Note: If you don't see the Forecast Reports folder, your organization may not have customizable forecasting enabled. Customizable Forecasting includes five standard report types. You can't create custom report types with Customizable Forecasting.

Special Features of Customizable Forecast Reports

Consider the following when running customizable forecast reports:

Standard Reports

Create a forecast history report if you want the report to include information about forecasts you already submitted. Select All Forecasts to show the state of the forecasts after each update. Choose Latest Forecasts to display only the current state of the forecasts.

Report Types

- When creating a custom forecast report, choose the **Customizable Forecast: Opportunity Forecasts** report type if you want to include information about the opportunities that contribute to your forecast.
- You can include custom user fields in custom opportunity forecast reports as columns and as column values. Custom user fields
 are available as either opportunity owner or account owner custom information. The custom field label will be used for both
 opportunity owner or account owner information, which can cause confusion if they are both on the same report and have
 similar field labels.
- If you have the "View All Forecasts" permission, forecast summary and forecast history reports include all users' data for the chosen forecast hierarchy, including users in branches of the forecast hierarchy that have no forecast manager.

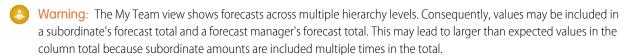


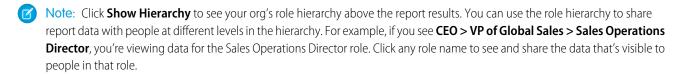
Available in: Salesforce Classic

Tips on Customizable Forecasting Reports

- You must specify a reporting interval by fiscal year or by date.
- Forecast reports include private opportunities.
- Opportunity amounts are included in forecast reports if the opportunity is set to close within the forecast period.
- Forecast reports include only opportunities that are set to close within the forecast period, except those assigned to the Omitted forecast category.
- Create a forecast summary report if you want to view information about the amounts in your forecast without details of the opportunities that contribute to that amount.
- If your organization uses territories, you can create custom forecast reports that summarize information by territory name.
- You can view an opportunity forecast report based on the forecast hierarchy by drilling down in your forecast hierarchy.
 - 1. Select the desired role or territory from the **Hierarchy** links.
 - 2. Choose the appropriate forecast users or teams from the Show Opportunities Owned By or Show Forecasts Owned By scopes at the top of the report.
 - 3. Click **Run Report** again to refresh the report data for your chosen parameters.

For example, to see your opportunity information, including your overrides on opportunity amounts or quantities and overrides from users below you in the forecast hierarchy, set the **Hierarchy** link to your forecast role. To see the opportunity information of someone beneath you in the forecasting hierarchy, drill down into that role.





SEE ALSO:

The Report Run Page **Limit Report Results**

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

EDITIONS

Available in: Salesforce Classic and Lightning Experience

- 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:

The Report Run Page 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.

IN THIS SECTION:

1. Tips for Working with Opportunity Reports

Opportunity reports can include all opportunity fields plus some extra columns for additional 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:

The Report Run Page Limit Report Results

Tips for Working with Opportunity Reports

EDITIONS

Available in: Salesforce Classic

Opportunity reports can include all opportunity fields plus some extra columns for additional detail.

- You can use the following columns to provide additional detail:
 - Age—For an open opportunity, the number of days since the opportunity was created.
 For a closed opportunity, the number of days between the creation date and the closed date.
 - 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

- If your organization uses Forecasts or Customizable Forecasts, you'll see forecast report options paired with opportunity report options in the standard reports folders and the custom reports wizard. Opportunities owned by users who do not have a role in your organization's hierarchy are omitted from opportunity, forecast, and sales reports. This restriction does not apply to Personal or Group Edition organizations, which do not have a role hierarchy. For reports that include the Partner column (except for the Partner Opportunities report), only the primary partner of the opportunity is displayed.
- 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 <code>Role</code>, you see your opportunities and opportunities owned by all of your subordinates in the role hierarchy. If you select <code>Territory</code>, 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, shows opportunities they
 own or for which they are on the opportunity team.
 - My territories—For organizations that use territory management, this option shows opportunities that belong to the territories
 to which you are assigned.

All opportunities—Shows all opportunities 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.

EDITIONS

Available in: Salesforce Classic

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

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"

Report Name	Description
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.
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 <i>Primary equals True</i> 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 Reports
Opportunity Report Types
Tips for Working with Opportunity Reports

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.

IN THIS SECTION:

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.

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.

EDITIONS

Available in: Salesforce Classic

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

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"

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
	• Stage
	• Age
	 Type
	Probability (%)
	• Lead Source
	• Fiscal Period
	• Next Step
	• Created Date
Opportunity Owner Information	• Opportunity Owner
	• Owner Role

EDITIONS

Available in: Salesforce Classic

Object Information Type	Columns
Account Information	• Account Name

SEE ALSO:

The Report Run Page

Limit Report Results

Opportunity Reports

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	TitleFirst NameLast Name
Contact Role: Phone/Fax/Email	• Phone • Email
Contact Role: Address	 Mailing Street Mailing City Mailing State/Province Mailing ZIP/Postal Code Mailing Country

EDITIONS

Available in: Salesforce Classic

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

SEE ALSO:

The Report Run Page 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.

The default settings for this report are:

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	TitleFirst NameLast Name
Contact Role: Phone/Fax/Email	• Phone • Email
Contact Role: Address	Mailing StreetMailing CityMailing State/ProvinceMailing ZIP/Postal CodeMailing Country

EDITIONS

Available in: Salesforce Classic

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

SEE ALSO:

The Report Run Page Limit Report Results **Opportunity Reports**

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

EDITIONS

Available in: Salesforce Classic

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

SEE ALSO:

The Report Run Page

Limit Report Results

Opportunity Reports

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.

EDITIONS

Available in: both Salesforce Classic Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

AND

Manage Custom Report Types"

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

EDITIONS

Available in: Salesforce Classic

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

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:

The Report Run Page

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 C	Columns
Opportunity Information	Opportunity Name

EDITIONS

Available in: Salesforce Classic

Available in all editions

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:

The Report Run Page

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 NameTeam Role

If your admin has created custom opportunity team fields, you can include them in this report.

SEE ALSO:

The Report Run Page

Limit Report Results

Opportunity Reports

Opportunities with Opportunity Teams and Products Report

EDITIONS

Available in: Salesforce Classic

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 NameTeam Role

EDITIONS

Available in: Salesforce Classic

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

SEE ALSO:

The Report Run Page

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 NameClose DateAmount
Opportunity Owner Information	• Opportunity Owner
Partner Information	Partner OwnerPartner RolePartner
Account Information	Account Owner

EDITIONS

Available in: Salesforce Classic

Object Information Type	Columns
	Account Name

SEE ALSO:

The Report Run Page

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

EDITIONS

Available in: Salesforce Classic

Object Information Type	Columns
	• Owner Role
Account: General Information	• Account Name

SEE ALSO:

The Report Run Page

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 ${\tt Historical}$ Stage and columns grouped by ${\tt As}$ of ${\tt Date}.$

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:

The Report Run Page

Limit Report Results

Opportunity Reports

Opportunities with Quotes and Quote PDFs Report

EDITIONS

Available in: both Salesforce Classic Lightning Experience

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 NameSyncing
Quote PDF Information	Quote PDF: Created DateQuote PDF: NameQuote PDF: DiscountQuote PDF: Grand Total

EDITIONS

Quotes are available in: Salesforce Classic

Quotes are available in:
Performance and
Developer Editions and in
Professional, Enterprise,
and Unlimited Editions with
the Sales Cloud

SEE ALSO:

The Report Run Page 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)

EDITIONS

Quotes are available in: Salesforce Classic

Quotes are available in:
Performance and
Developer Editions and in
Professional, Enterprise,
and Unlimited Editions with
the Sales Cloud

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 NameAmount
Quote Information	Quote NameDiscountSyncingStatus
Quote Line Item Information	 Quote Line Item: Discount Product: Product Name Line Item Number Sales Price List Price Quote Line Item: Subtotal Quote Line Item: Total Price

SEE ALSO:

The Report Run Page

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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:

The Report Run Page **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 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:

The Report Run Page **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.

IN THIS SECTION:

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.

EDITIONS

Available in: Salesforce Classic

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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: All Editions except **Database.com** (The edition determines which reports you see.)

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:

The Report Run Page 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

	Cases
Field	Description
# 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	The time the case entered an entitlement process.
Start Time	This field displays if an entitlement process applies to the case.
Entitlement Process	The time the case exited an entitlement process.
End Time	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 submitted via the Self-Service portal by anyone in his or her company.
	(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 a comment has been added to a case via the Self-Service portal and has not been reviewed by the case owner.
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)

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** Editions except **Database.com** (The edition determines which reports you see.)

Cases

Field	Description
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 <i>not</i> 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 <i>not</i> taken into account for this field. Holidays suspend business hours during specified dates and times. (Case History Report)
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 <i>not</i> 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	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)
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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** Editions except **Database.com** (The edition determines which reports you see.)

Field	Description
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.

IN THIS SECTION:

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. Reports for Answers

Create custom report types so users can analyze questions, replies, and votes.

3. 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.

4. 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.

5. Reports That You Can Run on Activities

Run standard or custom reports that display details about your tasks or events and their related contacts, accounts, opportunities, and other associated records.

6. Report on Sales Reps' Meetings with Customers

Get insight into how your sales teams are prioritizing their time. Create a custom report that shows which customers sales reps are spending their time with.

7. 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.

8. 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.

9. Idea Reports

Create custom report types so users can create reports about ideas, idea comments, and votes.

10. Report on Salesforce Knowledge Articles

Use Salesforce Knowledge custom reports to track how articles are created, maintained, and delivered.

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 create, edit, and delete reports:

 "Create and Customize Reports"

11. Live Agent Session Reports

Use Live Agent 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.

12. Report on Partners

Some opportunities involve partner relationships. Use this report to identify and analyze those relationships.

13. 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.

14. Enable the Account Owner Report

The Account Owner report lists all accounts and who owns them.

15. 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.

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.

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.

EDITIONS

Available in: Salesforce Classic

Territory management is available in: **Performance** and **Developer** Editions and in **Enterprise** and **Unlimited** Editions with the Sales Cloud.

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:

The Report Run Page Limit Report Results

Reports for Answers

Create custom report types so users can analyze questions, replies, and votes.



Note: Answers is scheduled for retirement in all Salesforce orgs as of Winter '18 release. For more information, see Chatter Answers to Retire in Winter '18. Starting with Summer '13, Answers isn't available in new orgs. Instead, you can use Chatter Questions, a Q&A feature that's seamlessly integrated into Chatter. With Chatter Questions, users can ask questions and find answers without ever needing to leave Chatter. Existing orgs will continue to have access to Answers if it was enabled before the Summer '13 release from now until Winter '18.

As an administrator, you can create custom report types so users can create reports about questions, replies, and votes. Custom report types are the only way to make reports about your answers community available for your users—Salesforce does not provide sample answers reports or a standard report folder for answers.

To create a folder of answers reports for your users:

- 1. Set up a custom report type for answers.
 - The primary object for your report is Questions.
 - There isn't a specific report category for answers, so you probably want to store your report in the Other Reports or Administrative Reports category.
 - When setting up object relationships, Questions can have a relationship with Replies, and Replies can have a relationship with Votes.

EDITIONS

Available in: Salesforce Classic

Answers is available in: Enterprise, Performance, Unlimited, and Developer Editions.

USER PERMISSIONS

To create or update custom report types:

"Manage Custom Report Types"

To delete custom report types:

"Modify All Data"

- 2. Create a new public folder for answers reports. This step requires the "Manage Public Reports" permission.
- **3.** Using your custom report type, create one or more new custom reports for answers. Assign the reports to the new answers reports folder you created.

After completing these steps, a folder of answers reports is available to your users on the Reports home page.

SEE ALSO:

Reports Give You Access to Your Data

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 and Lightning Experience

Available in: **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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create or update custom report types:

"Manage Custom Report Types"

To delete custom report types:

"Modify All Data"

For this field	Do this
	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:

Build a New Report

Reports That You Can Run on Activities

Run standard or custom reports that display details about your tasks or events and their related contacts, accounts, opportunities, and other associated records.

To View This Information	Use This Report
Activities related to another record type, such as opportunities	Click Create New Custom Report on the Reports tab and select the Activities report type.
Number of contacts	$\label{thm:count} \mbox{Activity reports. Add the Relation Count column to your reports.}$
related to an activity	For example, you have a task with five contacts. When you run the Tasks and Events report, it displays "5" to indicate the number of contacts related to the task.
One row per contact for each activity	The Activities with Contacts report. Select at least one field from the Contact object.
	For example, your report includes a task with five contacts. If you include the Last Name field from the contact object on the report, you see five rows (one row for each contact) related to this activity. If you don't add fields from the contact object, you see only one row for the activity.
	The Activities with Contacts report is the only report that:
	Displays all contacts on an activity
	 Can be filtered using attributes of contacts other than the primary contact
	Other reports show the primary contact and the total number of contacts on an activity, but they don't show details of the other contacts.
	The Activities with Contacts report doesn't indicate which contacts are primary or secondary.
An activity's primary contact in two different reports: Tasks and Events, and Activities with Cases	Either the Tasks and Events report or the Activities with Cases report.
	For example, if the primary contact for a task is Ron Dean, these reports display that name in the Contact field.
	The primary contact also appears in the Name field on the task detail page or event detail page.

EDITIONS

Available in Lightning
Experience in: Contact
Manager, Personal, Group,
Professional, Enterprise,
Performance, Unlimited,
and Developer Editions

Available in Salesforce Classic in: **All** Editions

Report on Sales Reps' Meetings with Customers

Get insight into how your sales teams are prioritizing their time. Create a custom report that shows which customers sales reps are spending their time with.

- **1.** Create a custom report type.
 - a. For Primary Object, choose Users.
 - **b.** Complete the remaining required fields: label, description, and category.
 - c. Click Next.
- 2. Define a relationship between activities and users.
 - a. Relate the Activity Relationships object to the Users object.
 - b. Click Save.
- **3.** Define the fields that appear in this report type.
 - a. In the Fields Available for Reports section, click Edit Layout.
 - **b.** Add the following Activity Relationships fields to the layout: Date, Subject, Name, Related To, Relation Count.
 - **c.** Change the label for the Name field to *Contact* and for the Related To field to *Opportunity*.
 - d. For each field, select the option to display the field by default.
 - e. Click Save.
- 4. Create a report.
 - **a.** In the Reports tab, select the report type that you created in steps 1-3.
 - **b.** Choose **Summary Format**, and create a grouping by Full Name.
 - **c.** Run and save the report.

You now have a report that you can use to help your sales teams optimize the time that they spend with customers.

Note: If Shared Activities isn't enabled, custom reports based on activity relationships don't include event organizers along with invitees, and they don't show events to which no one has been invited.

which customers sales reps are spending their time with.

EDITIONS

Available in Lightning
Experience in: **All** Editions

Available in Salesforce Classic in: **All** Editions

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

To view activities owned by you and users below you in the role hierarchy:

No permissions needed

To view all activities:

"View All Data"

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.



Note: This information applies to Collaborative Forecasts and not Customizable Forecasts. To create a forecasting custom report type, enable Collaborative Forecasts.

- 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 a Forecasting object, such as 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 and Lightning Experience

Available in: **Professional** (no custom field forecasts), **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Opportunity Splits available in: **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.



Note: This information applies to Collaborative Forecasts and not Customizable Forecasts. To create a forecasting custom report type, enable Collaborative Forecasts.

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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional** (no custom field forecasts), **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

Opportunity Splits available in: **Performance** and **Developer** Editions and in **Enterprise** and **Unlimited** Editions with the Sales Cloud

Primary Object

Use to create a report for...

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 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

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 appear in the ForecastingItemCategory
values.

Forecasting Items With Opportunities as a related object

Viewing 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 split 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.

Primary Object Use to create a report for... Viewing product family information for specific forecasting line items. For example, you can create a Forecasting Items summary report for each of your subordinates that includes the product families and total price for with Opportunity Product as a related object their forecasting items, with adjustment information and final forecast amounts. Note: This report type shows forecasting items only for the Product Family-Revenue and Product Family-Quantity forecast types. It includes opportunities with and without opportunity products specified. Viewing data about individual or team quotas. As a best practice, include all the default fields in the Forecasting Quotas 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. Viewing quota attainment. For example, you can use Forecasting Quotas and Forecasting Forecasting Quotas with Forecasting Items to create the custom report type. Then, when you create the report, include a team's quotas Items as a related object 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.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create or update custom report types:

"Manage Custom Report Types"

To delete custom report types:

"Modify All Data"

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.

IN THIS SECTION:

1. Create a Folder for Article Reports

Create a public folder where you can store article reports for your users.

2. 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

EDITIONS

Available in: Salesforce Classic, Lightning Experience

Salesforce Knowledge is available in **Performance** and **Developer** Editions and in **Unlimited** Edition with the Service Cloud.

Salesforce Knowledge is available for an additional cost in: **Professional**, Enterprise, and Unlimited Editions.

USER PERMISSIONS

To create or update custom report types:

"Manage Custom Report Types"

To create a public reports folder:

"Manage Public 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.

Article Reports

EDITIONS

Available in: Salesforce Classic, Lightning Experience

Salesforce Knowledge is available in **Performance** and **Developer** Editions and in **Unlimited** Edition with the Service Cloud.

Salesforce Knowledge is available for an additional cost in: **Professional**, **Enterprise**, and **Unlimited** Editions.

USER PERMISSIONS

To create or update custom report types:

"Manage Custom Report Types"

To create a public reports folder:

"Manage Public Reports"

Create a custom report type to report on Salesforce Knowledge article data.

To create a custom report:

1. From Setup, enter *Report Types* in the Quick Find box, then select **Report Types** and click **New Custom Report Type**.

See Create a Custom Report Type

2. In the Primary Object drop-down menu, select the article-related object you want to report on:

Primary Object	Description
Article types (Classic Knowledge only)	Compare information such as creation dates, published channels, and number of associated cases for your custom article type, such as an FAQ. To access any history and custom fields for the custom article type, add the article type's version as a secondary object.
Knowledge Articles	Compare information about individual articles, such as their creation dates, published channels, and number of associated cases. If you choose this custom report type, you can also include article view and vote statistics. 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 Searches	Analyze the number of searches per day, month, or year for each channel and role.
Knowledge Article Version	Compare information about individual translations, such as their creation dates, published channels, and number of associated cases. If you choose this custom report type, you can also include article view and vote statistics.
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 Keyword Search	See which keywords users are looking for in your knowledge base.
Knowledge Search Activity	 Analyze the number of searches per day, month, or year for each channel and language. For each search, see the: Date

EDITIONS

Available in: Salesforce Classic, Lightning Experience

Salesforce Knowledge is available in **Performance** and **Developer** Editions and in **Unlimited** Edition with the Service Cloud.

Salesforce Knowledge is available for an additional cost in: **Professional**, **Enterprise**, and **Unlimited** Editions.

USER PERMISSIONS

To create or update custom report types:

 "Manage Custom Report Types"

To create a public reports folder:

"Manage Public Reports"

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
 - Articles that appear in the search results
- For each article, see the average number of:
 - Clicks
 - Unique users who clicked it
- 3. Complete the required fields and click Save.

In the Store in Category drop-down menu, we recommend choosing **Customer Support Reports** or **Other Reports**. This is the category where users find the custom report type on the Reports tab.

4. Make your choices on the Define Report Records Set page.

Primary Object	Available Secondary Objects
Article types (Classic Knowledge only)	Article type Versions
Knowledge Articles	Article View Statistics, Article Vote Statistics, Case Article
Knowledge Article Searches	None
Knowledge Article Version	Article View Statistics, Article Vote Statistics
Article Version History	None
Knowledge Article Views	None
Knowledge Article Votes	None
Knowledge Keyword Search	None
Knowledge Search Activity	None

5. Click Save.

6. 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 508.

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.

