



Analyze Your Data

Salesforce, Spring '16



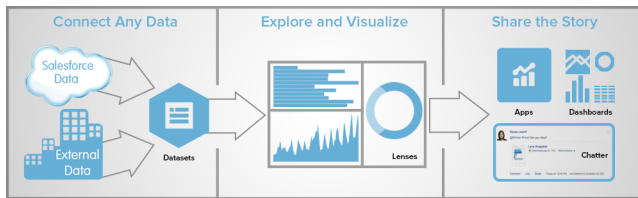
CONTENTS

ANALYTICS	1
Wave Analytics	1
Reports and Dashboards	224
WAVE ANALYTICS, REPORTS, AND DASHBOARDS RESOURCES	441
INDEX	443

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 dashboards. It's a better way to distribute insight to business users so they can understand and take action on changing information.



- [Learn Wave Analytics with Trailhead](#)
- [Wave Analytics Glossary](#)
- [Wave Analytics Limits](#)
- [Wave Analytics Limitations](#)

Set Up Wave Analytics

- [Wave Platform Setup on page 4 | PDF](#)
- [Wave Analytics Security Implementation Guide PDF](#)

Integrate Your Salesforce Data and External Data

- [Data Integration | PDF](#)
- [External Data API Guide | PDF](#)
- [External Data Format Reference | PDF](#)

Explore and Share Data

- [Key Elements of a Wave Dashboard](#)
- [Explore and Visualize Your Data](#)
- [Support Decisions with a Wave Analytics Dashboard](#)
- [Incorporate Wave Analytics Throughout Your Business](#)

Wave Analytics Apps

- [Work with Wave Analytics Apps](#)
- [The Sales Wave Analytics App](#)
- [The Admin Analytics App \(Pilot\)](#)

Customize Wave Analytics

- [Dashboard JSON Guide](#) | PDF
- [SAQL Reference](#) | PDF
- [Extended Metadata \(XMD\) Reference](#) | PDF

Wave Analytics Mobile

- [Salesforce Analytics for iOS Help](#)

Wave REST API

- [Wave REST API Developer's Guide](#) | PDF

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.

To access all of the Wave Analytics trails, start here: https://developer.salesforce.com/trailhead/trail/wave_analytics_explorer. You'll find the sign up for the special Developer Edition org in the Wave Analytics Basics trail.

SEE ALSO:

[Wave Analytics Limits](#)

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.

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.

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.

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.

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 Platform Setup

To set up your organization to use the Salesforce Wave Analytics platform, learn about Wave Analytics platform licenses and permissions. Then follow the same basic process you use to set up other Salesforce products.

 **Important:** 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.

 **Note:** Sales Wave app users should go to [The Sales Wave Analytics App](#) to get started with the app.

Each Analytics Cloud - Wave Analytics Platform license is a single-user license that provides access to the Salesforce Wave Analytics cloud. It includes a single Analytics Cloud - Sales Wave Analytics App single-user license. 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
Analytics Cloud - Sales Wave Analytics App	25 million rows when used without Analytics Cloud - Wave Analytics Platform license. Use of Sales 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.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

Wave Platform Setup Process

Enable Wave and assign permission set licenses to users. The first step when you set up Wave is simply to enable it. Then, you assign an Analytics Cloud - Wave Analytics Platform permission set license to each user. Each permission set license can be assigned to

only one user. Also, the user license that's associated with the user profile must support the Analytics Cloud - Wave Analytics Platform permission set license. Not all user licenses support the Wave platform permission set license.


Define user types and create and assign permission sets. Next, define the main user types in your organization and create permission sets to group related user permissions according to each user type's needs. After you create permission sets, assign them to users. Although you can assign a permission set to an individual user, for efficiency, you can also assign it to groups of users. You can assign multiple permission sets to a user.



[Walk Through It: create, edit, and assign a permission set](#)

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

 **Note:** 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.

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

IN THIS SECTION:

1. [Learn About the Analytics Cloud Permission Set License and User Permissions](#)
The Analytics Cloud - Wave Analytics Platform permission set license enables all the user permissions needed to explore data with and manage Salesforce Wave Analytics.
2. [Identify Wave Analytics Platform User Types](#)
Identifying types of Wave users and the tasks they need to perform helps assure that the setup process meets the majority of your team members' analytics needs.
3. [Enable Wave Analytics and Select Analytics Cloud - Wave Analytics Platform Permission Set License](#)
After Salesforce provisions you with the Wave Analytics platform license, you need to enable Wave and select the Wave Analytics permission set license and assign it to users.
4. [Create Wave Analytics Permission Sets](#)
To give users in your organization access to Wave features, create and assign one or more permission sets based on the level of analytics capabilities they need.
5. [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 the Wave functionality.
6. [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 the Analytics Cloud Permission Set License and User Permissions

The Analytics Cloud - Wave Analytics Platform permission set license enables all the user permissions needed to explore data with and manage Salesforce Wave Analytics.

Each Wave platform user needs an Analytics Cloud - Wave Analytics Platform permission set license to use Wave. 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. Monitoring dataflow and system jobs in the data monitor. Use discretion when assigning this user permission because it enables the user to access all Salesforce object data to which the Integration User has access. For more information see Salesforce Data Access .
"Manage Wave Analytics"	Accessing all Wave Analytics features. Provides Wave administrator-level capabilities.

 **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 a user can perform 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.

Identify Wave Analytics Platform User Types

Identifying types of Wave users and the tasks they need to perform helps assure that the setup process meets the majority of your team members' analytics needs.

The Wave Analytics Platform license includes the permissions users in your organization need to administer Wave as well as to import, work with, and explore data. To set up Wave in the most useful way, think through the different ways people need to access Wave features and categorize them into user types. Here are a few examples:

In most cases, you'll need to set up Wave to meet the needs of two basic types of people on your team:

- Users, who need to be able to view Wave dashboards, lenses, and datasets and occasionally import data to Wave.
- Administrators or managers, who need access to all of Wave functionality to create apps, dashboards, lenses, and datasets, edit data, and otherwise customize the Wave experience.

Accordingly, when you set up the Wave platform, creating two basic permission sets can probably meet the needs of most users in your organization:

EDITIONS

Available in: **Salesforce Classic and Lightning Experience**

Available in: **Developer Edition**

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

- 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 may need to create a “superuser” permission set who need capabilities to create and share apps and dashboards or a “data wizard” permission set for users who want 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 show you how to create and assign in our detailed setup instructions.

Enable Wave Analytics and Select Analytics Cloud - Wave Analytics Platform Permission Set License

After Salesforce provisions you with the Wave Analytics platform license, you need to enable Wave and select the Wave Analytics permission set license and assign it to users.

1. In the Salesforce Setup menu, under administer, click **Analytics Cloud | Getting Started**.
2. Click **Enable Analytics**.
3. In the Setup menu, click **Manage Users | Users**.
4. 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.
5. Position the cursor over **Permission Set License Assignments** and click **Edit Assignments**.
6. Select the Analytics Cloud - Wave Analytics Platform permission set license and click **Save**.
7. Repeat the above steps for all users in your organization who require access to Wave.


 **Note:** No data is available for queries until you load data into Wave Analytics.

Create Wave Analytics Permission Sets

To give users in your organization access to Wave features, create and assign one or more permission sets based on the level of analytics capabilities they need.

After assigning permission set licenses to users, the next step in the setup process is to create permission sets made up of 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 permission sets:

- *Manage Wave Analytics* permission set, for users who administer and manage Wave Analytics and need access to all Wave Analytics features.
- *View Wave Analytics* permission set, for users who explore data and occasionally upload new data to Wave Analytics and need access to a limited set of Wave Analytics features.

 **Note:** Customers who purchased Wave before October 20, 2015: See [Set up the Wave Analytics Platform With Licenses Purchased Before October 20, 2015](#) before following the instructions here.




[Walk Through It: create, edit, and assign a permission set](#)


USER PERMISSIONS

To create permission sets:


- “Manage Profiles and Permission Sets”

1.  **Note:** 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 new Wave Analytics platform license, follow these steps to create a permission set with the same functionality.


Create Manage Wave Analytics permission set, which gives users access to all Wave features.

- a. In the Setup menu, under administer, click **Manage Users | Permission Sets** and then click **New**.
- b. Enter `Manage Wave Analytics` in the Label field. This automatically creates the API name, as well, which you can change if you'd like. Note that the API name can't include spaces, end with an underscore, or have two consecutive underscores.
- c.  **Warning:** Follow this step precisely or some users in your organization may not be able to access Wave.

In the User License field, keep the default value `--None--`. Do *not* select a user license.
- d. Click **Save**. The permission set details page shows the new permission set. Now add user permissions to the set.
- e. Scroll to the bottom of the permission set details page, and click **System Permissions**. Then click **Edit**.
- f. Select "Manage Wave Analytics" user permissions and click **Save**. You've successfully created a permission set that enables a user to access all Wave features. You can now assign it to users, which is covered in Assign Permission Sets to Users. You should assign this permission set sparingly because it provides access to all Wave features, many of which are inappropriate for most analytics users.

2.  **Note:** 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 new Wave Analytics platform license, follow these steps to create a permission set with the same functionality.


Create View Wave Analytics permission set, which enables users to view Wave lenses, dashboards, and datasets and import data to Wave.

- a. In the Setup menu, under administer, click **Manage Users | Permission Sets** and then click **New**.
- b. Enter `View Wave Analytics` in the Label field. This automatically creates the API name, as well, which you can change if you'd like. Note that the API name can't include spaces, end with an underscore, or have two consecutive underscores.
- c.  **Warning:** Follow this step precisely or some users in your organization may not be able to access Wave.

In the User License field, keep the default value `--None--`. Do *not* select a user license.
- d. Click **Save**. The permission set details page shows the new permission set. Now add user permissions to the set.
- e. Scroll to the bottom of the permission set details page, and click **System Permissions**. Then click **Edit**.
- f. 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 upload data files from outside Salesforce. You can now assign it to users, which is covered in Assign Permission Sets to Users.

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 the Wave functionality.

1.  [Walk Through It: assign a permission set](#)
2. In the Setup menu, under administer, click **Manage Users | Permission Sets**.

USER PERMISSIONS

To assign a permission set to users:

- "Assign Permission Sets"

3.  **Warning:** Assign the Manage Wave Analytics permission set sparingly, since it provides access to all Wave features. It should only be assigned to users who administer or manage the Wave platform.

Click the permission set you want to assign to users. If you've followed the instructions in [Create Wave Analytics Permission Sets](#) on page 7, select either the Manage Wave Analytics or View Wave Analytics permission sets.

4. Click **Manage Assignments** and click **Add Assignments**.
5. Select the users who need access to Wave, and click **Done**.

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.

[Grant Users Access to the Flex Dashboard Designer](#)

To grant users access to the flex dashboard designer, enable the feature and assign the "Create and Edit Wave Analytics Dashboards" user permission to the users. If you assign the permission but don't enable the feature, the users can't access the new designer, but they can access the old one.

[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 for iOS](#)

Install the Analytics for iOS Connected App to allow your mobile clients to easily connect to your org. The Connected App gives you control over who's logging in and how your mobile clients share images and links.

[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](#)

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.

Enable Preview Thumbnails for All Lenses and Dashboards

Make it easier for users to find their lenses and dashboards on the home page and on app pages. Replace the generic icons with preview thumbnails of the contents of all lenses and dashboards, including those assets that have row-level security restrictions.

If no security predicate is defined for a dataset, then preview thumbnails always appear. If a security predicate is defined, then preview thumbnails don't show unless this setting is enabled. With the setting enabled, preview thumbnails appear for all assets.

Important: Even if row-level security applies to the dataset used by a lens or dashboard, the preview thumbnail could expose data from restricted rows. Row-level security restrictions take effect only after you've clicked through to the lens or dashboard.

1. From Setup, enter *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**.

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. Set up your Salesforce org as described in the [Wave Platform Setup](#) section.
2. From Setup, enter *Wave Analytics* in the Quick Find box, then select **Settings**. Select `Enable Wave Analytics for Communities`.
3. Assign your community members the Analytics Cloud - Wave Community Users permission set license.
4. Create a permission set that includes the "View Wave Analytics on Communities Pages" permission and assign it to your community members.
5. In Wave, create dashboards and save them in a Wave app. Designate that app for sharing with the community.
6. Embed your Wave dashboards in Visualforce pages on your community site. For more information, see [Add a Wave Analytics Dashboard to a Visualforce Page](#).
7. From the app containing your dashboards, give access to community members by selecting **Share**.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To modify settings:

- "Manage Wave Analytics"

EDITIONS

Available in: Salesforce Classic

Available in: **Developer Edition**

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

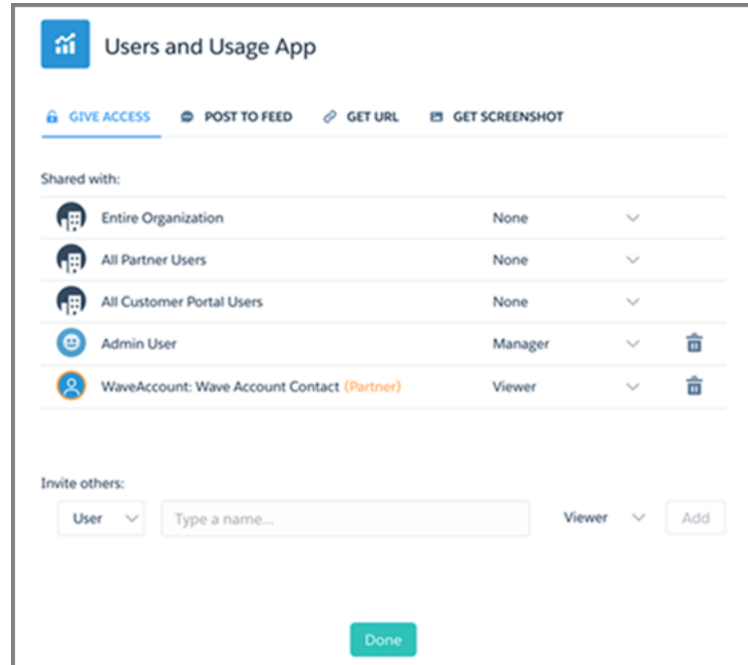
USER PERMISSIONS

To modify settings:

- "Manage Wave Analytics"

To view Wave dashboards and lenses:

- "View Wave Analytics on Communities Pages"



8. Invite community partners and customers to share the app. They appear highlighted in orange to distinguish them as external users, and Viewer is the only access available for them.

For information about building Communities, see [Getting Started With Communities](#).

Grant Users Access to the Flex Dashboard Designer

To grant users access to the flex dashboard designer, enable the feature and assign the “Create and Edit Wave Analytics Dashboards” user permission to the users. If you assign the permission but don’t enable the feature, the users can’t access the new designer, but they can access the old one.

1. From Setup, enter *Wave Analytics* in the Quick Find box, then select **Settings**.
2. Select **Enable Flex Dashboard Designer (Beta)**, 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 old or new designer for each dashboard.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To access the old or new designer:

- “Create and Edit Wave Analytics Dashboards”

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**.

Connected App for Wave Analytics for iOS

Install the Analytics for iOS Connected App to allow your mobile clients to easily connect to your org. The Connected App 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**, open the Salesforce Insights for iOS package.
4. 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.

Attribute Key	Mobile clients can share
SHARE_IMAGE_AIRDROP	Images via AirDrop
SHARE_IMAGE_CAMERAROLL	Images from Camera Roll
SHARE_IMAGE_OPEN	The image that's currently open on the mobile device
SHARE_IMAGE_S1	Images in Salesforce1

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To modify settings:

- "Manage Wave Analytics"

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To install and configure Connected App:


- "Manage Wave Analytics"

Attribute Key	Mobile clients can share
SHARE_IMAGE_EMAIL	Images via email
SHARE_LINK_AIRDROP	Links via AirDrop
SHARE_LINK_CLIPBOARD	The link that's currently in the clipboard of the mobile device
SHARE_LINK_S1	Links in Salesforce1
SHARE_LINK_EMAIL	Links via email
SPOTLIGHT_SEARCH	Mobile clients can see Wave Analytics assets in Spotlight search results

 **Note:** To apply custom attributes to connected users, you can revoke them and ask them to reconnect.

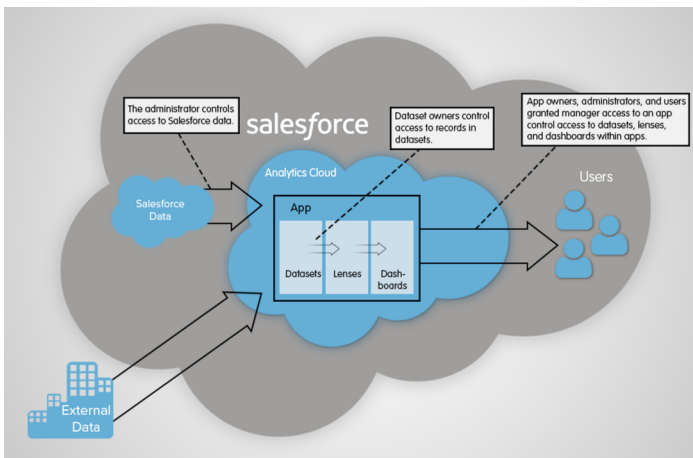
- To view and control how mobile clients connect to your Wave-enabled org, from Setup, enter *Connected Apps* in the *Quick Find* box, then select **Connect Apps OAuth Usage**.

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.

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 its 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

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, you might choose to restrict 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.

Wave Analytics uses the permissions of the Security User to access the User object and its fields when a user queries a dataset that has row-level security based on the User object. 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 you must 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 a user tries to query the dataset on which the predicate is applied.

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.

SEE ALSO:


[Control Access to Salesforce Objects and Fields](#)

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 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 platform for users in your organization or you can migrate to the new single-user Wave Analytics platform license.

-  **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](#).

! **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 queries 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.

 **Note:** When you create permission sets for Wave users, selecting any of the other Wave Analytics permissions automatically enables the “Wave Analytics” 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.

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 4. 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 App 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 Quota dataset so the app includes quotas data for your team.

When you create the Sales Wave app, 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, 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](#), which covers the steps described here.

EDITIONS

Available in: Salesforce Classic and Lightning Experience


Available in: **Developer Edition**


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

USER PERMISSIONS

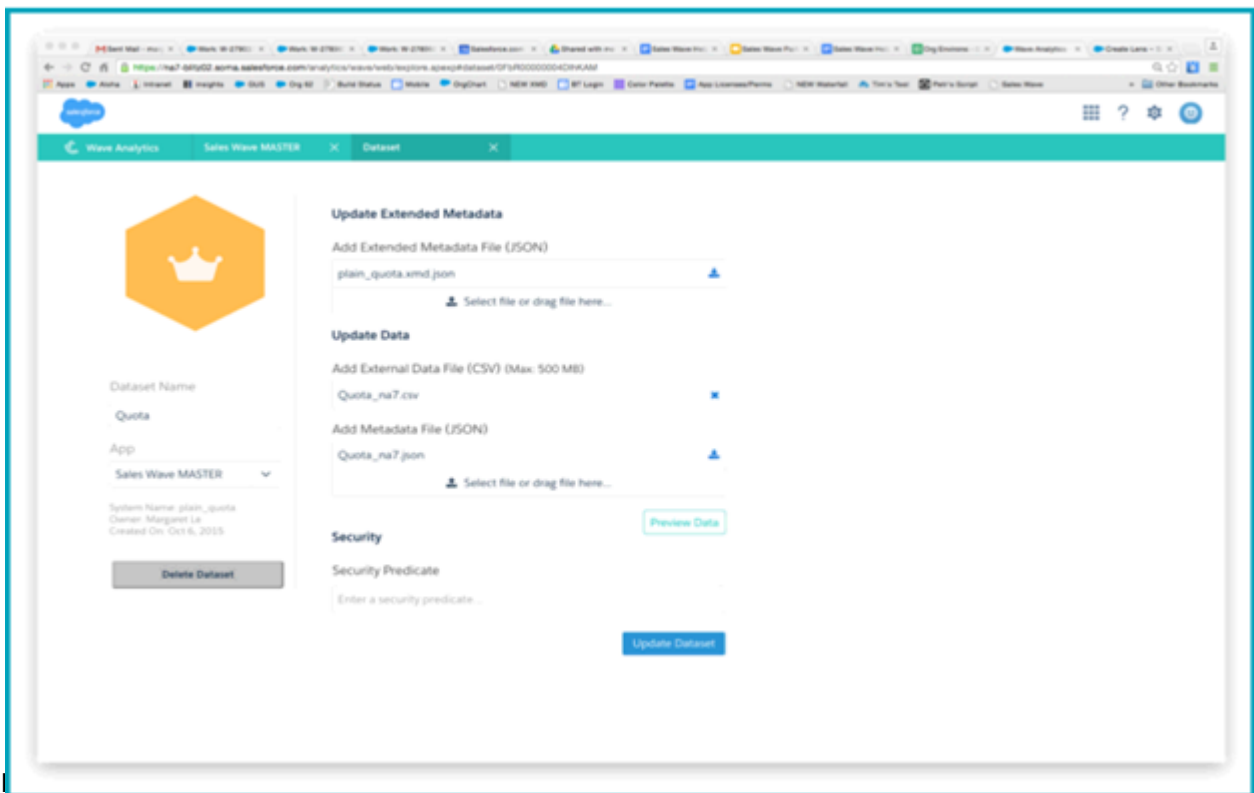
To create and manage Wave apps:

- “Manage Wave Analytics Templated Apps”
- “Edit Wave Analytics Dataflows”

 **Note:** When you work with .CSV files you want to import to Sales Wave, 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

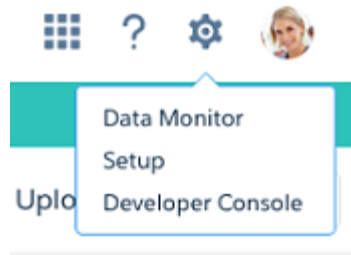
 **Note:** If your business doesn't track quotas at the user level, you can skip this step. You only need to complete this if you answer "yes" to the [configuration wizard](#) question about whether your business tracks quotas at the user level.

1. Create a .CSV file to include the following fields, in this order: QuotaAmount, StartDate (in yyyy-mm-dd format), OwnerName, and Username. For an example, see [Sales Wave App 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**

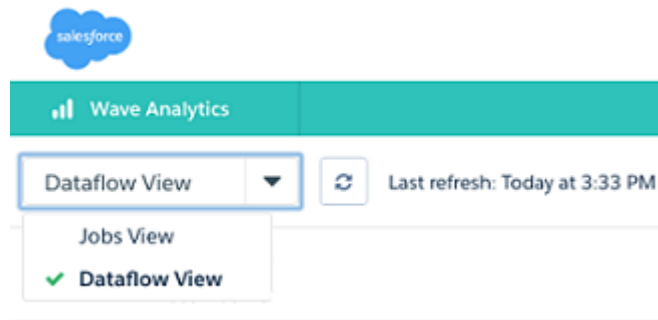


6. In the dialog box that opens, navigate to the .CSV file you created in Step 1, and double-click it.
7. Click **Update Dataset**.
8. If your fiscal period is different than calendar period, that is if it starts another date besides January 1, you also need to update the Quota Metadata file.
 - a. Copy the JSON from [Sales Wave 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 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**.

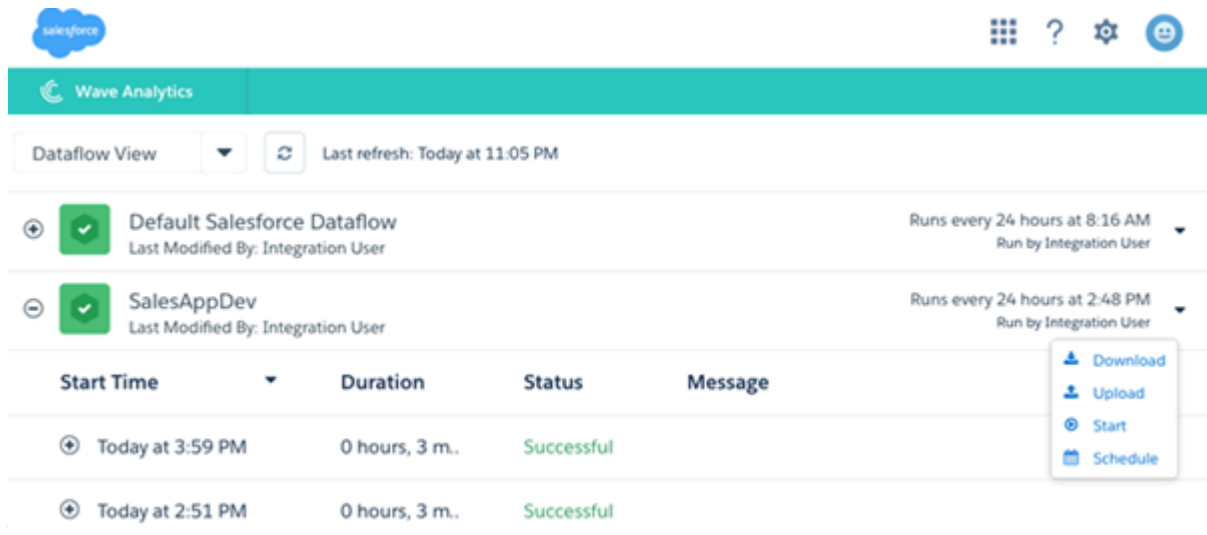
9. After uploading your quota data, re-run the dataflow to update the dashboards.
 - a. Click the Gear menu at the upper right of the Sales Wave screen and select Data Monitor.



- 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 [Keep Sales Wave App Data Up-to-Date by Scheduling a Dataflow](#).

Sales Wave App Example .CSV File

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

See [Update the Sales Wave App Quota \(Target\) Dataset](#) on page 205.

 **Example:**

```
QuotaAmount,StartDate,OwnerName,Username
5000000,2014-01-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-02-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-03-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-04-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-05-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-06-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-07-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-08-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-09-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-10-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-11-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-12-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-01-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-02-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-03-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-04-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-05-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-06-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-07-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-08-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-09-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-10-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-11-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-12-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
```

Sales Wave Quota Dataset JSON File

Use this file to update the Sales Wave 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 Sales Wave 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 following the instructions in [Update the Sales Wave App 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 (programmatically 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](#)

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.

[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.

[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](#)

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.”

[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 User Interface for External Data](#)

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.


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	Text	Number of seconds. If the date contains no seconds, value is '0.'
<date field name>_Minute	Text	Number of minutes. If the date contains no minutes, value is '0.'
<date field name>_Hour	Text	Number of hours. If the date contains no hours, value is '0.'
<date field name>_Day	Text	Day of the month.
<date field name>_Week	Text	Week number in calendar year.
<date field name>_Month	Text	Month number in calendar year.
<date field name>_Quarter	Text	Quarter number in calendar year.
<date field name>_Year	Text	Calendar year.
<date field name>_Week_Fiscal	Text	Week number in fiscal year.
<date field name>_Month_Fiscal	Text	Month number in fiscal year.
<date field name>_Quarter_Fiscal	Text	Quarter number in fiscal year.
<date field name>_Year_Fiscal	Text	Fiscal year.
<date field name>_sec_epoch	Numeric	Number of seconds that have elapsed since January 1, 1970 (midnight UTC/GMT).
<date field name>_day_epoch	Numeric	Number of days that have elapsed since January 1, 1970 (midnight UTC/GMT).

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 [sfcdigest transformation parameters](#) for Salesforce data or in the [metadata file](#) for external data.

 **Note:** 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:

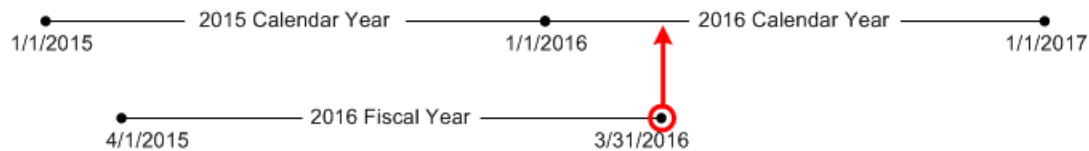
fiscalMonthOffset

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'.

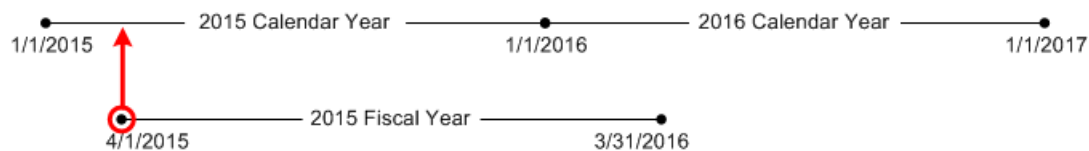
isYearEndFiscalYear

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..

[Wave Connector Setup for Administrators](#)

In some cases, administrators need to change Salesforce security settings so users can connect to the Wave Connector app.

SEE ALSO:

[Install the Wave Connector Excel App](#)

Upload User Interface for External Data

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.