Ø

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

EDITIONS

Available in: Salesforce Classic, Lightning Experience

Salesforce Knowledge is available in **Performance** and **Developer** Editions and in **Unlimited** Edition with the Service Cloud

Salesforce Knowledge is available for an additional cost in: **Professional**, **Enterprise**, and **Unlimited** Editions.

USER PERMISSIONS

To create or update custom report types:

 "Manage Custom Report Types"

To create a public reports folder:

"Manage Public 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 (Classic Knowledge only) Reports

The Article Type primary object allows you to add a secondary object of the article type version.

Table 15: Article Type Report Primary Object Fields

Tubic	13.74 dele Type Report Timary Object Telus
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	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.

EDITIONS

Available in: Salesforce Classic, Lightning Experience

Salesforce Knowledge is available in **Performance** and **Developer** Editions and in **Unlimited** Edition with the Service Cloud.

Salesforce Knowledge is available for an additional cost in: **Professional**, **Enterprise**, and **Unlimited** Editions.

USER PERMISSIONS

To create or update custom report types:

 "Manage Custom Report Types"

To create a public reports folder:

"Manage Public Reports"



Note: Currently you can't use Knowledge Article Version and Last Modified Date in the same report.

Table 16: 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.
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.
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).

Field	Description
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.
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.
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.

Field	Description
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 <code>Daily</code> , <code>Monthly</code> , and <code>Yearly</code> . 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 Version 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.
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.

Field	Description
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.
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.

Field	Description
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

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).

Field	Description
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.

Create a Salesforce Knowledge Search Activity Custom Report Type: Walkthrough

This walkthrough guides you through building a custom report type that supports the Knowledge Search Activity object. This data helps knowledge managers identify ways to monitor content performance and to deliver the most relevant content to end users.

Create a Knowledge Search Activity Report Type

This enables your users to create reports with the Knowledge search fields they need to monitor the performance of their content.



Walk through it: Knowledge Search Activity Custom Report Type

- 1. Click New Custom Report Type.
- 2. Select Knowledge Search Activity.
- 3. Enter a name for the report type.

Your users will select the report type by this name when they use it to create a new report.

4. Enter a description.

This helps your users identify the business objective of this new report type. You might want to enter, "Monitor the performance of your knowledge article content in search results."

5. Select a category.

This is where your users will find the new report type when they create new reports.

6. Select Deployed.

We'll get it active right away.

7. Click Next.

Don't worry, we're almost done.

- 8. Click Save.
- 9. Click Edit Layout.

Customize the fields that are available on reports built using this report type.

EDITIONS

Available in: Salesforce Classic, Lightning Experience

Salesforce Knowledge is available in **Performance** and **Developer** Editions and in **Unlimited** Edition with the Service Cloud.

Salesforce Knowledge is available for an additional cost in: **Professional**, **Enterprise**, and **Unlimited** Editions.

USER PERMISSIONS

To create or update custom report types:

"Manage Custom Report Types"

To create a public reports folder:

"Manage Public Reports"

Live Agent Session Reports

Use Live Agent 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 Live Agent 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 Live Agent session.

Customize Live Agent session reports to include columns of information about any of the following categories, as well as any custom fields related to Live Agent 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 Live Agent Session.
Assistance Flags Lowered (Supervisor)	Number of times a supervisor lowered an assistance flag during an agent's Live Agent Session.
Assistance Flags Raised	Number of times an agent raised an assistance flag during a Live Agent 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.
Live Agent Session ID	ID of the Live Agent session record.
Live Agent Session Name	Automatically generated ID of the Live Agent 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.

EDITIONS

Available in: Salesforce Classic

Live Agent is available in: **Performance** Editions and in **Developer** Edition orgs that were created after June 14, 2012

Live Agent is available in: **Unlimited** Edition with the Service Cloud

Live Agent is available for an additional cost in: **Enterprise** and **Unlimited** Editions

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: **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.

EDITIONS

Available in: Salesforce Classic

Available in: Salesforce for Wealth Management

USER PERMISSIONS

To create or update custom report types:

"Manage Custom Report Types"

To delete custom report types:

"Modify All Data"

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:

- 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To show or hide account owner report:

"Customize Application"

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Salesforce Files available in: **All** editions, **except** Customer Portals

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.



Walk Through It: Report on File Search Activity

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.
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 Create a Report

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)

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

"Create and Customize Reports"

AND

"Report Builder"

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 <i>third</i> summary grouping. The first column grouping becomes the <i>second</i> summary grouping. The second column grouping is removed.
	If you're using the report wizard, <i>both</i> 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 New Report

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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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"

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.

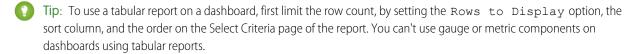
To create reports on an external object, choose the **Other Reports** category. To create reports on an external object with a child standard or custom object, select the category containing the custom report type that you created for that relationship.

- **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.
 - Ø

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 Build a Custom Summary Formula on page 533.
- **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.
- Only the first 254 characters in a rich text area or a long text area are displayed in a report.
- 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.



- **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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

"Create and Customize Reports"



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.

Group Your Report Data

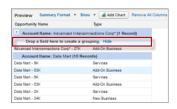
Group data in columns or rows in summary, matrix, and joined reports to display meaningful information. For example, group opportunities by Close Date to see closed opportunities or group cases by product to see the number of cases for each product. You can have groupings inside groupings.

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.



EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

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 New Report Report Fields

Customizing Reports

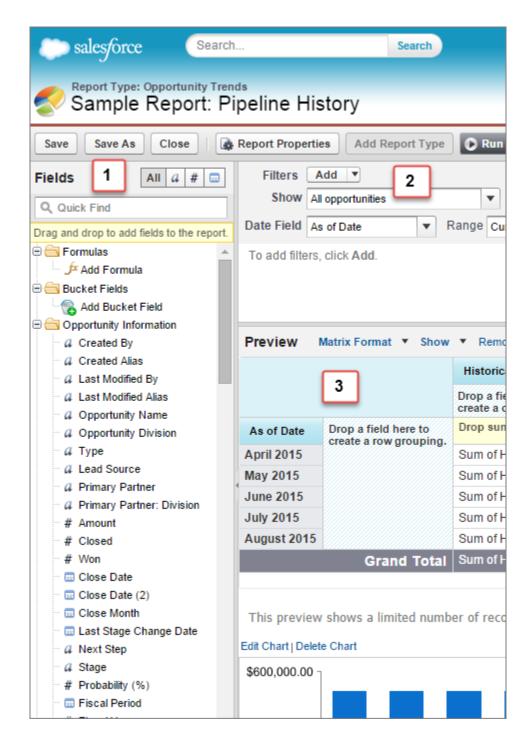
Report builder is a visual editor for reports. The report builder screen lets you work with report fields and filters, and shows you a preview of your report with just some of the data.

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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



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:

Dashboards Help You Visualize Complex Information

Create a Report

Build a New Report

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 and to see all field types, as well as custom summary formulas.
- Click a 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

• 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:

Work with Formulas in Reports Group Your Report Data Summarize Your Report Data Highlight Data Ranges

Summarize Your Report Data

A summary field contains numeric values for which you want to know the sum, the average, or the highest or lowest. Summary fields show at all grouping levels. In summary and matrix reports, they also appear at the grand total level.

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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

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

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:

The Report Run Page
Build a Custom Summary Formula
Build a New Report
Subtotal Report Results

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To subtotal report results:

 "Create and Customize Reports"

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.



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

Work with Formulas in Reports

Formulas let you create custom summaries based on calculated values using report data. These formulas can then be used as columns in your report.

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

To add a new formula to a summary or matrix report, double-click **Add Formula**. Once you define it and click **OK**, it's automatically added to the preview as a column for summary reports, and a summary field for matrix reports. With joined reports, you can add either a cross-block or a standard custom summary formula. To edit a formula, hover over the name in the Fields pane and click \mathscr{D} .

To remove a formula from the preview, grab its header and drag it back to the Fields pane, or click its menu and choose **Remove Formula**. Press **CTRL** to select multiple formulas. Removing a formula

from the preview doesn't delete it. To delete a formula, hover over the name and click **m**, or click **Delete Formula** from the field menu.

Dashboard and report charts that display values from custom 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 applies to currencies, numbers, and percentages.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

IN THIS SECTION:

1. Build a Custom Summary Formula

Create custom summary formulas for summary and matrix reports to calculate additional totals based on the numeric fields available in the report.

2. 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.

3. PARENTGROUPVAL and PREVGROUPVAL

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

4. Get the Most Out of Custom Summary Formulas

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

SEE ALSO:

Build a Custom Summary Formula Report Fields

Build a Custom Summary Formula

Create custom summary formulas for summary and matrix reports to calculate additional totals based on the numeric fields available in the report.

A formula is an algorithm that derives its value from other fields, expressions, or values. Custom summary formulas can contain 3900 or fewer characters. Custom summary formulas are available for summary, matrix, and joined reports. They can't be shared across multiple reports.

- 1. In report builder, click **Add Formula** in the Fields pane.
- **2.** Enter a name for your formula as it will appear on the report. The label must be unique. Optionally, enter a description.
- **3.** From the Format drop-down list, select the appropriate data type for your formula based on the output of your calculation.
- **4.** From the Decimal Places drop-down, 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. If you select **None** for Decimal Places, the effective values are:
 - For currency, displays up to 18 digits, with the default decimal-place precision for the currency
 - For percentages, displays up to 18 digits, with no decimal places
 - For numbers, displays up to 18 digits, without changing the decimal-place precision

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

- 5. Set the Where will this formula be displayed? option. The formula calculation will be displayed in the report at the level you select. To display the formula calculation at every level, including the Grand Total, select All summary levels.
- **6.** Build your formula:
 - **a.** Select one of the fields listed in the **Summary Fields** drop-down list. This field's value is used in your formula.
 - b. Select the kind of summary type to 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.** Click **Operators** to add operators to your formula.
- **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.
- 7. Click **Check Syntax** to see if your formula contains errors. Errors are highlighted by the cursor.

8. Click OK. Your formula isn't saved until you save the report.

SEE ALSO:

Build a New Report

Work with Formulas in Reports

Get the Most Out of Custom Summary Formulas

Build a Custom Summary Formula for a Joined Report

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:

Work with Formulas in Reports
PARENTGROUPVAL and PREVGROUPVAL

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.

TOTAL PRICE:SUM/PARENTGROUPVAL(TOTAL PRICE:SUM, GRAND SUMMARY)

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

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 only use this function in custom summary formulas for reports.
Use:	Summary and Joined: PARENTGROUPVAL(summary_field, grouping_level)
	<pre>Matrix: PARENTGROUPVAL(summary_field, parent_row_grouping, parent_column_grouping)</pre>
	Where summary_field is the summarized field value, grouping_level is the parent level for summary reports, and parent_row_level and parent_column_level are the parent levels for matrix reports.
Example:	

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 only use this function in custom summary formulas for reports.
Use:	<pre>PREVGROUPVAL(summary_field, grouping_level [, increment])</pre>
	Where summary_field is the name of the grouped row or column, grouping_level is the summary level, and increment is the number of rows or columns previous.
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:

Build a Custom Summary Formula
Work with Formulas in Reports
Build a Custom Summary Formula for a Joined Report

Get the Most Out of Custom Summary Formulas

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

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

- A summary formula can't reference another summary formula.
- 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.
- Dashboard and report charts that display values from custom 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 applies to currencies, numbers, and percentages.
- When a field is deleted or is unavailable (for example, because of field-level security), all custom summary formulas that contain the field are removed from the report.
- The summary types Sum, Largest Value, Smallest Value, and Average are not 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 custom summary formulas on matrix reports, Salesforce calculates results for all formulas where the Where will this
 formula be displayed? option is not set to All summary levels. The formula determines the value returned. If
 null is returned, the cell will be empty.
- Summary fields on tabular, summary, and matrix reports can display up to 21-digits. Summary fields of 21 digits will display precisely, but more than 21 digits may not be precise. If "#Too Big!" appears in report cells, check your formula for calculations that could result in more than 18 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 custom summary formulas are available for joined reports.

SEE ALSO:

Build a Custom Summary Formula Custom Summary Formulas with Joined Reports Build a Custom Summary Formula for a Joined Report

EDITIONS

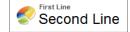
Available in: both Salesforce Classic and Lightning Experience

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

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:



EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** Editions except **Database.com**

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

To enter a unique report name used by the API:

"Customize Application"

To select a folder to store public reports:

"Manage Public Reports"

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 reporting with charts is supported in Lightning Experience, but tabular views of historical trend reports aren't available.

IN THIS SECTION:

1. 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.

2. 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.

3. 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.

4. Learn about 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.

EDITIONS

Available in: both Salesforce Classic Lightning Experience

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

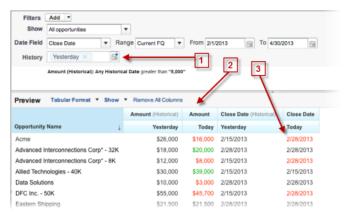
5. Limitations on Historical Trend Reporting

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.

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 summary report points out opportunity amounts and close dates that have changed since yesterday.



EDITIONS

Available in: both Salesforce Classic Lightning Experience

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

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

IN THIS SECTION:

1. 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.

2. 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.

3. 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.

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.

- 1. Create an opportunity history report.
 - **a.** On the Reports tab, click **New Report**.
 - **b.** Under **Select Report Type** > **Opportunities**, select *Opportunities* with *Historical Trending*.

In order to see *Opportunities with Historical Trending*, turn on Historical Trend Reporting for Opportunities in the Salesforce Setup. To learn how to turn on Historical Trend Reporting for Opportunities, see *Set Up Historical Trend Reporting* in the Salesforce help.

- c. Click Create.
- 2. For Show, select All Opportunities.

Note that the default value in the Historical Date Field 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.

EDITIONS

Available in: both Salesforce Classic Lightning Experience

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

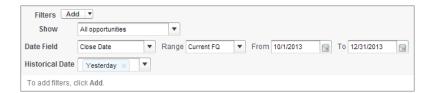
USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"



3. Click the down arrow in the headers of the Amount (Historical) and the Amount—Today columns. Click Summarize This Field, then select Sum.

The total amount appears at the bottom of each column.

- 4. Click the down arrow in the header of the Amount column and select Show Changes.
- 5. Click Run Report.
- **6.** In the Change column, observe the difference between the total value of the Amount (Historical) column and that of the Amount-Today column.
 - 1 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

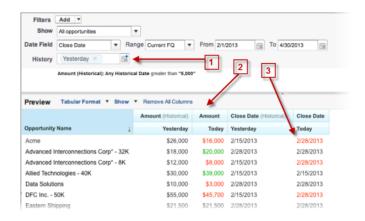
Identify Historical Deals Over a Given Value

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.** Under **Select Report Type** > **Opportunities**, select *Opportunities with Historical Trending*.

In order to see *Opportunities with Historical Trending*, turn on Historical Trend Reporting for Opportunities in the Salesforce Setup. To learn how to turn on Historical Trend Reporting for Opportunities, see *Set Up Historical Trend Reporting* in the 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.
 - 1 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.

EDITIONS

Available in: both Salesforce Classic Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

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.** Under **Select Report Type** > **Opportunities**, select *Opportunities* with *Historical Trending*.

In order to see *Opportunities with Historical Trending*, turn on Historical Trend Reporting for Opportunities in the Salesforce Setup. To learn how to turn on Historical Trend Reporting for Opportunities, see *Set Up Historical Trend Reporting* in the 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.
 - 1 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.
 - 1 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

EDITIONS

Available in: both Salesforce Classic Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

- "Create and Customize Reports"
 - AND
 - "Report Builder"

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.

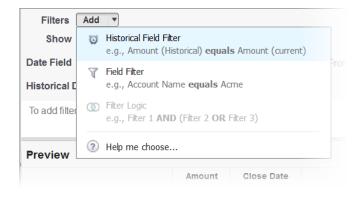
- 1. Create an opportunity history report.
 - a. On the Reports tab, click New Report.
 - **b.** Under **Select Report Type** > **Opportunities**, select *Opportunities* with *Historical Trending*.

In order to see *Opportunities with Historical Trending*, turn on Historical Trend Reporting for Opportunities in the Salesforce Setup. To learn how to turn on Historical Trend Reporting for Opportunities, see *Set Up Historical Trend Reporting* in the 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 $\mathtt{Amount} - \mathtt{Today}$ 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.

1 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

EDITIONS

Available in: both Salesforce Classic Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

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 guarter.

- 1. Create a forecast history report.
 - a. On the Reports tab, click New Report.
 - **b.** Under **Select Report Type** > **Forecasts**, select *Forecasting Items with Historical Trending*. Before you can select this report type, your organization must have enabled Historical Trending for Forecasting Items. If you do not see the report type listed, talk to 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.

EDITIONS

Available in: both Salesforce Classic Lightning Experience

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

USER PERMISSIONS

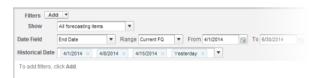
To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

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.

- 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.

EDITIONS

Available in: both Salesforce Classic Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

"Create and Customize Reports"

AND

Learn about 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.

EDITIONS

Available in: both Salesforce Classic Lightning Experience

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

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.



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

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. 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, this includes
 the standard preselected fields, which can't be disabled.
- Formula fields are not supported.
- Row limit filters are not supported.
- The summary report format is not 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.
- Dated exchange rates are not supported. When you run the historical trend report, it only uses the latest dated exchange rate.
- Internet Explorer 6 is not supported.
- You can't subscribe to historical trend reports.
- The Report Wizard is not 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.

(1) 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.

EDITIONS

Available in: both Salesforce Classic Lightning Experience

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

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*.

IN THIS SECTION:

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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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..

EDITIONS

Available in: both Salesforce Classic 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

- 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 Apexcode 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 Force.com Platform or Salesforce Platform One licenses will not be able to view it. Alternatively, if the Running User has a Force.com Platform or Salesforce Platform One license, users who have Salesforce licenses will be able to see the reporting snapshot. If you have users with Force.com Platform or Salesforce Platform One licenses, we recommend creating a separate reporting snapshot for them with a Running User that has a Force.com 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

SEE ALSO:

Report on Historical Data with Reporting Snapshots Create a Custom Report in Accessibility Mode

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.

 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.

EDITIONS

Available in: both Salesforce Classic 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

- 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.

SEE ALSO:

Standard Report Types

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.

- 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.

EDITIONS

Available in: both Salesforce Classic 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

• 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:

Report on Historical Data with Reporting Snapshots Build a Custom Summary Formula

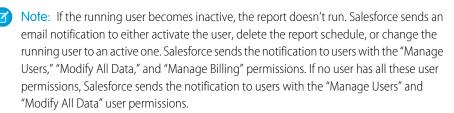
Schedule and Run a Reporting Snapshot

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.

- 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 will not run as scheduled if the user in the Running User field does not have access to the folder in which the source report is stored.



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. Furthermore, you can only select Users and Public Groups in the Search drop-down list.
- **5.** Schedule the reporting snapshot to run:
 - 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

EDITIONS

Available in: both Salesforce Classic 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

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. Note that if the target object's status is In Development, the running user must have the "Customize Applications" permission.

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.
 - (1) 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.

EDITIONS

Available in: both Salesforce Classic 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

- 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.
 Any records over 2000 are recorded in the Row Failures related list. If the Total Row Number is blank and the run history indicates "Some rows failed," the report likely contained more than 2000 summaries.
- The details of a failed run are available on the Row Failures related list for 14 days before they are automatically deleted. You cannot 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.

EDITIONS

Available in: both Salesforce Classic 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

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 cannot 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. Matrix reports cannot 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

Categorize Data Quickly with Buckets

Bucketing lets you quickly categorize report records without creating a formula or a custom field. When you create a bucket field, you define multiple categories (buckets) used to group report values.

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

Bucketing lets you quickly categorize report records without creating a formula or a custom field. For example, create a bucket field named Size based on the # Employees field. Then, create buckets that group records into "Large," "Medium," or "Small" ranges that you define. Bucket fields can be used like any other field to sort, filter, and group your report.

EDITIONS

Available in: Salesforce Classic

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

IN THIS SECTION:

1. Add a Bucket Field

Create a bucket field to contain the buckets into which you want to organize your report data.

2. Edit a Bucket Field

Set up your bucket field by specifying the buckets that it contains and the values that go in the buckets.

3. Edit a Numeric Bucket Field

A numeric bucket helps you sort data that can be described in terms of numbers.

4. 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.

5. Edit a Picklist Bucket Field

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

6. Picklist Bucketing Example: Industry Types

Use a picklist bucket field to sort your accounts by their industry.

7. Edit a Text Bucket Field

A text bucket helps you sort out values that consist of words or phrases.

8. Enter Values for Text Buckets

A text bucket field needs a descriptive name and two or more buckets containing text strings.

9. Text Bucketing Example: Strategic Accounts

Use a text bucket to sort accounts into general categories that you can act on.

Add a Bucket Field

Create a bucket field to contain the buckets into which you want to organize your report data. You can add up to five bucket fields per report, each with up to 20 buckets.

- 1. In the Fields pane of the report builder, double-click **Add Bucket Field** or drag it into the report preview. You can also click a column menu for a field in the report and select **Bucket this Field**.
- **2.** Edit the bucket field according to the field type.
 - Edit a Numeric Bucket Field
 - Edit a Picklist Bucket Field
 - Edit a Text Bucket Field

Fields with other field types are not supported.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

Edit a Bucket Field

Set up your bucket field by specifying the buckets that it contains and the values that go in the buckets.

- 1. In the report builder 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.
- **2.** Edit the bucket field according to the field type.
 - Edit a Numeric Bucket Field
 - Edit a Picklist Bucket Field
 - Edit a Text Bucket Field

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

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:

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

EDITIONS

Available in: Salesforce

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

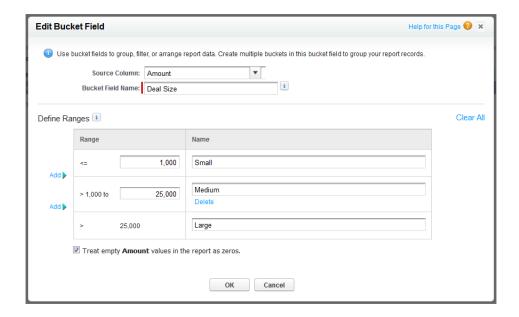
USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

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.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

- (1) 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.

EDITIONS

Available in: Salesforce Classic

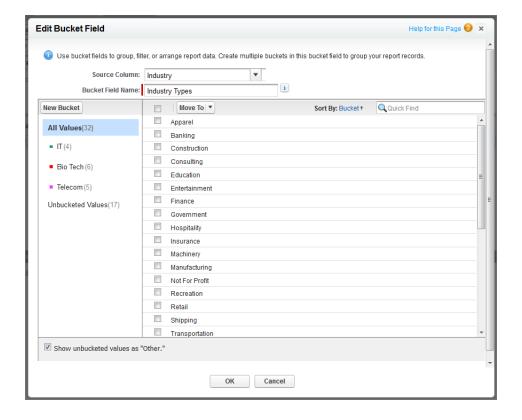
Available in: **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

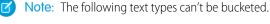


SEE ALSO:

Edit a Picklist Bucket Field

Edit a Text Bucket Field

A text bucket helps you sort out values that consist of words or phrases.



- 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:

- 2. Enter a bucket field name. This appears as the column name in the report. field describes the scope of the buckets. For example, a bucket field named "Region" could
- 1. For Source Column, select the field you want to bucket. Since a bucket field is intended to have multiple buckets within it, a good name for a bucket 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.



562

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

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

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**.

EDITIONS

Available in: Salesforce Classic

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

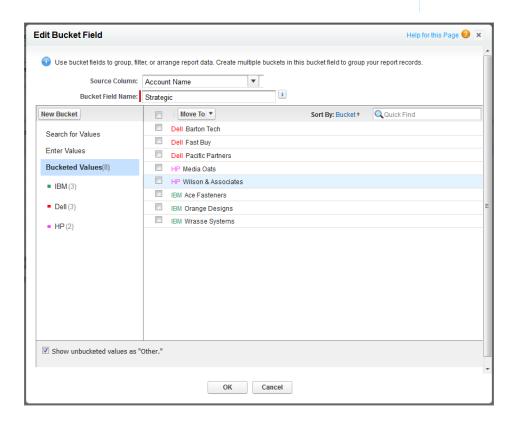
USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"



SEE ALSO:

Edit a Text Bucket Field

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.

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

To get started with joined reports, create a new or edit an existing report in report builder, click the Format drop-down, and choose Joined.

Most of the things you can do with summary or matrix reports you can also do with joined reports. For example, you can find, add, and remove fields; summarize fields; and run and save reports.



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.

IN THIS SECTION:

1. 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.

2. 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

3. Build a Custom Summary Formula for a Joined Report

You can add standard or cross-block custom summary formulas for joined reports to calculate additional totals based on the numeric fields available in the report.

4. Work with Blocks

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

5. 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.

SEE ALSO:

Build a New Report

How Joined Reports Work

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

"Create and Customize Reports"

AND

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.

1. Select Joined from the report format menu.

2. Click Add Report Type.

The Choose an Additional Report Type overlay appears and displays the report types that you can add to the existing report.

3. Select the report type.

The overlay displays a message that identifies the objects that are common to the selected report type and the types already included in the report.

4. Click OK.

The additional report type is added. Notice that:

- A new block appears in the report.
- 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.

SEE ALSO:

Build a Custom Summary Formula for a Joined Report How Joined Reports Work

Combine Different Types of Information in a Joined Report

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. To remove a block, drag it to the Fields pane. If there are multiple blocks based on the principal report type, you must remove them all.

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.
- 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 to a Joined Report

Add a Report Type to a Joined Report

What Can't I Do with Joined Reports?

Combine Different Types of Information in a Joined Report

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

Build a Custom Summary Formula for a Joined Report

You can add standard or cross-block custom summary formulas for joined reports to calculate additional totals based on the numeric fields available in the report.

1. Use one of these options to access the Add Summary Formula overlay from the Fields pane.

Option	Description
To create a standard custom summary formula	Go to the report type category, and double-click Add Formula
To create a cross-block custom summary formula	Go to the Common category and double-click Add Cross Block Formula

- **2.** Enter a name for the formula as you want it to appear in the report, and optionally a description. The name must be unique.
- **3.** From the Format drop-down list, select the appropriate data type for your formula based on the output of your calculation.
- **4.** From the Decimal Places drop-down, 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.
- 5. Set the Where will this formula be displayed? option.

The calculated value displays in the report block at either the Grand Total or the global grouping level, depending on which you select. To display the formula calculation at every level, including the Grand Total, select All summary levels. Optionally you can select to show the calculation at a global grouping level. You can have three global groupings in a joined report. The groupings apply across all blocks.

- **6.** Build your formula:
 - **a.** Select one of the fields listed in the **Summary Fields** drop-down list. This field's value is used in your formula. When creating a cross-block formula, the fields are grouped by block. You can also use Quick Find to search for a field. When creating a single-report type formula, the list displays numeric fields available for the report type.
 - b. Select the kind of summary type to 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.** Click **Operators** to add operators to your formula.
- **d.** Select the function category (All, Logical, Math, or Summary), choose the function you want to use in your formula, and click **Insert**.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

- "Create and Customize Reports"
 - AND

- e. Repeat these steps as necessary.
- 7. Click **Check Syntax** to see if your formula contains errors. Errors are highlighted by the cursor.
- **8.** Click **OK**. Your formula isn't saved until you save the report.

SEE ALSO:

Work with Formulas in Reports
Custom Summary Formulas with Joined Reports
Get the Most Out of Custom Summary Formulas
Joined Report Examples

Work with Blocks

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

IN THIS SECTION:

1. Add a Block to a Joined Report

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 a 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

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

Add a Block to a Joined Report

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

1. Click and hold a field from the Fields pane, then drag it to the empty area of the Preview pane.

EDITIONS

Available in: Salesforce Classic

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

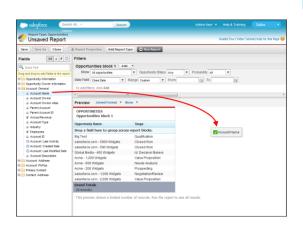
USER PERMISSIONS

To create, edit, and delete reports:

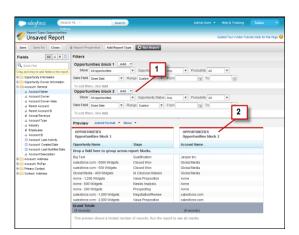
 "Create and Customize Reports"

AND

"Report Builder"



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).

To delete a block, click 💌 in the block header, then click **Remove Block**. Or just drag the block to the Fields pane.

Keep the following in mind when working with blocks.

- Adding a new report type to a joined report also adds a new block.
- You must choose a field from a report type category when creating a block. You can't use a field from the Common category. For example, if your report contains both the Opportunities and the Cases report types, choosing a field from the Opportunities category creates a new Opportunities block, while choosing a field from the Cases category creates a new Cases block.

SEE ALSO:

Add a Report Type to a Joined Report

Delete a Block

Build a Custom Summary Formula for a Joined Report

Work with Blocks

Combine Different Types of Information in a Joined Report

Reorder Blocks

You can reorder blocks in a 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. You should rename the blocks to avoid confusion.

When reordering a block, you move it to either the left or right of an existing block. You can't drag it to an empty area in the Preview pane.

To reorder a block:

• 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.

SEE ALSO:

Rename a Block

Work with Blocks

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

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.

1. Click the block's name.

The name now appears in an editable text box.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"



- 2. Enter the new name.
- **3.** Press Enter or click outside the block to apply the name.

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.

You can hide or show record count two ways.

- 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.
- 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.



SEE ALSO:

Show and Hide Report Details
Use a Summary Function in a Custom Summary Formula

Delete a Block

There are two ways to delete a block from a joined report.

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

SEE ALSO:

Work with Blocks

Combine Different Types of Information in a Joined Report

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

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.

EDITIONS

Available in: Salesforce Classic

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

Each joined report has a *principal* report type. By default, the principal type is the first one added to the report, and is identified in the Fields pane with a small dot beside its name. 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. In those fields, click i to see the name of the field in other 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. These fields appear in the Common Fields area in the Fields pane, and can be used to group report blocks.

IN THIS SECTION:

1. 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.

2. Custom Summary Formulas with Joined Reports

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

3. What Can't I Do with Joined Reports?

Most of the things you can do with summary or matrix 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.

SEE ALSO:

Add a Report Type to a Joined Report Build a Custom Summary Formula for a Joined Report Work with Blocks

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.

Creating a Sales Rep Performance Scorecard

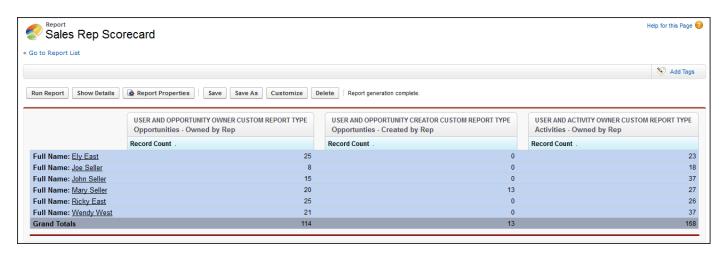
A sales rep scorecard lets your sales management team understand the performance and actions of your organization's sales reps. To create it, you need to have three separate custom report types, each of which creates a relationship between User (as the primary object) and one of the following three objects: 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.

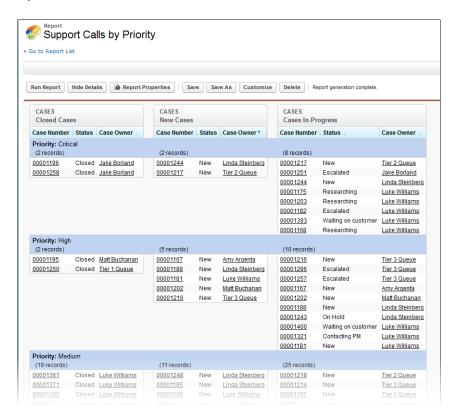


Reviewing Support Cases by Status

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

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.

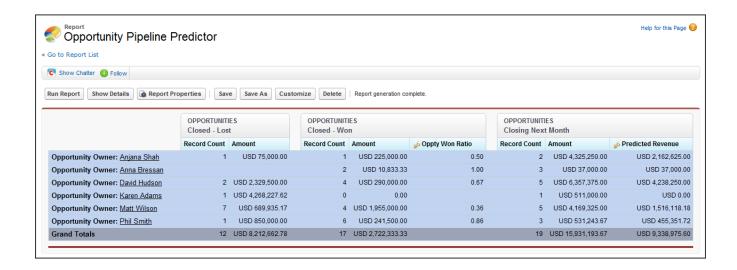
Block	Filters
Block 1	Opportunity Status equals Closed
	Date Field equals Close Date
	Range equals Current and Previous FY
	Stage equals Closed Lost
Block 2	Opportunity Status equals Closed Won
	Date Field equals Close Date
	Range equals Current and Previous FY
Block 3	Date Field equals Close Date
	Range 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:

- **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.

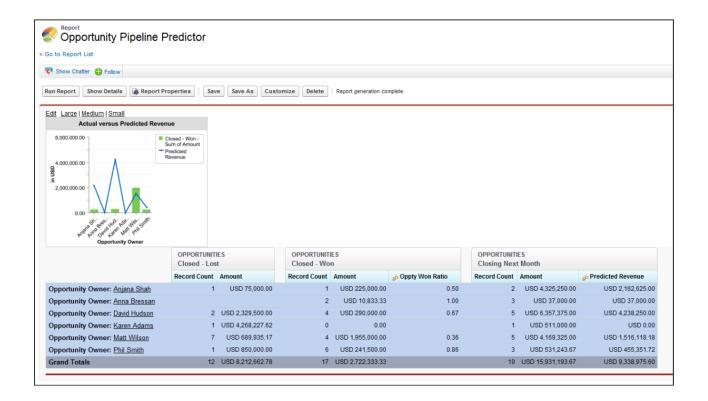


Adding a Chart to the Opportunity Pipeline Predictor Report

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 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

Custom Summary Formulas with Joined Reports

How Joined Reports Work

Work with Blocks

Combine Different Types of Information in a Joined Report

Custom 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. Custom summary formulas in joined reports support the same data formats, formula options, functions, and calculation display locations as they do with summary and matrix reports.

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

EDITIONS

Available in: Salesforce Classic

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

- The formulas aren't automatically added to the report when you create them. To add a formula, 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. Note that 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.
- Cross-block formulas aren't automatically added to the report when you create them. To add a formula, 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 "]").

SEE ALSO:

Build a Custom Summary Formula
Get the Most Out of Custom Summary Formulas
Joined Report Examples
Build a Custom Summary Formula for a Joined Report

What Can't I Do with Joined Reports?

Most of the things you can do with summary or matrix 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.

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 opportunity or activity reports.
- Create reporting snapshots based on joined reports.

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.

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

EDITIONS

Available in: Salesforce Classic

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

- 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
- Forecasts
 - Customizable Forecasting: Forecast History
 - Customizable Forecasting: Forecast Summary
 - Customizable Forecasting: Opportunity Forecasts
 - Forecast History
 - Forecasts
 - Quota versus Actual*
- 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.

SEE ALSO:

Choose a Report Type
Change the Principal Report Type
Combine Different Types of Information in a Joined Report
Standard Report Types

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.

IN THIS SECTION:

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 Different Data Sets 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 Different Data Sets 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 New Report

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 586 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

"Create and Customize Reports"

Field	Description	
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.	

Field	Description
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 on summary or matrix reports based on ranges and colors you specify. To enable conditional highlighting, your report must contain at least one summary field or custom summary formula.

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 exactly 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 exactly 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 non-stacked bar and column charts. Negative values on pie, donut, funnel, and stacked charts are not displayed. Groupings containing negative values are displayed in the legend, and negative values are reflected in the calculation of all summary values, including the total for donut 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.
- Dashboard and report charts that display values from custom 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 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 <i>third</i> summary grouping. The first column grouping becomes the <i>second</i> 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	The existing report becomes the first block and the chart remains in the report.
joined	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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

When you make this change	The effect is
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 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 Different Data Sets in One Chart

Show Different Data Sets 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 more columns to a vertical column chart.
- Add up to three more bars to a horizontal bar chart.

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

- **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

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

- 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 drop-down 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.** Once you've set up your report, click **Run Report** or **Save**.



- 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.
- Both X-Axis and Y-Axis must not be set to Auto. If set to Auto, you can't plot the chart.
- Filtered drill-down 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.

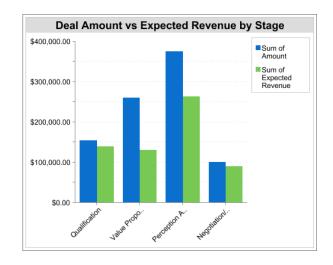
A combination chart plots multiple sets of data on a single chart.

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.

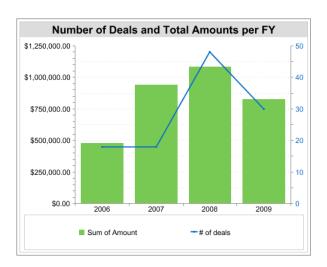


Line-on-Column

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.

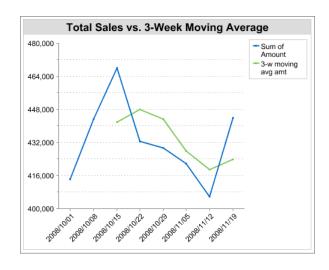


Line-on-Line

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 Different Data Sets 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.

IN THIS SECTION:

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

Highlight field values on summary or matrix reports based on ranges and colors you specify. To enable conditional highlighting, your report must contain at least one summary field or custom summary formula.

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
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 New Report
Combine Different Types of Information in a Joined Report

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

USER PERMISSIONS

To run reports:

"Run Reports"

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

Highlight Data Ranges

Highlight field values on summary or matrix reports based on ranges and colors you specify. To enable conditional highlighting, your report must contain at least one summary field or custom summary formula.

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 exactly 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 exactly 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 ${\tt High}\ {\tt Breakpoint}$ value.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

"Create and Customize Reports"

AND

"Report Builder"

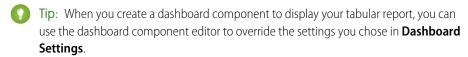
SEE ALSO:

Build a New 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.



EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

"Create and Customize Reports"

AND

"Report Builder"

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.
- Cross filters work like ordinary filters, but they have some special characteristics of their own.
 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

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 <code>Customer notes</code> 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:

The Report Run Page

Example: Report on Related Objects with Cross Filters

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: When Salesforce disables TLS 1.0, we're ending support for Connect for Office. This change means that there's no guarantee that Connect for Office can establish a connection between Salesforce and Microsoft® Word or Excel. Even if sales reps can establish a connection, Salesforce no longer provides support or maintenance when there's an issue.

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.

EDITIONS

Available in: Salesforce Classic

Available in all editions

USER PERMISSIONS

To access Salesforce reports from Excel:

"Run Reports"

AND

"Export Reports"

IN THIS SECTION:

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.

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.

(1) Important: When Salesforce disables TLS 1.0, we're ending support for Connect for Office. This change means that there's no guarantee that Connect for Office can establish a connection between Salesforce and Microsoft* Word or Excel. Even if sales reps can establish a connection, Salesforce no longer provides support or maintenance when there's an issue.

The system requirements for Connect for Office are:

- Microsoft® Office 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.

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.