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

5. [Reschedule the Dataflow](#)

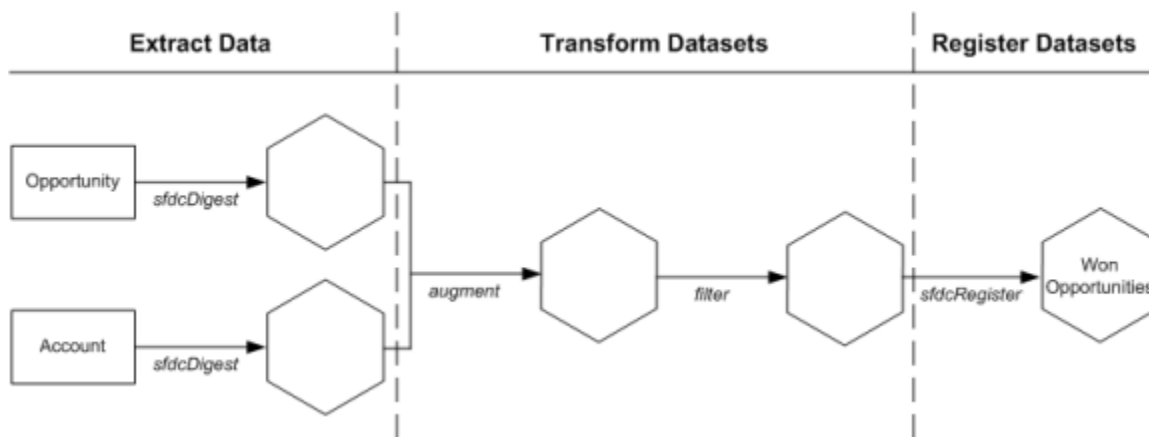
The dataflow job runs on a daily schedule. You can change the time that the dataflow job runs. You might change the time to ensure that the data is available by a particular time or to run the job during non-business hours.

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.

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 Monitor** to open the data monitor.
The Jobs view of the data monitor appears 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

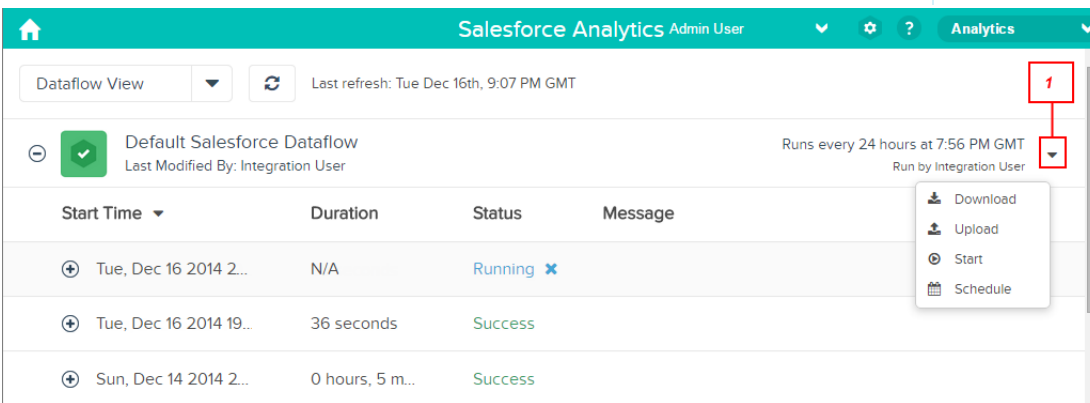
Available in: **Developer Edition**

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

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 transformations to the dataflow definition file.
For example, based on the design in the [previous step](#), you can add the following transformations:

```
{
  "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": "sfcdcDigest",
    "parameters": {
        "object": "Account",
        "fields": [
            { "name": "Id" },
            { "name": "Name" }
        ]
    }
},
"Transform_Augment_OpportunitiesWithAccountDetails": {
    "action": "augment",
    "parameters": {
        "left": "Extract_Opportunities",
        "left_key": [ "AccountId" ],
        "relationship": "OpptyAcct",
        "right": "Extract_AccountDetails",
        "right_key": [ "Id" ],
        "right_select": [
            "Name"
        ]
    }
},
"Transform_Filter_Opportunities": {
    "action": "filter",
    "parameters": {
        "filter": "StageName:EQ:Closed Won",
        "source": "Transform_Augment_OpportunitiesWithAccountDetails"
    }
},
"Register_Dataset_WonOpportunities": {
    "action": "sfcdcRegister",
    "parameters": {
        "alias": "WonOpportunities",
        "name": "WonOpportunities",
        "source": "Transform_Filter_Opportunities"
    }
}
}

```

 **Note:** The JSON keys and values are case-sensitive. Each bolded key in the previous example JSON contains one action, 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.


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 data monitor, 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.

Start the 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 Monitor** to open the data monitor.
The Jobs view of the data monitor appears by default.
2. Select **Dataflow View**.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

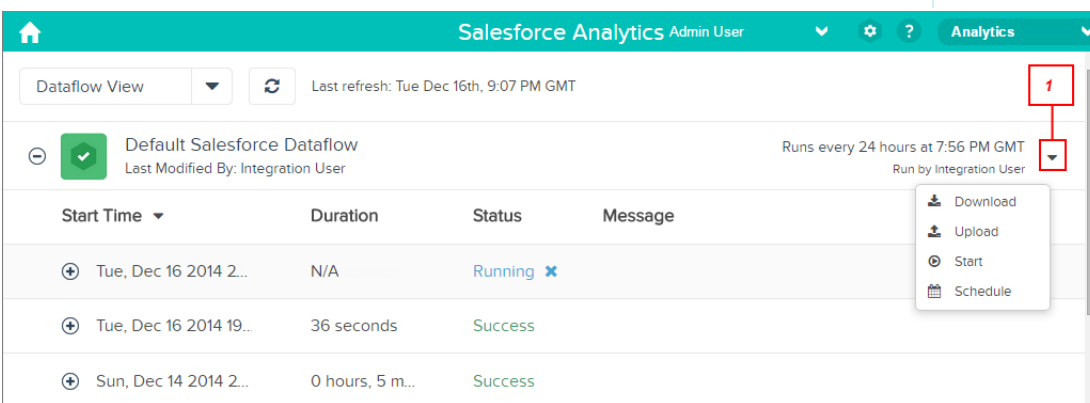
Available in: **Developer Edition**

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

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 greyed out while the dataflow job runs.

- After the job completes, Wave Analytics sends an email notification to the user who last modified the dataflow definition file. 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.

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

The Dataflow view of the data 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 and view the run-time details about every transformation that is processed.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To access the data monitor:



- “Edit Wave Analytics Dataflows,” “Upload External Data to Wave Analytics,” or “Manage Wave Analytics”

The screenshot displays the 'Salesforce Analytics' interface. At the top, there is a 'Dataflow View' dropdown menu (1) and a refresh button (2). Below this, a dataflow job named 'SalesEdge ELT Workflow' is shown, which runs every 24 hours at 6:59 PM GMT. A table lists the job's history:

Start Time	Duration	Status	Message
Thu, Nov 13 2014 22:46:06 GMT	9 seconds	Failed	Killed by: cc@na2.insights
Thu, Nov 13 2014 19:07:24 GMT	35 seconds	Success	

Below the main table, a detailed view of a node is shown, with a red box (3) highlighting the 'Start Time' column. The detailed view table is as follows:

Node Name	Start Time	Duration	Node Type	Status	Input Rows Processed
102	Thu Nov 13 19:07:25 GMT 2014	0 hours, 0 minutes	sfdcDigest	success	61
103	Thu Nov 13 19:07:38 GMT 2014	0 hours, 0 minutes	sfdcDigest	success	23
104	Thu Nov 13 19:07:50 GMT 2014	0 hours, 0 minutes	flatten	success	23


1. In Wave Analytics, click the gear button (), and then click **Data Monitor** to open the data monitor.
The Jobs view shows by default.
2. Select **Dataflow View** (1).
3. Click the Refresh Jobs button () 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.

4. If the dataflow job fails, expand the job node (3) and review the run-time details for every transformation that was processed.
5. If there's a problem with the dataflow logic, edit the dataflow definition file, upload it, and then run the dataflow again.

Reschedule the Dataflow

The dataflow job runs on a daily schedule. You can change the time that the dataflow job runs. You might change the time to ensure that the data is available by a particular time or to run the job during non-business hours.

1. In Wave Analytics, click the gear icon () and then click **Data Monitor** to open the data monitor.
The Jobs view shows by default.
2. Select **Dataflow View**.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

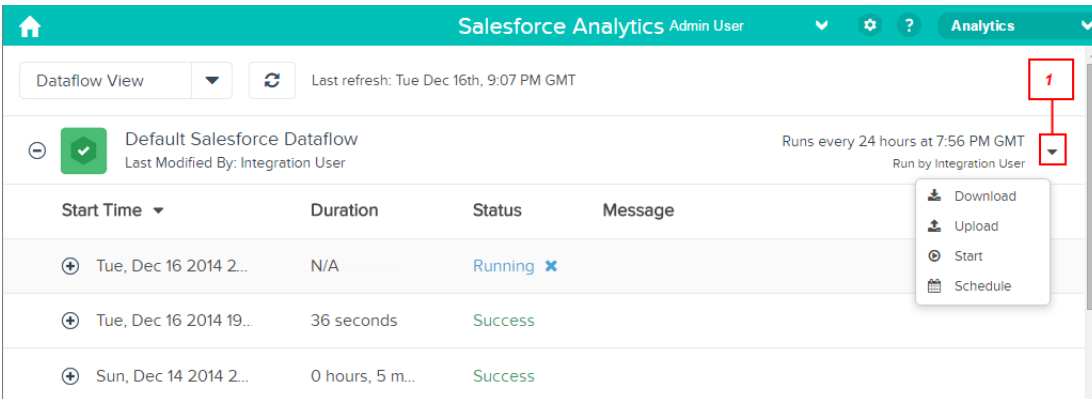
Available in: **Developer Edition**

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

USER PERMISSIONS

To start a dataflow job:

- "Edit Wave Analytics Dataflows"



3. Click **Schedule** in the actions list (1) to reschedule the dataflow job. The Dataflow Schedule dialog box appears.
4. Select the time that you want the dataflow to run.
5. Click **Save**.
6. Click **Done**.

Transformations

A *transformation* refers to the manipulation of data. You can add transformations to a dataflow to extract data from Salesforce objects or datasets, transform datasets that contain Salesforce or external data, and register datasets.

For example, you can use transformations to join data from two related datasets and then register the resulting dataset to make it available for queries.

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.

[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.

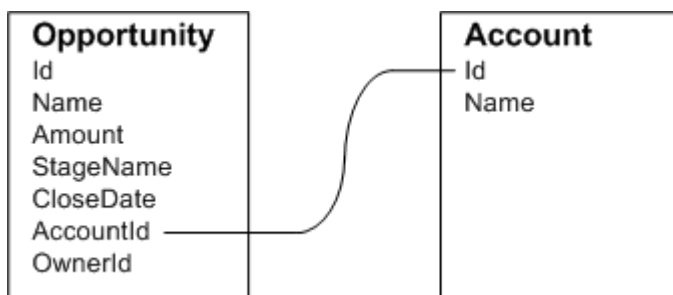
When you create the transformation, you identify each input dataset as the left or right dataset and specify the relationship between them. Wave Analytics combines all the columns of the left dataset with only the specified columns from the right dataset. (Keep in mind that each dataset can't have more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

```
"left_key": [ "Id" ], "right_key": [ "Prod_ID" ]
```

A key can be a single-column key or a composite key. For a match condition based on a composite key, the keys for both datasets must have the same number of columns, specified in the same order.

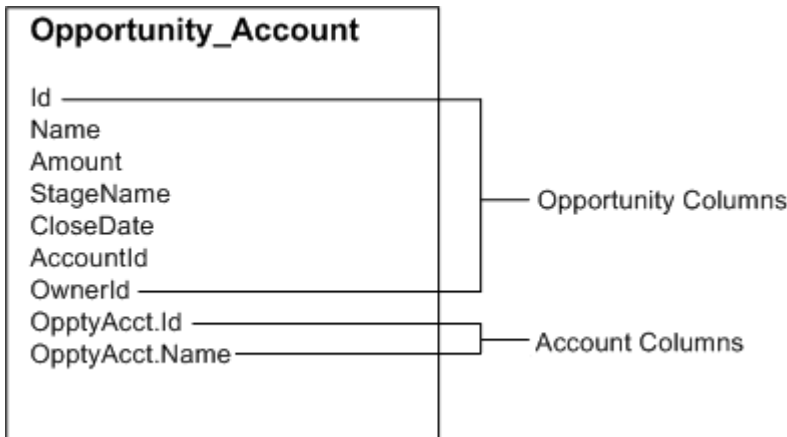
-  **Tip:** To augment more than two datasets, augment two datasets at a time. For example, to augment three datasets, augment the first two datasets, and then augment the resulting dataset with the third dataset.
-  **Example:** Let's look at an example of the augment transformation. In this example, you want to extract data from the Opportunity and Accounts objects, and then match the data based on the account ID field.



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.

Product		Price		
Id	Name	ProdId	Pricebook	UnitPrice
Prod1	Table	Prod1	Standard	1000
Prod2	Chair	Prod2	Standard	450
<i>null</i>	Bench	<i>null</i>	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.

Product		Price		
Id	Name	ProdId	Pricebook	UnitPrice
Prod1	Table	Prod1	Standard	1000
Prod2	Chair	Prod2	Standard	450
""	Bench	""	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.

Product		Price		
Id	Name	ProdId	Pricebook	UnitPrice
Prod1	Table	Prod1	Standard	1000
Prod2	Chair	Prod2	Standard	450
Prod2	Chair	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.

Product		Price		
Id	Name	ProdId	Pricebook	UnitPrice
Prod1	Table	Prod4	Standard	1000
Prod2	Chair	Prod5	Standard	450
Prod3	Bench			

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		Price		
Id	Name	ProdId	Pricebook	UnitPrice
Prod1	Table	Prod1	Standard	1000
Prod2	Chair	Prod2	Standard	450
Prod3	Bench	Prod3	Custom	800
		Prod3	Standard	700

Although there are multiple rows for Prod3 in the Price dataset, 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		Price		
Id	Name	ProdId	Pricebook	UnitPrice
Prod1	Table	Prod1	Standard	1000
Prod2	Chair	Prod2	Standard	450
Prod3	Bench	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", "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
<code>operation</code>	No	<p>Indicates what the transformation does if it matches multiple rows in the right dataset with a row in the left. Valid values:</p> <ul style="list-style-type: none"> <code>LookupSingleValue</code>. Returns values from one of the matched rows. If you don't specify the <code>operation</code> parameter, the transformation uses this operation.


Parameter	Required?	Value
		<ul style="list-style-type: none"> LookupMultiValue. Returns values from all matched rows. <p>For more information about each operation, see Special Cases for Matching Records with the augment Transformation.</p>
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	<p>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:</p> <pre>["Key Column1", "Key Column2", ..., "Key ColumnN"]</pre> <p> Note: The left or right key can't be a multi-value field.</p>
right	Yes	Node in the dataflow definition file that identifies the right dataset. This is one of two input sources for this transformation.
relationship	Yes	Relationship between the left and right datasets. The dataflow adds the relationship to the beginning of the right column names in the output dataset to make the column names unique and descriptive.
right_select	Yes	An array of column names from the right dataset that you want to include in the output dataset. The dataflow adds the relationship as a prefix to the column name to determine the name of the right column in the output dataset.
right_key	Yes	Key column in the right dataset used to augment the datasets. If you use a composite key, the left and right keys must have the same number of columns in the same order.

SEE ALSO:

[augment Transformation](#)

computeExpression Transformation


The `computeExpression` transformation enables you to add derived fields to a dataset. The values for derived fields aren't extracted from the input data source. Instead, 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:

```
"CreateDerivedFields": {
  "action": "computeExpression",
  "parameters": {
    "source": "sourceNode",
    "mergeWithSource": false,
    "computedFields": [
      {
        "name": "Derived_A",
        "type": "Text",
        "label": "Derived Field A",
        "saqlExpression": "\"hello \""},
      {
        "name": "Derived_B",
        "type": "Text",
        "label": "Derived Field B Dependent on Field A",
        "saqlExpression": "Derived_A + \"world\""
      }
    ]
  }
}
```

- You can choose whether the resulting dataset includes only the derived fields, or includes the input and derived fields.
-  **Example:** Let's look at an example. You want to create a dataset based on Salesforce opportunity data. You create a dataflow that extracts the `Id` and `Amount` fields from the `Opportunity` object. In addition, you also want to add the following derived fields to the dataset: `ModifiedId`, `SalesTax`, `FinalPrice`, and `ValueCategory`. For the derived fields, you will:
 - Append "SFDC" to each opportunity `Id` to get a new modified `Id`.
 - Calculate the sales tax based on an 8% tax rate.
 - Calculate the final price by adding the amount and sales tax.

- Categorize opportunities into low-, medium-, and high-value buckets based on the calculated final price.

You create the following dataflow definition.

```
{
  "salesData": {
    "action": "sfcdigest",
    "parameters": {
      "object": "Opportunity",
      "fields": [
        {"name": "Amount"},
        {"name": "Id"}]
    }
  },
  "Derived_Fields": {
    "action": "computeExpression",
    "parameters": {
      "source": "salesData",
      "mergeWithSource": true,
      "computedFields": [
        {
          "name": "ModifiedId",
          "type": "Text",
          "sqlExpression": "\"SFDC\" + Id",
        },
        {
          "name": "SalesTax",
          "type": "Numeric",
          "precision": 18,
          "defaultValue": "0",
          "scale": 5,
          "sqlExpression": "Amount * 0.08",
        },
        {
          "name": "FinalPrice",
          "type": "Numeric",
          "precision": 18,
          "defaultValue": "0",
          "scale": 5,
          "sqlExpression": "Amount + SalesTax",
        },
        {
          "name": "ValueCategory",
          "type": "Text",
          "sqlExpression": "
            case
              when FinalPrice < 1000 then \"Low\"
              when FinalPrice >= 1000 and FinalPrice < 2000 then \"Medium\"
              else \"High\"
            end"
        }
      ]
    }
  },
  "Register_CategorizedSales": {
    "action": "sfcdRegister",
    "parameters": {
      "alias": "Categorized_Sales",
      "name": "Categorized_Sales",
      "source": "Derived_Fields"
    }
  }
}
```

```
}
}
```

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
<code>source</code>	Yes	Node in the dataflow definition file that identifies the input source for this transformation.
<code>mergeWithSource</code>	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.
<code>computedFields</code>	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: <pre>"type": "Text"</pre>
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.
sqlExpression	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: <pre>"sqlExpression": "toDate (birth_day, \ "yyyy-M-d\ ") "</pre>
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. Example: <pre>"precision": 10</pre>
scale	Yes (for Numeric fields only)	The number of digits to the right of the decimal point in a numeric value. Must be


Parameter	Required?	Value
		less than the precision value. Value must be between 0 and 17 characters, inclusive. Example: <pre>"scale": 2</pre>
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 an `sfdcDigest` 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.

Wave processes the `computeRelative` transformation differently than other transformations. Wave merges the logic from this transformation into the associated `sfdcDigest` transformation. As a result, you don't have to register the `computeRelative` transformation to add the derived fields to the final dataset. Instead, register only `sfdcDigest`—the registered dataset has fields from both transformations. For example, the `sfdcDigest` transformation extracts data from three fields (Id, CloseDate, and Amount). The `computeRelative` transformation generates two derived fields (PreviousCloseDate and PreviousAmount). The resulting dataset contains all five columns.

Consider the following guidelines when creating a `computeRelative` transformation.

- This transformation supports only the `sfdcDigest` node as input—you can't use any other transformation node as input.
- Specify this transformation directly after the associated input `sfdcDigest` transformation in the dataflow JSON.
- You can use this transformation to analyze trends in Salesforce data, not external data.
- You can use this transformation in a replication dataflow. For more information about replication, see [Getting Started with Wave Replication Tipsheet](#).

 **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
00630000003SZA4AA	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.

OpportunityCreatedDate

Determines the date that the opportunity was first created. You can use this date with the actual close date to determine the number of days required to close the sale. The goal is to shorten the sales cycle to recognize revenue.

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 apply sfdcRegister on sfdcDigest to get the fields from both transformations.

You create the following dataflow definition.

```
{
  "extractOppHistory": {
    "action": "sfdcDigest",
    "parameters": {
      "object": "OpportunityHistory",
      "fields": [
        {"name": "OpportunityId"},
        {"name": "CreatedDate"},
        {"name": "StageName"},
        {"name": "Amount"},
        {"name": "CloseDate"}
      ]
    }
  }
}
```

```

    }
  },
  "computeTrending": {
    "action": "computeRelative",
    "parameters": {
      "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": "extractOppHistory"
  }
}

```

The dataflow runs and creates the following dataset.

Sales Pipeline Trending Dataset								
OpportunityId	CreatedDate	StageName	Amount	CloseDate	OpportunityCreatedDate	ValidToDate	AmountPrev	CloseDatePrev
00630000003Jh6AAK	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
00630000003Jh6AAK	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
00630000003Jh6AAK	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
00630000003SYADA44	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
00630000003SZA444	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	sfdcDigest 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 and only one partition-by field.
orderBy	Yes	Field used to sort the records within each partition and the sort order: ascending (asc) or descending (desc). Specify one and only one sort field.

Parameter	Required?	Value
		<p>Example:</p> <pre>"orderBy": [{ "name": "CreatedDate", "direction": "asc" }]</pre>
computedFields	Yes	<p>A list of definitions for derived fields.</p> <p>Example:</p> <pre>"computedFields": [{ "name": "OpportunityCreatedDate", "expression": { "sourceField": "CreatedDate", "offset": "first()", "default": "current()" } }, { "name": "AmountPrev", "expression": { "sourceField": "Amount", "offset": "previous()", "default": "0" } }]</pre> <p>See computedFields.</p>

computedFields


The following table describes the attributes in the `computedFields` section.


Parameter	Required?	Value
name	Yes	<p>API name of the derived field to add to the dataset. The name must be unique in the dataset.</p> <p> Note: If the name is not unique, the dataflow fails to run.</p>
expression	Yes	<p>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.</p>

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

expression

The following table describes the attributes in the `expression` section.

Parameter	Required?	Value
sourceField	Yes	API name of the input field from the <code>sfdcDigest</code> transformation that's used in the expression.
offset	Yes	<p>The function used in the expression. You can use the following functions:</p> <p>current () Gets the value from the current record. Example: <pre>"offset": "current ()"</pre></p> <p>first () Gets the value from the first record in the partition, like the first <code>CreateDate</code> for an opportunity. Example: <pre>"offset": "first ()"</pre></p> <p>next () Gets the value from the next record. Example: <pre>"offset": "next ()"</pre></p> <p>previous () Gets the value from the previous record. Example: <pre>"offset": "previous ()"</pre></p> <p> 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, <code>next ()</code> can't access the value of a derived field in the next row.</p>

Parameter	Required?	Value
		 Tip: To get the correct results when using the <code>previous()</code> and <code>next()</code> functions, the <code>computeRelative</code> 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 <code>current()</code> as a default value. Examples: <pre>"default": "3000-01-01T00:00:00.000Z"</pre> <pre>"default": "current()"</pre>


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	For example, if the date column is <code>CloseDate</code> , the generated epoch second column is <code>CloseDate_sec_epoch</code> . This column provides the number of seconds that have elapsed since January 1, 1970 (midnight UTC/GMT).
<date_column_name>_day_epoch	For example, if the date column is <code>CloseDate</code> , the generated epoch day column is <code>CloseDate_day_epoch</code> . 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": "sfcdigest",
    "parameters": {
      "object": "OpportunityHistory",
      "fields": [
        { "name": "OpportunityId" },
        { "name": "CloseDate" },
        { "name": "StageName" },
        { "name": "Amount" }
      ]
    }
  },
  "Calculate_Delta": {
    "action": "delta",
    "parameters": {
      "dimension": "OpportunityId",
      "epoch": "CloseDate_day_epoch",
      "inputMeasure": "Amount",
      "outputMeasure": "DeltaAmount",
      "source": "Extract_Opportunities"
    }
  },
  "Register_Dataset": {
    "action": "sfdcregister",
    "parameters": {
      "alias": "OppHistoryDelta",
      "name": "OppHistoryDelta",
      "source": "Calculate_Delta"
    }
  }
}
```


To calculate the delta values for each opportunity amount, the delta transformation sorts the records by the dimension (OpportunityId) first, and then by time (CloseDate_day_epoch) as shown here.

OpportunityID	CloseDate	StageName	Amount
1	1/1/2014	New	100
1	3/1/2014	ClosedWon	100
2	1/1/2014	New	100
2	2/1/2014	ClosedWon	200

After the records are sorted, for each dimension (OpportunityId), the transformation compares the previous value to the next value to determine the delta for each record. The transformation creates the following dataset.

OpportunityId	CloseDate	StageName	Amount	DeltaAmount
1	1/1/2014	New	100	0
1	3/1/2014	ClosedWon	100	0
2	1/1/2014	New	100	0
2	2/1/2014	ClosedWon	200	100

For the first record of each dimension, the transformation inserts '0' for the delta value.

 **Note:** If an opportunity contains multiple changes on the same day, you must sort the records on a shorter time interval. In this case, sort on CloseDate_sec_epoch column. Otherwise, records might not be sorted correctly, which means delta values will be incorrect.

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.



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": "OpportunityVersusFinalSales",
      "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": "sfcdcDigest",
    "parameters": {
      "object": "Opportunity",
      "fields": [
        { "name": "Id" },
        { "name": "Name" },
        { "name": "Amount" },
        { "name": "StageName" },
        { "name": "CloseDate" },
        { "name": "AccountId" },
        { "name": "OwnerId" }
      ]
    }
  },
  "Filter_Opportunities": {
    "action": "filter",
    "parameters": {
      "filter": "StageName:EQ:Closed Won",
      "source": "Extract_Opportunities"
    }
  }
}
```

```

    }
  },
  "Register_My_Won_Oppportunities_Dataset": {
    "action": "sfdcRegister",
    "parameters": {
      "alias": "MyWonOppportunities",
      "name": "MyWonOppportunities",
      "source": "Filter_Oppportunities"
    }
  }
}

```

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: <code>"filter": "StageName:EQ:Closed Won"</code>

Filter Expression Syntax	Description
dim:R:val0:val1	True if the left dimension falls within the specified range between val0 and val1. Example: <code>"filter": "EmployeeId:R:100:1000"</code>
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'). Example: <code>"filter": "LastName:R:Li"</code>
dim:R::val	True if the dimension is less than or equal to the value based on binary sort order. Example: <code>"filter": "LastName:R::Levy"</code>
dim:N:val	True if the dimension and value are not equal. Example: <code>"filter": "RoleName:N:Manager"</code>
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. Example: <code>"filter": "Lead Status:EQ:Open Contacted"</code>
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. Example: <code>"filter": "Lead Status:EQ:Qualified,Rating:EQ:Hot"</code>

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 node in the hierarchy. Each record contains all ancestors up the hierarchy chain for each node.

The flatten transformation stores the list of ancestors in two formats in the resulting dataset. 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	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

 **Example:** Let's look at an example. In this 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, you will 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"
  }
},
"Augment_User_FlattenUserRole": {
  "action": "augment",
  "parameters": {
    "left": "Extract_User",
    "left_key": [ "RoleId" ],
    "relationship": "Role",
    "right": "Flatten_UserRole",
    "right_key": [ "Id" ],
    "right_select": [
      "Id",
      "Name",
      "Roles",
      "RolePath"
    ]
  }
},
"Augment_Opportunity_UserWithRoles": {
  "action": "augment",
  "parameters": {
    "left": "Extract_Opportunity",
    "left_key": [ "OwnerId" ],
    "right": "Augment_User_FlattenUserRole",
    "relationship": "Owner",
    "right_select": [
      "Name",
      "Department",
      "Role.Id",
      "Role.Name",
      "Role.Roles",
      "Role.RolePath"
    ]
  }
}

```

```

    ],
    "right_key": [ "Id" ]
  }
},
"Register_OpportunityWithRoles_Dataset": {
  "action": "sfdcRegister",
  "parameters": {
    "alias": "OppRoles",
    "name": "OppRoles",
    "source": "Augment_Opportunity_UserWithRoles",
    "rowLevelSecurityFilter": "'Owner.Role.Roles' == \"\$User.UserRoleId\" || 'OwnerId'
== \"\$User.Id\""
  }
}
}
}

```

To flatten the Salesforce role hierarchy, the flatten transformation uses the following input fields from the UserRole object.

Id

Id identifies each node in the Salesforce role hierarchy.

ParentRoleId

ParentRoleId identifies the parent as defined in the role hierarchy.

After traversing through each parent-child relationship in the UserRole object, the flatten transformation generates one record for each role ID. Each record contains all ancestor roles for each role in the hierarchy. The flatten transformation generates two output columns—Roles and RolePath—to store all ancestor roles for each role.

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
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.

Parameter	Required?	Value
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: <code>Sales Manager 1\Regional Manager 1\Vice President 1\CEO</code> .
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"
  }
}
}
}

```

Tips for Using the sfdcDigest Transformation

- Be selective when choosing fields because a dataset can contain a maximum of 5,000 fields.
- To extract data from a Salesforce object, the sfdcDigest transformation runs a SOQL query. The length of the SOQL query cannot exceed 20,000 characters. No results are returned if the SOQL query exceeds the maximum. To reduce the SOQL query length, break 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 the half of the fields and create another sfdcDigest transformation to extract the remaining fields.