(1) Important: When Salesforce disables TLS 1.0, we're ending support for Connect for Office. This change means that there's no guarantee that Connect for Office can establish a connection between Salesforce and Microsoft® Word or Excel. Even if sales reps can establish a connection, Salesforce no longer provides support or maintenance when there's an issue.

EDITIONS

Available in: Salesforce Classic

Available in all editions

- 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.
- Note: 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 must add to the end of your password. For example, if your password is mypassword, and your security token is xxxxxxxxxxxx, you must enter mypasswordxxxxxxxxxxx to log in.

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.

(1) Important: When Salesforce disables TLS 1.0, we're ending support for Connect for Office. This change means that there's no guarantee that Connect for Office can establish a connection between Salesforce and Microsoft® Word or Excel. Even if sales reps can establish a connection, Salesforce no longer provides support or maintenance when there's an issue.

- 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.

EDITIONS

Available in: Salesforce Classic

Available in all editions

USER PERMISSIONS

To access reports in Excel:

- "Run Reports"
 - AND
 - "Export Reports"

(1) Important: When Salesforce disables TLS 1.0, we're ending support for Connect for Office. This change means that there's no guarantee that Connect for Office can establish a connection between Salesforce and Microsoft* Word or Excel. Even if sales reps can establish a connection, Salesforce no longer provides support or maintenance when there's an issue.

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.

Upgrade the Report Wizard

Report builder, a powerful drag-and-drop editor, is the standard tool for creating and editing reports. If your organization is still using the old report wizard, you should upgrade to report builder.

- All profiles get access to the report builder by default. (You may continue to see the "Report Builder" permission in permission sets and profiles and the PermissionSet and Profile objects in the API, though the upgrade overrides those settings.)
- The old report wizard is available only to users in Accessibility Mode.
- Group and Professional Edition organizations can use report builder.
- You get scatter charts, a new chart type for reports.

New organizations automatically get the latest version of report builder. If you don't see the Report Builder Upgrade section on the User Interface Settings page, the upgrade has already been enabled for your organization.

Assigning the "Report Builder" permission to all users through profiles or permission sets isn't the same thing as enabling report builder for your entire organization. To enable report builder for your organization, follow these steps.

EDITIONS

Available in: Salesforce Classic

Available in: **All** Editions except **Database.com**

USER PERMISSIONS

To modify report and dashboard settings:

"Customize Application"

- (1) Important: Upgrading does not affect any of your existing reports. However, once you upgrade, you can't return to the old report wizard.
- 1. From Setup, enter *Reports* in the Quick Find box, then select **Reports and Dashboards Settings**.
- 2. Review the Report Builder Upgrade section of the page and click **Enable**. If you don't see the button, report builder has already been enabled for your entire organization.

3. Confirm your choice by clicking Yes, Enable Report Builder for All Users.

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.

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
- Cross filters
- Historical field filters
- Standard filters (except for scope and date filters)
- 1. From the 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. If you'd like 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.
 - 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.
 - **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.

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 entering filter criteria, see Filter Operators Reference and Add 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**. In new reports, the Run Report button doesn't appear until you save your report.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To add or edit a filter

 "Create and Customize Reports"

AND

"Report Builder"

To lock or unlock filters so that users can't edit them while viewing a report in Lightning Experience

 "Create and Customize Reports"

AND

"Report Builder"

To edit a filter while viewing a report in Lightning Experience

"Run Reports"

- Ø
- **Note:** In the Lightning Experience Report Builder, save your report before running it or risk losing some changes. Specifically, be sure to save after adding:
- Row limit filters
- Cross filters
- Formula fields
- Bucket fields
- **◎** E

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.

IN THIS SECTION:

Add Filter Logic

Filter logic governs how and when filters apply to your report.

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.

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 while 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.

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.

1. 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

To create custom list views:

"Read" on the type of record included in the list

To create, edit, or delete public list views:

 "Manage Public List Views"

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 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 only 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

Role hierarchy filters aren't available on reports based on custom report types.

Filter by Role Hierarchy in Lightning Experience

1. If necessary, from the report run page, click Υ to open the Filters panel.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To add or edit a filter

 "Create and Customize Reports"

AND

"Report Builder"

To lock or unlock filters so that users can't edit them while viewing a report in Lightning Experience

 "Create and Customize Reports"

AND

"Report Builder"

To edit a filter while viewing a report in Lightning Experience

"Run Reports"

- 2. From the Filters panel, click Role Hierarchy.
- 3. From the Role Hierarchy modal, filter by a role. To find specific roles, consider using the search box.
- **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.

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.

- 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 £v4. 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/<rp>

• 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.

EDITIONS

Available in: Lightning Experience

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

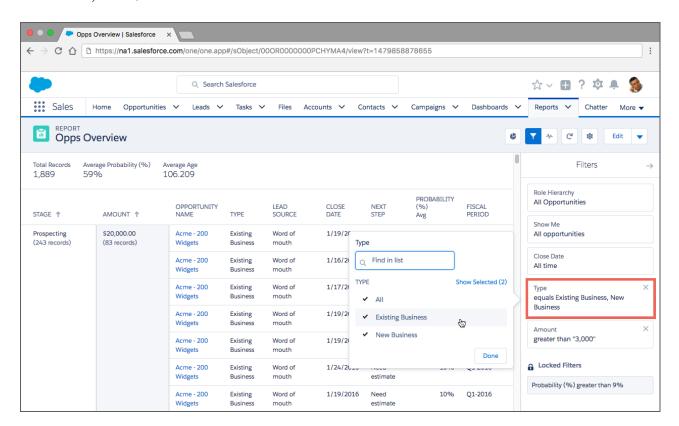
USER PERMISSIONS

To filter a report in Lightning Experience:

"Run Reports"

0

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.)

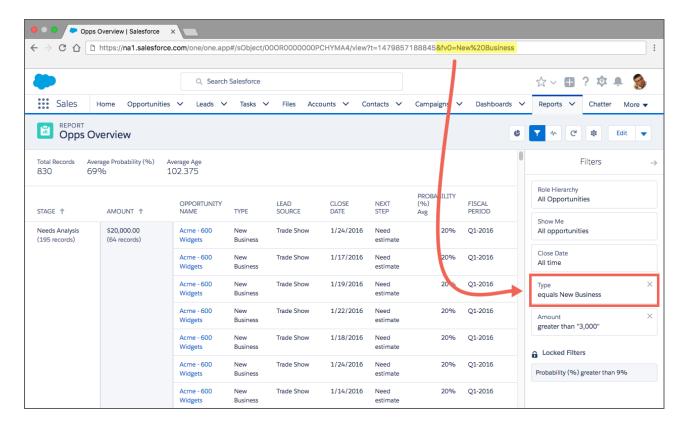


Instead of loading the report, editing the filter, and reloading the report, create a bookmark with a filter value parameter for New Business in the URL. Take note of the unfiltered opportunities report URL.

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 URl 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 URl encoded. The full, bookmarked URL reads:

https://nal.salesforce.com/one/one.app#/s0bject/000R000000PCHYMA4/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. Standard filters (role hierarchy filters, scope filters, date filters), cross filters, and row limit filters aren't supported. Standard, cross, and row limit filters don't count when listing filters fv0 is the first field filter.
- Chart filters aren't supported.
- 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.

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 and Date Field. Show filters the object around common groupings (like "My accounts" or "All accounts"). Date Field filters by a field (such as Created Date or Last Activity) and a date range (such as "All Time" or "Last Month").

Filter Type	Description
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. With tabular, summary, and matrix reports, you can drag a field from the Fields pane to the Filters pane to add a report filter.
Filter Logic	Add Boolean conditions to control how field filters are evaluated. You must 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 <i>Accounts with Opportunities</i> , click Add Opportunity Filter and create the <i>Opportunity Name equals ACME</i> 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.

IN THIS SECTION:

Example: Report on Related Objects with Cross Filters

Use a cross filter to fine-tune your results by including or excluding records from related objects, without having to write formulas or code. You can apply cross filters by themselves, or in combination with field filters.

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: Report on Related Objects with Cross Filters

Use a cross filter to fine-tune your results by including or excluding records from related objects, without having to write formulas or code. You can apply cross filters by themselves, or in combination with field filters.

Watch a Demo: Using Cross Filters in Reports (Salesforce Classic)

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.

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 Nameequals ACME subfilter to see just those opportunities. You can create up to five subfilters for each cross filter.

- 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.

IN THIS SECTION:

1. Create a Cross Filter

Use cross filters to include or exclude records in your report results based on related objects and their fields.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

2. Example: Using 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.

3. Example: Using 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.

4. Example: Using Multiple Cross Filters

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.

5. Tips for Working with Cross Filters

Cross filters work like ordinary filters, but they have some special characteristics of their own.

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 Demo: Using Cross Filters in Reports (Salesforce Classic)

- 1. In the Filters pane of report builder, click **Add** > **Cross Filter**.
- **2.** Select a parent object from the drop-down 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.
- **3.** Choose with or without.
- **4.** Select a child object from the drop-down or search by its name. The drop-down list contains all eligible child objects of your selected parent object.
- **5.** Optionally add subfilters:
 - a. Click Add Your Related Child Object Filter.
 - **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.
- 6. Click OK.

SEE ALSO:

Example: Using Multiple Cross Filters
Example: Using WITH in Cross Filters
Example: Using WITHOUT in Cross Filters

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

Example: Using 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.

- 1. Create a new report. For the report type, click **Accounts & Contacts**, select **Accounts**, and click **Create**.
- 2. In the report builder's Filters pane, set the appropriate standard filters.
- 3. Create a field filter where Billing State/Province equals CA and click OK.
- 4. Click Add > Cross Filter and specify Accounts with Cases.

When you click **Run Report**, the results will include only California accounts with an associated case.

SEE ALSO:

Create a Cross Filter

Example: Using 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:

- Create a new report. For the report type, click Accounts & Contacts, select Accounts, and click Create.
- 2. In the report builder's Filters pane, set the appropriate standard filters.
- 3. Create a field filter where Billing State/Province equals CA and click OK.
- **4.** Click Add > Cross Filter and specify Accounts without Contacts.

When you click **Run Report**, the results will include only California accounts without an associated contact.

SEE ALSO:

Create a Cross Filter

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

Example: Using Multiple Cross Filters

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 problem escalations because you want to ensure your support team takes care of them.

- 1. Create a new report. For the report type, click **Accounts & Contacts**, select **Accounts**, and click **Create**.
- 2. In the report builder's Filters pane, set the appropriate standard filters.
- **3.** To limit your report to customer accounts, click **Add** > **Field Filter** and specify *Type equals Customer*.
- **4.** To see which accounts have no activities, click **Add** > **Cross Filter** and specify *Accounts* without *Activities*.
- **5.** To see only accounts without completed activities, add a subfilter to your cross filter:
 - a. Click Add Activities Filter and specify Status equals Completed.
 - **b.** Click **OK**.
- **6.** To include your accounts that currently have cases, add another cross filter but this time specify *Accounts with Cases*.
- 7. To exclude cases that were not escalated, add a subfilter to this cross filter:
 - a. Click Add Cases Filter and specify Type equals Problem.
 - b. Click Add Cases Filter again, but this time specify Status equals Escalated.
 - c. Click OK.

When you run the report, it will include only customer accounts without completed activities with escalated cases.

SEE ALSO:

Create a Cross Filter

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

Tips for Working with Cross Filters

Cross filters work like ordinary filters, but they have some special characteristics of their own.

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. Therefore, choosing a report type of Accounts with
 Partners and adding a cross filter for Accounts without Partners will yield no results.

SEE ALSO:

Example: Using Multiple Cross Filters

Create a Cross Filter

Example: Using WITH in Cross Filters

Example: Using WITHOUT in Cross Filters

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"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. 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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

 "Create and Customize Reports"

AND

"Report Builder"

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 while 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.		
not equal to	Shows results that don't have the value you enter. This operator is especially useful for eliminating empty fields. For example, "Email not equal to <blank>."</blank>		
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.		

Operator	Uses
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, 2016, filter using a relative date: Close Date equals this year.



Mote: Capitalization doesn't matter in relative date filter operators. THIS YEAR works, as do This Year and this year.

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 <i>n</i> 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 <i>n</i> times seven days.
n WEEKS AGO	Starts at 12:00:00 AM on the first day of the week that started <i>n</i> weeks before the start of the current week and continues for seven days.
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.

Relative Date Value	Range
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 <i>n</i> th month.
LAST n MONTHS	Starts at 12:00:00 AM on the first day of the month that started <i>n</i> 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 <i>n</i> 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 90 DAYS	Starts at 12:00:00 AM 90 days before the current day and continues up to the current second. (The range includes today.)
NEXT 90 DAYS	Starts at 12:00:00 AM on the day after the current day and continues for 90 days. (The range does not include today.)
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.)
NEXT n DAYS	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 <i>n</i> 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.
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.

Relative Date Value	Range
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 <i>n</i> 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 <i>n</i> 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 nth 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.
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.

Relative Date Value	Range
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 <i>n</i> 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. Reports can't be filtered on custom long text area fields. Only the first 255 characters are shown for custom long text area fields in list views.

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.

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 <code>Lead Source equals Email</code> or <code>Web</code> and you change the picklist value Web to Referral, your report filter changes to <code>Lead Source equals Email</code>. 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".

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.

ii the user's currency is invalia, 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 5000000* 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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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.

IN THIS SECTION:

1. Subscribe to Get Refreshed Report Results in Lightning Experience

In Lightning Experience, subscribe to up to 5 reports and receive refreshed report results by email on a schedule that you set: daily, weekly, or monthly.

2. 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 Salesforce1 notifications, Chatter, or email.

3. 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 to Get Refreshed Report Results in Lightning Experience

In Lightning Experience, subscribe to up to 5 reports and receive refreshed report results by email on a schedule that you set: daily, weekly, or monthly.

For example, every Monday morning you arrive at your desk and run your case overview report to determine case priority. Instead of manually running the report, subscribe to it and have it emailed to you every Monday morning at 8:00am.

- 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, then click Subscribe on the report again.
- From the Edit Subscription modal, set the subscription schedule.To review the cases your team closes each day, receive a case report daily at 5:00 PM.
- 3. Click Save.
- **4.** Make sure that the subscription is active if you're ready to start receiving notifications.
- 5. Click Save.

Your report subscription starts, and you begin receiving report results by email according to the schedule you set.

EDITIONS

EDITIONS

Experience

Available in: **Group**, **Professional**, **Enterprise**,

Performance, Unlimited,

and **Developer** Editions

Available in: both Salesforce Classic and Lightning

Available in: Lightning Experience

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

USER PERMISSIONS

To subscribe to reports in Lightning Experience:

"Subscribe to Reports"

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 Salesforce1 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 Salesforce1 in-app notification
 - Post to Chatter
 - Send an email notification
 - 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.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To subscribe to reports in Salesforce Classic:

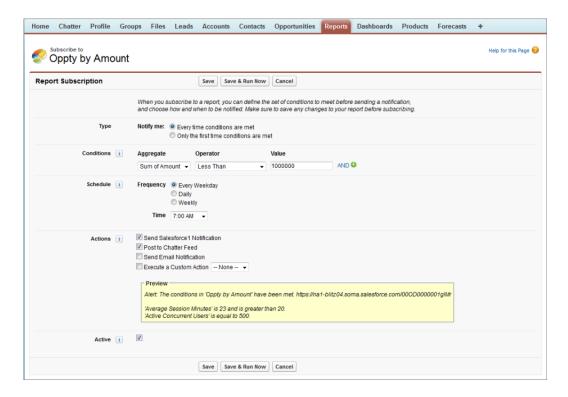
"Run Reports"

To enable or disable Salesforce Classic report notification subscriptions:

"Customize Application"

(3)

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 Salesforce1 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)

IN THIS SECTION:

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

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	То
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.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To schedule reports:

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.

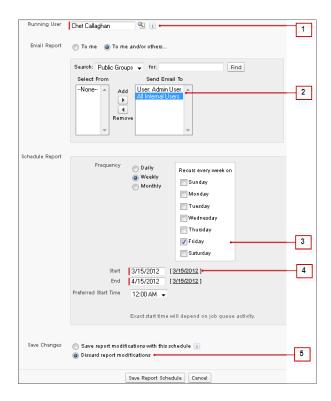
6. Click **Save Report Schedule**. You can choose:

Choose	То
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.



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

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To schedule reports:

"Schedule Reports"

To view all scheduled report for the organization:

 "View Setup and Configuration"

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

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To schedule reports:

"Schedule Reports"

To delete a scheduled report run:

"Modify All Data"

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To schedule reports:

"Schedule Reports"

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

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To schedule reports:

"Schedule Reports"

To delete a scheduled report run:

"Modify All Data"

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

- 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).

EDITIONS

Available in: Salesforce Classic

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

• 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 Manage a Report's Schedule

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

Export a Report

To work with report data in a dedicated tool, such as a spreadsheet, export report data as a Microsoft Excel * (.xls) or comma-separated values (.csv) file.

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 drop-down list.

Excel does not 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.

- From Reports, next to the report you want to export, click > Export.
 - If you set the "Do not 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 "Do not save
 encrypted pages to disk" under Internet options.
 - Reports in the joined format can't be exported, but you can choose a printable view.
- **2.** Set an Encoding appropriate for your language.
- 3. Set the Format to Excel Format .xls or Comma Delimited .csv.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To export reports:

"Export Reports"

4. Click Export.

5. If prompted by a browser dialog, choose to save the file. You can export up to 256 columns and 65,536 rows of data in one report.

Your exported report downloads.



🕜 Note: For security purposes, Salesforce can require users to pass a CAPTCHA user verification test to export data from their org. This simple text-entry test prevents malicious programs from accessing your org's data. To pass the test, users must correctly type the two words displayed in the overlay's text box. The words entered in the text box must be separated by a space.

SEE ALSO:

Run Reports in the Background Print a Report

Share a Report

Share your reports with colleagues so everyone's working with the same info.



Note: As you work with folders, keep these Lightning Experience limitations in mind.

Features Not Available in Lightning Experience

Feature	Notes about Lightning Experience Availability
Standard Folders, containing default reports and dashboards that come packaged with Salesforce	
Move Reports between Folders	Users can move dashboards between folders from the Lightning Experience dashboard editor, but must switch to Salesforce Classic to move reports.
Create and Delete Report and Dashboard Folders	Folders created in Salesforce Classic and their contents are available in Lightning Experience. Users can save inside these folders when creating or cloning a report or dashboard.
Share Report and Dashboard Folders	Sharing permissions set on folders in Salesforce Classic are obeyed in Lightning Experience, but sharing permissions can't be set nor changed.
Search for Report and Dashboard Folders	Users can't search for report and dashboard folders in Lightning Experience. Global search results include reports and dashboards, but not report and dashboard folders.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: All editions except Database.com

IN THIS SECTION:

1. Share Reports and Dashboards

Reports and dashboards are shared through folders. You share the folder, not the report or dashboard itself. To let others work with your report or dashboard, give them Viewer, Editor or Manager access to the folder where the report or dashboard is stored.

2. 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.

3. 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.

Share Reports and Dashboards

Reports and dashboards are shared through folders. You share the folder, not the report or dashboard itself. To let others work with your report or dashboard, give them Viewer, Editor or Manager access to the folder where the report or dashboard is stored.

Ø

Note:

- Users can't share report and dashboard folders in Lightning Experience. All sharing permissions set on folders in Salesforce Classic are obeyed in Lightning Experience.
- This set of sharing documentation is for orgs with Enhanced Analytics Folder Sharing enabled. Enhanced Analytics Folder Sharing grants report and dashboard folders detailed access levels. If your org was created after the Summer '13 Salesforce release, you already have Enhanced Analytics Folder Sharing. If your org doesn't have Enhanced Analytics Folder Sharing enabled, refer to *Turn On Enhanced Sharing for Reports and Dashboards* in the Salesforce help.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

IN THIS SECTION:

1. Share a Report or Dashboard Folder

To enable others to use a report or a dashboard, share the folder it is in. Give each user, public group, or role the degree of access to the folder that makes sense for them.