IN THIS SECTION:

[Extracting Incremental Changes to Data \(Pilot\)](#)

You can configure the sfdcDigest transformation to extract only records that changed in a Salesforce object since the last dataflow run. Use incremental extraction to decrease the time required to extract the data.

[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](#)


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.

Extracting Incremental Changes to Data (Pilot)

You can configure the `sfdcDigest` transformation to extract only records that changed in a Salesforce object since the last dataflow run. Use incremental extraction to decrease the time required to extract the data.

 **Note:** Incremental extraction with the `sfdcDigest` transformation is currently available through a pilot program. Any unreleased services or features referenced in this or other press releases or public statements are not currently available and may not be delivered on time or at all. Customers who purchase our services should make their purchase decisions based upon features that are currently available.

There are two types of extractions: full and incremental. With a full extraction, the `sfdcDigest` transformation extracts all records from the Salesforce object. With incremental extraction, the transformation extracts only records that have changed since the last dataflow run. Unless configured otherwise, the `sfdcDigest` transformation runs full extractions.

When you enable incremental extraction, the `sfdcDigest` transformation performs different actions on the dataset based the types of record changes in the Salesforce object since the last dataflow run:

Record Change in Salesforce Object	Change to Dataset
Deleted record	Deletes corresponding record in the dataset.
New record	Inserts the new record into the dataset.
Updated record	Updates the corresponding record in the dataset.

Consider the following guidelines when enabling incremental extraction for an `sfdcDigest` transformation:

- Use incremental extraction for as many instances of the `sfdcDigest` transformation in the dataflow as possible.
- If necessary, when the `sfdcDigest` transformation is configured to run an incremental extraction, you can use the `fullRefreshToken` parameter to run a one-time, full extraction. You might use this option if the data in the Salesforce object and dataset become out of sync.
- The `sfdcDigest` transformation extracts all records from the object if you enable incremental extraction and it's the first time running the dataflow.
- You must use a full extraction in the following cases because incremental extraction is not supported:
 - The Salesforce object contains a formula field that references an external data object.
 - The Salesforce object fields or field types changed since the last dataflow run.
 - The filter in the `sfdcDigest` transformation changed since the last dataflow run.
 - The Salesforce object is one of the following objects:
 - CallCenter
 - CaseTeamMember
 - CategoryNode
 - CollaborationGroup
 - CollaborationGroupMember
 - CollaborationGroupMemberRequest
 - Division
 - Domain
 - DomainSite
 - Group

- GroupMember
- Profile
- Site
- Territory
- Topic
- User
- UserRole
- UserTerritory



Example: Let's consider an example. You'd like to incrementally extract data from the Opportunity object. As a result, you configure the following dataflow definition:

```
{
  "Extract_Opportunities": {
    "action": "sfcdigest",
    "parameters": {
      "object": "Opportunity",
      "incremental": true,
      "fields": [
        { "name": "Id" },
        { "name": "Name" },
        { "name": "StageName" },
        { "name": "CloseDate" },
        { "name": "AccountId" },
        { "name": "OwnerId" }
      ]
    }
  },
  "Register_Opportunities_Dataset": {
    "action": "sfcdregister",
    "parameters": {
      "alias": "Opportunities",
      "name": "Opportunities",
      "source": "Extract_Opportunities"
    }
  }
}
```

Later, you realize that one of the opportunity records is missing. You decide to run a one-time full extraction by adding the `fullRefreshToken` parameter as shown in the following dataflow definition.

```
{
  "Extract_Opportunities": {
    "action": "sfcdigest",
    "parameters": {
      "object": "Opportunity",
      "incremental": true,
      "fullRefreshToken": "bb",
      "fields": [
        { "name": "Id" },
        { "name": "Name" },
        { "name": "StageName" },

```

```

        { "name": "CloseDate" },
        { "name": "AccountId" },
        { "name": "OwnerId" }
      ]
    }
  },
  "Register_Opportunities_Dataset": {
    "action": "sfdcRegister",
    "parameters": {
      "alias": "Opportunities",
      "name": "Opportunities",
      "source": "Extract_Opportunities"
    }
  }
}

```

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.

SEE ALSO:

[sfdcDigest Transformation](#)

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
=	<p>Filter condition is true if the value in the field equals the specified value. String comparisons using the equals operator are case-insensitive.</p> <p>Example:</p> <pre>"filterConditions": [{ "field": "OwnerId", "operator": "=", "value": "a07B00000012HYu" }]</pre>
!=	<p>Filter condition is true if the value in the field does not equal the specified value.</p> <p>Example (using backslashes to escape double quotes in a string value):</p> <pre>"filterConditions": [{ "field": "Nickname__c", "operator": "!=", "value": "\"Sammy\"" }]</pre>

Operator	Comment
>	<p>Filter condition is true if the value in the field is greater than the specified value.</p> <p>Example:</p> <pre data-bbox="818 401 1446 632">"filterConditions": [{ "field": "Amount", "operator": ">", "value": "100000" }]</pre>
<	<p>Filter condition is true if the value in the field is less than the specified value.</p> <p>Example (using a date literal):</p> <pre data-bbox="818 800 1446 1066">"filterConditions": [{ "field": "CloseDate", "operator": "<", "value": "THIS_MONTH", "isQuoted": false }]</pre>
>=	<p>Filter condition is true if the value in the field is greater than or equal to the specified value.</p> <p>Example:</p> <pre data-bbox="818 1241 1446 1476">"filterConditions": [{ "field": "Amount", "operator": ">=", "value": "100000" }]</pre>
<=	<p>Filter condition is true if the value in the field is less than or equal to the specified value.</p> <p>Example (using a SOQL function):</p> <pre data-bbox="818 1650 1446 1843">"filterConditions": [{ "field": "CALENDAR_YEAR (CreatedDate)", "operator": "<=", "value": "2015", "isQuoted": true }]</pre>

Operator	Comment
	<pre data-bbox="824 260 1435 331"> }]</pre>
LIKE	<p data-bbox="824 382 1435 508">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.</p> <ul data-bbox="824 529 1435 802" style="list-style-type: none"> • 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. <p data-bbox="824 823 909 844">Example:</p> <pre data-bbox="824 865 1435 1096"> "filterConditions": [{ "field": "FirstName", "operator": "LIKE", "value": "Chris%" }]</pre>
IN	<p data-bbox="824 1150 1435 1243">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 <code>isQuoted</code> to true.</p> <p data-bbox="824 1264 909 1285">Example:</p> <pre data-bbox="824 1306 1435 1537"> "filterConditions": [{ "field": "StageName", "operator": "IN", "value": ["Closed Won", "Closed Lost"] }]</pre>
NOT IN	<p data-bbox="824 1587 1435 1650">Filter condition is true if the value in the field does not equal any of the values in the specified list.</p> <p data-bbox="824 1671 909 1692">Example:</p> <pre data-bbox="824 1713 1435 1873"> "filterConditions": [{ "field": "BillingState", "operator": "NOT IN", "value": ["California", "New York"] }]</pre>

Operator	Comment
	<pre data-bbox="821 258 1440 331"> }]</pre>
INCLUDES	<p data-bbox="821 380 1440 443">For picklist or multi-select picklist fields only. Filter condition is true if the value in the picklist field includes the specified value.</p> <p data-bbox="821 464 911 485">Example:</p> <pre data-bbox="821 506 1440 737"> "filterConditions": [{ "field": "BillingState", "operator": "INCLUDES", "value": ["California"] }]</pre>
EXCLUDES	<p data-bbox="821 785 1440 848">For picklist or multi-select picklist fields only. Filter condition is true if the value in the picklist field excludes the specified value.</p> <p data-bbox="821 869 911 890">Example:</p> <pre data-bbox="821 911 1440 1142"> "filterConditions": [{ "field": "BillingState", "operator": "EXCLUDES", "value": ["California", "New York"] }]</pre>

Let's look at a few examples of structured filters.



Example: Let's look at an example with a basic structured filter. To perform pipeline analysis on opportunities in fiscal quarter 2 of fiscal year 2015, you create this dataflow definition file to create the relevant dataset.


```


{
  "Extract_Filtered_Opportunities": {
    "action": "sfcdigest",
    "parameters": {
      "object": "Opportunity",
      "fields": [
        { "name": "Id" },
        { "name": "Name" },
        { "name": "AccountId" },
        { "name": "Amount" },
        { "name": "StageName" },
        { "name": "CloseDate" },
        { "name": "OwnerId" },
        { "name": "FiscalYear" },
        { "name": "FiscalQuarter" },
        { "name": "SystemModstamp" }
      ]
    }
  }
}
```

```

    ],
    "filterConditions": [
      {
        "field": "FiscalYear",
        "operator": "=",
        "value": "2015"
      },
      {
        "field": "FiscalQuarter",
        "operator": "=",
        "value": "2"
      }
    ]
  }
},
"Register_Opportunities_Dataset": {
  "action": "sfdcRegister",
  "parameters": {
    "alias": "Opportunities_2015Q2",
    "name": "Opportunities_2015Q2",
    "source": "Extract_Filtered_Opportunities"
  }
}
}

```

 **Note:** If you do not specify a logical operator for multiple filter conditions—as is the case in this example—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.

```

{
  "Extract_Opportunities": {
    "action": "sfdcDigest",
    "parameters": {
      "object": "Opportunity",
      "fields": [
        { "name": "Id" },
        { "name": "Name" },
        {
          "name": "Amount",
          "defaultValue": 0
          "precision": 18
          "scale": 2
        },
        { "name": "StageName" },
        {

```

```

        "name": "CloseDate",
        "fiscalMonthOffset": 9,
        "firstDayOfWeek": "2",
        "isYearEndFiscalYear": true
    },
    { "name": "AccountId" },
    { "name": "OwnerId" },
    {
        "name": "OpportunitySupportTeamMembers__c",
        "type": "Text",
        "isMultiValue": true,
        "multiValueSeparator": ",",
        "precision": 255
    }
]
}
},
"Register_Opportunities_Dataset": {
    "action": "sfdcRegister",
    "parameters": {
        "alias": "Opportunities",
        "name": "Opportunities",
        "source": "Extract_Opportunities"
    }
}
}
}

```

SEE ALSO:

[sfdcDigest Transformation](#)

Unsupported Salesforce Objects and Fields

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.

- AuthProvider
- BrandTemplate
- ChatterConversation
- ChatterConversationMember
- ChatterMessage
- ConnectedApplication
- ContentFolderLink
- ContentWorkspace
- ContentWorkspaceDoc

- CorsWhitelistEntry
- CustomNotDeployed__OwnerSharingRule
- EmailDomainKey
- EmailServicesAddress
- EmailServicesFunction
- EmailTemplate
- EnvironmentHub
- EnvironmentHubInvitation
- EnvironmentHubMemberRel
- FeedPollChoice
- FeedPollVote
- KnowledgeArticleVersion
- LoginGeo
- LoginHistory
- NetworkActivityAudit
- NetworkModeration
- OrganizationProperty
- OrgWideEmailAddress
- OutboundField
- PackageLicense
- PartnerNetworkSyncLog
- PermissionSet
- PermissionSetLicense
- Profile
- ReputationLevel
- ReputationLevelLocalization
- ReputationPointsRule
- SearchPromotionRule
- SelfServiceUser
- SetupAssistantProgress
- SsoUserMapping
- TenantSecret
- Territory2ModelHistory
- TwoFactorInfo
- UserLogin
- UserPackageLicense
- UserProvAccount
- UserProvAccountStaging
- UserProvisioningConfig
- UserProvisioningLog

- UserProvisioningRequest
- UserProvisioningRequestOwnerSharingRule
- UserProvisioningRequestShare
- UserProvMockTarget
- 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 cannot extract data from these fields:

Object	Unsupported Fields
Account	CleanStatus
ActionPlanItem	ItemId
AuthSession	<ul style="list-style-type: none"> • LoginGeold • LoginHistoryId
CaseArticle	KnowledgeArticleId
Contact	<ul style="list-style-type: none"> • CanAllowPortalSelfReg • CleanStatus
ContentDocument	ParentId
ContentFolderLink	ParentEntityId
CustomPerson__p	Title
DocumentAttachmentMap	ParentId
EmailMessage	ActivityId
EmailRoutingAddress	EmailServicesAddressId
EnvironmentHubMember	EnvironmentHubId
ExternalEventMapping	EventId
InstalledMobileApp	ConnectedApplicationId
Lead	CleanStatus
KnowledgeArticle	MasterLanguage
KnowledgeArticleVersion	<ul style="list-style-type: none"> • IsOutOfDate • TranslationCompletedDate

Object	Unsupported Fields
	<ul style="list-style-type: none"> • TranslationExportedDate • TranslationImportedDate
Network	<ul style="list-style-type: none"> • CaseCommentEmailTemplateld • ChangePasswordEmailTemplateld • ForgotPasswordEmailTemplateld • WelcomeEmailTemplateld
Organization	<ul style="list-style-type: none"> • SelfServiceEmailUserOnCaseCreationTemplateld • SelfServiceNewCommentTemplateld • SelfServiceNewPassTemplateld • SelfServiceNewUserTemplateld • WebToCaseAssignedEmailTemplateld • WebToCaseCreatedEmailTemplateld • WebToCaseEmailTemplateld • WebToLeadEmailTemplateld
PermissionSet	<ul style="list-style-type: none"> • PermissionsEditEvent • PermissionsEditTask
PermissionSetLicense	<ul style="list-style-type: none"> • MaximumPermissionsEditEvent • MaximumPermissionsEditTask
Profile	<ul style="list-style-type: none"> • PermissionsEditEvent • PermissionsEditTask
ThirdPartyAccountLink	SsoProviderId
User	LastPasswordChangeDate
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
<code>object</code>	Yes	API name of the Salesforce object from which you want to extract data. This object is the input source for this transformation. The <code>sfdcDigest</code> transformation doesn't support extraction from all Salesforce objects.
<code>incremental</code>	No	Performs an incremental extraction, which extracts only changes to the Salesforce object since the last dataflow run. Valid values: true or false.
<code>fullRefreshToken</code>	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 <code>fullRefreshToken</code> parameter is included in the dataflow definition. To run a full extraction again, change the value of the <code>fullRefreshToken</code> parameter to a different value.
<code>fields</code>	Yes	An array of names of all fields from which you want to extract data from the specified

Parameter	Required?	Value
		Salesforce object. The sfdcDigest transformation doesn't support extraction from all field types. See Field Attributes.
filterConditions	No	A filter that restricts the records extracted from the specified Salesforce object. The sfdcDigest transformation extracts all records from the Salesforce object for which the filter is true. You can specify a structured or advanced filter. See Filter Conditions Parameters.
complexFilterConditions	No	For advanced filters only. A SOQL WHERE clause used to filter records extracted from the specified Salesforce object.

Field Attributes

The following table describes the attributes in the `fields` section. It also describes optional attributes that you can provide to override the field metadata. You can override the metadata that the sfdcDigest transformation extracts from a Salesforce object to make the data appear differently in a dataset. For example, 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: <pre>"type": "Text"</pre>

Attribute	Required?	Value
fiscalMonthOffset	No	<p>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.</p> <p>Example:</p> <pre>"fiscalMonthOffset": 9</pre> <p> Note: This attribute also controls whether Wave Analytics generates fiscal date fields. To generate fiscal date fields, set <code>fiscalMonthOffset</code> to a value other than '0'.</p> <p>For more information, see Date Handling in Datasets.</p>
isYearEndFiscalYear	No	<p>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.</p> <ul style="list-style-type: none"> • 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. <p>Example:</p> <pre>"isYearEndFiscalYear": true</pre> <p>This field is relevant only when <code>fiscalMonthOffset</code> is greater than 0.</p> <p>For more information, see Date Handling in Datasets.</p>
firstDayOfWeek	No	<p>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.</p>

Attribute	Required?	Value
		Example: <code>"firstDayOfWeek": "0"</code> For more information, see Date Handling in Datasets .
isMultiValue	No	For text fields only. Indicates whether the specified field has multiple values. Example: <code>"isMultiValue": false</code>
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: <code>"multiValueSeparator": ";"</code>
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: <code>"precision": 10</code>
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: <code>"scale": 2</code>

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	<p>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:</p> <pre>{ "field": "<field name>", "operator": "<operator>", "value": "<value>", "isQuoted": true false}</pre>
operator	No	<p>The purpose depends on the context.</p> <ul style="list-style-type: none"> • <code>operator</code> can be used as a comparison operator—like <code>=</code>, <code><</code>, and <code>IN</code>—that compares the field value against a constant value. • <code>operator</code> can also be used as a logical operator (AND, OR, or NOT) that links multiple filter conditions together. <p>In the example below, the bold <code>operator</code> is the logical operator. The other instances of <code>operator</code> are comparison operators.</p> <pre>"filterConditions": [{ "operator": "OR", "conditions": [{ "field": "StageName", "operator": "=", "value": "Closed Won" }, { "field": "Probability", "operator": ">=", "value": "90" }] }]</pre>

Parameter	Required?	Value
		<pre> }]</pre>
value	No	The value used in a filter condition.
isQuoted	No	<p>Indicates whether you quoted the string value in a filter condition.</p> <p>Example with quoted values:</p> <pre> "filterConditions": [{ "field": "StageName", "operator": "IN", "value": "('Closed Won', 'Closed Lost')", "isQuoted": true }]</pre> <p>Example with non-quoted values:</p> <pre> "filterConditions": [{ "field": "StageName", "operator": "IN", "value": ["Closed Won", "Closed Lost"], "isQuoted": false }]</pre> <p>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.</p>
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](#)

[Extracting Incremental Changes to Data \(Pilot\)](#)


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.
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": "OwnerId' == '\$User.Id'"  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 its security predicate setting in the dataflow. To change the security predicate for an existing dataset, edit the dataset in the user interface.


SEE ALSO:

[sfdcRegister Transformation](#)

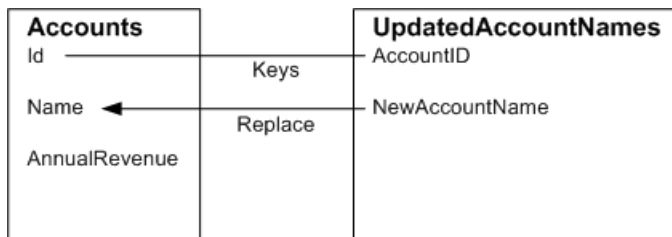
update Transformation

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

When you create the transformation, you specify the keys that are used to match records between the two datasets. To dictate which field in the lookup dataset updates the field in the source dataset, you also map the corresponding fields from both datasets.

 **Example:** Let's look at an example. You have an existing Accounts dataset that contains account information—Id, Name, and AnnualRevenue. Unfortunately, some of the account names in the dataset are now incorrect because of a series of mergers and acquisitions. To quickly update the account names in the dataset, you perform the following tasks.

1. Create a .csv file that contains the new account names and associated account IDs for accounts that have name changes.
2. Upload the .csv file to create a dataset called UpdatedAccountNames.
3. Create a dataflow definition file to update account names in the Accounts dataset by looking up the new account names in the UpdatedAccountNames dataset.



You create the following dataflow definition file.

```

{
  "Extract_AccountDetails": {
    "action": "sfdcDigest",
    "parameters": {
      "object": "Account",
      "fields": [
        { "name": "Id" },
        { "name": "Name" },
        { "name": "AnnualRevenue" }
      ]
    }
  },
  "Extract_UpdatedAccountNames": {
    "action": "edgemart",
    "parameters": { "alias": "UpdatedAccountNames" }
  },
  "Update_AccountRecords": {
    "action": "update",
    "parameters": {
      "left": "Extract_AccountDetails",
      "right": "Extract_UpdatedAccountNames",
      "left_key": [ "Id" ],
      "right_key": [ "AccountID" ],
      "update_columns": { "Name": "NewAccountName" }
    }
  }
}

```



```

    },
    "Register_UpdatedAccountRecords": {
      "action": "sfdcRegister",
      "parameters": {
        "alias": "Accounts",
        "name": "Accounts",
        "source": "Update_AccountRecords"
      }
    }
  }
}

```



Example: Let's look at another example, where a composite key is used to match records between both datasets. In this case, you match records using the account ID and account name fields.

You create the following dataflow definition file.

```

{
  "Extract_AccountDetails": {
    "action": "sfdcDigest",
    "parameters": {
      "object": "Account",
      "fields": [
        { "name": "Id" },
        { "name": "Name" },
        { "name": "AnnualRevenue" }
      ]
    }
  },
  "Extract_UpdatedAccountNames": {
    "action": "edgemart",
    "parameters": { "alias": "UpdatedAccountNames" }
  },
  "Update_AccountRecords": {
    "action": "update",
    "parameters": {
      "left": "Extract_AccountDetails",
      "right": "Extract_UpdatedAccountNames",
      "left_key": ["Id", "Name"],
      "right_key": ["AccountId", "NewAccountName"],
      "update_columns": {
        "Name": "NewAccountName",
        "CreatedDate": "NewCreatedDate",
        "AnnualRevenue": "NewAnnualRevenue"
      }
    }
  },
  "Register_UpdatedAccountRecords": {
    "action": "sfdcRegister",
    "parameters": {
      "alias": "Accounts",
      "name": "Accounts",
      "source": "Update_AccountRecords"
    }
  }
}

```

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 match records in the other dataset. If you use a composite key, the left and right keys must have the same number of columns in the same order. For an example, see update Transformation on page 92.
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	<p>An array of corresponding columns between the left and right datasets. Use the following syntax: <code>"update_columns": { "LeftColumn1": "RightColumn1", "LeftColumn2": "RightColumn2", ... "LeftColumnN": "RightColumnN" }</code>. The value from right column replaces the value from the corresponding left column. The field types of the left and right column must match.</p> <p> Note: If you specify a column name that does not exist, the dataflow fails.</p> <p>If you do not specify this parameter, the transformation updates the left dataset</p>

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 `sfcdigest` 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": "sfcdigest",
  "parameters": {
    "object": "Opportunity",
    "fields": [
      { "name": "Name" },
      { "name": "Amount" }
    ]
  },
  "schema": {
    "objects": [
      {
        "label": "Sales Opportunities",
        "fields": [
          {
            "name": "Amount",
            "label": "Opportunity Amount"
          }
        ]
      }
    ]
  }
}
```

Object Attributes

You can override the following object attributes.

Object Attribute	Type	Description
label	String	The display name for the object. Can be up to 40 characters. Example: <code>"label": "Sales Data"</code>
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: <code>"description": "The SalesData object tracks basic sales data."</code>
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: <code>"name": "Amount"</code> <code>"name": "Role.Name"</code>
label	String	The display name for the field. Can be up to 255 characters. Example: <code>"label": "Opportunity Amount"</code>
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: <code>"description": "The Amount field contains the opportunity amount."</code>
isSystemField	Boolean	Indicates whether this field is a system field to be excluded from query results. Example: <code>"isSystemField": false</code>


Field Attribute	Type	Description
format	String	<p>The display format of the numeric value.</p> <p>Examples:</p> <p><code>"format": "\$#,##0.00"</code> (Numeric)</p> <p>For more information about valid formats, see Numeric Formats.</p>

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 one digit
.	This symbol is the default decimal separator. Use the <code>decimalSeparator</code> field to set the decimal separator to a different symbol.
-	Minus sign
,	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	<ul style="list-style-type: none"> Change the object label to "Sales Opportunities" Add an object description
Id field	<ul style="list-style-type: none"> Change the field label to "Opportunity Id" Hide the field from queries
Amount field	<ul style="list-style-type: none"> Change the field label to "Opportunity Amount" Change the format to \$#,##0.00
CloseDate field	<ul style="list-style-type: none"> Change the field label to "Closing Date"

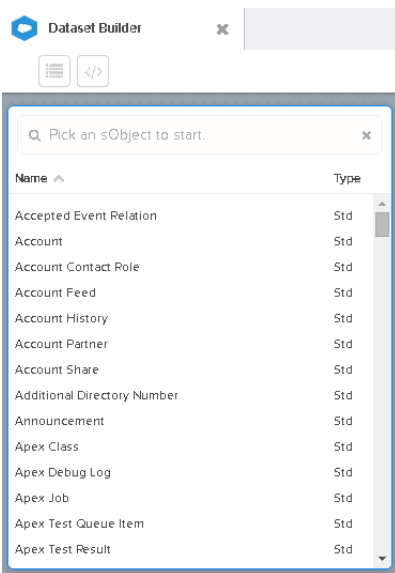
To override the attributes, you add the Schema section with the override values to sfdcDigest in the dataflow definition file.


```
{
  "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.



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 its 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.

EDITIONS

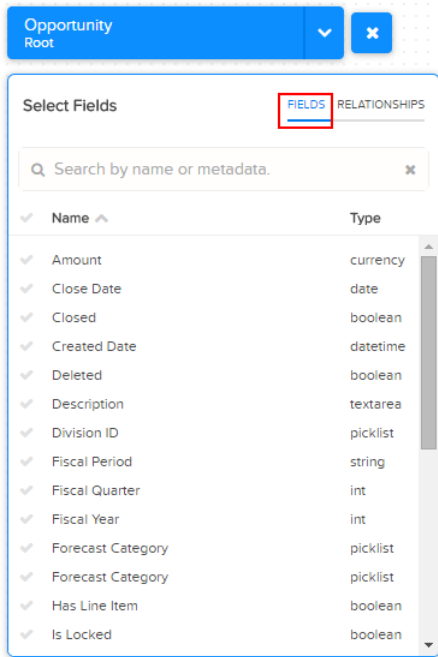
Available in: Salesforce Classic and Lightning Experience


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

USER PERMISSIONS

To access the dataset builder:

- "Edit Wave Analytics Dataflows"

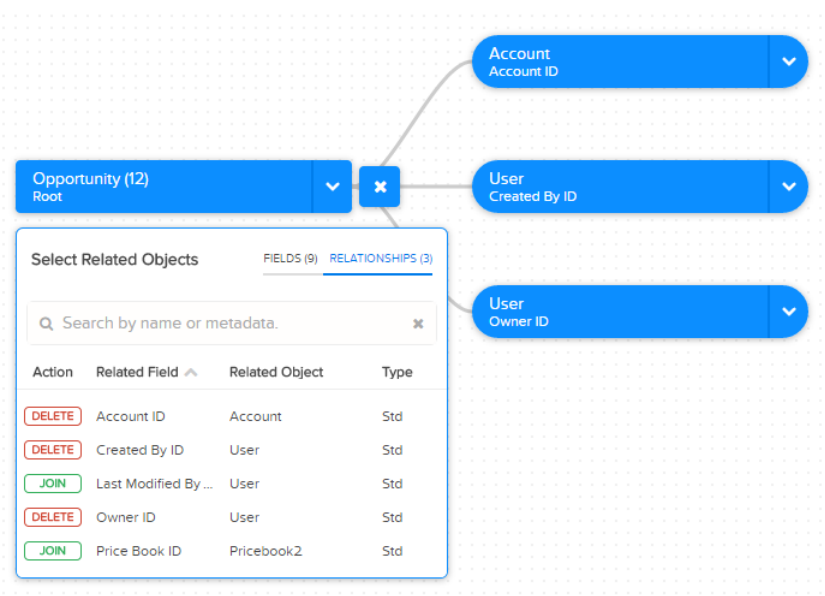


 **Note:** You can view this dialog box for any object included in the canvas.


- In the Fields tab, select the fields from which you want to extract data. To locate fields more quickly, you can search for them or sort them by name or type.

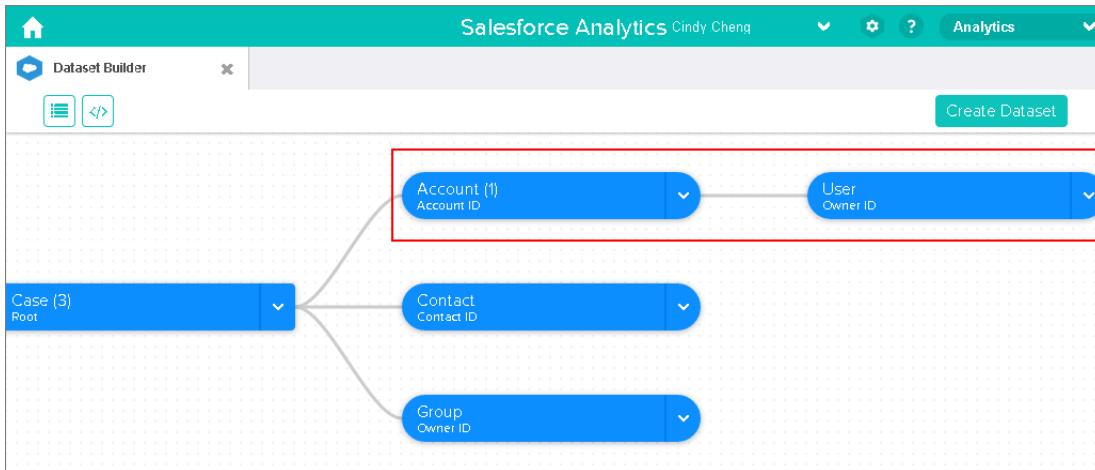
 **Important:** You must select at least one field for each object that you add to the canvas. If you add an object and don't add any of its fields, the dataflow fails at run time.

- 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.

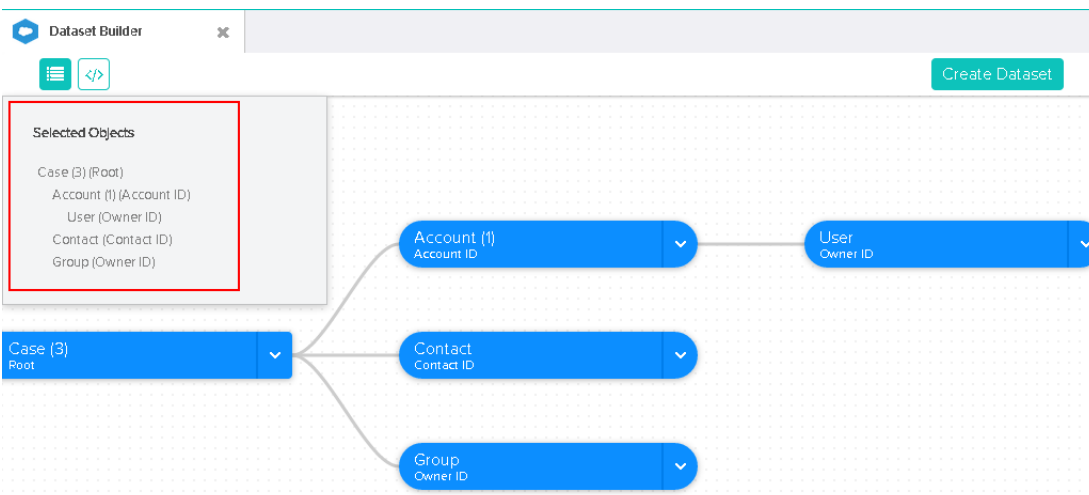


8. For each related object, select the fields from which you want to extract data.

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.

11. To view the associated JSON, click .

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.

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..

[Wave Connector Setup for Administrators](#)

In some cases, administrators need to change Salesforce security settings so users can connect to the Wave Connector app.

SEE ALSO:

[Install the Wave Connector Excel App](#)

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

The screenshot shows the Microsoft Excel interface with the Wave Connector application window open. The Excel spreadsheet contains a list of opportunity statuses in column N, such as 'ect- OpportunityStatus', 'mer Open Opply', 'ect Open Opply', etc. The Wave Connector window is titled 'Salesforce Wave Connector fo...' and features a 'Get Started' section. It provides instructions on how to use the connector to import data from Excel to Salesforce Analytics Cloud. A text box for naming the dataset is filled with 'EventMarketingLeadTrackin'. Below this, the 'Data Selection' section indicates that 15 columns and 1048576 rows are selected. A table lists the data fields and their types: Lead Count (Numeric), CampaignID (Text), StartDate (Date), Product (Text), Location (Text), LeadSource (Text), EmailAddress (Text), AccountID (Text), and Customer (Text). A 'Submit Data' button is located at the bottom of the connector window.

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.

 **Note: Excel Online in Office 365 users:** Your administrator has to change Salesforce security settings to give you access to the Connector. Administrators can go to [Wave Connector Setup for Administrators](#) on page 104 to learn how.

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.

If the Connector doesn't open, check with your Salesforce administrator. Your organization may use a custom domain, which an administrator needs to configure for the Connector following instructions in [Wave Connector Setup for Administrators](#) on page 104.

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.

SEE ALSO:

[Wave Connector Setup for Administrators](#)

Wave Connector Setup for Administrators

In some cases, administrators need to change Salesforce security settings so users can connect to the Wave Connector app.

The Wave Connector app gives users a fast, easy way to import data from Excel 2013 into Salesforce Wave Analytics. Users of Excel 2013 on the desktop in a standard Salesforce domain access the Connector through the Microsoft Apps for Office store without requiring any additional setup work by administrators.

 **Note: Wave Connector users:** The steps described here can only be done by your Salesforce administrator. Users should go to [Install the Wave Connector Excel App](#) on page 102 .

There are two circumstances that require administrators to change Salesforce security settings so users can access the Connector:

- **If your organization includes users of Excel Online, available in Office 365:** Administrators need to add a URL to the CORS whitelist so users can connect to the app.
- **If your organization created a custom Salesforce domain using the [My Domain](#) feature:** Administrators need to download the custom Wave Connector manifest and create a network share catalog using Microsoft Sharepoint. This lets users access the Connector through a private app catalog available only to your organization.

Add Wave Connector URL to CORS Whitelist

Users of the Office 365 version of Excel, Excel Online, require you to change Salesforce settings to enable them to access the Wave Connector. Otherwise they see an error message if they try to launch the app. To give Excel Online users in your organization access to the Connector, add the Connector URL to the CORS whitelist in Security settings.

1. From Setup, choose Security Controls | CORS.
2. Choose New.
3. Enter `https://waveconnector.force.com` as an origin URL pattern.

Whitelist an Origin

To add an origin to the CORS whitelist, enter a URL pattern that identifies the origin.
The URL pattern must include `https://` and a domain name, and may include a port.
The wildcard character (*) is supported and must be in front of a second-level domain name, for example, `https://*`.

CORS Whitelist Origin Edit Save Save & New Cancel

Origin URL Pattern

Complete URL:
`https://waveconnector.force.com`

CORS


Add Access to a Wave Connector Manifest for Custom Salesforce Domains

Wave Connector users in a custom Salesforce domain require you to set up a private app catalog to give them access to the Connector


1. Download the Wave Connector custom manifest from <https://waveconnector.force.com/analytics-excel-connector/manifest/MyManifest.apexp>. It should download automatically.
2. Follow the instructions at [MSDN](#) for creating a network share catalog to set up a private Sharepoint app catalog for your organization.

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](#).

 **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

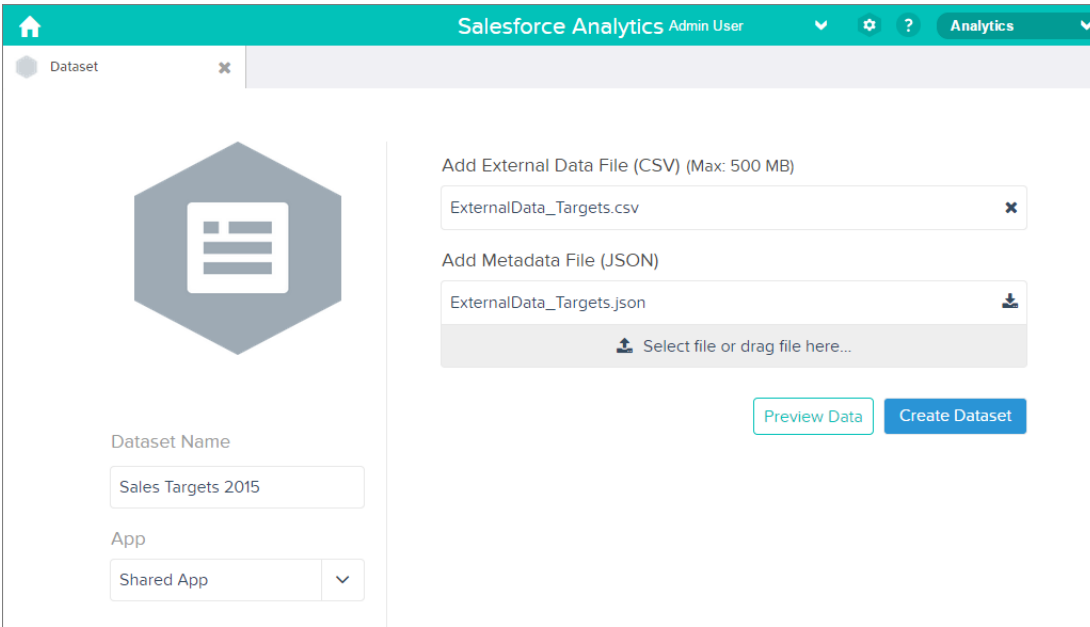
Available in: **Developer Edition**


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

USER PERMISSIONS

To upload external data:

- "Upload External Data to Wave Analytics"

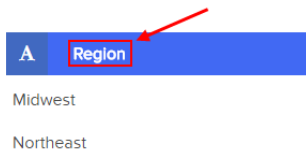


 **Note:** Instead of using the generated metadata file, if you want, you can upload a different metadata file that you created from scratch. If you upload your own metadata file, the **Preview Data** button is disabled.

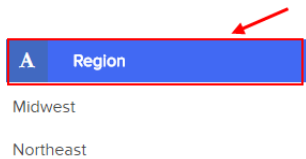
6. Perform the following tasks to change the metadata attributes in the generated metadata file.
 - a. Click **Preview Data** to view and change the required metadata attributes.

AccountOwner	Region	Target	TargetDate
Tony Santos	Midwest	10000	1/1/2011
Lucy Timmer	Northeast	50000	1/1/2011
Lucy Timmer	Northeast	0	12/1/2013
Bill Rolley	Midwest	15000	1/1/2011
Keith Laz	Southwest	35000	1/1/2011
Lucy Timmer	Southeast	40000	1/1/2011

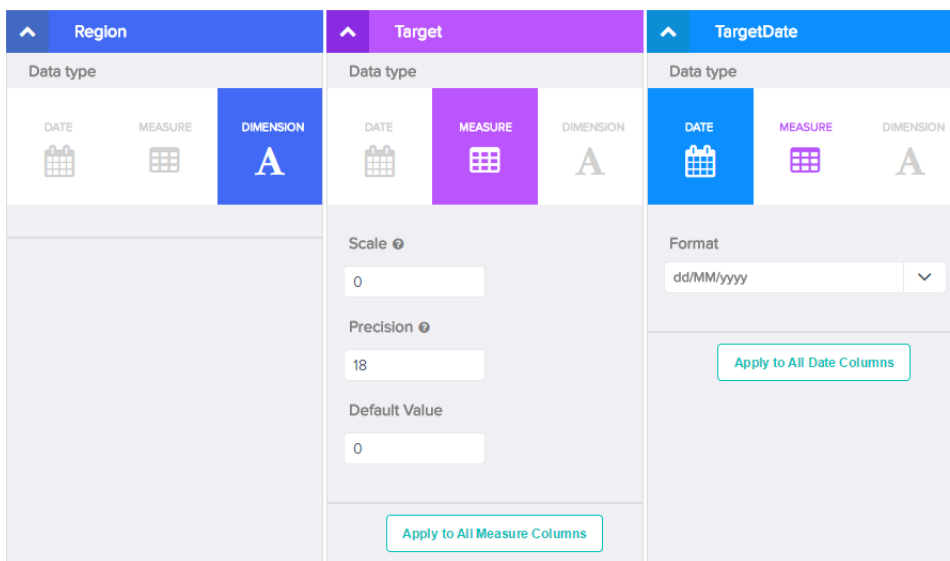
- b. Click a column name to change it. The column name is the display name in the dataset. The column name cannot exceed 40 characters.





- 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 in: **Developer Edition**

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


USER PERMISSIONS


To access the data monitor:


- “Edit Wave Analytics Dataflows,” “Upload External Data to Wave Analytics,” or “Manage Wave Analytics”

Dataflow	Start Time	Duration	Status	Message
Default Salesf...	Today at 7:39 AM	24 seconds	Successful	
Default Salesf...	Yesterday at 7:39 AM	0 hours, 10...	Failed	Queue wait time exceeded limit
Sales_Targets...	2 Days Ago at 9:34 AM	0 hours, 1 ...	Failed	Error executing node load [roo

Node Name	Start Time	Duration	Node Type	Status
load	2 Days Ago at 9:34 AM	0 hours, 1 minute	sfdcFetch	error
digest	N/A	N/A	csvDigest	pending
optimize-register	N/A	N/A	optimizer	pending
register	N/A	N/A	sfdcRegister	pending

1. In Wave Analytics, click the gear button () and then click **Data Monitor** to open the data monitor. The Jobs view appears by default. The Jobs view displays dataflow and upload jobs. The Jobs view 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.

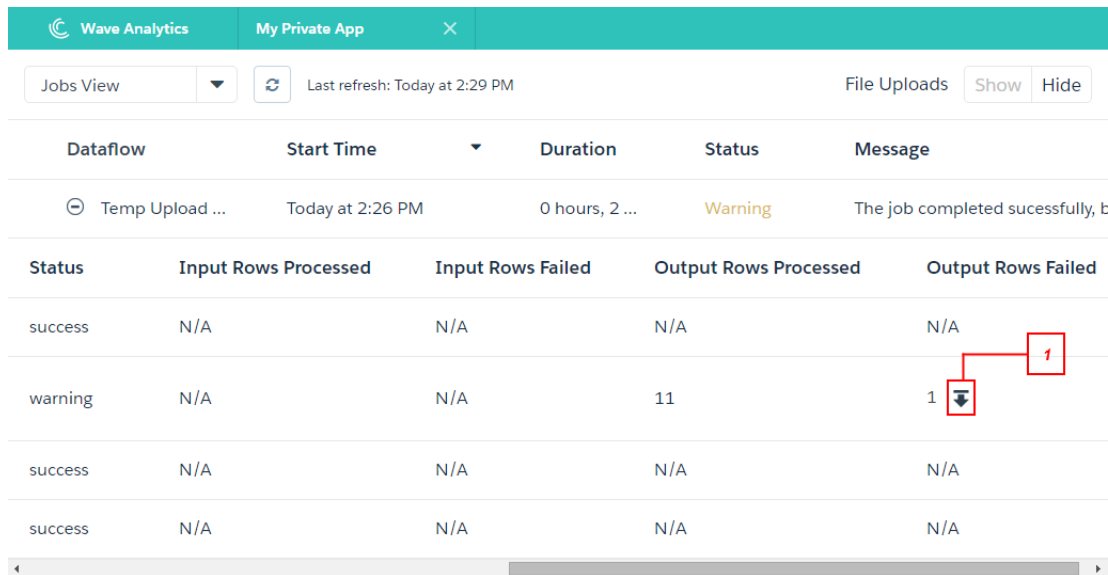
- To see the latest status of a job, click the Refresh Jobs button ().

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.

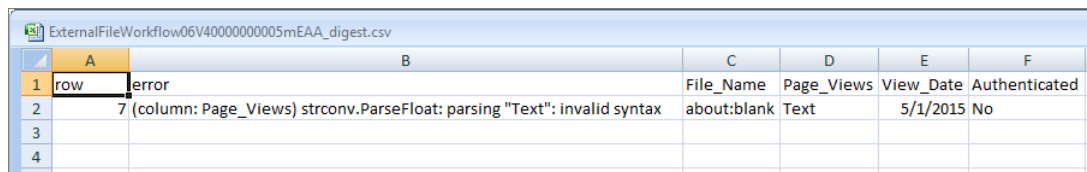
- To view the run-time details for a job, expand the job node (3).
- The run-time details display under the job. In the run-time details section, scroll to the right to view information about the rows that were processed.
- To troubleshoot a job that has failed rows, view the error message. Also, click 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 screenshot shows the Wave Analytics interface. At the top, there's a teal header with 'Wave Analytics' and 'My Private App'. Below that, a 'Jobs View' dropdown and a refresh button are visible. The main content area shows a table of jobs. One job, 'Temp Upload ...', is highlighted with a 'Warning' status. Below the job list, there's a detailed view for this job. It shows columns for 'Status', 'Input Rows Processed', 'Input Rows Failed', 'Output Rows Processed', and 'Output Rows Failed'. The 'Output Rows Failed' column shows '11'. A red box highlights a download button (1) next to the '1' in the 'Output Rows Failed' column.

The error log contains a list of failed rows.



The screenshot shows an external file workflow error log. The file name is 'ExternalFileWorkflow06V4000000005mEAA_digest.csv'. The log contains a table with columns A through F. Row 1 shows an error message: '7 (column: Page_Views) strconv.ParseFloat: parsing "Text": invalid syntax'. Row 2 shows the error details: 'File_Name: about:blank, Page_Views: Text, View_Date: 5/1/2015, Authenticated: No'. Rows 3 and 4 are empty.

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

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.

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.
App	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 <i>Extended Metadata (XMD) Reference</i> 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. Refer to <i>External Data Format Reference</i> 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 <i>External Data Format Reference</i> for information about metadata files.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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


USER PERMISSIONS

To edit a dataset:

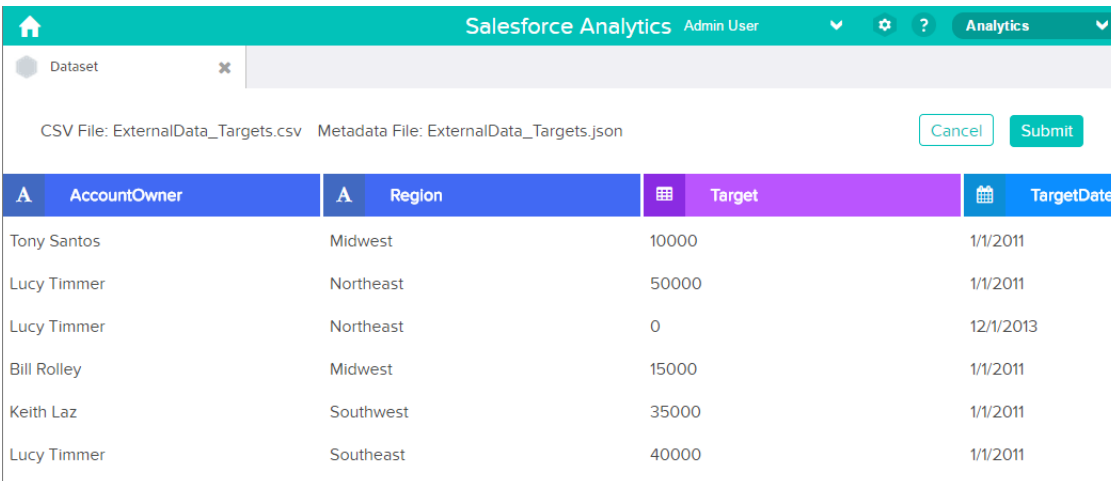
- "Edit Wave Analytics Dataflows"

Option	Description
Security Predicate	<p>Add a security predicate if you'd like to apply row-level security on the dataset.</p> <p>For information about predicates, see Row-Level Security for Datasets.</p>

- 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.

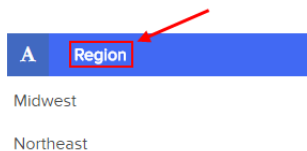
 **Note:** The **Preview Data** button is disabled if you uploaded your own metadata file.

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

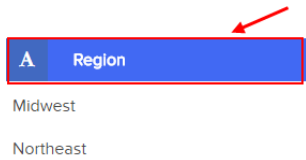


AccountOwner	Region	Target	TargetDate
Tony Santos	Midwest	10000	1/1/2011
Lucy Timmer	Northeast	50000	1/1/2011
Lucy Timmer	Northeast	0	12/1/2013
Bill Rolley	Midwest	15000	1/1/2011
Keith Laz	Southwest	35000	1/1/2011
Lucy Timmer	Southeast	40000	1/1/2011

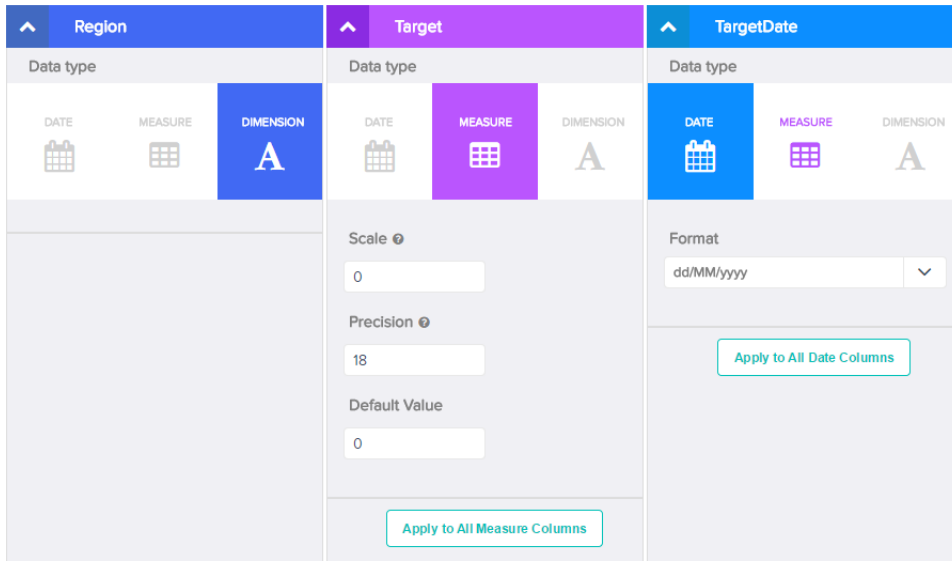
- For each column:
 - Click a column name to change it. The column name is the display name in the dataset. The column name cannot exceed 40 characters.





- 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  to download the metadata file, edit the file, and then upload it.
- 9. Click **Update Dataset**.
- 10. Click **Continue** to dismiss the confirmation message.

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.

```
{
  ...
  "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.

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.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To delete a dataset:


- "Edit Wave Analytics Dataflows"

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

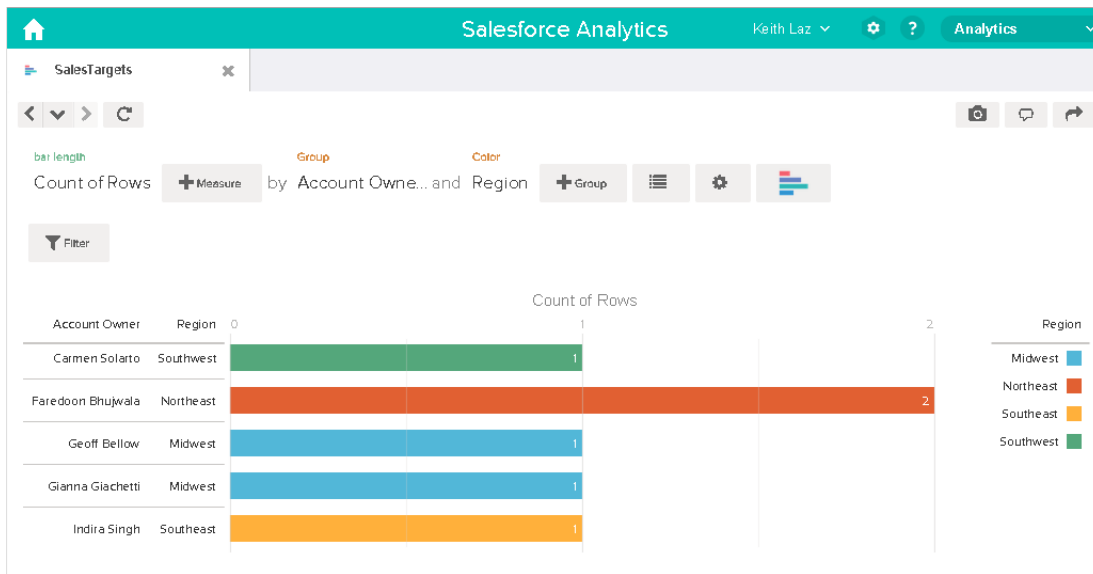
First, determine what data you want to include in the dataset. For this example, you will create a Targets dataset that contains all sales targets.

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",
          "fullyQualifiedName": "Targets.Target",
          "label": "Target",
          "type": "Numeric",
          "precision": 16,
          "scale": 0,
          "defaultValue": "0",
          "format": null
        },
        {
          "name": "TargetDate",
          "fullyQualifiedName": "Targets.TargetDate",
          "label": "TargetDate",
          "description": "",
          "type": "Date",
          "format": "dd/MM/yy HH:mm:ss",
          "isSystemField": false,
          "fiscalMonthOffset": 0
        }
      ]
    }
  ]
}
```

```
]
}
```

Create the Dataset

Now that you updated the metadata file with the predicate, you can create the dataset.

 **Warning:** If you wish to perform the steps in this sample implementation, perform the steps in a non-production environment. Ensure that these changes do not impact other datasets that you already created.

To create the dataset, perform the following steps.

1. In 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**

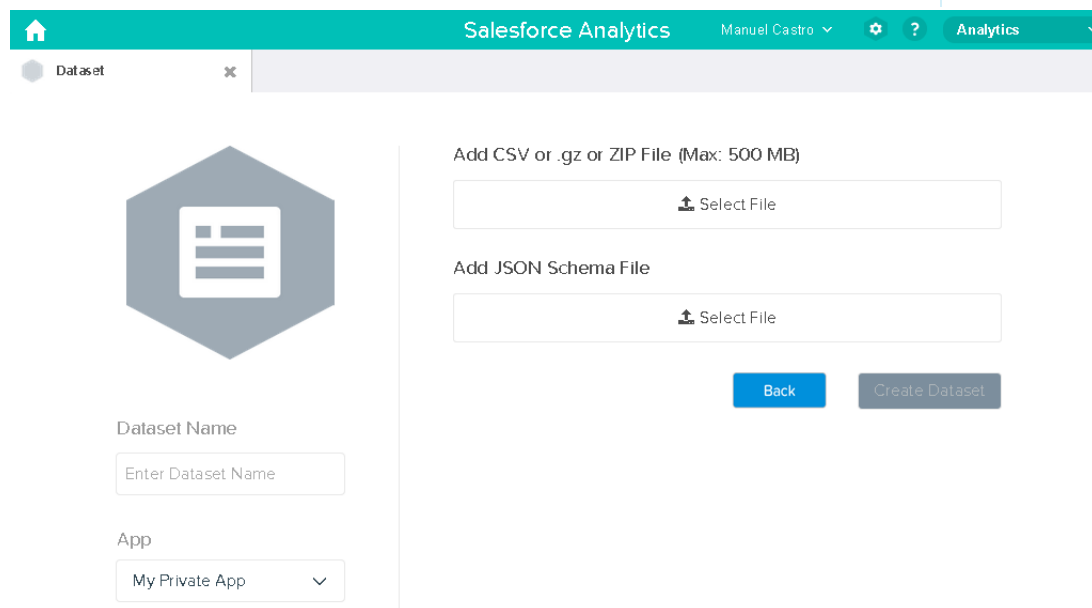
Available in: **Developer Edition**

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

USER PERMISSIONS

To upload a CSV and metadata file:

- “Upload External Data to Wave Analytics”



Dataset

Add CSV or .gz or ZIP File (Max: 500 MB)

Select File

Add JSON Schema File

Select File

Back Create Dataset

Dataset Name



Enter Dataset Name

App

My Private App

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 () 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.

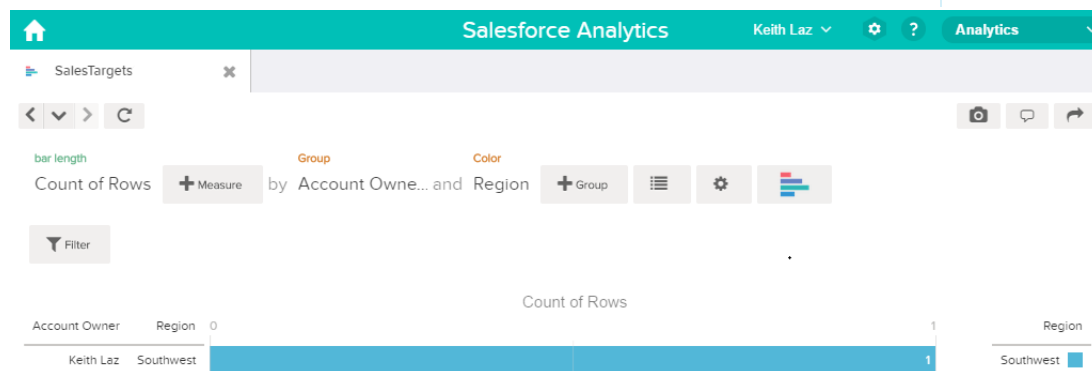
1. Log in to Wave Analytics as Keith.
2. Open the SalesTargets dataset.
As shown in the following lens, notice that Keith can see only his sales target.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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



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](#)

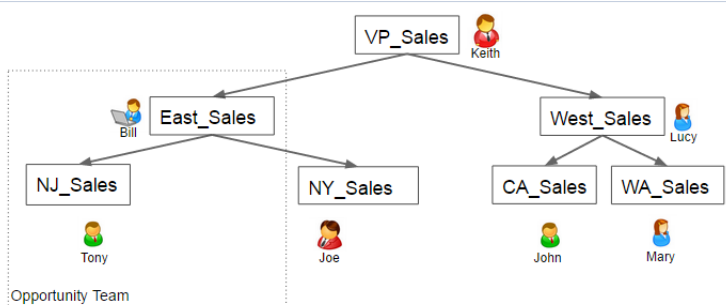
You must verify that the predicate is applied properly and that each user can see the appropriate opportunities.