2. 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; and Managers can control access.

3. 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 report and dashboard folders.

4. User Permissions for Sharing Reports and Dashboards

Each level of access to a report or dashboard folder consists of a combination of specific user permissions. As an administrator, you can further fine-tune users' access to dashboards and reports by assigning or removing one or more permissions.

Share a Report or Dashboard Folder

To enable others to use a report or a dashboard, share the folder it is in. Give each user, public group, or role the degree of access to the folder that makes sense for them.



Note: To give someone 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 is 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.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

USER PERMISSIONS

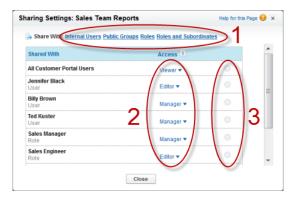
To share a report folder:

 "Manager" folder access or "Manage Reports in Public Folders"

To share a dashboard folder:

 "Manager" folder access or "Manage Dashboards in Public Folders"

(1) Share your folder by user, by user group, or by role. (2) Choose the access level you want each user, group, or role to have. (3) Stop sharing the folder with the user, group, or role.



IN THIS SECTION:

- 1. Share a Report or Dashboard with an Individual User

 If you only have one or two people to share a report or dashboard folder with, give them access individually.
- 2. Share a Report or Dashboard with a Group

If more than one or two people need access to an analytics folder, share the folder with a public group. That way you don't have to keep track of which individuals you've shared it with.

3. Share a Report or Dashboard by Role

When you give people access to a report or dashboard folder based on their role, you can control who has access without having to keep track of users or groups.

SEE ALSO:

Access Levels for Report and Dashboard Folders User Permissions for Sharing Reports and Dashboards

Share a Report or Dashboard with an Individual User

If you only have one or two people to share a report or dashboard folder with, give them access individually.

- 1. On the Reports tab, hover over a report folder in the left pane, click , and then select **Share**.
- 2. Select Internal Users.
 - Note: Internal users don't include customer portal or partner portal users.
- **3.** Find the user you want, click **Share**, and choose an access level. To search, just start entering a name.



4. Click **Done**, review your changes, and click **Close**.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

USER PERMISSIONS

To share a report folder with internal users:

 "Manager" folder access or "Manage Reports in Public Folders"

To share a dashboard folder with internal users:

"Manager" folder access or "Manage Dashboards in Public Folders"

Share a Report or Dashboard with a Group

If more than one or two people need access to an analytics folder, share the folder with a public group. That way you don't have to keep track of which individuals you've shared it with.

- 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

When you give people access to a report or dashboard folder based on their role, you can control who has access without having to keep track of users or groups.

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 to the folder. This means 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 will be able to 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.
 - Select **Roles** if you want to give access to all users who have that role.
 - Select Roles and Subordinates if you want to give access to those users plus everyone
 with a role below them in the role hierarchy.
- **3.** Find the role you want, click **Share**, and choose a level of access. To search, enter a name.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

USER PERMISSIONS

To share a report folder with public groups:

 "Manager" folder access or "Manage Reports in Public Folders"

To share a dashboard folder with public groups:

 "Manager" folder access or "Manage Dashboards in Public Folders"

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

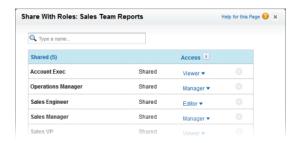
USER PERMISSIONS

To share a report folder with a role or role and subordinates:

 "Manager" folder access or "Manage Reports in Public Folders"

To share a dashboard folder with a role or role and subordinates:

 "Manager" folder access or "Manage Dashboards in Public Folders"



4. Click **Done**, review your changes, and click **Close**.

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; and 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 if you need them. When you create a folder, it is accessible only to you and users with administrative permissions until you share it.

IN THIS SECTION:

1. Viewer Access Levels to Report and Dashboard Folders

With Viewer access you can see the data in a report or dashboard, but you can't make any changes, except by cloning it into a new report or dashboard. All users have at least Viewer access to report and dashboard folders that have been shared with them. (Some users may have administrative user permissions that give them greater access.)

2. Editor Access to Report and Dashboard Folders

When you are an Editor on a folder, you can view and modify the reports and dashboards it contains, and move them to and from any other folders you have Editor or Manager access to.

3. Manager Access to Report and Dashboard Folders

With Manager access, you can do everything Viewers and Editors can do, plus control other users' access to it, change its properties, or delete it

SEE ALSO:

Share a Report or Dashboard Folder
Compare Access Levels for Report and Dashboard Folders

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

Viewer Access Levels to Report and Dashboard Folders

With Viewer access you can see the data in a report or dashboard, but you can't make any changes, except by cloning it into a new report or dashboard. All users have at least Viewer access to report and dashboard folders that have been shared with them. (Some users may have administrative user permissions that give them greater access.)

For example, 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 be able 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: both Salesforce Classic and Lightning Experience

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

Editor Access to Report and Dashboard Folders

When you are an Editor on a folder, you can view and modify the reports and dashboards it contains, and move them to and from any other folders you have Editor or Manager access to.

For example, Allison, a sales manager, wants to provide a different sales dashboard for each of three regional teams. She'll need Editor access to the folder that contains the Master Sales dashboard so she can move the underlying reports into the right folders and then modify them to show the appropriate data. She'll also need the Create and Customize Dashboards user permission.



Note: You can't give Editor access to standard report folders. All users get Viewer access to these folders by default.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

Manager Access to Report and Dashboard Folders

With Manager access, you can do everything Viewers and Editors can do, plus control other users' access to it, change its properties, or delete it.

For example, 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 its creator, he automatically 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. All users get Viewer access to these folders by default.

EDITIONS

Available in: Salesforce Classic

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

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 report and dashboard folders.

	Viewer	Editor	Manager
View reports or dashboards in the folder	~	~	~
Run reports in the folder	~	~	~
Refresh dashboards in the folder	~	~	~

EDITIONS

Available in: Salesforce Classic

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

	Viewer	Editor	Manager
Change, move, or delete your reports or dashboards in the folder	✓	~	~
See who has what level of access to the folder	~	~	~
Change, move, or delete reports or dashboards that were created by someone else in the folder		~	~
Note: If Enhanced Folder Sharing is turned on for your organization, then you also need the "View All Data" permission to change these dashboards.			
Change the folder's name			~
Change the folder's sharing settings			~

User Permissions for Sharing Reports and Dashboards

Each level of access to a report or dashboard folder consists of a combination of specific 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 analytics 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 My Personal Dashboards folder. Create dashboards and save into any shared folder if sharing rights allow.	
Create and Customize Reports	Create, edit, and delete reports in My Personal Custom Reports folder. Create reports and save into 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 you created in shared folders.	
Edit My Reports	Edit, move, save, and delete reports you created in shared folders.	
Manage Dashboards in Public Folders	Create, edit, delete dashboards ¹ , and manage their sharing in all public dashboard folders, which does not include others' personal folders. This permission allows users to edit and share dashboards in all folders, including hidden folders. They also get these permissions:	

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

	Create and Customize Dashboards
	• Create Dashboard Folders
	Edit My Dashboards
	• View Dashboards in Public Folders
	(¹ To edit a dynamic dashboard, users also need Manage Dynamic Dashboards and View My Team's Dashboards.)
Manage Reports in Public Folders	Create, edit, delete reports, and manage their sharing in all public report folders, which does not include 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, which does not include others' personal folders.
View Reports in Public Folders	View reports in public report folders, which does not include others' personal folders.

SEE ALSO:

Share a Report or Dashboard Folder
Compare Access Levels for Report and Dashboard Folders

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.

IN THIS SECTION:

1. 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.

SEE ALSO:

Export a Report
Schedule a Report for Refresh

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.

IN THIS SECTION:

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:

The Report Run Page

Create a Custom Report in Accessibility Mode

Create a Report

Customizing Reports

Report Run Page Options

Export a Report

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

USER PERMISSIONS

To export reports to the background:

 "Background Report Export"

To view reports exported to the background by other users:

 "Background Report Export"

AND

"View All Data"

To manage reports exported to the background by other users:

 "Background Report Export"

AND

"Modify All Data"

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.

- 1. Create or run a report.
- 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."

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

USER PERMISSIONS

To export reports to the background:

 "Background Report Export"

To view reports exported to the background by other users:

"Background Report Export"

AND

"View All Data"

To manage reports exported to the background by other users:

 "Background Report Export"

AND

"Modify All Data"

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.

- 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.
 - Note: For security purposes, Salesforce can require users to pass a CAPTCHA user verification test to export data from their org. This simple text-entry test prevents malicious programs from accessing your org's data. To pass the test, users must correctly type the two words displayed in the overlay's text box. The words entered in the text box must be separated by a space.
- 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.

(1) 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.

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.



IN THIS SECTION:

1. 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.

2. 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.

3. 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.

4. Limits on Report Charts in Pages

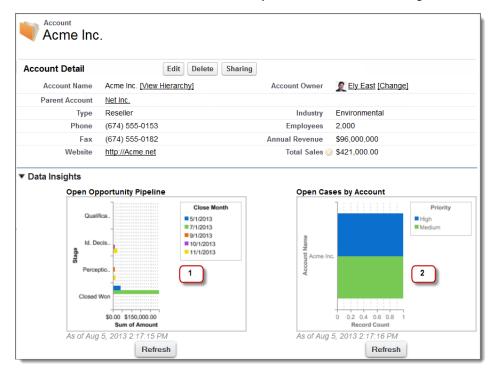
Consider these limits when embedding charts in detail pages.

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



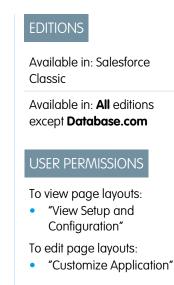
- 1. The pipeline chart shows opportunities for the account in various stages including two that were won. The account executive can hover on the chart to get the value of each opportunity (Values are highlighted if the source report chart has hover enabled). She is able to click the chart, go to the 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.

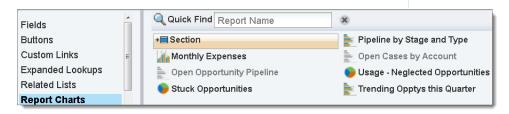
Add a Report Chart to a Page Layout

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

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.

IN THIS SECTION:

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.



USER PERMISSIONS

To view page layouts:

"View Setup and Configuration"

To edit page layouts:

"Customize Application"

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.

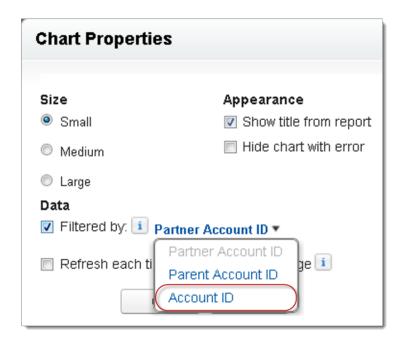
Make a Chart Filterable

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.

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.
- 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

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.

IN THIS SECTION:

1. Find a Report

Search 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.

2. Get the Information You Need From the Reports Tab List View

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.

3. Print a Report

Print a report from the run page of a report using your browser's print function.

4. 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.

5. 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.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

6. Move a Report or Dashboard Between Folders

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.

7. Delete a Report

You can delete a selected report from the Reports tab or from its run page.

Find a Report

Search 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.

Search looks up all folders. Filters are set to All Items and All Types to display all of 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.

2. Refine results using these options:

To refine results by Do this

Type

Select a filter:

- Reports
- Dashboards
- All Types

Type filters don't apply to report templates in standard report folders.

View

Select a filter:

- All Items
- Recently Viewed
- Items I Created
- Items I'm Following

View filters.

A selected folder

Select a folder, then type in the search box.

Filters are set to All Items and All Types to display all of the folder's contents.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To search for reports:

"Run Reports"

To search for dashboards:

 "Run Reports" AND access to dashboard folder

To refine results by	Do this
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.

SEE ALSO:

Dashboards Help You Visualize Complex Information Get Started with Dashboards

Get the Information You Need From the Reports Tab List View

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.

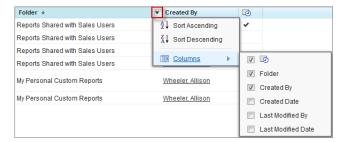
Customize your list view as follows:

Option	Description
To resize a column	Click and drag its right margin to the preferred size.
To hide a column	Click 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.

Available in: Salesforce Classic Available in: All editions except Database.com USER PERMISSIONS

To view the Reports tab:
• "Run Reports"

Select Column Header for Sorting and Hiding Columns



SEE ALSO:

Reports Give You Access to Your Data

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.

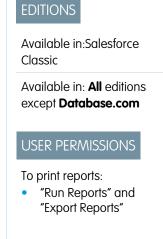
SEE ALSO:

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 .



EDITIONS

Available in: Salesforce Classic

Available in all editions

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.

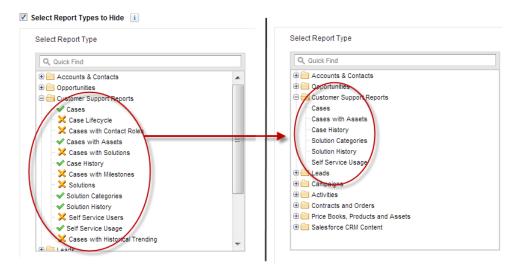
The green check mark next to a report means it's visible to everyone.

3. To hide the report type, click the check mark to change it to an X.



Types"

Create New Report



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.

Move a Report or Dashboard Between Folders

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 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 into 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 a 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.

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.

SEE ALSO:

The Report Run Page

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To delete reports in My Personal Custom Reports folder:

 "Create and Customize Reports"

To delete reports in public folders:

"Manage Public Reports"

Report Limits

As you report on your data, be aware of these limits and restrictions.

Salesforce Reports and Dashboards Limits

Feature	Personal Edition	Contact Manager		Polesiand Edition	Enterprise Edition	Unlimited and Reference 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 dasł	hboard		
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 ^{2, 3, 4}	1 ²	2 ²	1 ^{2, 3, 4}
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 ^{2, 3, 5}	1 ²	2 ²	1 ^{2, 3, 5}

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

Salesforce retains historical data for the previous three months, plus the current month.

The following Salesforce Reports and Dashboards limits apply to all supported editions.

Report Limits

• The report builder preview shows a maximum of 20 rows for summary and matrix reports, and 50 rows for tabular.

EDITIONS

¹ These limits apply to the report builder. If you're using the report wizard, the limit 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.

• 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.

- Reports display a maximum of 2,000 rows. To view all the rows, export the report to Excel or use the printable view for tabular and summary reports. For joined reports, export is not available, and the printable view displays a maximum of 20,000 rows.
 - Summary and matrix reports 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.
- 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 Salesforce1, they're converted to tabular reports.
- 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.
- 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.
- 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 254 characters in a rich text area or a long text area are displayed in a report.
- 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 is not supported for joined reports.

Dashboard Limits

- A dashboard filter can have up to 50 options. By default, 10 options are enabled. Contact Salesforce to extend your limit.
- Each dashboard can have up to 20 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.
- A dashboard table or chart can display up to 20 photos.
- You must wait at least one minute between dashboard refreshes.

Report Type Limits

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 additional 56 objects. However, users will 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

- 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

Only the first 255 characters in a custom text field count for filtering purposes.

Embedded Report Charts Limits

- 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.
- 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

• Only the first 255 characters are shown for custom long text area fields in list views.

Bucket and Bucket Field Limits

- 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

- 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, this includes the standard preselected fields, which can't be disabled.
- Formula fields are not supported.
- Row limit filters are not supported.
- The summary report format is not 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.
- Dated exchange rates are not supported. When you run the historical trend report, it only uses the latest dated exchange rate.
- Internet Explorer 6 is not supported.

- You can't subscribe to historical trend reports.
- The Report Wizard is not 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

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 2,000 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.

Reports and Dashboards API Limits

The following limits 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 trend reports are only supported for matrix 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.

Troubleshoot Reports

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

IN THIS SECTION:

- 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

4. Why can't I see formula options in report builder?

If you can't see the Add Formula option in the Fields pane of report builder, change your report format to summary, matrix, or joined. Formulas don't show up for tabular reports.

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 are some common report limits. To see a full list of analytics limits, see the Salesforce Limits Quick Reference Guide.

7. Why am I getting an "obsolete report" error message?

You may be trying to report on data that isn't available.

8. Improve Report Performance

Many factors can cause a report to perform poorly or to time out. You can address most of them with simple changes, such as using the correct filter operators, applying more filters, and reducing the amount of data.

9. 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 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 2,000 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 f Add Formula option in the Fields pane of report builder, change your report format to summary, matrix, or joined. Formulas don't show up for tabular reports.

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. 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 2,000 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 are some common report limits. To see a full list of analytics limits, see the Salesforce Limits Quick Reference Guide.

Feature	Limit	Editions Info
Bucket fields per report	Up to 5 (each bucket field can have 20 buckets)	$\label{thm:performance} A vailable in \text{Enterprise}, \text{Unlimited}, \text{Performance}, \text{and} \text{Developer} \text{Editions}.$
Matrix reports	Up to 400,000 summarized values. Data exceeding this limit is not displayed.	Available in all editions.
Scheduled reports per organization	One or two per hour, based on edition	Available in Professional, Enterprise, Unlimited, Performance, and Developer Editions.
Field filters per report	Up to 20	Available in all editions.
Formulas per report	Up to 5	Available in all editions.
Rows displayed in a report	Up to 2,000. To view all the rows, export the report to Excel or use the printable view for tabular and summary reports. For joined reports, export is not available, and the printable view displays a maximum of 20,000 rows.	Available in all editions.
Rows displayed in the report builder preview	Up to 20 rows for summary and matrix reports, and up to 50 rows for tabular	Available for all editions.

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.

Improve Report Performance

Many factors can cause a report to perform poorly or to time out. You can address most of them with simple changes, such as using the correct filter operators, applying more filters, and reducing the amount of data.

Watch this video for quick tips on Making Your Reports Run Faster (Salesforce Classic).

A report can take anywhere from a few seconds to several minutes to run. Occasionally, a report is so complex that it exceeds the timeout limit and fails. If the report is slow to load, it's usually because it is:

- Querying too many objects
- Dealing with intricate lookups
- Has too many fields

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

For example, say that you run a sales team and want to see how many Leads each of your salespeople has. You build a report that returns all the leads 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 Leads assigned to your team. If you need your report to run even faster, consider filtering by time. Do you need to see Leads from last year, or the year before? If not, add a second filter which scopes your report to this year only. Now your report runs as fast as lightning!

If you can't view a report and want to edit it to avoid the time-out, append ?edit=1 to the report URL. Doing so gets you into edit mode, where you can adjust the criteria. Try these tips to get your reports to run more efficiently.

- When filtering, use the equals or not equal to operators instead of contains or does not contain. For example, use Account Owner equals John James, not Account Owner contains John. Choose AND rather than OR for filter logic
- To narrow your report's date range, use time frame filters. For example, use Last 30 Days instead of Current FY.
- Set time frame filters by choosing a Date Field and Range to view. Only records for that time frame are shown.
- Reduce the number of fields in the report by removing unnecessary columns or fields.
- If you receive an error message saying that your activity report has too many results, 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".
- Add time filters, scope filters, and filter criteria to the report to further narrow the results.

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.

Get the Right Data into Your Report

A well-designed report helps you turn raw data into information people can act on. Summarize values and group them into meaningful categories, filter out irrelevant facts, and gather data about multiple objects at once.

Communicate Your Report Data Effectively

A report produces information in the form of a table. You can set up that table to help people scan quickly for the information they need. You can also add a graph to help users understand the data visually, and you can display the report's data on a dashboard.

SEE ALSO:

Show Report Data Graphically Show Report Data in Tables

The Report Run Page

The Report Run page lets you control what happens to your report when you run it. From the report run page, 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.