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.



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

Action	Opportunity Name	Account Name	Amount	Close Date	Stage	Opportunity Owner Alias
<input type="checkbox"/> Edit Del +	Acc - 1000 Widgets	Acc_salesrep		9/4/2014	Prospecting	Tony
<input type="checkbox"/> Edit Del +	Acme - 1,200 Widgets	Acme	\$140,000.00	6/14/2012	Value Proposition	Keith
<input type="checkbox"/> Edit Del +	Acme - 200 Widgets	Acme	\$20,000.00	10/13/2012	Prospecting	Keith
<input type="checkbox"/> Edit Del +	Acme - 600 Widgets	Acme	\$70,000.00	8/10/2012	Needs Analysis	Keith
<input type="checkbox"/> Edit Del +	ESales_01	East_Sales_acc_01		9/4/2014	Prospecting	Bill
<input type="checkbox"/> Edit Del +	Global Media - 400...	Global Media	\$40,000.00	7/13/2012	Id. Decision Makers	Keith
<input type="checkbox"/> Edit Del +	salesforce.com - 1...	salesforce.com	\$100,000.00	6/14/2012	Negotiation/Review	Keith
<input type="checkbox"/> Edit Del +	salesforce.com - 2...	salesforce.com	\$20,000.00	8/12/2012	Value Proposition	Keith
<input type="checkbox"/> Edit Del +	salesforce.com - 50...	Global Media	\$50,000.00	5/12/2012	Closed Won	Keith
<input type="checkbox"/> Edit Del +	salesforce.com - 50...	Global Media	\$500,000.00	5/12/2012	Closed Won	Keith
<input type="checkbox"/> Edit Del +	West_Sales_01	West_Sales_Acc_01		9/4/2014	Prospecting	Lucy

EDITIONS

Available in: Salesforce Classic and Lightning Experience

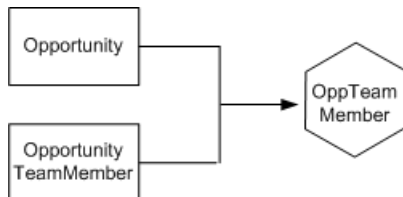
Available in: **Developer Edition**

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

Acc - 1000 Widgets is the only opportunity shared by an opportunity team. Bill is the Sales Manager for this opportunity. Tony is the opportunity owner.

Design the Dataflow to Load the Data

Now it's time to figure out how the dataflow will extract the Salesforce data and load it into a dataset. You start by creating this high-level design for the dataflow.



The dataflow will extract data from the Opportunity and OpportunityTeamMember objects, join the data, and then load it into the OppTeamMember dataset.

Now let's implement that design in JSON, which is the format of the dataflow definition file. A dataflow definition file contains transformations that extract, transform, and load data into a dataset.

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

```
{
  "Extract_OpportunityTeamMember": {
    "action": "sfdcDigest",
    "parameters": {
      "object": "OpportunityTeamMember",
      "fields": [
        { "name": "Name" },
        { "name": "OpportunityId" },
        { "name": "UserId" }
      ]
    }
  },
  "Extract_Opportunity": {
    "action": "sfdcDigest",
    "parameters": {
      "object": "Opportunity",
      "fields": [
        { "name": "Id" },
        { "name": "Name" },
        { "name": "Amount" },
        { "name": "StageName" },
        { "name": "AccountId" },
        { "name": "OwnerId" }
      ]
    }
  },
  "Augment_OpportunityTeamMember_Opportunity": {
    "action": "augment",
    "parameters": {
      "left": "Extract_OpportunityTeamMember",
```

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

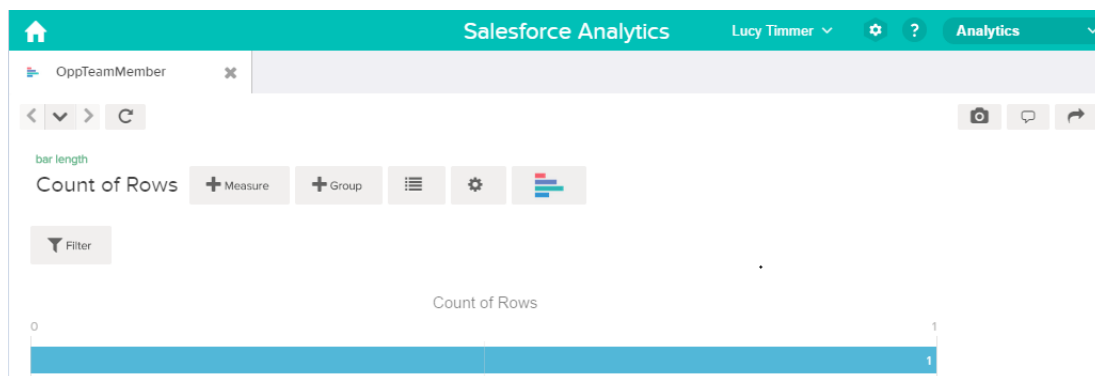
```

    "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.

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

This predicate compares the UserId column in the dataset against the ID of the user running a query against the dataset. The UserId column in the dataset contains the user ID of the team member associated with each opportunity. To determine the ID of the user running the query, 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.

```
{
  "Extract_OppportunityTeamMember": {
    "action": "sfdcDigest",
    "parameters": {
      "object": "OppportunityTeamMember",
      "fields": [
        { "name": "Name" },
        { "name": "OppportunityId" },
        { "name": "UserId" }
      ]
    }
  },
  "Extract_Oppportunity": {
    "action": "sfdcDigest",
```

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: [Developer Edition](#)

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

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: [Developer Edition](#)

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


```

    "parameters": {
      "object": "Opportunity",
      "fields": [
        { "name": "Id" },
        { "name": "Name" },
        { "name": "Amount" },
        { "name": "StageName" },
        { "name": "AccountId" },
        { "name": "OwnerId" }
      ]
    }
  },
  "Augment_OpportunityTeamMember_Opportunity": {
    "action": "augment",
    "parameters": {
      "left": "Extract_OpportunityTeamMember",
      "left_key": [
        "OpportunityId"
      ],
      "relationship": "TeamMember",
      "right": "Extract_Opportunity",
      "right_key": [
        "Id"
      ],
      "right_select": [
        "Name", "Amount"
      ]
    }
  },
  "Register_Dataset": {
    "action": "sfdcRegister",
    "parameters": {
      "alias": "OppTeamMember",
      "name": "OppTeamMember",
      "source": "105_Augment_OpportunityTeamMember_Opportunity",
      "rowLevelSecurityFilter": "'UserId' == \"\$User.Id\""
    }
  }
}


```

Create the Dataset

Now that you have the final dataflow definition file, you can create the dataset.

 **Warning:** If you wish to perform the steps in this sample implementation, verify that you have all required Salesforce objects and fields, and perform the steps in a non-production environment. Ensure that these changes do not impact other datasets that you already created. Also, always make a backup of the existing dataflow definition file before you make changes because you cannot retrieve old versions of the file.

To create the dataset, perform the following steps.

1. In 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

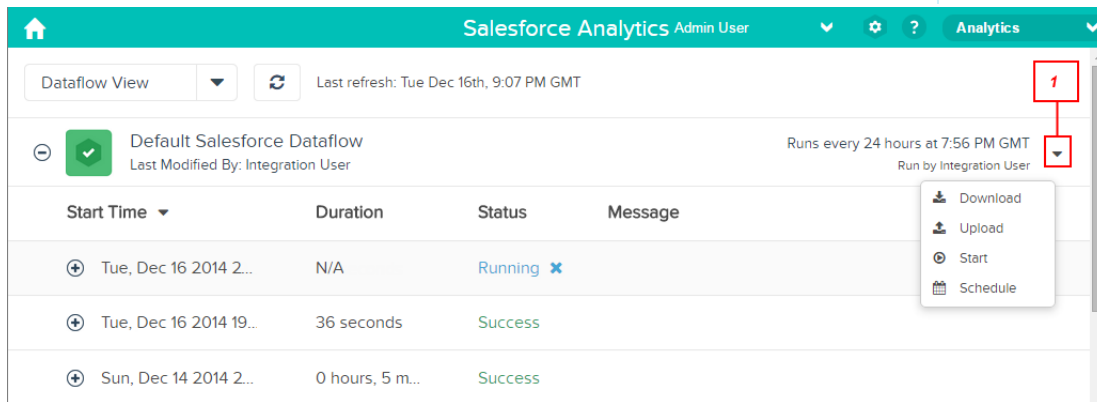
Available in: **Developer Edition**


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

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 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.

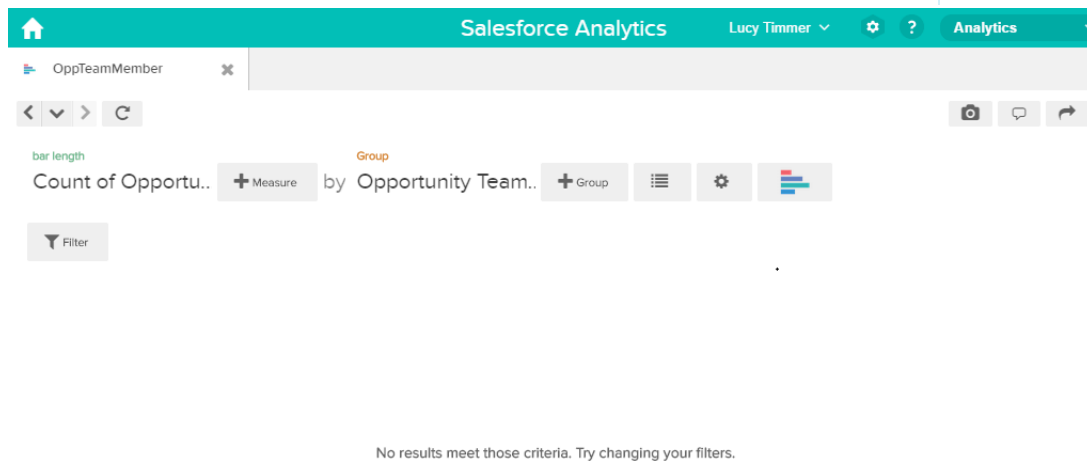
1. Log in to Wave Analytics as Lucy.
2. Open the OppTeamMember opportunity.
Notice that Lucy can't view the opportunity associated with the opportunity team anymore because she is not a member of the team.

EDITIONS

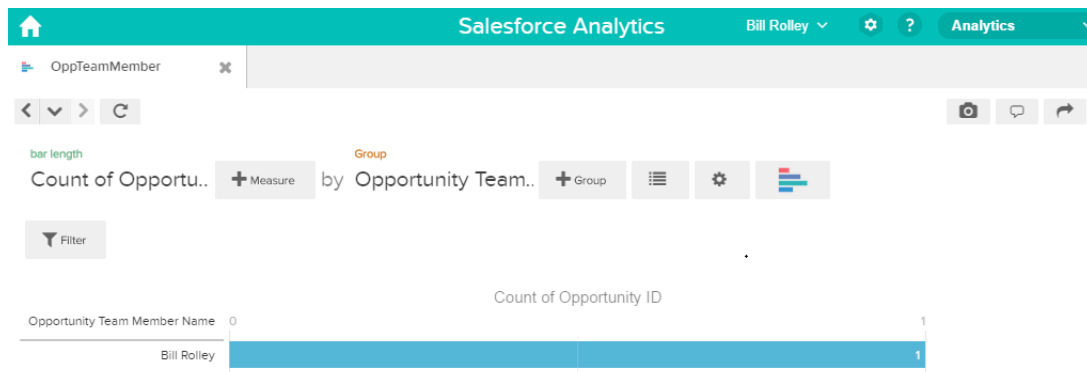
Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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



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.

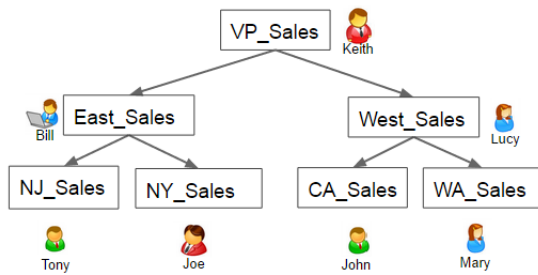
In this example, your Salesforce organization has the following role hierarchy and users.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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



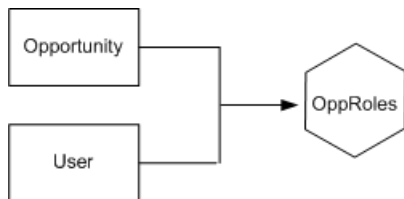
Also, your organization contains the following opportunities, most of which are owned by Keith.

All Opportunities Edit | Delete | Create New View List Feed

Action	Opportunity Name	Account Name	Amount	Close Date	Stage	Opportunity Owner Alias
Edit Del +	Acc - 1000 Widgets	Acc_salesrep		9/4/2014	Prospecting	Tony
Edit Del +	Acme - 1,200 Widgets	Acme	\$140,000.00	6/14/2012	Value Proposition	Keith
Edit Del +	Acme - 200 Widgets	Acme	\$20,000.00	10/13/2012	Prospecting	Keith
Edit Del +	Acme - 600 Widgets	Acme	\$70,000.00	8/10/2012	Needs Analysis	Keith
Edit Del +	ESales_01	East_Sales_acc_01		9/4/2014	Prospecting	Bill
Edit Del +	Global Media - 400 ...	Global Media	\$40,000.00	7/13/2012	Id. Decision Makers	Keith
Edit Del +	salesforce.com - 1,...	salesforce.com	\$100,000.00	6/14/2012	Negotiation/Review	Keith
Edit Del +	salesforce.com - 2,...	salesforce.com	\$20,000.00	8/12/2012	Value Proposition	Keith
Edit Del +	salesforce.com - 50 ...	Global Media	\$50,000.00	5/12/2012	Closed Won	Keith
Edit Del +	salesforce.com - 50 ...	Global Media	\$500,000.00	5/12/2012	Closed Won	Keith
Edit Del +	West_Sales_01	West_Sales_Acc_01		9/4/2014	Prospecting	Lucy

Design the Dataflow to Load the Data

Now it's time to figure out how the dataflow will extract the data and load it into a dataset. You start by creating this high-level design for the dataflow.



The dataflow will extract data from the Opportunity and User objects, join the data, and then load it into the OppRoles dataset.

Now let's implement that design in JSON, which is the format of the dataflow definition file. A dataflow definition file contains transformations that extract, transform, and load data into a dataset.

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

```
{
  "Extract_Opportunity": {
    "action": "sfdcDigest",
```

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

```

    "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",

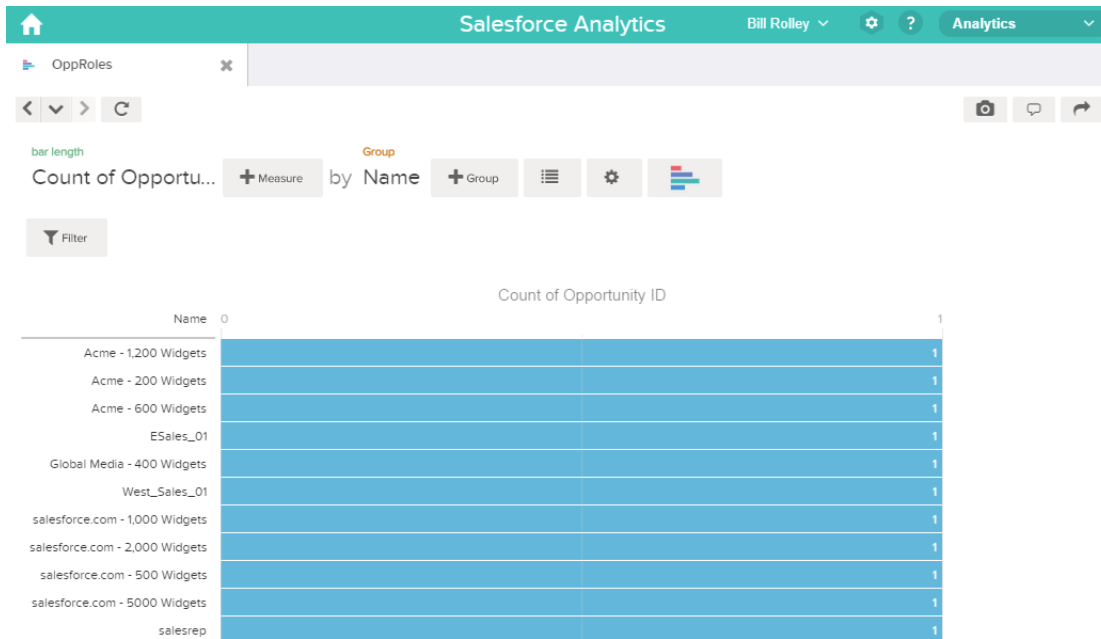
```

```

    "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.


```
'ParentRoleIDs' == "$User.UserRoleId" || 'OwnerId' == "$User.Id"
```

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

 **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"	<ul style="list-style-type: none"> • 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.
'OwnerId' == "\$User.Id"	<ul style="list-style-type: none"> • OwnerId refers to the dataset column that contains the user ID of the owner of each opportunity. • \$User.Id refers to the Id column of the User object. 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.

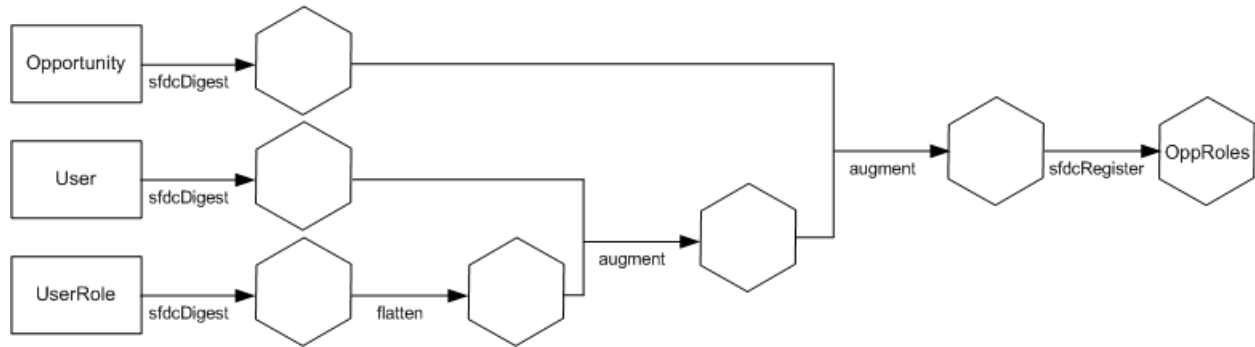
- Add a column in the dataset that stores a comma-separated list of the role IDs of all parents for each opportunity owner. When you defined the predicate in the previous step, you temporarily referred to this column as "ParentRoleIDs." To add the column, you redesign the dataflow as shown in the following diagram:

EDITIONS

Available in: **Salesforce Classic and Lightning Experience**

Available in: **Developer Edition**

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



The new dataflow design contains the following changes:

- Extracts the role IDs from the UserRole object.
 - Uses the Flatten transformation to generate a column that stores a comma-separated list of the role IDs of all parents of each user. When you determined the predicate in the previous step, you temporarily referred to this column as "ParentRoleIDs."
 - Link the new column to the OppRoles dataset.
- Add the predicate to the Register transformation that registers the OppRoles dataset.

You modify the dataflow as shown below.

```

{
  "Extract_Opportunity": {
    "action": "sfdcDigest",
    "parameters": {
      "object": "Opportunity",
      "fields": [
        { "name": "Id" },
        { "name": "Name" },
        { "name": "Amount" },
        { "name": "StageName" },
        { "name": "AccountId" },
        { "name": "OwnerId" }
      ]
    }
  },
  "Extract_User": {
    "action": "sfdcDigest",
    "parameters": {
      "object": "User",
      "fields": [
        { "name": "Id" },
        { "name": "Username" },
        { "name": "LastName" },
        { "name": "FirstName" },
        { "name": "Name" },
        { "name": "CompanyName" },
        { "name": "Division" },
        { "name": "Department" },
        { "name": "Title" },
        { "name": "Alias" },
        { "name": "CommunityNickname" },
      ]
    }
  }
}
  
```

```

        { "name": "UserType" },
        { "name": "UserRoleId" }
    ]
}
},
"Extract_UserRole": {
    "action": "sfdcDigest",
    "parameters": {
        "object": "UserRole",
        "fields": [
            { "name": "Id" },
            { "name": "ParentRoleId" },
            { "name": "RollupDescription" },
            { "name": "OpportunityAccessForAccountOwner" },
            { "name": "CaseAccessForAccountOwner" },
            { "name": "ContactAccessForAccountOwner" },
            { "name": "ForecastUserId" },
            { "name": "MayForecastManagerShare" },
            { "name": "LastModifiedDate" },
            { "name": "LastModifiedById" },
            { "name": "SystemModstamp" },
            { "name": "DeveloperName" },
            { "name": "PortalAccountId" },
            { "name": "PortalType" },
            { "name": "PortalAccountOwnerId" }
        ]
    }
},
"Flatten_UserRole": {
    "action": "flatten",
    "parameters": {
        "multi_field": "Roles",
        "parent_field": "ParentRoleId",
        "path_field": "RolePath",
        "self_field": "Id",
        "source": "Extract_UserRole"
    }
},
"Augment_User_FlattenUserRole": {
    "action": "augment",
    "parameters": {
        "left": "Extract_User",
        "left_key": [
            "UserRoleId"
        ],
        "relationship": "Role",
        "right": "Flatten_UserRole",
        "right_key": [
            "Id"
        ],
        "right_select": [
            "Roles",
            "RolePath"
        ]
    }
}

```

```


    }
  },
  "Augment_Opportunity_UserWithRoles": {
    "action": "augment",
    "parameters": {
      "left": "Extract_Opportunity",
      "left_key": [
        "OwnerId"
      ],
      "right": "Augment_User_FlattenUserRole",
      "relationship": "Owner",
      "right_select": [
        "Name",
        "Role.Roles",
        "Role.RolePath"
      ],
      "right_key": [
        "Id"
      ]
    }
  },
  "Register": {
    "action": "sfdcRegister",
    "parameters": {
      "alias": "OppRoles",
      "name": "OppRoles",
      "source": "Augment_Opportunity_UserWithRoles",
      "rowLevelSecurityFilter": "'Owner.Role.Roles' == \"\$User.UserRoleId\" || 'OwnerId'
== \"\$User.Id\""
    }
  }
}

```


 **Note:** In this example, the dataset has columns Owner.Role.Roles and OwnerId. A user can view the values of these columns for each record to which they have access.

Create the Dataset

Now that you have the final dataflow definition file, you can create the dataset.

 **Warning:** If you wish to perform the steps in this sample implementation, verify that you have all required Salesforce objects and fields, and perform the steps in a non-production environment. Ensure that these changes do not impact other datasets that you already created. Also, always make a backup of the existing dataflow definition file before you make changes because you cannot retrieve old versions of the file.

To create the dataset, perform the following steps.

1. In 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

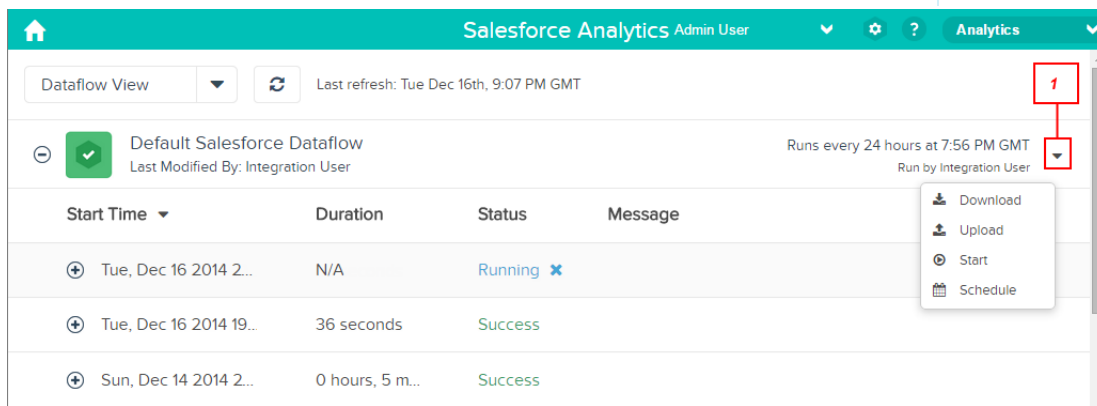
Available in: **Developer Edition**


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

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.

EDITIONS

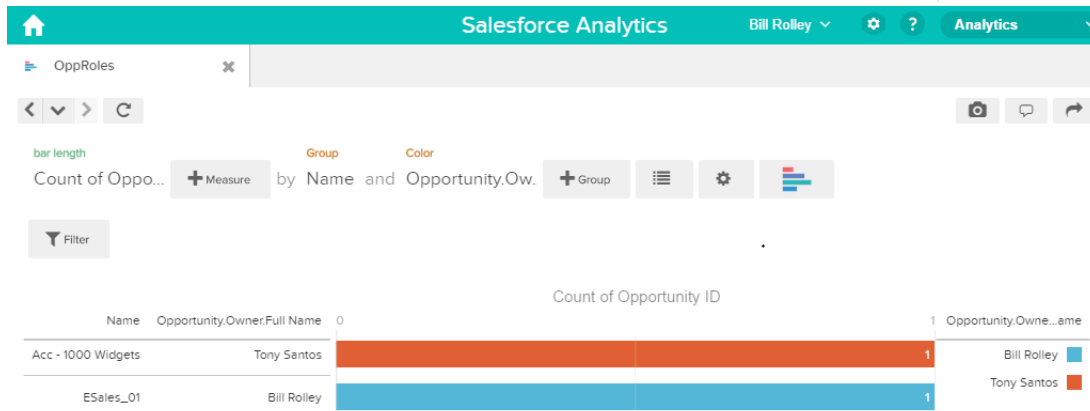
Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

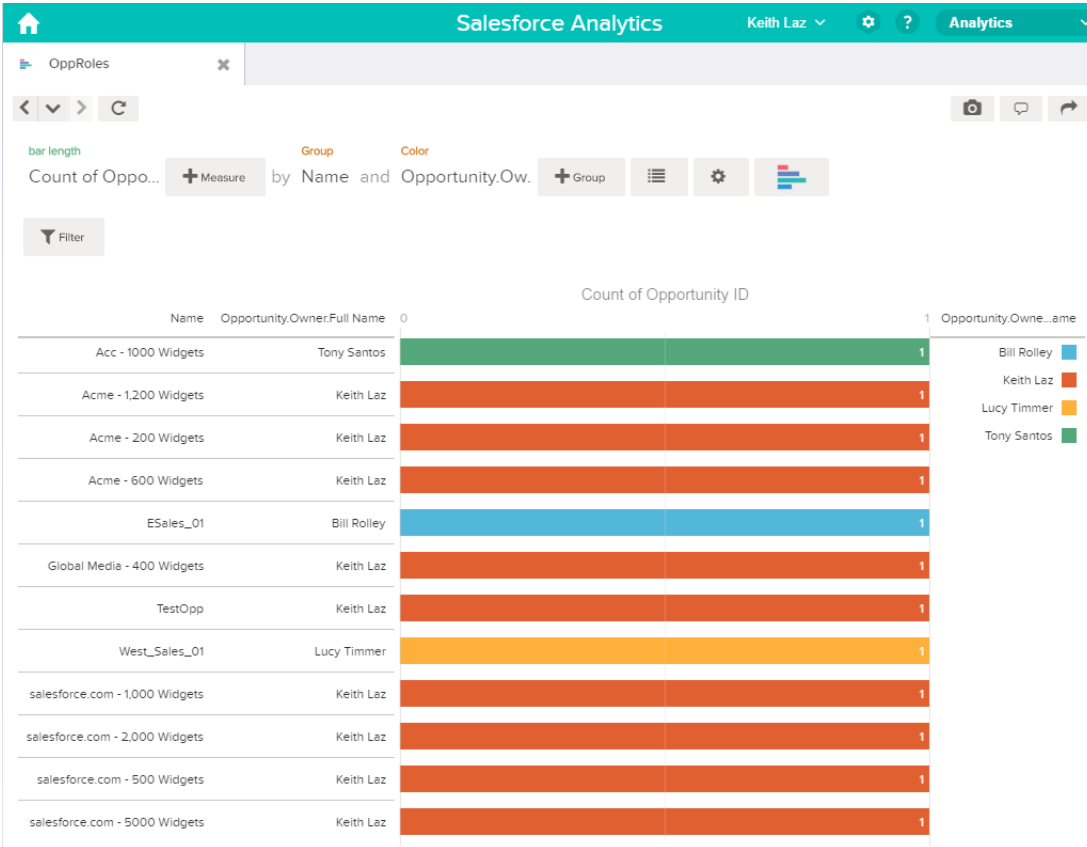
Available for an extra cost in: **Enterprise, Performance, and Unlimited Editions**

USER PERMISSIONS

- ""
- ""



3. Log out and now log in as Keith.
As expected, Keith can still see all opportunities.



Predicate Expression Syntax

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](#)

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.	<ul style="list-style-type: none"> • <code>'Owner' == "Amber"</code> • <code>'Stage Name' == "Closed Won"</code>
number literal	Can be a float or long datatype. Do not enclose in quotes.	<ul style="list-style-type: none"> • <code>'Expected_Revenue' >= 2000.00</code> • <code>'NetLoss' < -10000</code>

Value Type	Requirements	Predicate Expression Examples
field value	<p>When referencing a field from the User object, use the <code>\$User.[field]</code> syntax. Use the API name for the field.</p> <p>You can specify standard or custom fields.</p> <p>When you define a predicate for a dataset, you must have read access on all User object fields used to create the predicate expression.</p> <p>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.</p> <p> Note: By default, the Security User does not have access permission on custom fields of the User object.</p> <p>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.</p>	<ul style="list-style-type: none"> • <code>'Owner.Role' == "\$User.UserRoleId"</code> • <code>'GroupID' == "\$User.UserGroupId__c"</code>

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
<code>\b</code>	One backspace character
<code>\n</code>	New line
<code>\r</code>	Carriage return
<code>\t</code>	Tab
<code>\Z</code>	CTRL+Z (ASCII 26)
<code>\"</code>	One double-quote character
<code>\\</code>	One backslash character
<code>\0</code>	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: <code>'Expected_Revenue' >= 2000.00</code>
"	Double quote	Encloses a string value or field value in a predicate expression. Example predicate expression: <code>'OpportunityOwner' == "Michael Vesti"</code>
()	Parentheses	Enforces the order in which to evaluate a predicate expression. Example predicate expression: <code>('Expected_Revenue' > 4000 'Stage Name' == "Closed Won") && 'isDeleted' != "False"</code>
\$	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: <code>'Owner.Role' == "\$User.UserRoleId"</code>
.	Period	Separates the object name and field name in a predicate expression. Example predicate expression: <code>'Owner' == "\$User.UserId"</code>

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: <code>'Stage Name' == "Closed Won"</code>
!=	Not equals	True if the operands are not equal. String comparisons that use the not equals operator are case-sensitive. Example predicate expression: <code>'isDeleted' != "False"</code>
<	Less than	True if the left operand is less than the right operand. Example predicate expression: <code>'Revenue' < 100</code>
<=	Less or equal	True if the left operand is less than or equal to the right operand.
>	Greater than	True if the left operand is greater than the right operand.
>=	Greater or equal	True if the left operand is greater than or equal to the right operand.
in	Multi-value list filter	True if the left operand exists in the list of strings substituted for a multi-value picklist (field value). Example predicate expression: <code>'Demog' in ["\$User.Demographic__c"]</code> In this example, <code>Demographic__c</code> is of type <code>multiPicklistField</code> . 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

Review the samples to see how to structure a predicate expression.

The samples are based on the following Opportunity dataset.

Opportunity	Expected_Rev	Owner	OwnerRoleID	Stage_Name	IsDeleted
OppA	2000.00	Bill	20	Prospecting	True
OppB	3000.00	Joe	22	Closed Won	False
OppC	1000.00	可爱的花	36	Closed Won	False
OppD	5000.00	O'Fallon	18	Prospecting	True
OppE		Joe	22	Closed Won	True


Let's take a look at some examples to understand how to construct a predicate expression.

Predicate Expression	Details
'OwnerRoleID' == "\$User.UserRoleId"	Checks column values in the User object.
'Expected_Rev' > 1000 && 'Expected_Rev' <= 3000	
'Owner' = "Joe" 'Owner' = "Bill"	
('Expected_Rev' > 4000 'Stage Name' == "Closed Won") && 'isDeleted' != "False"	Parentheses specify the order of operations.

Predicate Expression	Details
'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](#)

Start, Stop, and Reschedule Dataflow Jobs

The dataflow job runs on a daily schedule. You can reschedule the time that the job runs. If you don't want to wait for the scheduled run, you can start a job to load the data immediately. In addition, you can stop a job.

 **Note:** You cannot run more than one instance of a dataflow at a time.

1. In Wave Analytics, click the gear icon () and then click **Data Monitor** to open the data monitor.
The data monitor has two views: Dataflow view and Job view. By default, the Dataflow view displays.
2. To perform tasks on the dataflow, click the actions list (1) for the dataflow.

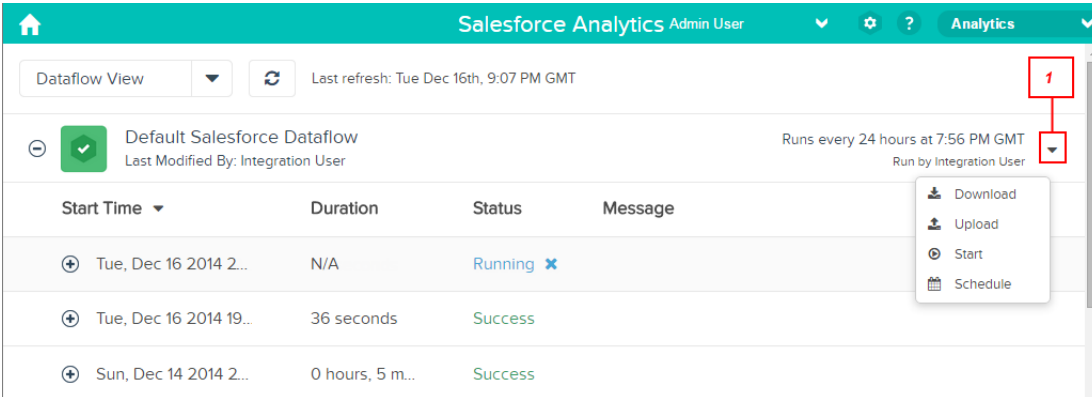
USER PERMISSIONS

To access the data monitor:

- "Edit Wave Analytics Dataflows" or "Manage Wave Analytics"

To start a dataflow job:

- Edit Wave Analytics Dataflows



The actions list displays the following options: **Download**, **Upload**, **Start**, and **Schedule**.

3. Click **Start** in the actions list to run the dataflow job now.
The dataflow job is added to the job queue. The **Start** button is greyed out while the dataflow job is running.
4. To stop a dataflow job that is currently running, click **X** 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.
5. Click **Schedule** in the actions list to change the time the dataflow is scheduled to run.

Data Monitor

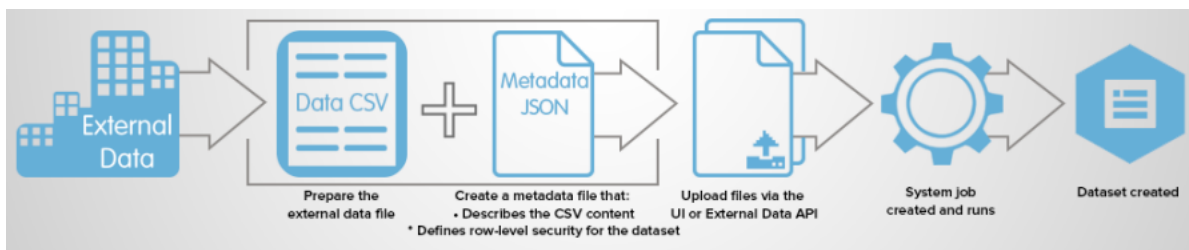
The data monitor provides options for you to manage jobs.

SEE ALSO:

- [Start the Dataflow](#)
- [Reschedule the Dataflow](#)
- [Monitor a Dataflow Job](#)
- [Monitor an External Data Upload](#)

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.

[Get Oriented in Wave Analytics](#)

Go to Wave Analytics by opening the Force.com app menu in the Salesforce header and then clicking **Wave Analytics**.

[Collections of Data in Wave Analytics](#)

Several kinds of data collections are available from the home page. Datasets, lenses, dashboards, and apps represent levels of data refinement—everything 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, or downloading a screenshot of it. 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 screenshot option.

[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.

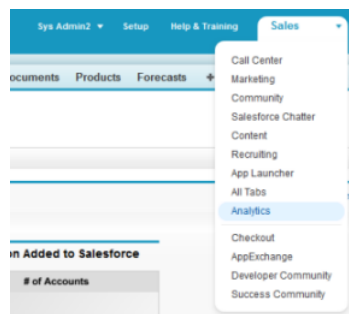


- View—View the data set.
- Explore—Get to know the boundaries and shape of your data. Play with different views, zoom in and zoom out, and see what you get with different charting options.
- Refine—Narrow your view to the most important data. Decide the level of detail and categorization that's easiest 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.

Get Oriented in Wave Analytics

Go to Wave Analytics by opening the Force.com app menu in the Salesforce header and then clicking **Wave Analytics**.

1. Open Wave Analytics.



The home page is your home base for exploring all the data that you have access to.

Collections of Data in Wave Analytics

Several kinds of data collections are available from the home page. Datasets, lenses, dashboards, and apps represent levels of data refinement—everything from raw data that's uploaded from your source systems to highly curated, packaged views of your data.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To access Wave Analytics:

- "Use Wave Analytics"

- 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* 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.


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.

[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.

[Return to a Previous View by Using History](#)

Exploring your data can lead you down many paths. Not all of them will be fruitful, and that's OK. Explorer keeps a full history of your activity in a lens. If you've changed your lens in undesired ways, or just want to see those nifty animations again, use history to backtrack quickly to a previous state in your visualization.

[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.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To view visualizations:

- "Use Wave Analytics"

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.

1. 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

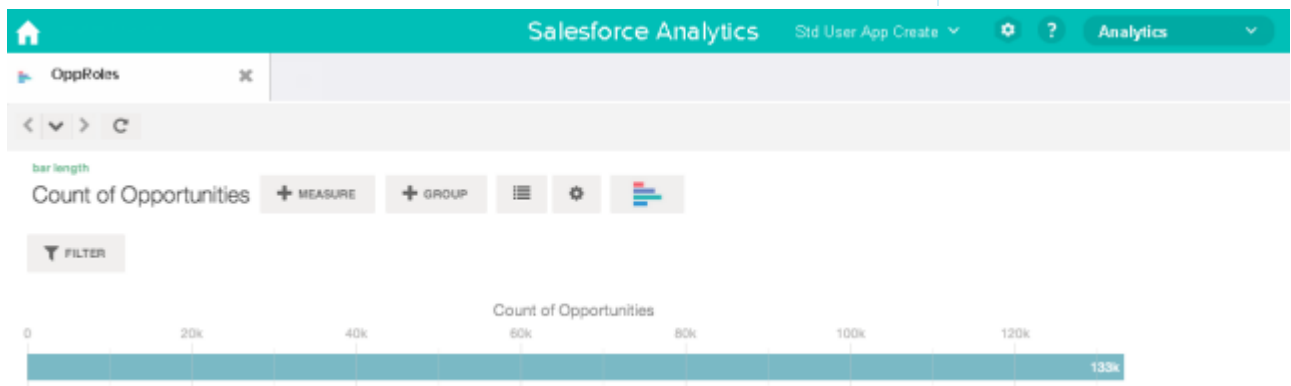
Available in: **Developer Edition**

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

USER PERMISSIONS

To view visualizations:

- "Use Wave Analytics"



Group Your Data into Categories

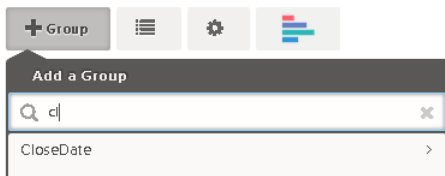
Group data to organize it so that the data is displayed in useful categories. For example, to look for seasonal patterns in your sales cycle, group opportunity data by the month when deals close.

1. Click the **Group** button.

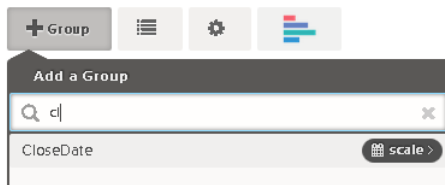


2. Using the search box, type the first few letters of the dimension that you want to group by, such as Close Date.

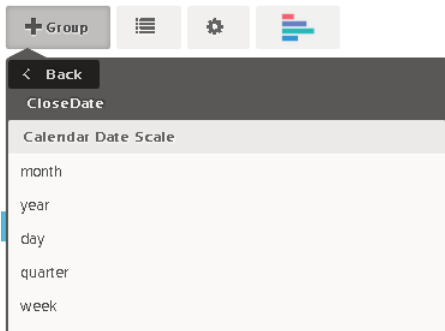
The dimensions in your data determine which categories you can group by.



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.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

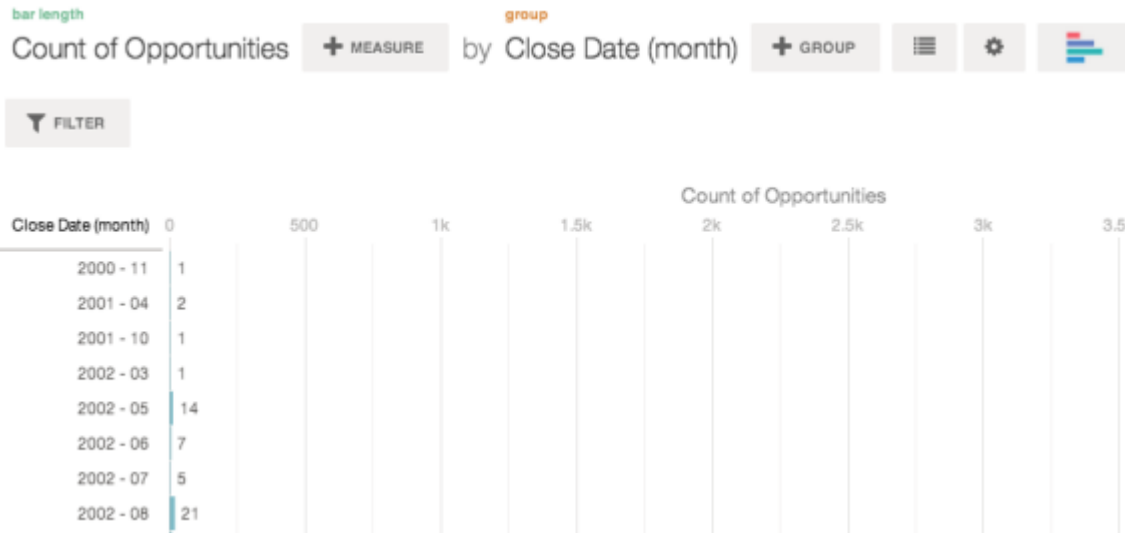
Available in: **Developer Edition**

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

USER PERMISSIONS

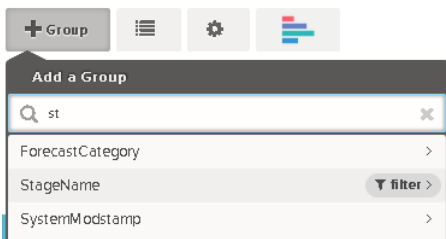
To view visualizations:

- "Use Wave Analytics"



- 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.



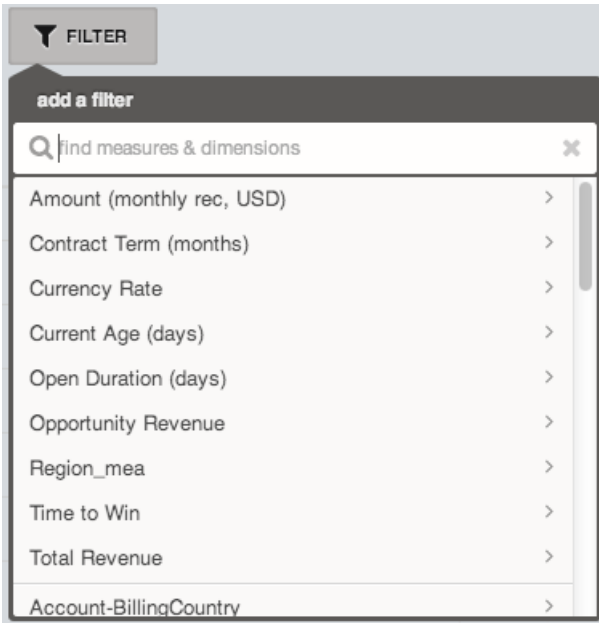
Filter Your Data

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

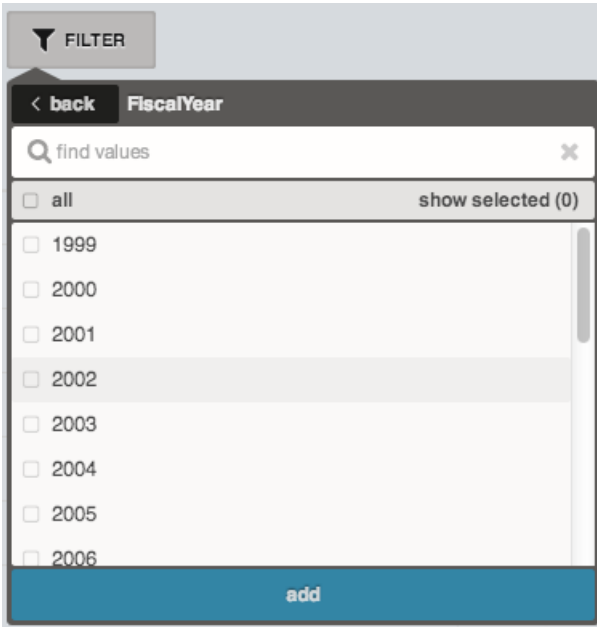
Available in: **Developer Edition**

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

USER PERMISSIONS

To view visualizations:

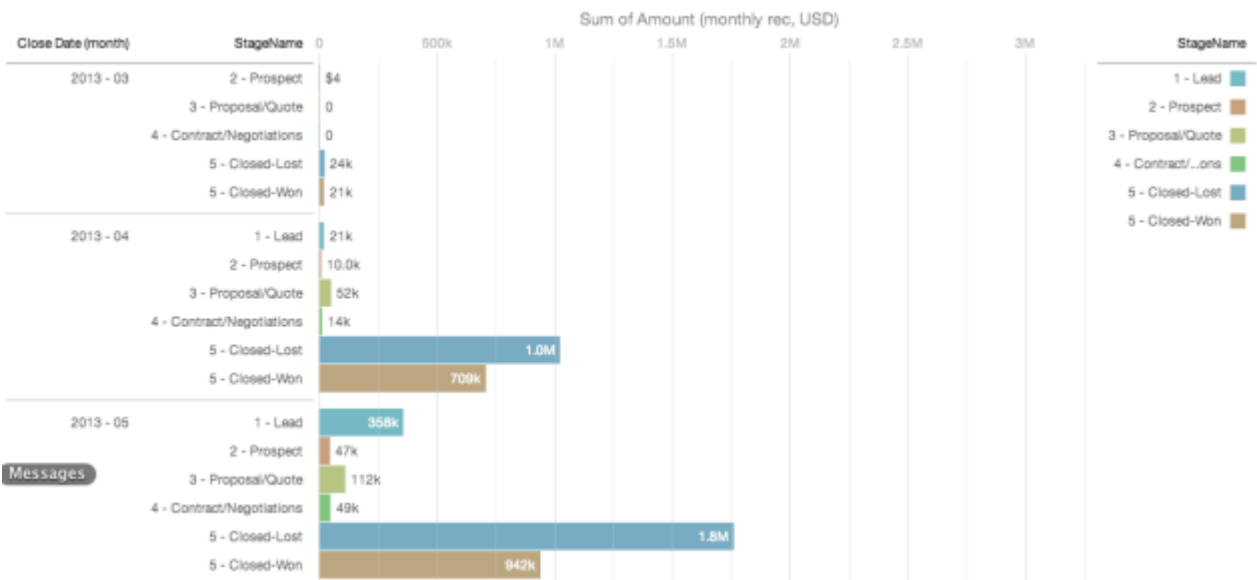
- "Use Wave Analytics"



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).

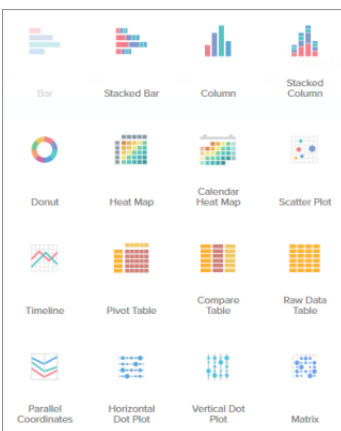
Change the Chart Type

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. Only visualizations that are valid for the underlying query are displayed.



2. Hover over a chart type to see how many measures and dimensions that type of visualization is suitable for. For example, a pie chart can have one measure and one or two dimensions.
3. Click a chart type, such as Stacked Bar.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To view visualizations:

- "Use Wave Analytics"

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

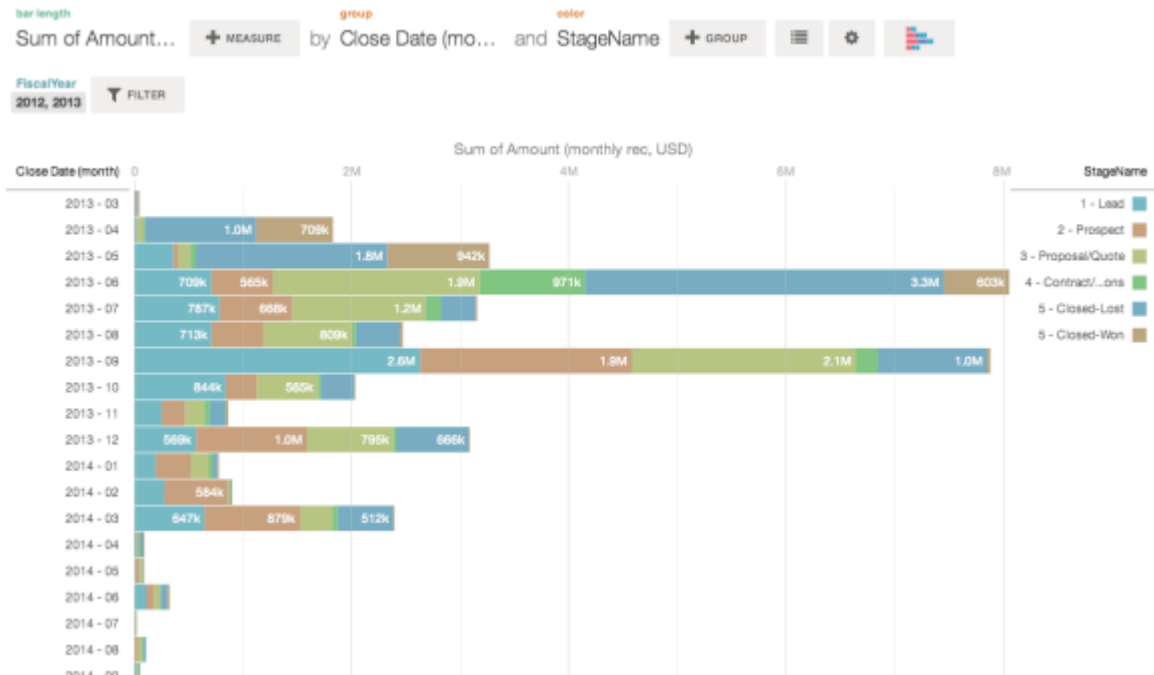
Available for an extra cost in: **Enterprise, Performance, and Unlimited** Editions

USER PERMISSIONS

To view visualizations:

- "Use Wave Analytics"

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 in: **Developer Edition**

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

USER PERMISSIONS

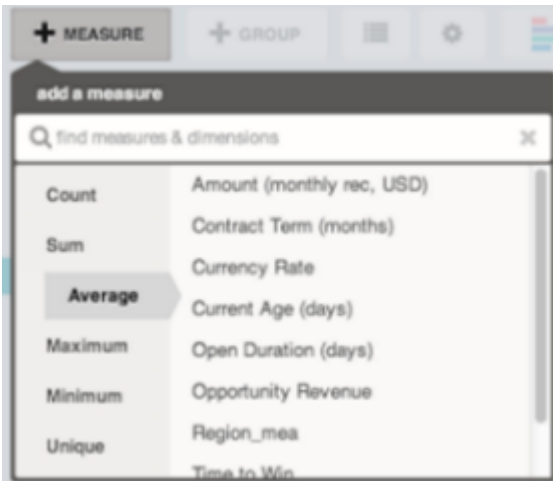
- To view visualizations:
- “Use Wave Analytics”



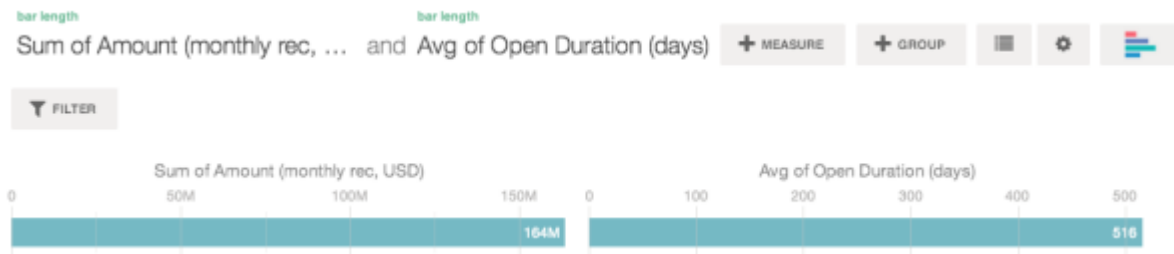
- To add a second measure, click the **Measure** button.



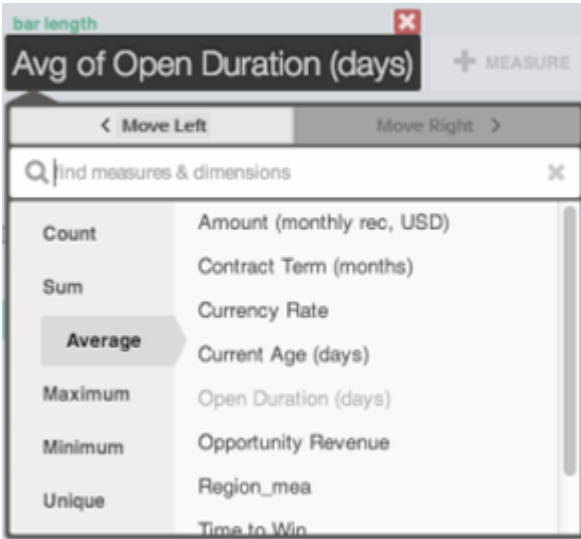
- Click the aggregation **Average**, and then click the Open Duration (days) measure.



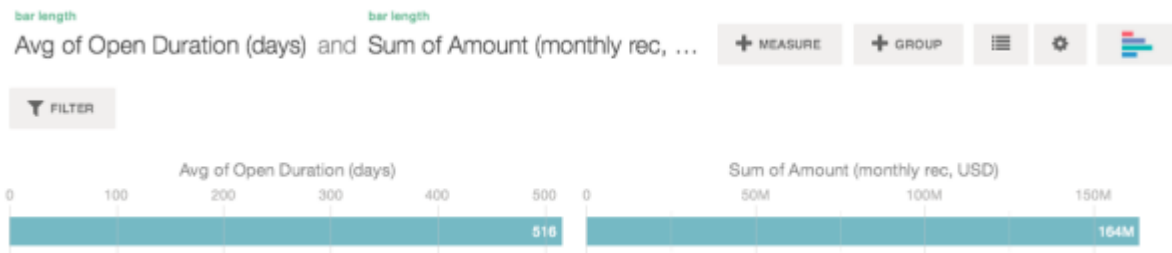
The new measure is displayed on the right.



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



The two visualizations swap positions.



- 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.

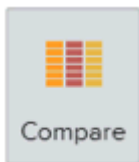


2. In the dialog box, choose a scale.

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**.
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 in: **Developer Edition**

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

USER PERMISSIONS

To view visualizations:

- "Use Wave Analytics"

EDITIONS

Available in: Salesforce Classic and Lightning Experience

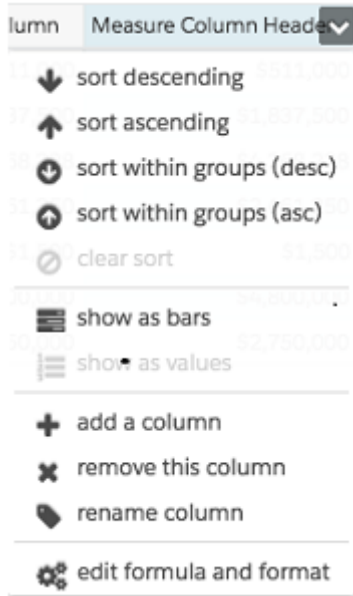
Available in: **Developer Edition**

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

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**.