SEE ALSO:

Report Run Page Options Limit Report Results

Report Run Page Options

Customize the report run page to support the specific information you need for the way you work. The following fields and buttons are available from the report run page, depending on your setup.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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"

Option	Description		
Customize	Opens the report builder to help you customize the report.		
Date Field	Choose date-specific information about the records: created date, last updated date, and date of last activity.		
Delete	Deletes the report. Deleted reports are moved to the Recycle Bin.		
Display Currencies Using	Determines the currency type for report subtotals and amounts in the "converted" column of any field. Available only for organizations using multiple currencies.		
	Amounts in reports are shown in their original currencies, and report totals are displayed in your personal currency. To change the currency go to the report builder and click Show > Currencies Using , then select an active currency. For any amount, you can also choose to display the "converted" column (for example, "Annual Revenue (converted)"), which will show amounts in the selected currency.		
Division	Determines the division of records to include in the report. SelectCurrent to show records in your current working division. Reports that are already scoped (such as My Cases or My Opportunity Team's Accounts) include records in all divisions. Available only if your organization uses divisions to segment data and you have the "Affected by Divisions" permission. If you do not have the "Affected by Divisions" permission, your reports include records in all divisions.		
Drill Down	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.		
Export Details	Exports data from the report to a Microsoft $^{\circ}$ Excel spreadsheet or CSV (comma-separated values) file.		
Hierarchy	Provides options to view the report data based on role or territory hierarchy. View data for an individual user, a role, or an entire organization. Available only for Professional, Enterprise, Unlimited, Performance, and Developer Editions.		
Printable View	Lets you display the report in Microsoft Excel format, for easy printing.		
Relationship to Opportunity	Specifies a scope based on opportunity ownership or opportunity team; limits the data in an opportunity report.		
Range	Specifies a time period for a report. You can limit the data by specific dates using any standard or custom date field defined for the type of record in the report. Works with $From$ and To fields.		
Run Report	Regenerates the report data based on current report settings.		
Save	Saves changes to one of your existing custom reports.		
Save As	Saves parameter changes to any custom, standard, or public report under a new name, without altering the original report.		
Show	Specifies which items to include in the report: just the records you own, or all records to which you have access. Use this option to see team data in your report.		

Option	Description		
Show/Hide Details	Toggles the display to show all information or just summary information.		
Subscribe	Lets you subscribe to receive report notifications, whenever certain metrics meet conditions you specify.		
Summarize information by	Provides subtotals for the fields you specify.		
Units	Choose to view the age of a case in days, hours, or minutes. 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.		

SEE ALSO:

The Report Run Page Creating a Custom Report

Get Started with Reports

Set up your reporting environment, use the report builder to create a basic report, and organize your reports to make it easy to find information when you need it. In most cases, you can start with a standard report and customize it to your needs.

You can report on any data you have read or read/write access to. The available data varies depending on your sharing model, but you can almost always report on data owned by or shared with users below you in the role hierarchy, regardless of your sharing model. (Certain sharing models do not allow this.) Salesforce offers two ways to report on information owned by you or users who report to you:

- Some reports have a View filter. Choose the "My team's" option, such as "My Team's Opportunities," to view data owned by you and users below you in the role hierarchy.
- Some reports, such as opportunity and activity reports, have **Hierarchy** links that allow you to drill down to different data sets based on the role or user hierarchy. If your organization uses territory management, in some reports these links are based on the territory hierarchy.

You can choose to sort the report by record owner or role when customizing the report.

SEE ALSO:

Improve Report Performance
Build a New Report
Reports Give You Access to Your Data
Standard Report Types

Reports Give You Access to Your Data

Use reports to leverage the data your organization has accumulated over time. You can examine your organization's data in almost infinite combinations, display it in easy-to-understand formats, and share the resulting insights with others via dashboards.

A *report* returns a set of records that meets certain criteria, and displays it in organized rows and columns. Report data can be filtered, grouped, and displayed graphically as a chart. Reports are stored in folders, which control who has access.

SEE ALSO:

Customizing Reports
Standard Report Types

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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.

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

IN THIS SECTION:

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 Your 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 — just 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. 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. Run and Then Read a Dashboard

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!

5. Share a Dashboard

A dashboards is the best way to share a curated, interactive set of data with colleagues.

6. Organize Dashboards

Keep your dashboards 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.

7. Dashboard Limits

As you build dashboards of your data, be aware of these limits and restrictions.

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. Dashboards Help You Visualize Complex Information

Dashboards help you make decisions based on real-time data you've gathered with reports. Use dashboards to help users identify trends, sort out quantities, and measure the impact of their activities.

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.

IN THIS SECTION:

1. Build a Lightning Experience Dashboard

Build a Lightning Experience dashboard to provide a graphical view of the data in your reports.

2. Build a Salesforce Classic Dashboard

Build a dashboard to provide a graphical view of the data in your reports.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

3. Working with 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.

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.

Build a Lightning Experience Dashboard

Build a Lightning Experience dashboard to provide a graphical view of the data in your reports.

This topic is about creating dashboards for reports in Lightning Experience. For information on creating dashboards in Salesforce Classic or Wave Analytics, review these articles:

- Build a Salesforce Classic Dashboard
- Build Wave 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. If necessary, create the custom reports containing the data you want to display.
 - (1) Important: Be sure to store these reports in folders that your intended dashboard viewers can access
- 2. From the Dashboards tab, click **New Dashboard**.
- **3.** Name the dashboard. Optionally, provide a short description.
- **4.** To add components to the dashboard, click **Add**.
- **5.** Click **Save**, then click **Done**.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To create dashboards:

"Run Reports" AND
 "Manage Dashboards"

To edit and delete dashboards you created:

"Run Reports" AND
 "Manage Dashboards"

To edit and delete dashboards you didn't create:

 "Run Reports," "Manage Dashboards," AND "View All Data"

To create, edit, and delete dynamic dashboards:

Build a Salesforce Classic Dashboard

Build a dashboard to provide a graphical view of the data in your reports.

This topic is about creating dashboards for reports in Salesforce Classic. For information on creating dashboards in Salesforce Classic or Wave Analytics, review these articles:

- Build a Lightning Experience Dashboard
- Build Wave 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**.

SEE ALSO:

Delete a Dashboard
Install the CRM Sample Dashboards from AppExchange

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create dashboards:

"Run Reports" AND
 "Manage Dashboards"

To edit and delete dashboards you created:

"Run Reports" AND "Manage Dashboards"

To edit and delete dashboards you didn't create:

 "Run Reports," "Manage Dashboards," AND "View All Data"

To create, edit, and delete dynamic dashboards:

Working with 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.

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**.

IN THIS SECTION:

Add a Dashboard Component

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

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.

Edit Dashboards with Keyboard Shortcuts in Lightning Experience

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 Dashboards in Accessibility Mode in Salesforce Classic

In Salesforce Classic, you can edit dashboards in Accessibility mode.

SEE ALSO:

Add a Dashboard Filter

Build a Salesforce Classic Dashboard

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create dashboards:

"Run Reports" AND
 "Manage Dashboards"

To edit and delete dashboards you created:

"Run Reports" AND "Manage Dashboards"

To edit and delete dashboards you didn't create:

 "Run Reports," "Manage Dashboards," AND "View All Data"

To create, edit, and delete dynamic dashboards:

Add a Dashboard Component

Add components by dragging a component type onto the dashboard, then dropping a data source (report, s-control, or Visualforce page) onto it.

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.

1 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.
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.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create dashboards:

"Run Reports" AND
 "Manage Dashboards"

To edit and delete dashboards you created:

"Run Reports" AND
"Manage Dashboards"

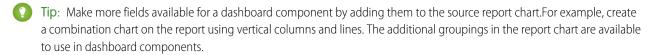
To edit and delete dashboards you didn't create:

"Run Reports," "Manage Dashboards," AND "View All Data"

To create, edit, and delete dynamic dashboards:

- **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 **a** 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.



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 x for a data source to remove it from the component.
 - Click x 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

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: "Manage Dynamic Dashboards" is available after Enhanced Analytics Folder Sharing is turned on. Enhanced Analytics Folder Sharing grants report and dashboard folders detailed access levels. If your org was created after the Summer '13 Salesforce release, you already have Enhanced Analytics Folder Sharing. If your org doesn't have Enhanced Analytics Folder Sharing enabled, refer to *Turn On Enhanced Sharing for Reports and Dashboards* in the Salesforce help.

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.

IN THIS SECTION:

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.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To edit and delete dashboards you created:

"Run Reports" AND
"Manage Dashboards"

To edit and delete dashboards you didn't create:

 "Run Reports" AND "Manage Dashboards"

To create, edit, and delete dynamic dashboards:

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.
- **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

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	

Component Type	Image	Description
		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.
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.
		(1) 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

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.

IN THIS SECTION:

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

Create a Custom Report in Accessibility Mode

Show Different Data Sets 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

IN THIS SECTION:

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

IN THIS SECTION:

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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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.

IN THIS SECTION:

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. 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

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.

IN THIS SECTION:

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
- 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

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

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 Ву

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.

Chart

Combination Select this option to plot additional values on this chart. The chart type you chose must allow combination charts.

EDITIONS

Available in: Salesforce Classic

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

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.

SEE ALSO:

Formatting Settings for Dashboard Bar Chart Components

Data Settings for Dashboard Funnel Chart Components

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.
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 <code>Auto</code> . 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.

EDITIONS

Available in: Salesforce Classic

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

SEE ALSO:

Formatting Settings for Funnel Dashboard Components

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**.

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 <code>Auto</code> . 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 <code>Displayed?</code> option set to a grouping level other than <code>All summary levels</code> , 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.

EDITIONS

Available in: Salesforce Classic

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

SEE ALSO:

Scatter Charts

Modify a Dashboard Component

Formatting Settings for Dashboard Scatter Chart Components

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.

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 <code>Auto</code> . 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.

EDITIONS

Available in: Salesforce Classic

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

SEE ALSO:

Formatting Settings for Dashboard Gauge Components

Data Settings for Dashboard Line Chart Components

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.
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

EDITIONS

Available in: Salesforce Classic

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

Setting	Description	
	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

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

EDITIONS

Available in: Salesforce Classic

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

Setting	Description
	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

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.
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 <i>Auto</i> . 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.

EDITIONS

Available in: Salesforce Classic

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

SEE ALSO:

Formatting Settings for Pie and Donut Dashboard Components

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.

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.

EDITIONS

Available in: Salesforce Classic

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

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.

IN THIS SECTION:

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.
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.

EDITIONS

Available in: Salesforce Classic

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

Formatting Settings for Funnel Dashboard Components

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 <i>5</i> 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.

EDITIONS

Available in: Salesforce Classic

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

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**.

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.	

EDITIONS

Available in: Salesforce Classic

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

Setting	Description
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

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	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	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.

EDITIONS

Available in: Salesforce Classic

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

SEE ALSO:

Modify a Dashboard Component

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.

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.	
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.	

EDITIONS

Available in: Salesforce Classic

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

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.

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.

EDITIONS

Available in: Salesforce Classic

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

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

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.

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 <i>5</i> 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 %	Display the percentage value for each wedge of pie and donut charts.	

EDITIONS

Available in: Salesforce Classic

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

Setting	Description	
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

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.

EDITIONS

Available in: Salesforce Classic

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

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 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.
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.

Setting	Description
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

Edit Dashboards with Keyboard Shortcuts in Lightning Experience

USER PERMISSIONS		EDITIONS
To view and refresh dashboards:	"Run Reports" AND access to dashboard folder	Available in: Lightning Experience
To create dashboards:	"Run Reports" AND "Manage Dashboards"	Available in: Professional ,
To edit and delete dashboards you created:	"Run Reports" AND "Manage Dashboards"	Enterprise, Performance, Unlimited, and Developer
To edit and delete dashboards you didn't create:	"Run Reports," "Manage Dashboards," AND "View All Data"	Editions
To create, edit, and delete dynamic dashboards:	"Run Reports" AND "Manage Dynamic Dashboards"	

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	Action
Tab	Focus on next item in the dashboard

Keyboard Shortcut	Action
Shift + Tab	Focus on previous item in the dashboard
Spacebar	Select a component to move or a component corner to resize
***	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.

Edit Dashboards in Accessibility Mode in Salesforce Classic

USER PERMISSIONS		EDITIONS
To view and refresh dashboards:	"Run Reports" AND access to dashboard folder	Available in: both Salesforce Classic
To create dashboards:	"Run Reports" AND "Manage Dashboards"	Available in: Professional ,
To edit and delete dashboards you created:	"Run Reports" AND "Manage Dashboards"	Enterprise, Performance, Unlimited, and Developer
To edit and delete dashboards you didn't create:	"Run Reports," "Manage Dashboards," AND "View All Data"	Editions
To create, edit, and delete dynamic dashboards:	"Run Reports" AND "Manage Dynamic Dashboards"	

In Salesforce Classic, you can edit dashboards in Accessibility mode.

(1) 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 (1), (2), and (1) 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.

IN THIS SECTION:

- 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

(1) 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.

To set dashboard properties:

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.
 - (1) 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

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

USER PERMISSIONS

To create dashboards:

"Run Reports" AND "Manage Dashboards"

To create, edit, and delete dynamic dashboards:

"Run Reports" AND
"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

(1) 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.



- 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

Dashboard Component Properties in Accessibility Mode

(1) 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.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create dashboards:

"Run Reports" AND
 "Manage Dashboards"

To edit and delete dashboards you created:

"Run Reports" AND "Manage Dashboards"

To edit and delete dashboards you didn't create:

 "Run Reports," "Manage Dashboards," AND "View All Data"

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.
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 Dashboard Component Types

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.

Filter Your 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 — just 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.

A filter has a name, a filter operator and one or more values. Each filter can have up to 10 options. Each dashboard can have up to 3 filters. Contact Salesforce to increase the filter options limit. A maximum of 50 filter options is possible.

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.



- Filters can't be added 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 formula fields in a dashboard filter.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To install packages:

 "Download AppExchange Packages"

To create dashboards:

 "Run Reports" AND "Manage Dashboards"

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To add filters to a dashboard:

"Run Reports" AND "Manage Dashboards"

To view, refresh, and filter dashboards:

 "Run Reports" AND access to dashboard folder

• In Lightning Experience, you can't edit filtered dashboards, which means you can't add new dashboard filters. (You can change existing filters while viewing the dashboard, though!) Switch to Salesforce Classic to edit filtered dashboards.

IN THIS SECTION:

1. Add a Dashboard Filter

To create a dashboard filter, select a field that contains the type of information you want to filter on, then define how the filter returns the data.

2. Apply a Dashboard Filter

Filter a dashboard to analyze the information interactively. When you filter a Salesforce Classic dashboard, the filtered view is preserved so that the next time you see the dashboard, data is filtered by the same view. In Lightning Experience, dashboards always open with no filters applied.

3. Dashboard Filter Examples

Filters on a dashboard allow you to choose different views of data. When you filter a Salesforce Classic dashboard, the filtered view is preserved so that the next time you see the dashboard, data is filtered by the same view. In Lightning Experience, dashboards always open with no filters applied. Without dashboard filters, you'd have to create multiple dashboards, each with its own set of filtered reports.

Add a Dashboard Filter

To create a dashboard filter, select a field that contains the type of information you want to filter on, then define how the filter returns the data.

The field you select for the filter may 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 may 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 Filtered By. If equivalent fields are available for a component, you can select a different field to filter the component.

- 1. Click Edit on your dashboard, then click Add Filter.
 - In Lightning Experience, you can't edit filtered dashboards, which means you can't add new dashboard filters. (You can change existing filters while viewing the dashboard, though!) Switch to Salesforce Classic to edit filtered dashboards.
- **2.** Select the field to filter on from the Field drop-down. The drop-down shows fields that can be used to filter all components. (To see equivalent fields for your selection, click i).)
- **3.** Give your filter a display label to identify it. If your filter has many equivalent fields, we recommend using a label that works for all components.
- **4.** Under Filter Options, select an operator and provide one or more values to filter by.

 Use the **Group Name** 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.

EDITIONS

Available in: Salesforce Classic

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

EDITIONS

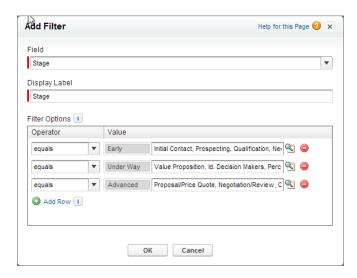
Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To add filters to a dashboard:

"Run Reports" AND
 "Manage Dashboards"



- Note: Options with multiple values are treated as OR conditions.
- **5.** Click **OK**. When you click OK, you can see how the filter is applied on each component. If equivalent fields are available for a component, you can select which one to use to filter the component.

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. When you filter a Salesforce Classic dashboard, the filtered view is preserved so that the next time you see the dashboard, data is filtered by the same view. In Lightning Experience, dashboards always open with no filters applied.

- 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.
- Open a dashboard.
 In Salesforce Classic, if you applied any filters last time you used this dashboard, notice that they are still selected. In Lightning Experience, the dashboard always opens with no filters applied.
- **2.** Select an option from the filter drop-down. Each filter has one or more options you can choose to narrow your selection further.

EDITIONS

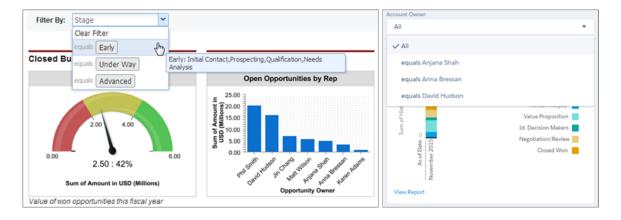
Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To view, refresh, and filter dashboards:

 "Run Reports" AND access to dashboard folder



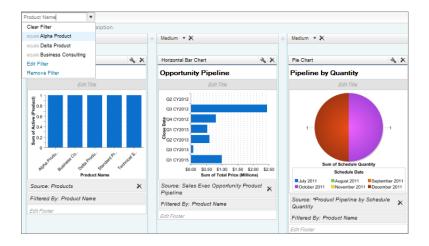
3. To see data unfiltered, select Clear Filter or All from the filter drop-down.

Dashboard Filter Examples

Filters on a dashboard allow you to choose different views of data. When you filter a Salesforce Classic dashboard, the filtered view is preserved so that the next time you see the dashboard, data is filtered by the same view. In Lightning Experience, dashboards always open with no filters applied. 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.



EDITIONS

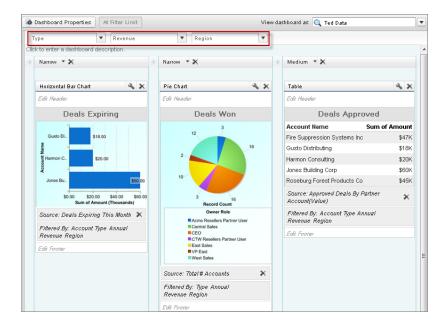
Available in: Salesforce Classic

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

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.



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.)

IN THIS SECTION:

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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.

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.
 - A specified user 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 logged in user** Dashboard readers see data as themselves, according to their own access to data.
- **4.** If necessary, specify who dashboard readers view the dashboard as.
- 5. Optionally, select Allow dashboard viewers to change this (dynamic dashboard) 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.
- **6.** 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.

EDITIONS

Available in: Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete dashboards:

"Run Reports"

AND

"Manage Dashboards"

To specify who people view a Lightning Experience dashboard as:

"View My Team's Dashboards"

OR

"View All Data"

To enable dashboard readers to choose who they view the dashboard as:

"View My Team's Dashboards"

OR

"View All Data"

Choose a Running User in Salesforce Classic

Select a running user to specify which data to display in a dashboard.

Each dashboard has a running user, whose security settings determine which data to display in a dashboard.



Note: The "Manage Dynamic Dashboards" permission, required for selecting a running user, isn't available in **Professional** Edition.

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.

Users with a Salesforce Platform or Salesforce Platform One user license can only view a dashboard if the dashboard running user also has the same license type. Users with user licenses that don't have full dashboard access can only see dashboards with running users of the same license type. Consider creating separate dashboards for users with different license types.

To select a dashboard's running user in Salesforce Classic:

- 1. Edit a dashboard.
- 2. Click 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.
 - Users with "Enable Other User's Dashboard" can edit the dashboard if they have access to it, even if they aren't the running user and don't have "View All Data."
 - 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.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete dashboards:

"Run Reports" AND "Manage Dashboards"

To create, edit, and delete dynamic dashboards:

"Run Reports" AND "Manage Dynamic Dashboards"

To enable choosing a different running user for the dashboard:

"View My Team's Dashboards" OR "View All Data"

- 6. In the View dashboard as field, enter a running user.
- 7. Save your dashboard.

IN THIS SECTION:

Provide Individualized Views of a Dashboard in Salesforce Classic

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

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.

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:

EDITIONS

Available in: Salesforce Classic

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

- 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 Your 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.

Your organization can have up to 5 dynamic dashboards for Enterprise Edition, 10 for Unlimited and Performance Edition, and 3 for Developer Edition.

Note:

- You can't save dynamic dashboards to personal 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.
 - Users with "Enable Other User's Dashboard" can edit the dashboard if they have access to it, even if they aren't the running user
 and don't have "View All Data."

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

Choose a Running User in Salesforce Classic

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete dynamic dashboards:

"Run Reports" AND
"Manage Dynamic
Dashboards"

To enable choosing a different running user for the dashboard:

"View My Team's Dashboards" OR "View All Data"

Run and Then Read a Dashboard

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!

IN THIS SECTION:

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. Schedule a Dashboard Refresh

In Enterprise Edition, Unlimited Edition, and Performance Edition, you can schedule dashboards to refresh daily, weekly, or monthly.

3. 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.

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To refresh dashboard data:

 "Run Reports" AND access to dashboard folder

Schedule a Dashboard Refresh

In Enterprise Edition, Unlimited Edition, and Performance Edition, you can schedule dashboards to refresh daily, weekly, or monthly.



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 <code>Scheduled Jobs</code> in the <code>Quick Find</code> 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 ru

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.



Note: Dashboard emails that contain images compatible with Lotus Notes are substantially larger and the image quality can be lower.

4. Schedule the refresh.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To schedule and email a dashboard refresh:

"Schedule Dashboards"

To delete the schedule to refresh a dashboard:

"Modify All Data"

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

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.

EDITIONS

Available in: Lightning Experience

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

USER PERMISSIONS

To expand dashboard components:

"Run Reports"

Share a Dashboard

A dashboards is the best way to share a curated, interactive set of data with colleagues.



Note: As you work with folders, keep these Lightning Experience limitations in mind.

Features Not Available in Lightning Experience

Feature	Notes about Lightning Experience Availability
Standard Folders, containing default reports and dashboards that come packaged with Salesforce	
Move Reports between Folders	Users can move dashboards between folders from the Lightning Experience dashboard editor, but must switch to Salesforce Classic to move reports.
Create and Delete Report and Dashboard Folders	Folders created in Salesforce Classic and their contents are available in Lightning Experience. Users can save inside these folders when creating or cloning a report or dashboard.
Share Report and Dashboard Folders	Sharing permissions set on folders in Salesforce Classic are obeyed in Lightning Experience, but sharing permissions can't be set nor changed.
Search for Report and Dashboard Folders	Users can't search for report and dashboard folders in Lightning Experience. Global search results include reports and

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

Feature Notes about Lightning Experience Availability

dashboards, but not report and dashboard folders.

IN THIS SECTION:

1. Manage Access to Reports and Dashboards

Access to reports and dashboards is controlled through settings for the folder they are stored in. To give others access to your reports or dashboards, place them in a shared folder. To prevent others from accessing your reports, place them in personal folders that only you can access.

2. Move Dashboards Between Folders in Lightning Experience

If your org is using Enhanced Analytics Folder Sharing for Reports and Dashboards, then folders govern who has access to the reports and dashboards saved inside. (If your org was created after the Summer '13 Salesforce release, you already have Enhanced Analytics Folder Sharing.) Move dashboards between folders to both organize and control access to dashboards.

3. Print a Dashboard

Print dashboards using your browser's print option.

Manage Access to Reports and Dashboards

Access to reports and dashboards is controlled through settings for the folder they are stored in. To give others access to your reports or dashboards, place them in a shared folder. To prevent others from accessing your reports, place them in personal folders that only you can access.

To access report or dashboard folders go to the Reports tab. On the tab, in the folders pane, click

and select **New Report Folder** or **New Dashboard Folder** to create a folder, or click on a folder to edit.

Consider the following when you edit reports or dashboards you didn't create or when you're try to share reports with others:

- You can modify the contents of a folder only if the folder access level is set to Read/Write. Only
 users with the "Manage Public Reports" or "Manage Public Dashboards" can delete or change
 a Read Only folder. Regardless of permissions or folder settings, you can't edit any unfiled folders
 or others' personal folders.
- 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.
 Note that public reports and report folders are not available with Personal, Contact Manager,
 and Group Editions.
- To make a report public, run the report and click Save As. Give the report a name and choose
 a public report folder.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** Editions except **Database.com**

Report folders not available in: Contact Manager, Group, and Personal Editions

USER PERMISSIONS

To create, edit, and delete public report folders:

"Manage Public Reports"

To create, edit, and delete public dashboard folders:

 "Manage Dashboards" AND "View All Data"

IN THIS SECTION:

1. Access to Report Folders

Folder access is controlled by permissions. Users must have certain permissions to access public, hidden, or shared report folders.

2. Access to Dashboard Folders

Folder access is controlled by permissions. Users must have certain permissions to access public, hidden, or shared dashboard folders.

SEE ALSO:

Access to Dashboard Folders
Access to Report Folders

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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** Editions except **Database.com**

Report folders not available in: **Contact Manager**, **Group**, and **Personal** Editions

Access Level	Permissions to Access Read-Only Folders	Permissions to Access Read/Write Folders
Read	Any of the following:	Any of the following:
	"View Reports in Public Folders"	• "Run Reports"
	• "Run Reports"	"Manage Public Reports"
	"Manage Public Reports"	• "View All Data"
	• "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 All Data"
	"View Reports in Public Folders"	
	• "View All Data"	

Access Level	Permissions to Access Read-Only Folders	Permissions to Access Read/Write Folders
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:	Any of the following:
	"View Reports in Public Folders"	• "Run Reports" (for shared users)
	"Run Reports" (for shared users)	 "Manage Public Reports" (for shared users)
	"Manage Public Reports"	• "View All Data"
	• "View All Data"	
Write New	"Manage Public Reports"	"Create and Customize Reports" (for shared users)
Modify/Delete	"Manage Public Reports"	"Manage Public Reports"

Access to Dashboard Folders

Folder access is controlled by permissions. Users must have certain permissions to access public, hidden, or shared dashboard folders.

The following tables show the permissions that users must have to access the three different types of dashboard folders: public, hidden, and shared 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 to Access Read-Only Folders	Permissions 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"

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: All Editions except **Database.com**

Report folders not available in: Contact Manager, Group, and Personal **Editions**

Access Level	Permissions to Access Read-Only Folders	Permissions to Access Read/Write Folders
Modify/Delete	All of the following:	All of the following:
	• "Run Reports"	• "Run Reports"
	"Manage Dashboards"	"Manage Dashboards"
	• "View All Data"	• "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 to Access Read-Only Folders	Permissions to Access Read/Write Folders
Read	Both of the following:	Both of the following:
	• "Run Reports"	• "Run Reports"
	• "View All Data"	• "View All Data"
Write New	All of the following:	All of the following:
	• "Run Reports"	• "Run Reports"
	"Manage Dashboards"	• "Manage Dashboards"
	• "View All Data"	• "View All Data"
Modify/Delete	All of the following:	All of the following:
	• "Run Reports"	• "Run Reports"
	"Manage Dashboards"	• "Manage Dashboards"
	• "View All Data"	• "View All Data"

Shared Folders

Access Level	Permissions to Access Read-Only Folders	Permissions to Access Read/Write Folders
Read	"Run Reports"	"Run Reports"
Write New	All of the following:	Both of the following:
	• "Run Reports"	• "Run Reports"
	"Manage Dashboards"	"Manage Dashboards"
	• "View All Data"	
Modify/Delete	All of the following:	All of the following:
	• "Run Reports"	• "Run Reports"
	"Manage Dashboards"	"Manage Dashboards"
	• "View All Data"	• "View All Data"

Move Dashboards Between Folders in Lightning Experience

If your org is using Enhanced Analytics Folder Sharing for Reports and Dashboards, then folders govern who has access to the reports and dashboards saved inside. (If your org was created after the Summer '13 Salesforce release, you already have Enhanced Analytics Folder Sharing.) Move dashboards between folders to both organize and control access to dashboards.

- 1. Open a dashboard.
- 2. Click Edit.
- **3.** Open the Properties menu by clicking * .
- **4.** Under **Folder**, clear the dashboard's current folder and enter the destination folder.
- **5.** From the Properties window, click **Save**. Then, from the Dashboard Builder, click **Save** again.

The dashboard moves into the destination folder. Users' access rights to the dashboard change accordingly.

EDITIONS

Available in: Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete public dashboard folders:

"View All Data"

AND

"Manage Dashboards"

To move dashboards you created from one folder to another in Lightning Experience

"Run Reports"

AND

"Manage Dashboards"

To move dashboards you didn't create from one folder to another in Lightning Experience

"View All Data"

AND

"Run Reports"

AND

"Manage Dashboards"

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 and remove the browser-imposed headers and footers.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To print dashboards:

"Run Reports" AND access to dashboard folder

Organize Dashboards

Keep your dashboards 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.

IN THIS SECTION:

1. Delete a Dashboard

It's a good idea to delete dashboards that you no longer need.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

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.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To delete dashboards:

"Run Reports" AND
 "Manage Dashboards"

To delete dashboards created by another user:

 "Manage Dashboards" AND "Modify All Data"

Dashboard Limits

As you build dashboards of your data, be aware of these limits and restrictions.

Salesforce Reports and Dashboards Limits

Feature	Personal Edition	Contact Manager		Polesiand Edition	Enterprise Edition	Unlimited and Reference 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 dasl	hboard			
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							

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

Feature	Personal Edition	Contact Manager	Group Edition	Professional Edition	Enterprise Edition	Unlimited and Performance Edition	Developer Edition
Reporting snapshots	N/A			1 ^{2, 3, 4}	1 ²	2 ²	1 ^{2, 3, 4}
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 ^{2, 3, 5}	1 ²	2 ²	1 ^{2, 3, 5}	

¹ These limits apply to the report builder. If you're using the report wizard, the limit is 10.

Salesforce retains historical data for the previous three months, plus the current month.

The following Salesforce Reports and Dashboards limits apply to all supported editions.

Report Limits

- The report builder preview shows a maximum of 20 rows for summary and matrix reports, and 50 rows for tabular.
- 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.
- Reports display a maximum of 2,000 rows. To view all the rows, export the report to Excel or use the printable view for tabular and summary reports. For joined reports, export is not available, and the printable view displays a maximum of 20,000 rows.
 - Summary and matrix reports 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.
- 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 Salesforce1, they're converted to tabular reports.
- 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.
- 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.

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.

• 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 254 characters in a rich text area or a long text area are displayed in a report.
- 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 is not supported for joined reports.

Dashboard Limits

- A dashboard filter can have up to 50 options. By default, 10 options are enabled. Contact Salesforce to extend your limit.
- Each dashboard can have up to 20 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.
- A dashboard table or chart can display up to 20 photos.
- You must wait at least one minute between dashboard refreshes.

Report Type Limits

- 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 additional 56 objects. However, users will 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

- 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

Only the first 255 characters in a custom text field count for filtering purposes.

Embedded Report Charts Limits

- 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.
- 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

• Only the first 255 characters are shown for custom long text area fields in list views.

Bucket and Bucket Field Limits

- 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

- 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, this includes the standard preselected fields, which can't be disabled.
- Formula fields are not supported.
- Row limit filters are not supported.
- The summary report format is not 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.
- Dated exchange rates are not supported. When you run the historical trend report, it only uses the latest dated exchange rate.
- Internet Explorer 6 is not supported.
- You can't subscribe to historical trend reports.
- The Report Wizard is not 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

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 2,000 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.

Reports and Dashboards API Limits

The following limits 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 trend reports are only supported for matrix 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.

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 706 and Refresh Dashboard Data on page 710.

Dashboards Help You Visualize Complex Information

Dashboards help you make decisions based on real-time data you've gathered with reports. Use dashboards to help users identify trends, sort out quantities, and measure the impact of their activities.

Watch a Demo: • An Overview of Dashboards (Salesforce Classic)



Note: Dashboards in Group Edition organizations are view-only.

IN THIS SECTION:

1. Get Started with Dashboards

After you've found the data that you need, use a dashboard to find patterns, stay up-to-the-minute on changes, and share knowledge that you and your coworkers can act on in real time.

2. View Dashboard Lists

The dashboard list contains all the dashboards you can view.

SEE ALSO:

Filter Your Dashboard Build a Salesforce Classic Dashboard

Get Started with Dashboards

Provide Individualized Views of a Dashboard in Salesforce Classic



Available in: Salesforce Classic

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

Get Started with Dashboards

After you've found the data that you need, use a dashboard to find patterns, stay up-to-the-minute on changes, and share knowledge that you and your coworkers can act on in real time.

- Users with a Salesforce Platform or Salesforce Platform One user license can only view a
 dashboard if the dashboard running user also has the same license type. Users with user licenses
 that don't have full dashboard access can only see dashboards with running users of the same
 license type. Consider creating separate dashboards for users with different license types.
- Dashboards in Group Edition organizations are view-only.

Clicking the Dashboards tab displays the dashboard you viewed most recently. The top of the page shows the time the dashboard was refreshed last and the user whose permissions determine what data is visible on the dashboard. If you can't access a dashboard, verify your folder permissions.

SEE ALSO:

Dashboards Help You Visualize Complex Information Build a Salesforce Classic Dashboard Refresh Dashboard Data

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To view and refresh dashboards:

 "Run Reports" AND access to dashboard folder

To create dashboards:

"Run Reports" AND
"Manage Dashboards"

To edit and delete dashboards you created:

"Run Reports" AND
"Manage Dashboards"

To edit and delete dashboards you didn't create:

 "Run Reports," "Manage Dashboards," AND "View All Data"

View Dashboard Lists

The dashboard list contains all the dashboards you can view.

From the Dashboards tab, click Go to Dashboard List.
 The enhanced Reports tab lists your recently viewed dashboards in that dashboard's folder.

- **2.** On the dashboard list page, select a folder to view a list of dashboards stored in that folder, or search with filters.
- 3. Click a dashboard name to display the dashboard.
- **4.** If you're working in Salesforce Classic, Click next to a dashboard name to edit or delete a dashboard.
 - Note: Dashboards in Group Edition organizations are view-only.
- **5.** If Chatter is enabled, click or × to follow or stop following a dashboard in your Chatter feed.

SEE ALSO:

Dashboards Help You Visualize Complex Information Get Started with Dashboards

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To view and refresh dashboards:

 "Run Reports" AND access to dashboard folder

To create dashboards:

"Run Reports" AND
 "Manage Dashboards"

To edit and delete dashboards you created:

"Run Reports" AND "Manage Dashboards"

To edit and delete dashboards you didn't create:

 "Run Reports," "Manage Dashboards," AND "View All Data"

WAVE ANALYTICS, REPORTS, AND DASHBOARDS RESOURCES

In addition to online help, Salesforce creates video demos, guides, and tip sheets to help you learn about our features and successfully administer Salesforce.

Wave Analytics

	For End Users	For Admins
Guides and Tip Sheets		
Wave Analytics Setup Guide		✓
Wave Analytics Security Implementation Guide	✓	✓
Wave Analytics Data Integration	✓	✓
Videos		
 Create the Sales Wave App, Part 1: Using the Configuration Wizard (Lightning Experience) 		✓
Create the Sales Wave App, Part 2: Upload the Quota CSV File and Schedule a Dataflow (Lightning Experience)		✓

Reports and Dashboards

For End Users	For Admins
✓	
	✓
✓	
✓	
✓	
✓	
✓	
✓	
	✓
	✓
	•

	For End Users	For Admins
Sample CRM Dashboards	✓	
What Are Joined Reports?		✓
Using Cross Filters in Reports		✓
Using Bucket Fields		✓
Videos		
Tips for Scheduling Reports (Salesforce Classic)		✓
Make Data-Driven Decisions with Reports and Dashboards (Lightning Experience)		✓
An Overview of Dashboards (Salesforce Classic)	✓	
 Getting Dashboards to Display the Right Data (Salesforce Classic) 	✓	
 Analytics Folder Sharing (Salesforce Classic) 	✓	
Building Matrix Reports (Salesforce Classic)	✓	
Using Cross Filters in Reports (Salesforce Classic)	✓	
Getting Started with Buckets (Salesforce Classic)	✓	
Getting Started with Report Builder (Salesforce Classic)	✓	
● Introducing Joined Reports in Salesforce (Salesforce Classic)	✓	
Making Your Reports Run Faster (Salesforce Classic)	✓	
Report Notifications (Salesforce Classic)	✓	
Embedding Charts Anywhere (Salesforce Classic)	✓	

INDEX

\312	Article Types
A	report 504
A	Articles
access, controlling 32	report 504, 507
Accounts	reporting 504
reports 463	Assets
Activities	reports 489
reports 464	augment transformation
activities reports 500	description 62
activity 515	example 62
add action to Salesforce object 361, 365	JSON syntax 62
add field 130, 135	
add object 130, 135	В
admin 29	Bar chart
advanced filter in sfdcDigest transformation 100	dashboard component 681
Age	binding 318
cases 490–492	bindings
aggregation	steps 297, 300
definition 9, 200	box widget 319
aggregation change 208	browse 7
analytics 19, 447	Builder permission set license 33, 35
Analytics Cloud Builder permission set license 33, 35	bulk actions
Analytics Cloud Explorer permission set license 33, 35	assign to table widget (Wave dashboard designer) 374
Analytics Cloud Wave Platform license 19, 22, 33, 35	configure (Wave dashboard designer) 368
Analytics Cloud Wave Platform permission set license 22	create Visualforce page (Wave dashboard designer) 372
Analytics Cloud Wave Platform permission sets 25	define Apex class (Wave dashboard designer) 370
Analytics Cloud Wave Platform setup 25	example (Wave dashboard designer) 369
android 19	
Annotations 14, 28	C
Answers	calculated fields
reports 496	compare table (Wave dashboard designer) 273
API 447	calculating deltas 79
API limits 451	Call Center
app 376, 378	reports 497
append transformation	Campaigns
description 61	reports 467
example 61	Cases
JSON syntax 61	reports 490–492
parameters 62	change chart scale 211
AppExchange	change chart type 207, 211, 213, 221
Sample CRM Dashboards 700	change measures 208
Application Lifecycle Management 353, 355–356, 449	chart 320–321
apply action to Salesforce page layout 363	chart widget
Approval history reports	adding (Wave dashboard designer) 257
creating 497	Charts
creating 177	chart formatting tab 590
	CHAIL IOITHALLING LAD 270