Opportunity Owner	Opportunity Amount ^A	Expected Revenue ^B	Expected Revenue ^C
Ely East	134,381,275		
Bill West	89,291,860		
James Barry	78,821,529		
Cindy Central	77,106,000		
Kasey Central	76,590,000		
Ricky East	74,472,500		
Wendy West	69,786,500		
Tom Reseller	43,385,600		
Jamie Preston	9,355,000		
Sean Silver	4,060,000		
Barry Bronze	4,060,000		
Ted Data	3,039,560		

Notice that 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 formula and format** from the drop-down menu.

Edit Column Details

Enter a custom column name, define a formula for the column, or set the data format for the column...

column name


formula

data format

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

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

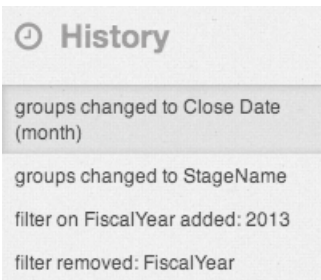
Symbol	Examples	Calculation
+	A+B, A+100	Addition, applied per row.
-	A-B, A-100	Subtraction, applied per row.
/	A/B, A/100	Division, applied per row.
*	A*B, A*100	Multiplication, applied per row.
()	(B-A)/(A*100)	Parentheses for grouping operations.



 **Tip:** Use Compare Tables for simple, ad hoc calculations. If a particular calculated value is needed frequently or is more complex, consider performing the calculation at the source data level.

Return to a Previous View by Using History

Exploring your data can lead you down many paths. Not all of them will be fruitful, and that's OK. Explorer keeps a full history of your activity in a lens. If you've changed your lens in undesired ways, or just want to see those nifty animations again, use history to backtrack quickly to a previous state in your visualization.

1. Click the **View History** button  in the header. A sidebar lists a history of all the actions that you've taken in the lens.



2. Select any item in the list to go to that point. Or use the buttons to **Step Backward**  and **Step Forward**  through the list.

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.



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.



EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To view visualizations:

- "Use Wave Analytics"

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**


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

USER PERMISSIONS

To view visualizations:

- "Use Wave Analytics"

- 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

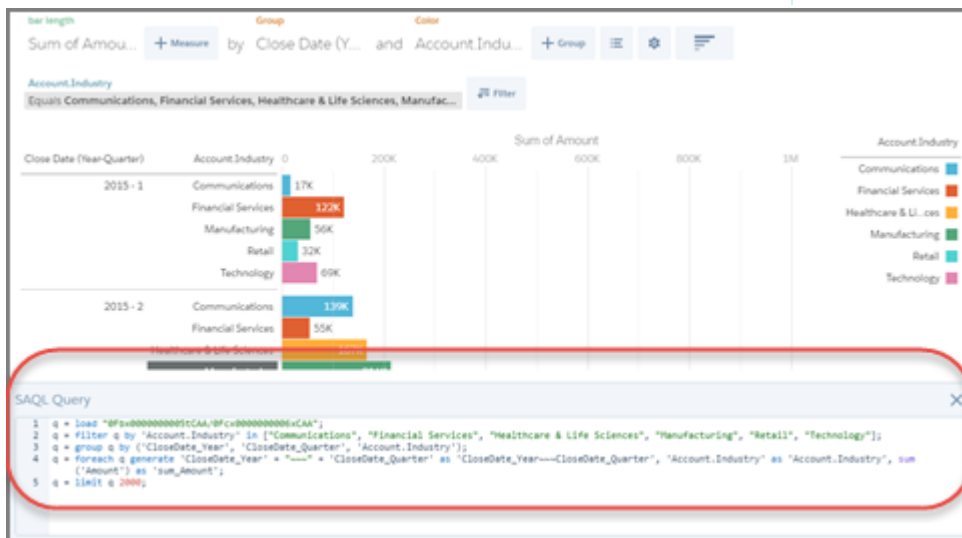
Wave Analytics uses Salesforce Analytics Query Language (SAQL) behind the scenes in lenses and dashboards to gather data for visualizations. Open the SAQL viewer to watch the underlying SAQL query being written as you explore or to copy it for use elsewhere.

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



- Select **On**.

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



For more information about SAQL, see the [Wave Analytics SAQL Reference](#).

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To view visualizations:

- "Use Wave Analytics"

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.

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 in: **Developer Edition**

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

USER PERMISSIONS

To view visualizations:

- "Use Wave Analytics"

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

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, or downloading a screenshot of it. 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 screenshot option.

1. Click **Share**.



2. Click the tab for the sharing method that you want:
 - a. **Post to Feed** posts to Chatter an image and link to the lens, dashboard, app. Select **User** or **Group** feed, enter the name, and then type your comment. You can also remove the image from your post by hovering over it and clicking the **x**. Note that posted images are public.
 - b. **Get URL** provides a unique URL to the asset. Copy the link and then paste it wherever you want to share it.
 - c. **Get Screenshot** gives you a static image to share without giving access. Click the image to download it as a .png file.
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**.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To view visualizations:

- “Use Wave Analytics”

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To view visualizations:

- “Use Wave Analytics”

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 database query.

Selectors

A selector lets you filter down to a particular value that matters to you. For example, you can set the maximum and minimum sliders in a range selector to make the dashboard show only the amounts that fall within a specific range. Watch the charts and numbers on the page change when you set the sliders. Selectors are often tied to each other in sequence so that you can progressively drill deeper. For example, you can use one selector to look at sales of a product, then use the next selector to filter sales by region, and then use the next selector to filter sales regions by account. There are different types of selectors, such as list, selectors, toggle, date, and range selectors.

Key Performance Indicators

Some information is best represented by a single number. Some number elements can be affected by filtering actions that you take on charts or selectors. For example, in a bar chart that shows individual salespeople's pipelines, clicking a specific bar can cause a number element to change to reflect the specific bar's value instead of the total value.

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.

Build Wave Dashboards

You've explored your data, picking out the useful themes and patterns. You've created one or more lenses. Now you can design a dashboard to help your audience turn data into decisions.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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



Here's what a dashboard can look like when it's fully fleshed out. We'll show you step by step some of the basic concepts to get started creating dashboards, along with some best practices and design tips.

No two dashboards are quite the same—every dashboard tells a unique story. The universe of stuff that you can show in a dashboard is infinite, so we won't try to tell you exactly what to do.



Tip: Although steps are laid out in order, you don't have to follow the sequences exactly, and some steps are optional.

IN THIS SECTION:

1. [Plan Your Dashboard](#)

Building a dashboard that delivers real, ongoing value requires planning and making choices. Plan your dashboard to help narrow your audience's focus to a manageable set of questions for basing actions upon, and to support ongoing updates so that your audience can see a coherent picture over time.

2. [Sketch Your Dashboard](#)

It's usually a good idea to sketch your dashboard on paper or a whiteboard before you start building. Doing so helps you anticipate ways that your audience might use the dashboard and address potential gaps.

3. [Build the Dashboard](#)

When you've thought through the purpose and design of your dashboard, you're ready to create the dashboard and assemble its pieces.

4. [Convert Your Dashboard to the Flex Designer \(Beta\)](#)

The Flex Dashboard Designer offers snap-to-grid for quickly creating visually aesthetic dashboards, and improved drag-and-drop for positioning widgets on the designer canvas. When resizing the browser, column widths automatically adjust, while maintaining their relative sizes to each other. To increase your workspace, hide lenses and property panels while editing a dashboard. And the Flex Designer offers integrated tools for creating lenses within the designer.

SEE ALSO:

[What Is a Dashboard?](#)

Plan Your Dashboard

Building a dashboard that delivers real, ongoing value requires planning and making choices. Plan your dashboard to help narrow your audience's focus to a manageable set of questions for basing actions upon, and to support ongoing updates so that your audience can see a coherent picture over time.

It's a good idea to start with a written plan for the dashboard. Use your plan to define the dashboard's focus up front, and then stick to the plan. Identify the specific question, business process, or user group that the dashboard will support, and eliminate anything that doesn't support those.

1. Who is the audience?

Will the dashboard be used by the executive team, the marketing team, a sales team, or another group? It's important to ensure that each dashboard presents data that's specific to a single audience.

For example, for a sales manager planning a dashboard for a weekly sales team meeting, the audience is the attendees of that meeting.

2. What does the audience need to measure?

What questions does the audience need to answer to meet its goals? Continuing our sales manager example, let's assume that the audience wants to know what each salesperson's pipeline looks like and how the overall pipeline is doing.

3. How often does the audience need to update its knowledge?

Does the executive team need quarterly financials? Does the marketing team want to monitor daily lead generation numbers?

To understand some of the key tasks involved in creating a dashboard, in the next sections, we'll create a dashboard for a sales team that meets weekly to go over its sales pipeline status.

SEE ALSO:

[What Kind of Dashboard Should I Create?](#)

Sketch Your Dashboard

It's usually a good idea to sketch your dashboard on paper or a whiteboard before you start building. Doing so helps you anticipate ways that your audience might use the dashboard and address potential gaps.

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.

Consider these widely accepted best practices as you sketch out your own dashboard.

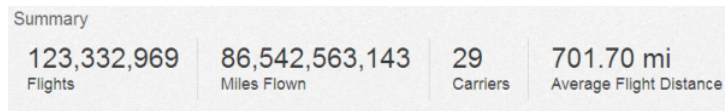
1. Prioritize elements, top left to bottom right.

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.

- a. Place more general, actionable information in charts near the top left, and place charts with supporting information lower.
- b. Group filters together at the top or left so that they are quickly noticeable.
- c. 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.

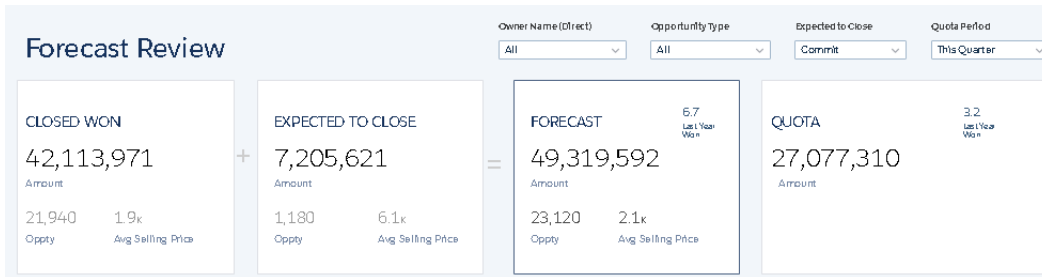
Highlight facts that the audience wants to know at a glance.



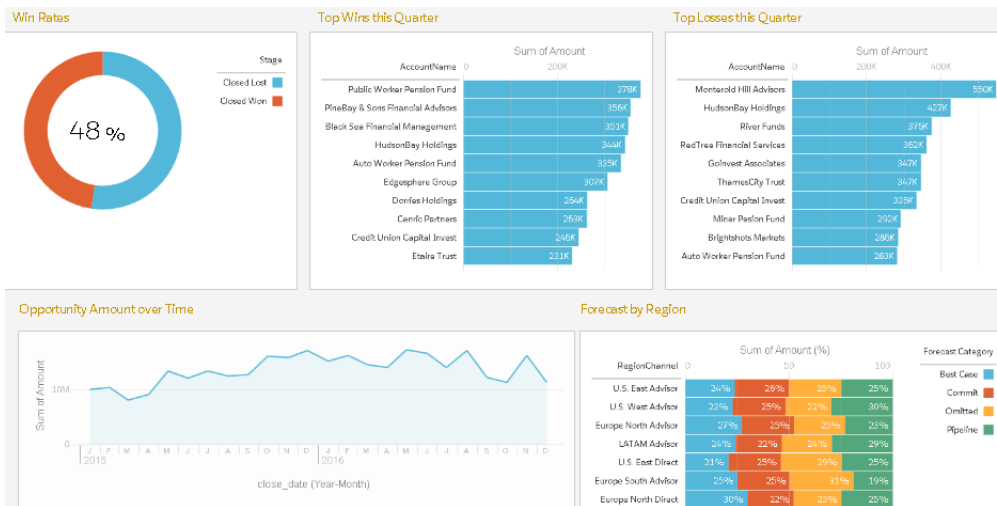
2. Choose chart types based on the characteristics of the data, not for look or variety.

For example, if most of your charts display value changes over time, it’s OK if they’re all line graphs.

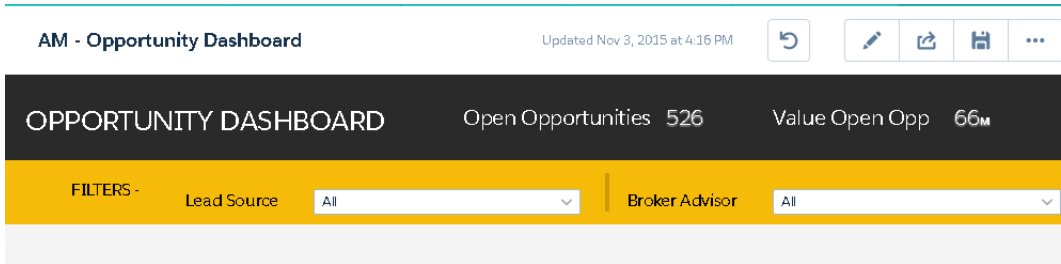
3. Use containers to frame and organize related elements in the dashboard.



4. Apply labels to sections and charts to annotate the dashboard.



5. Use colors to define sections.



6. Don't overload the dashboard—leave some empty space.

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.

Build the Dashboard

When you've thought through the purpose and design of your dashboard, you're ready to create the dashboard and assemble its pieces.

A Wave dashboard is a collection of widgets that work together to tell a data story from multiple angles. Widgets are the units that contain the information that you want to show to your audience. There are different types of widgets, such as visualizations based on queries, tables, filters, or images.

In the next sections, we'll take a look at an example that illustrates how to build a sales pipeline dashboard with different types of widgets.

IN THIS SECTION:

1. [Create the Dashboard](#)

When you create a 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 original dashboard designer or, if your org has enabled it, the new flex dashboard designer. In our example, we'll use the original designer.

2. [Add Key Performance Indicators to the 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 very useful. Add Number widgets to provide key indicators to a dashboard.

3. [Add Visualizations to the 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 are filtered in the same way. You can turn faceting on or off for each widget.

5. [Add Images to the 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 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. [Refine the Dashboard](#)

Use visual groupings, colors, labels, and links to clean up your dashboard and focus your audience's attention.

8. [See Details About a Widget's Lens](#)

To ensure that a widget is showing the right data, review details about the widget's lens.

9. [Filter or Remove Clipped Lenses from the 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:

[Wave Keyboard Shortcuts](#)

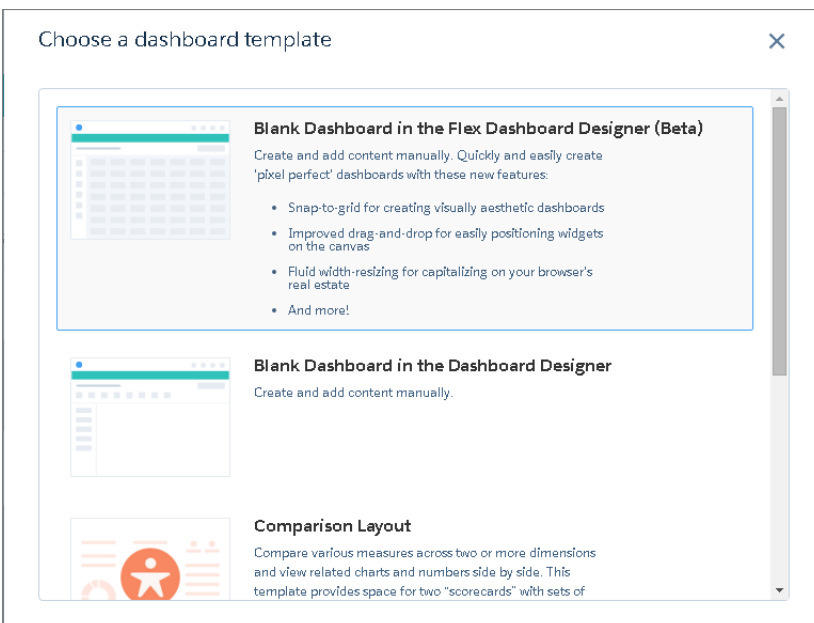
Create the Dashboard

When you create a 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 original dashboard designer or, if your org has enabled it, the new flex dashboard designer. In our example, we'll use the original designer.

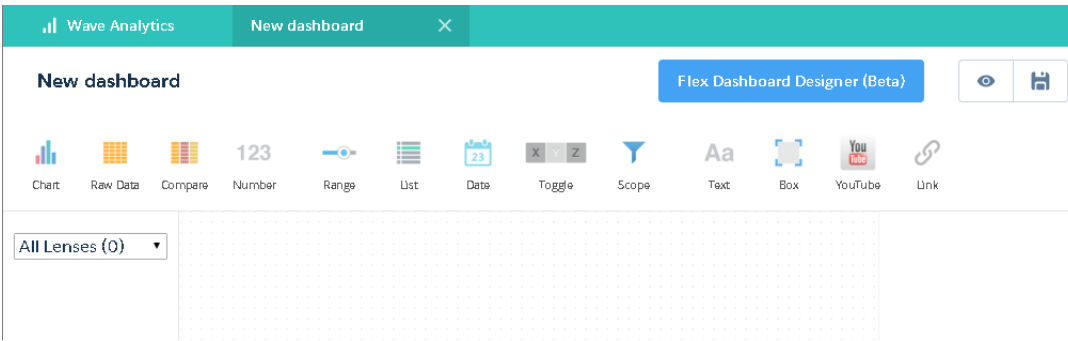
This topic is about creating dashboards in Wave Analytics. For information on creating dashboards for reports, review the article [Create a Salesforce Classic Dashboard](#).


 **Note:** To learn about differences between the original designer and the new flex dashboard designer, see the Spring '16 Release Notes.

1. From the home page, click **Create > Dashboard**.
The layout templates appear.



2. Choose a layout template, or select an option to create a blank dashboard in the original or new dashboard designer. In this case, we'll create a blank dashboard using the original designer..
3. Select **Blank Dashboard in the Dashboard Designer**, and then click **Continue**.
The designer opens.



- Click the Save button () to save the dashboard to an app.

Save ✕

Title *

Description


App

[Save](#)

- Click **Save**.

Add Key Performance Indicators to the 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 very useful. Add Number widgets to provide key indicators to a dashboard.

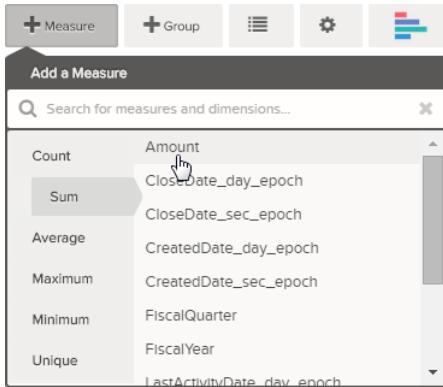
- Open the Opportunities dataset, and click  to calculate the total value of all opportunities in the pipeline.
 - In the measure picker, select `sum` from the left column.
 - Pick a dataset field from the right column that you would like to sum. In this case, select `Amount`.

EDITIONS

Available in: Salesforce Classic and Lightning Experience


Available in: **Developer Edition**

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



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  to clip this lens to the designer.

Tip: If you have multiple dashboards, the lens is clipped to whichever dashboard you had open last.

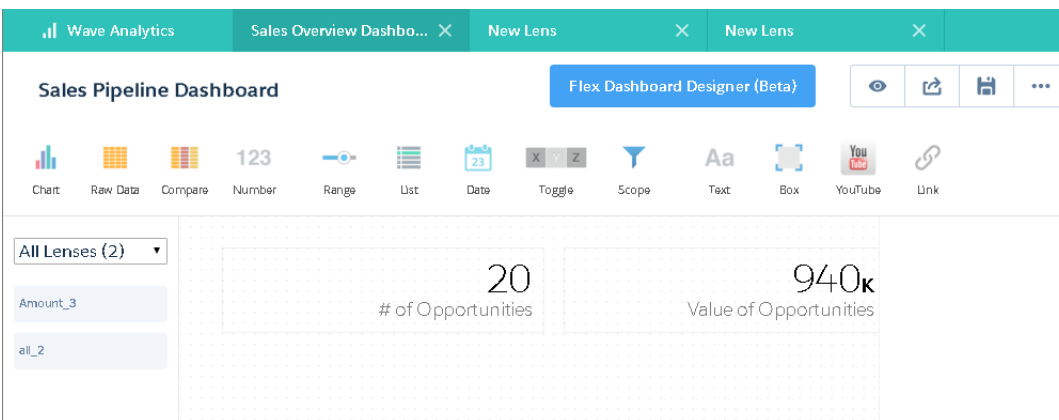
4. In the dashboard designer, click the number widget (`123`) to create an empty number widget, and then select the first clipped lens that contains a sum of amount. The number that appears in the number widget is the first measure (on the far left) that you specified in the lens.

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 Dashboard

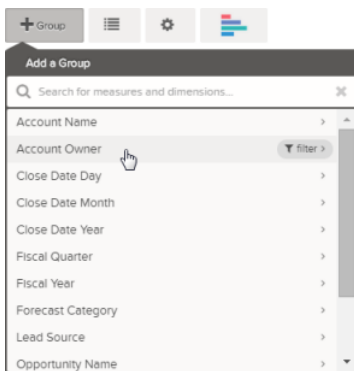
Add charts to visualize and interact with the data to tell a story about it. Each chart is based on the underlying query in a lens.


In the next steps, we'll create two interactive charts 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.

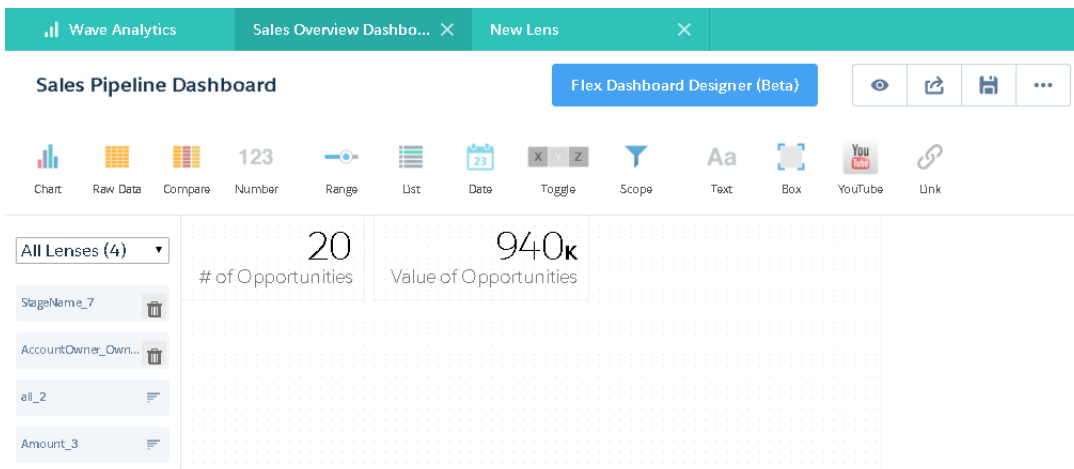
1. Verify that the Sales Pipeline dashboard is open.
You must have the dashboard open to clip lenses to it.

 **Note:** If no dashboard is open, Wave creates a new dashboard.

2. From the home page, open the Opportunities lens.
3. Group the data by Account Owner.



4. Click  to clip the lens to the dashboard.
5. 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 designer.
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 designer.



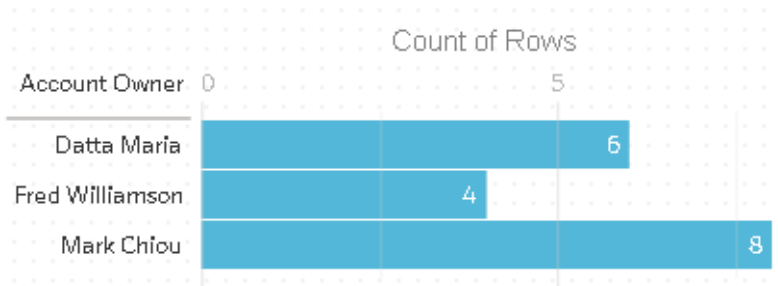
EDITIONS

Available in: Salesforce Classic and Lightning Experience

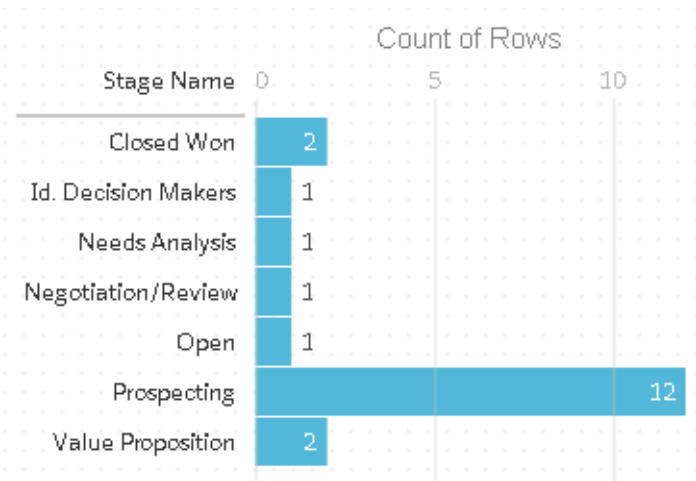
Available in: **Developer Edition**

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

- In the dashboard 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.

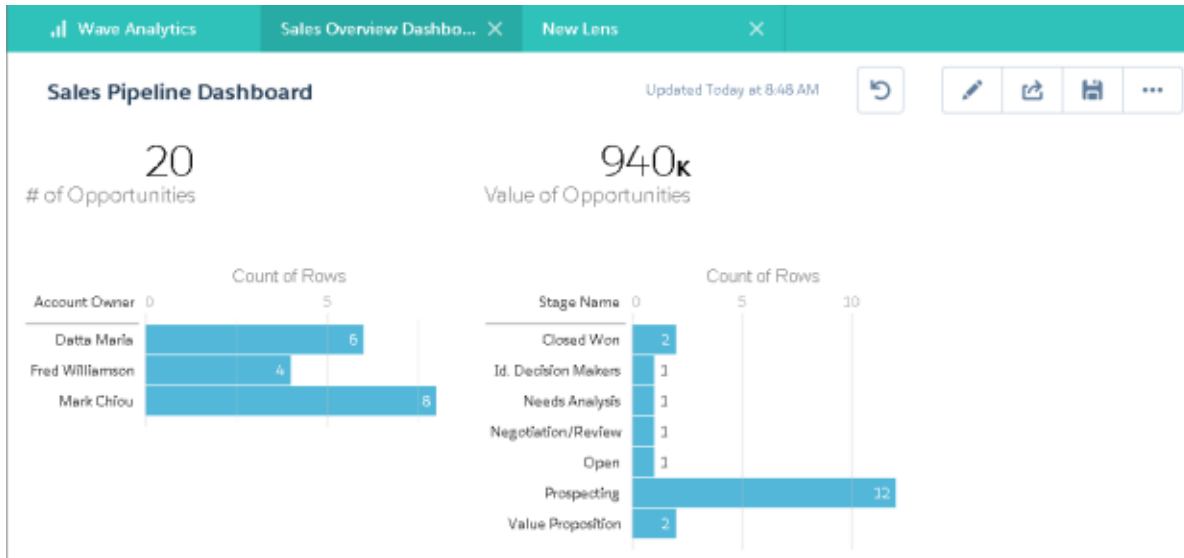


- Create another chart based on the other clipped lens—the one grouped by stage name.
The new chart widget shows how many opportunities are in each stage.



- Click  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 are filtered in the same way. You can turn faceting on or off for each widget.

Now we will look at how faceting works with the number and chart widgets that we added to our dashboard.

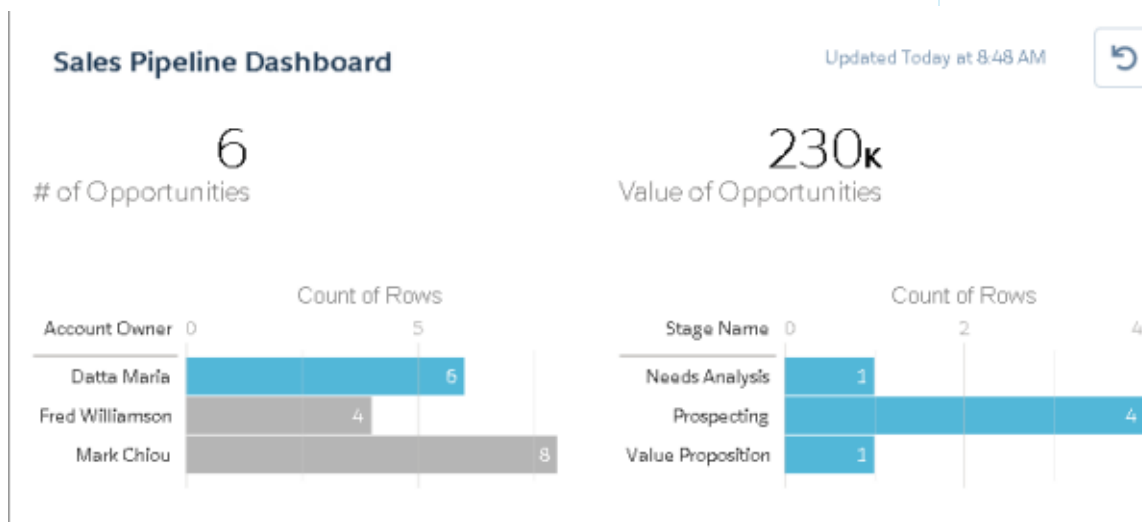
1. In the Sales Pipeline dashboard, click the first bar in the chart that shows the number of opportunities grouped by account owner. Notice that first bar is now highlighted and the results of the other widgets changed. Because of faceting, the number widgets and the other chart now show the results 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.