Charts (continued)	container widget
combination 587	adding (Wave dashboard designer) 275
combination chart examples 588	create app 376, 378
combining small groups 590	create measures from dimensions 82
customizing properties 583	create recipe 149, 151, 153–155, 157, 159, 161–164, 166
hover details 590	creating 50, 136
See Reports 660	creating a dataset 51, 123
types 226–227, 229, 231, 233–234, 236, 240, 243–247, 330,	creating datasets 138
334–335, 670–676, 679	creating with dataflow 51
classic designer	creating with external data 138
comparing designers 249	creating with the dataset builder 123
clipped lens 316	cross-block custom summary formula
clone 223	example 574
clone visualization 223	Cross-block custom summary formula 578
Cloning	Custom Report Type
dashboards 725	Salesforce Knowledge 505
collaborate 11	custom report type activities 500
	* **
Column chart	Custom Report Types
dashboard component 681	approval history 497
column types	forecasting 501
Date 40	forecastinig 501
Numeric 40	process instance 497
Text 40	Custom summary formulas
Communities	joined reports 567
Wave Analytics 350	reports 533
computeExpression transformation description 69	Custom views
computeExpression transformation example 69	field filters 601, 615
computeExpression transformation JSON syntax 69	Customizable forecasts
computeExpression transformation parameters 71	reports 472
computeRelative transformation description 73	customizations 406, 430
computeRelative transformation example 73	Customizing
computeRelative transformation JSON syntax 73	reports 523
computeRelative transformation parameters 76	6
configurator 394, 424, 427	D
configuring 52, 133–134	dashboard
Connect for Office	steps (Wave dashboard designer) 279
Excel integration 595	Dashboard
installing 595	overview 720
connected app 19	Dashboard builder
Connected App 29	adding components 664
connections	customizing dashboards 664
data source (Wave dashboard designer) 299	editing components 667
considerations for datasets 168	editing dashboards 664
Contacts	editing filters 701
contacts related to a task, viewing in reports 499	filters overview 700, 703
contacts related to an event, viewing in reports 499	dashboards
reports 463	building with classic designer 312
Shared Activities, viewing in reports 499	create with classic designer 313
5. Sied rearries, nerving irreports 122	links (Wave dashboard designer) 278

dashboards (continued)	Dashboards (continued)
steps 280	running user 697, 704
widget interactivity 297	running user, selecting 706
widget interactivity, bindings 300	Salesforce Classic 663
Dashboards	Sample CRM Dashboards on AppExchange 700
adding components 664–665	schedule refresh 711
applying filters 702	scheduled jobs queue 711
Bar chart	searching 643
683, 688–689	setting properties 697
dashboard component 683, 688–689	share 713
bar chart component 681, 683, 688–689	table component 687, 693
build 661	using charts 226–227, 229, 231, 233–234, 236, 240, 243–247,
cloning 725	330, 334–335, 670–676, 679
column chart component 681	view as 704–705
component types 669	viewing 725
components 667	viewing dashboard lists 726
create table 687, 693	dashboards (Wave dashboard designer)
creating 662–663	adding widgets 253
customizing 664	building 250
deleting 720	creating 251
Donut chart	design best practices 323
686, 692	faceting 298
dashboard component 686, 692	filtering 261
donut chart component 686, 692	layout properties 310
drag and drop 647	layout selections 312
dynamic dashboards 707	layouts 307
dynamic dashboards, setting up 709	properties 253
editing 664	step properties 297
editing components 667	widget properties 291
editing filters 701	widgets, grouping 275
email 711	data 38, 402–403
expand component 713	data exploration
filters overview 700, 703	clip lens 223
funnel chart component 682, 689	concepts 9, 200
gauge component 684, 690	data integration
Lightning Experience 662	description 40
line chart component 684, 691	external data 40
metric component 685, 691	Salesforce data 40
move 647	data integration dataflow 49
move between folders 718	data monitor 56, 142
overview 724	data prep 149, 153, 161–166
Pie chart	data sources
686, 692	connections (Wave dashboard designer) 299
dashboard component 686, 692	data storage 451
pie chart component 686, 692	data storage limits 451
printing 719	dataflow 51–52, 55, 58, 126–127, 130–135
read 710	dataflow definition file
refreshing data 710	sample with append 61
run 710	sample with augment 62

dataflow definition file (continued)	Email
sample with filter 85	dashboards 711
sample with flatten 87	report 619–620, 623–624
dataflow description 49	embed 33, 336–338, 341, 349
dataflow designing 51	embed in Salesforce pages 430
dataflow jobs 56, 451	empty keys 64
dataflow jobs limits 451	enable null measures 45
dataset 149, 151, 153, 161–166	enabling 130
dataset builder 123	Enabling 26–29
dataset builder description 50, 136	Error
Dataset definition 9, 200	obsolete report 655
dataset recipe 148, 151	event monitoring wave app 445
dataset recipe description 148	Event Monitoring Wave app 444
datasets	Event Monitoring Wave App 439, 444, 446–447
column types 40	Event Monitoring Wave dashboards 446
creating 40	Event Monitoring Wave Permissions 440–443
description 40	Events
filtering datasets 85	multiple contacts related to an event, viewing 499
datasets delete 148	reports, viewing multiple contacts related to an event 499
datasets description 41, 50, 136	example 82–83, 92, 112, 114, 116
datasets edit 144, 147	example JSON syntax 79
datasets row-level security example 169, 173, 181	excel data 50, 136
date columns 47	explorer
date widget	clip lens 223
adding (Wave dashboard designer) 263	Explorer data collections within home page 9, 200
dates description 47	Explorer permission set license 33, 35
Deleting	export as image 225
dashboards 720	Exporting
reports 648	reports 625
delta transformation 79, 81	extend 336
delta transformation description 79	extended metadata 147
description 56, 59, 83, 92, 112, 114, 116	external data 50, 136, 138, 141–142, 198, 451
designer 449	External Data API 51
developer 447	external data limits 451
devices	external data row-level security example 169
layout selection 312	external data upload 50
dim2mea transformation 82–83	external data uploads 142
dim2mea transformation description 82	extracting data from datasets 83
dimension	-
definition 9, 200	F
dimensions	facet
description 40	across datasets (Wave dashboard designer) 299
disable 134	dashboards (Wave dashboard designer) 298
Download 27	static steps (Wave dashboard designer) 299
Dynamic dashboards	steps 297
setting up 709	faceting 318
г	FAQ
E	control access, reports 714
edgemart transformation 83, 85	error, obsolete report 655

FAQ (continued)	Folders (continued)
limits, reports 655	permissions 715–716
report data 654	reports 715
reports, control access 714	Reports 627
reports, limits 655	Sharing 627
reports, obsolete report error 655	Forecasts
reports, old name in role hierarchy 656	custom report type 501
role hierarchy, old role name in report 656	reports 501
field metadata overrides 101	formula_ field 159
Field Service Wave Analytics app 433–434, 436–437	Formulas
field-level security 31	joined reports 567
fields 41	PARENTGROUPVAL 534
fields overriding attributes 119	PREVGROUPVAL 534
File 519	reports 533
Files	summary functions, overview 534
reports 469	summary functions, using 534
filter	Funnel chart
global filter (Wave dashboard designer) 259	dashboard component 682, 689
initial filter (Wave dashboard designer) 300–301	das isodia component sozy sos
on dates (Wave dashboard designer) 263	G
on dimensions (Wave dashboard designer) 265	Gauge 684, 690
range (Wave dashboard designer) 269	generating metadata file 141
toggle (Wave dashboard designer) 271	get started 3–4
filter expressions	get URL to share 17, 224
syntax 86	getting started 35, 38, 379, 381, 383, 391, 394, 397, 400, 402–404,
filter recipe 154–155, 157	409–411, 413, 421, 424, 427–429, 433–434, 436–437, 439,
filter transformation	444–445, 447
description 85	global filter widget
example 85	adding (Wave dashboard designer) 259
filter expressions syntax 86	group by 203
JSON syntax 85	group by 203
parameters 86	Н
filters 93–94, 100	
Filters	handling 47
	Hiding
reports 599, 602 flatten transformation	report types 646
	1
description 87	
example 87	Ideas
JSON syntax 87	reports 503
parameters 91	image widget
Flatten transformation 181	adding (Wave dashboard designer) 276
Folder	images
analytics 628–633	Wave dashboard designer 276
dashboard 628–633	incorporate 336
reports 628–633	incremental 132
sharing 628–633	integrate with Salesforce 406
Folders	Integration User description 31
dashboards 716	integrations 406, 430
Dashboards 627	iOS 19

	list widget
ining reports	adding (Wave dashboard designer) 265
joined reports	custom values, adding 287
examples 574	Live Agent
Joined reports	reports 517
cross-block custom summary formulas 567	sessions 517
custom summary formulas 567	Live Agent session
JSON syntax 82–83, 92, 112, 114, 116	reports 517
K	M
key performance indicator	
adding (Wave dashboard designer) 255	matching records 64
keyboard 10, 248	measure
knowledge 515	definition 9, 200
Miomeage 313	measure change 208
L	measures
labels	description 40
	Metric 691
adding (Wave dashboard designer) 278	mobile 19
layout 19	monitor 132
Layouts	monitoring 56
properties in Wave dashboard designer 310	monitoring uploads 142
layouts (Wave dashboard designer)	Move
creating 307	dashboards 647
selection process 312	reports 647
Leads	multiple matches 64
reports 473	
lens	N
clip 223	navigate 6
lens definition 9, 200	navigation 377
License Management App 353, 355–356, 449	Notifications 12
licenses 19, 22, 33, 35	null keys 64
Lightning Pages	null measures 43
standard components 338	number widget
limitations 102, 407, 432	adding (Wave dashboard designer) 255
limits 41, 136	numeric values 43
limits API calls 451	Humene values 43
Line chart	0
dashboard component 684, 691	object-level security 31
link widget	
adding (Wave dashboard designer) 278	objects overriding attributes 119
links	open Wave 4
dashboards 278	operations 64
lenses 278	Opportunities
steps 278	creating custom reports 477
links to Salesforce records 17, 358	creating reports 477
List View	custom reports 474
Customize 644	Opportunities report 479
Dashboards 644	Opportunities with Competitors report 481
Reports 644	Opportunities with Contact Roles and Products report 481 Opportunities with Contact Roles report 480

Opportunities (continued)	properties
Opportunities with Opportunity Teams and Products report 484	dashboards (Wave dashboard designer) 253 default widget 253
Opportunities with Opportunity Teams report 484	Wave dashboard designer layouts 310
Opportunities with Partners report 485	Wave dashboard designer steps 297
Opportunities with Products report 486	Wave dashboard designer widgets 291
Opportunities with Quotes and Quote Line Items report 488	
Opportunities with Quotes and Quote PDFs report 487	Q
Opportunity Field History report 483	queries 451
Opportunity History report 483	queries limits 451
Opportunity Trends report 487	quick actions menu 17, 358, 361, 363, 365
reports 474	Quotes
standard reports 474	Opportunities with Quotes and Quote Line Items report 488
using reports	Opportunities with Quotes and Quote PDFs report 487
476	R
using standard reports 476	K
Overview	range 43
reports 660	range widget
Salesforce Reports and Dashboards 456	adding (Wave dashboard designer) 269
P	recipe 149, 151, 153–155, 157, 159, 161–166
	recipe preview 151
packaging 447	record ownership example 169
Page layout	registering 112
report charts 637–642	Relationship groups
parameters	reports 518
append transformation 62	remove fields from datasets 114
filter transformation 86	replication 130–132, 134–136
flatten transformation 91	replication description 126–127
Partners	replication settings 133
reports 518	report 515
permission set licenses for Wave Analytics 19, 33, 35	Report 519
permission sets for Wave Analytics 22	Report builder
personalize 11	adding and removing fields 529
pin apps 5	asynchronous loading 526
predicate considerations 168	changing currencies 529
predicates 168	charting tips 586
prepare data 148	charts 583
present 6	conditional highlighting 593
presentation 377	customizing reports 460
principal report type	editing reports 460
changing 566	filter 599, 602
Printing	formulas 532
reports 645	grouping fields 525
process 9, 199	preview loading 526
Process Instance Abject 497	reordering columns 529
Process Instance object 497 Products	sorting columns 529
	summary fields 530
reports 489	working with fields 529

Report Builder	Reports (continued)
upgrading 598	changing currencies 529
Report schedule	chart formatting tab 590
report 623	chart properties 583
Report type specific custom summary formula 578	charts 637–642
Report types 456	choosing a report format 521
Report Types	combination chart examples 588
hiding 646	combination charts 587
Reporting	conditional highlighting 593
adding blocks 569	creating 523
blocks 568, 571–572	creating charts 583
custom summary formula tips 536	creating cross filters 607
custom summary formulas and joined reports 578	creating with the report wizard 524
deleting blocks 572	cross filters 606, 610
hide details 592	cross filters example 608–609
improving report performance 656	custom options 657
moving blocks 570	customizing 523
naming blocks 571	customizing reports 460
record count 572	data 654
renaming blocks 571	deleting 537, 648
reordering blocks 570	drag and drop 647
show details 592	editing reports 460
working with blocks 568	email 619–620, 623–624
Reporting snapshot	error, obsolete report 655
about 546	events, viewing multiple contacts related to 499
creating 547	exception reporting 606, 610
mapping fields 550	exporting 625
running 551	exporting to the background 635–636
scheduling 551	field filters 601, 615
setting up 547	files 469
terminology 546	filter definitions 605
troubleshooting 554	filter logic 605
Reporting snapshots	filter types 605
defining 549	Filter Via URL 603
managing 553	filters 599, 602
reports	formulas 532–533
control access 714	grouping 565, 573
Reports	grouping fields 525
about 660	historical 482, 537–545
adding and removing fields 529	joined format 566
adding report types 566	joined report format 565, 573
analytics 545	joined reports 521
approval history 497	limiting results 594
bucket fields 557	limits 655
bucketing, numeric 559	matrix reports 521
bucketing, picklist 560–561	move 647
bucketing, text 562, 564	Notifications 617–618
build 459	Opportunity 482, 537–543, 545
Case 544	organize 642, 719

Reports (continued)	Reports schedule (continued)
outer joins 606, 610	report 624
overview 457, 649, 660	Reports Tab
PARENTGROUPVAL 534	list view 644
partners 518	reschedule 58
pinning folders 645	Role hierarchy
PREVGROUPVAL 534	role name in report, old 656
principal report type 565–566, 573	role hierarchy example 181
printing 645	row-level security
publicly shared 537	example, determining 170
read 625	row-level security on datasets 167
record page layout 637–642	row-level security record ownership example 181
reordering columns 529	run 131
report blocks 565, 573	run apps 6
report builder 527	run-time monitoring 56
role name, old 656	running a recipe 165
row limit example 610	running a recipe 103
run 625	S
Salesforce CRM Content 469	
	Sales Wave Analytics app 35, 38, 379, 381, 383, 391, 394, 397, 400,
Salesforce Knowledge 504, 507	402–404, 406–407, 409
Salesforce Knowledge fields 508	Sales Wave Analytics App Permissions 389
saving 537	Sales Wave Analytics Permissions 388, 390
scheduled jobs queue 622–624	Sales Wave Permissions 387
scheduling 619–620, 622–624	Salesforce CRM Content
searching 643	reports 469
selecting a report type 461	Salesforce data 31–32, 50, 136
share 626	Salesforce data row-level security example 173, 181
Shared Activities, viewing in 499	Salesforce Knowledge
sorting columns 529	report 504, 507
standard report types 580	reporting 504, 508
subtotaling results 531	Salesforce Knowledge Article Types
summary fields 530	report 504
summary functions, overview 534	Salesforce Knowledge Articles
summary functions, using 534	report 504, 507
summary reports 521	reporting 504
tab 660	Salesforce Reports and Dashboards
tabular reports 521	Overview 456
tabular reports in dashboards 594	sample with delta 79
tasks, viewing multiple contacts related to 499	sample with dim2mea 82
trending 482, 537–545	sample with edgemart 83
troubleshoot 652	sample with sfdcDigest 92
unscheduling 624	sample with sfdcRegister 112
upgrading report builder 598	sample with sliceDataset 114
using charts 226–227, 229, 231, 233–234, 236, 240, 243–247,	sample with update 116
330, 334–335, 670–676, 679	SAQL 448
working with 657	SAQL viewer 222
working with fields 529	save 224
Reports schedule	save visualization 224
delete 624	schedule 131

Schedule	start 55
change 623	static step
dashboard refresh 711	creating 287
delete 623	static steps
report 622–624	facet (Wave dashboard designer) 299
view 623	steps
Scheduled jobs queue	binding 297, 300
viewing 622–624, 711	cloning 289
screenshot 224	creating with explorer 282
search 7, 515	creating with SAQL editor 284
Search 519	description 280
security 30	faceting 297
security on datasets 167	properties in Wave dashboard designer 297
security on Salesforce data 31	static, creating 287
security policy examples 167	Wave dashboard designer 279
Security User description 31	structured filter in sfdcDigest transformation 94
Self-Service	swap chart position 208
reports 490	
Service Wave Analytics app 410–411, 413, 421, 424, 427–430, 432	T
Service Wave Analytics Permissions 419	Table
Service Wave Permissions 417, 419, 421	dashboard component 687, 693
set up 349	table widget
set up Event Monitoring Wave 440–443	adding (Wave dashboard designer) 273
set up Sales Wave Analytics 388, 390	bulk action example (Wave dashboard designer) 369
set up Sales Wave Analytics App 389	bulk actions (Wave dashboard designer) 368
set up Service Wave Analytics 419, 421	tables
setup 29	compare table (Wave dashboard designer) 273
setup Sales Wave 387	values table (Wave dashboard designer) 273
setup Service Wave 417	Tabs
sfdcDigest transformation 92–93, 101–102, 106	reports 660
sfdcRegister transformation 112–113	Tasks
share 11, 336, 375	multiple contacts related to a task, viewing 499
share dashboard 17, 224	reports, viewing multiple contacts related to a task 499
share lens 17, 224	team membership example 173
Shared Activities	Territories
reports, viewing details in 499	reports 495
Sharing	text widget
dashboards 628–633	adding (Wave dashboard designer) 278
Dashboards 627	toggle widget
folders 628–633	adding (Wave dashboard designer) 271
Folders 627	custom values, adding 287
reports 628–633	track notifications 5, 7
Reports 627	transformations
sliceDataset transformation 114–115	append transformation 61
SoftPhone calls	augment transformation 62
reports 497	filter transformation 85
Solutions	flatten transformation 87
reports 490–492	troubleshooting 56, 141
sort order 207	

U	Wave dashboard keyboard shortcuts 321
	Wave dashboards 14
update dataflow 135	Wave distribution 356
update transformation 116, 118	Wave Everywhere 336
use history 220	wave for b2b marketing app 447
use in row-level security 181	Wave for Communities 27, 350
user permissions list 21	Wave home 5–7
user profile 31	Wave meatadata 356
Users	Wave Mobile 29
reports 466	Wave Packaging 353, 355–356, 449
V	Wave Public API 448
V	whitelist 33, 349
Videos 727	widget
view data 202, 222	static steps 287
Visualforce	steps, cloning 289
dashboard components 667	steps, creating with explorer 282
visualization	steps, creating with SAQL editor 284
adding (Wave dashboard designer) 257	widgets
141	adding to dashboard 253
W	chart (in Wave dashboard designer) 257
walkthrough 515	container (in Wave dashboard designer) 27
wave 3	date (Wave dashboard designer) 263
Wave 19, 337–338, 341, 447	default properties 253
Wave Analytics	filter (Wave dashboard designer) 261
Communities 350	global filter (Wave dashboard designer) 259
Wave Analytics Trailhead 2	image (Wave dashboard designer) 276
Wave Analytics tutorial 2	interactivity 297
wave apps 135, 379	interactivity, bindings 300
Wave apps 375, 377	link (Wave dashboard designer) 278
wave assets 8	list (Wave dashboard designer) 265
Wave Change Sets 353, 355, 449	number (Wave dashboard designer) 255
Wave dashboard 33, 349	properties in Wave dashboard designer 291
Wave dashboard designer	range (Wave dashboard designer) 269
comparing designers 249	steps 280
converting from classic designer 325	table (Wave dashboard designer) 273
dashboards, building 250	text (Wave dashboard designer) 278
dashboards, creating 251	toggle (Wave dashboard designer) 271
dashboards, faceting 298	window functions 217
layouts, creating 307	WITIGOW TUTICUOTS 217
layouts, selecting 312	X
steps 279	xmd 147
•	XIIIU 14/