EDITIONS

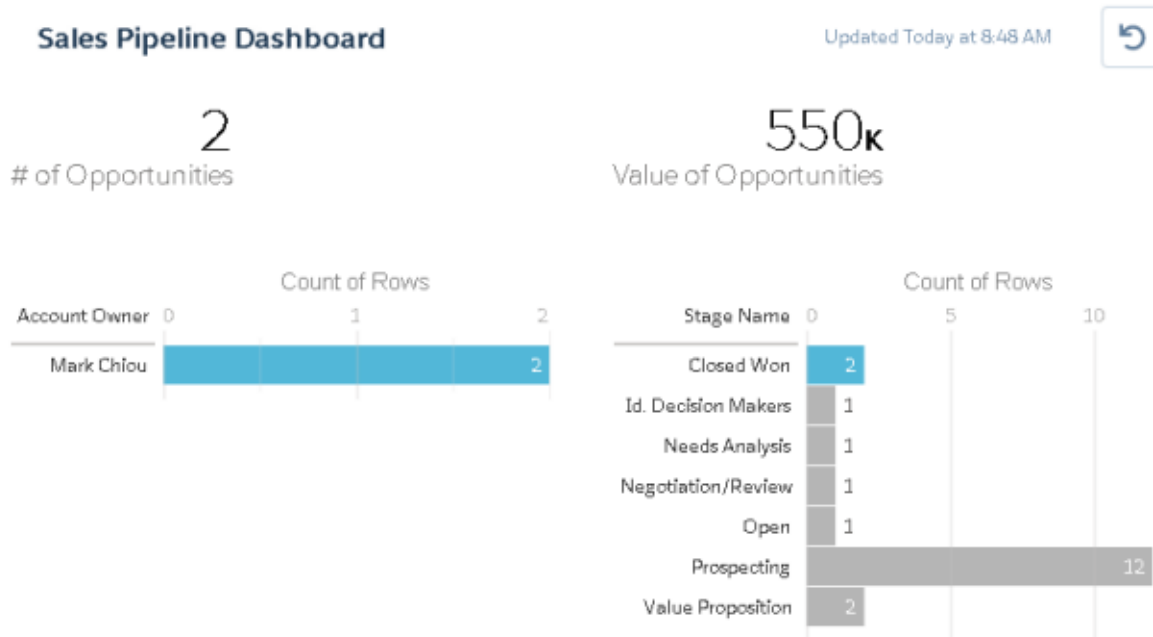
Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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



- 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.
- 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.



- 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 JSON Guide](#).

Add Images to the 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.

- 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.
- Copy the record ID for the document in the URL in the address bar.
- In the dashboard, click the box widget and paste the record ID into the **Image Document ID** field in widget properties.

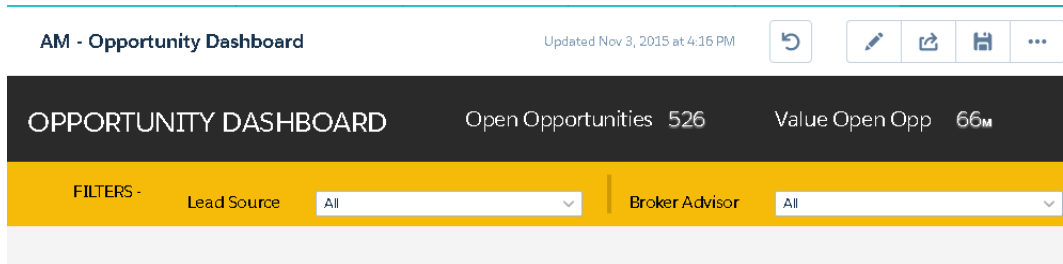
Note: You can also use the box widget to shade a section of the dashboard or add a background color to the dashboard. For example, you can use colored areas to highlight key metrics and filters.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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



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 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 below.

Destination Type

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 and filters applied by a dashboard viewer are transferred to the linked lens or dashboard.

 **Note:** When transferring states, Wave Analytics checks to make sure that the destination has lenses that are compatible with the lenses from the source dashboard.

On a lens-by-lens basis, these conditions prevent dashboard state from being transferred:

- 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 message that makes up the clickable part of the link.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

Refine the Dashboard

Use visual groupings, colors, labels, and links to clean up your dashboard and focus your audience's attention.

Use the following tips to enhance the design of your dashboard:

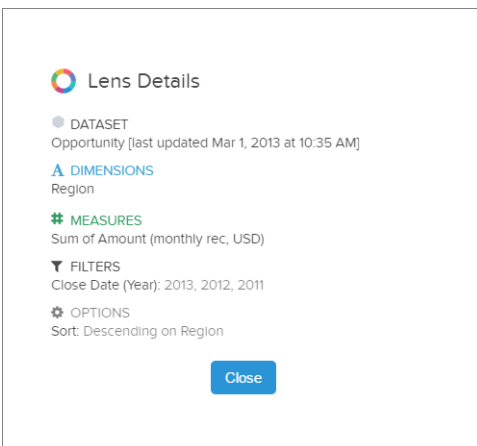
- Use the box widget in the original designer or the container widget in the flex dashboard designer to add background colors and banners, or to add your company logo to the dashboard. You can also use a border to group related widgets in a dashboard.
- Use text widgets to clearly annotate the dashboard, like labeling charts, sections, and filters.

See Details About a Widget's Lens

To ensure that a widget is showing the right data, review details about the widget's lens.

1. When viewing the dashboard in presentation mode, not edit mode, hover over a widget and

click  .



Lens details display.

2. Review the dataset and applied dimensions, measures, filters, and other details about the widget's lens.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

EDITIONS

Available in: Salesforce Classic and Lightning Experience

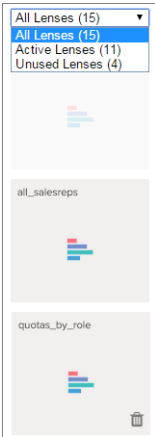
Available in: **Developer Edition**

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

Filter or Remove Clipped Lenses from the 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 .

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.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

Convert Your Dashboard to the Flex Designer (Beta)

The Flex Dashboard Designer offers snap-to-grid for quickly creating visually aesthetic dashboards, and improved drag-and-drop for positioning widgets on the designer canvas. When resizing the browser, column widths automatically adjust, while maintaining their relative sizes to each other. To increase your workspace, hide lenses and property panels while editing a dashboard. And the Flex Designer offers integrated tools for creating lenses within the designer.

If **Enable Flex Dashboard Designer (Beta)** is selected in the Wave Analytics settings, you can choose to use the new designer when creating or editing dashboards. You can also switch at any time by clicking the **Flex Dashboard Designer (Beta)** button when editing the dashboard.

The first time you open an existing dashboard in the Flex Designer, Wave makes a copy of the dashboard for the Flex Designer. After a dashboard is converted, you can no longer open it in the original designer. But with two copies, you can choose to work in either designer.

When the copy opens in the Flex Designer, widgets might shift in the designer canvas because of the new grid. If needed, realign them. The snap-to-grid feature makes it quick and easy. There are two additional known issues that can occur during conversion.

- Box widgets are removed. Workaround: Add a Container widget.
- YouTube widgets are removed. Workaround: Use a link widget to link to a YouTube video.

Key Differences Between the Original and the Flex Dashboard Designer (Beta)

Consider the following differences between the original and the Flex Dashboard Designer.

- In the Flex Designer, click the **Create New Lens** button in a dashboard to add a lens to the dashboard. For more information, see [Create Lenses in the Flex Designer](#).
- The Flex Designer uses snap-to-grid, letting you quickly align and resize widgets.

- The Flex Designer allows you to hide lenses and properties to give you more workspace when designing a dashboard.
- The YouTube widget is deprecated in the Flex Designer. If you open an existing dashboard, the YouTube widget is removed from the dashboard. If needed, you can use a Link widget to link to a YouTube video.
- The Container widget in the Flex Designer replaces the Box widget. You can add a background image to the Container widget, just as you can to the Box widget. But the Container widget extends the behavior of the Box widget by letting you group widgets within the container and move them all at once. If you open an existing dashboard in the Flex Designer, the Box widget is removed from the dashboard. If needed, you can add a Container widget.
- No widgets in the Flex Designer can overlap, except for a widget inside a Container widget.
- To add a widget in the Flex Designer, drag it onto the canvas, and then drop the lens on top. You can also drag a lens directly to the canvas—the widget uses the chart type specified for the lens.
- The following chart type widgets are not supported in the Beta release of the Flex Designer: Calendar Heat Map, Heat Map, Parallel Coordinates, Horizontal Dot Matrix, Vertical Dot Matrix, and Pivot Table. You can clip any chart type, whether or not it's supported. When you drag the widget onto canvas, the chart converts to a Horizontal Bar Chart, except for the Pivot Table which converts to a Compare Table.
- The Flex Dashboard Designer (Beta) doesn't support mobile layouts. The current version of Wave Analytics for iOS doesn't support dashboards created or saved in the Flex Dashboard Designer (Beta).

IN THIS SECTION:


[Create Lenses in the Flex Designer](#)

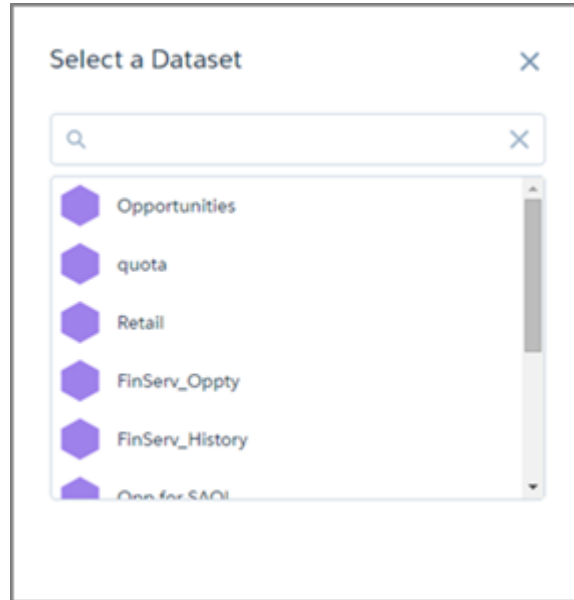
With the Flex Dashboard Designer (Beta), you can add lenses to your dashboard using tools within the designer. Convert the calculated columns of Compare Tables into visualizations, or customize your queries in the SAQL editor. Both approaches provide an easy way to set up your charts by specifying which measures to show and in what order.

Create Lenses in the Flex Designer

With the Flex Dashboard Designer (Beta), you can add lenses to your dashboard using tools within the designer. Convert the calculated columns of Compare Tables into visualizations, or customize your queries in the SAQL editor. Both approaches provide an easy way to set up your charts by specifying which measures to show and in what order.

To create lenses in the Flex Designer, follow these steps.

1. Click Show Lenses () to open the lenses panel.
2. Click **Create New Lens**.
3. Select a dataset. Datasets appear in most recently used order. Type in the search box if your dataset isn't easy to find.



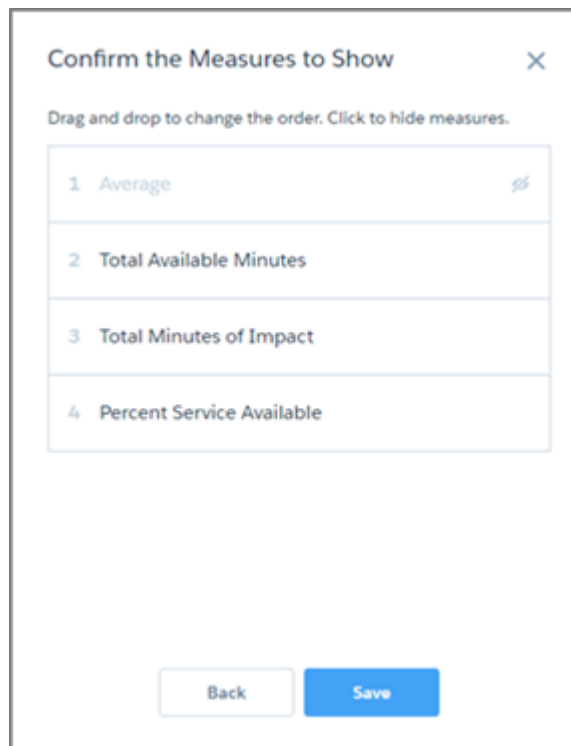
4. Explore and create visualizations.

From here, you have three choices.

a. If your lens is not a Compare Table, you can simply click **Save** and the lens is added to the designer's available lenses.

b. To use formula fields, select the Compare Table () from the charts gallery and create calculated columns.

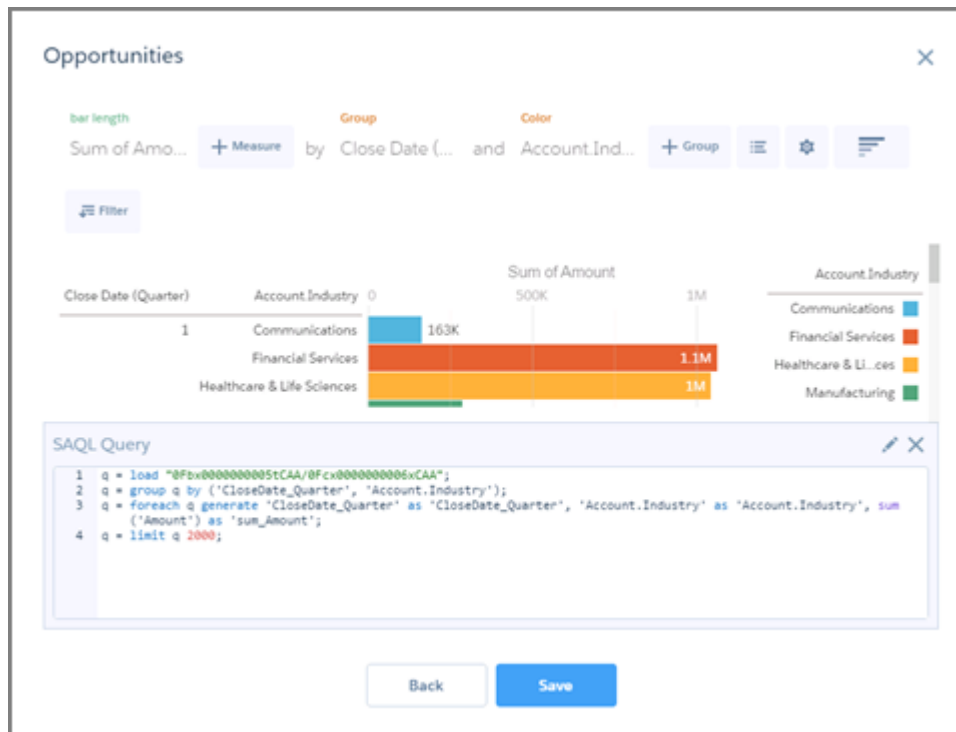
When you save, you're prompted to confirm or change how you'd like the columns with measures to appear in the visualization.



Slide the measures to change their order. You can also click a measure to hide or show it. The order affects how a measure is used in a visualization. For example, in a Scatter Plot, the first measure is used as the x-axis, the second as the y-axis, and the third as the size of the bubble.

From there, **Save** adds the lens to the designer's available lenses.

- c. To refine your query, select **Show SAQL** in the Options menu (⚙️), and click the Edit icon (✎️).



The SAQL editor provides contextually smart auto-completion suggestions. When you make changes in the SAQL editor, the **Next** button is disabled until you select **Run Query** and the editor verifies that the query is valid.

One use of a dashboard is to provide a focus for your audience. A dataset can contain vast amounts of data. Some of that data is relevant to specific questions that your audience has; some of that data is irrelevant noise. Data that's critical to one set of questions is sometimes noise in relation to another set of questions.

Keep in mind that even though we call it a dashboard, we're actually doing much more with this interface than we do with a traditional dashboard in a car. This dashboard doesn't just tell us what's changed; it invites us to dig deeper into the underlying data. It's like using the speedometer in a car to tell us not just how fast we're going but how our speed has changed at various points in our trip, how traffic conditions ahead of us are changing, and how those changes affect our arrival time.

SEE ALSO:

[Build Wave Dashboards](#)

What Kind of Dashboard Should I Create?

Dashboards can support many kinds of knowledge seeking: operational, strategic, or analytical. Sometimes a dashboard serves a combination of purposes, but it's a good idea to keep your dashboard focused on a single purpose.



Tip: Whatever their purpose, the best dashboards are self-contained and present a full picture so that the audience does not have to scroll or click to other resources.

Operational Dashboards

An operational dashboard is for keeping track of fast-changing data that can make a difference in day-to-day operations. An operational dashboard is consulted 24/7. Its audience expects real-time or near-real-time data.

For example, to monitor server uptime and utilization or to track the number of support cases that your call center takes each week, an operational dashboard provides the best solution.

Strategic Dashboard

A strategic dashboard presents an overview of business developments. Executives can use it to track key performance indicators and support decisions about overall direction. The audience expects a strategic dashboard to expose problems and opportunities, not to list issues for immediate resolution.

For example, a strategic dashboard for sales executives can show total sales figures grouped by region, total revenue for the quarter, or headcount grouped by department.

Analytical Dashboards

An analytical dashboard supports diving deep into operational or strategic data to find trends and make predictions. The audience wants to look at massive amounts of data, run what-if scenarios, and discover new questions that it hasn't thought of yet.

The audience typically works with an analytical dashboard on a monthly, quarterly, or even annual basis. Access to day-to-day updates isn't as important here as it is for operational or strategic dashboards.

SEE ALSO:

[Plan Your Dashboard](#)

Wave Keyboard Shortcuts

You can do some basic actions from your keyboard. The dashboard designer shortcuts vary based on whether you are using the original or flex dashboard designer.

Lens Keyboard Shortcut	Description
l	Clip to designer
s	Save
n	Clone in new tab
d	Delete
CRTL+E (Windows); CMD+E (Mac)	Open Expert Editor Mode (JSON editor)

Original Dashboard Designer Keyboard Shortcut	Description
Arrow keys	Move the selected item around.
x	Delete the selected item from the canvas.
[Put the selected item beneath other items.
]	Put the selected item on top of other items.

Flex Dashboard Designer Keyboard Shortcut	Description
Arrow keys	Move the selected item around.
c	Share the dashboard.
d	Delete the dashboard.
e	Toggles 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.

JSON Editor Keyboard Shortcut	Description
CRTL+E (Windows); CMD+E (Mac)	Switch to runtime
CRTL+3 (Windows); CMD+3 (Mac)	Reload JSON from runtime
CRTL+F (Windows); CMD+F (Mac)	Search (RegExp, case-sensitive, or whole word searches available)

SEE ALSO:

[Build the Dashboard](#)

Incorporate Wave Analytics Throughout Your Business

The great 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. In addition, custom menus in lenses and dashboards allow you to perform common Salesforce actions directly from Wave.

IN THIS SECTION:

[Embed Your Dashboard in a Salesforce Page](#)

Add a Wave Analytics dashboard to a detail page layout. On an account detail page, for example, show your users 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.

[Embed Wave Dashboards in Lightning Pages \(Pilot\)](#)

Add a Wave Analytics dashboard component to a Lightning Home or Record Page to provide an interactive visualization of your data. Users can drill in and explore the dashboard within the frame on the Lightning Page or in a Wave Analytics window.

[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.

[Enable Actions and Links With Custom Actions Menus](#)

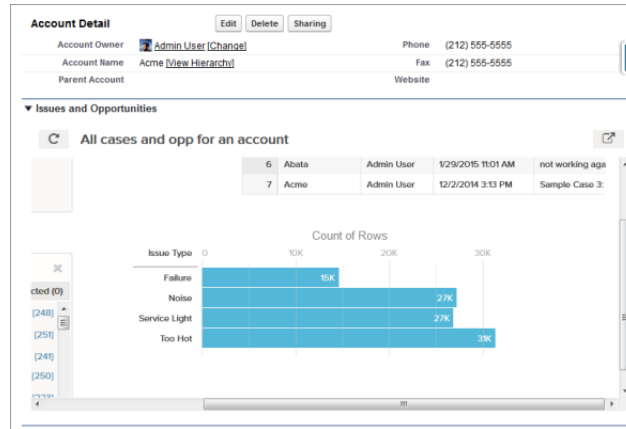
Custom menus in Wave Analytics lenses and dashboards let users take advantage of Salesforce actions and open records from Salesforce or other websites.

[Incorporate Visualizations through Chatter Posts and Sharing Links](#)

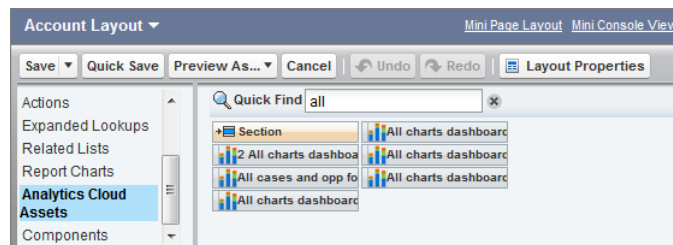
Share a visualization with your colleagues by posting to Chatter, or by getting its unique URL. A Chatter post provides an image and a link to the asset—lens, dashboard, or app—in Wave Analytics. Colleagues with the link and access to the asset can drill in and explore the information that's presented.

Embed Your Dashboard in a Salesforce Page

Add a Wave Analytics dashboard to a detail page layout. On an account detail page, for example, show your users 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.



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 and width by double-clicking the element or clicking the wrench icon next to it (🔧).

Use field mapping to map data fields in the dashboard to the object's fields, so that the dashboard shows only the data that's relevant for the record being viewed. You can specify fields from multiple datasets. Identify each dataset by its system name, which is listed on the dataset's edit page. Identify the data fields by row, and for the values, specify either the field names from the Salesforce object or specific values. The JSON field mapping follows this format:

```
{
  "dataset1_system_name": {
    "row_name": ["$field_name"],
    "row_name": ["$field_name"]
  },
  "dataset2_system_name": {
    "row_name": ["$field_name"],
    "row_name": ["$field_name"]
  }
}
```

For example, if the dashboard shows data from a dataset named Service, with the dimensions Account and Industry, the field mapping would be defined as:

```
{
  "service": {
    "account": ["$Name"],
  }
}
```

```

    "industry": [{"$Industry"}]
  }
}

```

Additional properties of Wave Analytics Assets:

- The `show title` checkbox gives you control over the visibility of the dashboard title.
- The `hide on error` checkbox 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.

 **Note:** To filter by a dimension, be sure to create a list widget in the dashboard for the dimension. Only dimensions that have a list selector on the dashboard can be included in the `filter` attribute JSON.

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 use the original page layout editor to open a layout that was created in the enhanced page layout editor, the original page layout editor will delete any Wave Analytics dashboard as soon as you save the layout.

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.

You can add one dashboard per Visualforce page. However, the dashboard can include links to other dashboards or Wave Analytics assets. You can also set filters to be applied at run time, control the visibility of the dashboard's title, and specify whether the dashboard appears on the page, if an error occurs.

To add a dashboard to your Visualforce page, configure the `<wave:dashboard>` component. With the `filter` attribute, you can use JSON to filter dataset fields at run time. With the `openLinksInNewWindow` attribute, you can specify where links from the dashboard to other assets are opened.

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.

```

<apex:page sidebar="true" standardController="account">

  <wave:dashboard dashboardId="0FKB000000006Y7OAI"
    showTitle="true"
    height="475px"
    openLinksInNewWindow="true"
    filter="{ 'opportunities': { 'Id': [ '!Account.Id' ] } }" />

</apex:page>

```

```

<apex:page sidebar="true" standardController="account">

```

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

```

<wave:dashboard dashboardId="0FKB000000006Y70AI"
  showTitle="true"
  height="1500px"
  openLinksInNewWindow="false"
  filter="{ 'service': { 'account': [' {!Account.Name}'], 'industry': [
' {!Account.Name}'] }}" />
</apex:page>

```

Note: To filter by a dimension, be sure to create a list widget in the dashboard for the dimension. Only dimensions that have a list selector on the dashboard can be included in the `filter` attribute JSON.

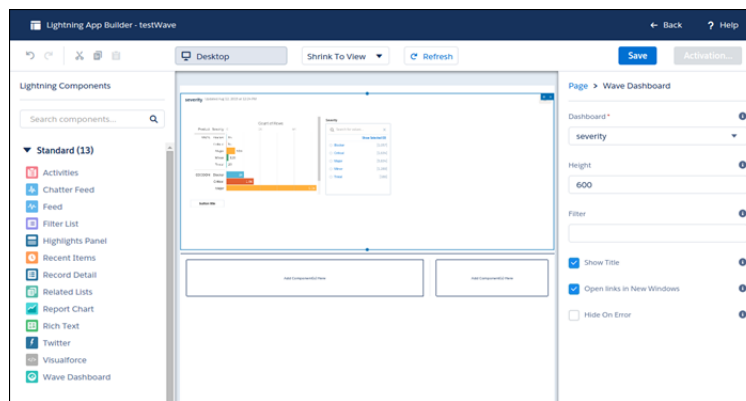
For more information about embedding Wave Analytics, see the `wave:dashboard` component in the [Visualforce Developer Guide Standard Component Reference](#).

Embed Wave Dashboards in Lightning Pages (Pilot)

Add a Wave Analytics dashboard component to a Lightning Home or Record Page to provide an interactive visualization of your data. Users can drill in and explore the dashboard within the frame on the Lightning Page or in a Wave Analytics window.

Note: Wave Analytics for the Lightning App Builder is in pilot. Contact your Salesforce account executive to enable this feature.

To add a dashboard to your Lightning Page, select the **Wave Dashboard** component and then select the dashboard to display. You can control the visibility of the dashboard's title and specify whether the dashboard appears on the page if an error occurs. With the **Open Links in New Windows** attribute, you can specify where links from the dashboard to other assets are opened.



With the **Filter** attribute, you can use JSON to filter dataset fields at run time. For example:

```


{datasetDeveloperName1:{field1:[value]},datasetDeveloperName1:{field1:[value1,value2],field2:["value3,with comma"]}}

```

Note: To filter by a dimension, be sure to create a list widget in the dashboard for the dimension. Only dimensions that have a list selector on the dashboard can be included in the `filter` attribute JSON.

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.

Your partner and customer users need the Analytics Cloud - Wave Community Users permission set license and a permission set that includes the “View Wave Analytics on Communities Pages” permission. For set up information, see [Enable Wave Analytics for Communities](#).

You give Community users access to view a dashboard by selecting **Share** from the dashboard or its app. In their community, they can drill in and explore the dashboard, while all other areas of Wave remain inaccessible. Security predicates can give you additional control over the visibility of your data.













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, share, or delete any Wave asset. They also can't upload data. This table summarizes differences between internal Wave users and external Community users.

EDITIONS

Available in: Salesforce Classic

Available in: **Developer Edition**

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

	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		

Enable Actions and Links With Custom Actions Menus

Custom menus in Wave Analytics lenses and dashboards let users take advantage of Salesforce actions and open records from Salesforce or other websites.

Custom actions menus in Wave lenses and dashboards let your team use quick actions on Salesforce objects as well as open records from Salesforce or elsewhere on the web. Developers or administrators edit the [Extended Metadata \(XMD\)](#) file for a dataset to create menus that are part of the following Wave chart and table interface elements:

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To view visualizations:

- "Use Wave Analytics"

To edit a dataset's extended metadata (XMD) file:

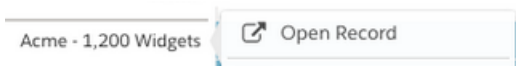
- "Edit Wave Analytics Dataflows"

Opportunity Name
 Equals Opportunity for Pearson1363 Filter

#	Opportunity Name	Amount	ValidFromDate
1	Opportunity for Pearson1363		02 20
2	Opportunity for Pearson1363	\$	02 20
3	Opportunity for Pearson1363	\$	02 30

- Open Record
- Log a Call
- New Task
- New Event
- Post
- File
- New Case
- New Note
- Link
- Poll
- Question

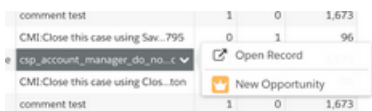
- Value labels on charts (such as "Opportunity for Sanders" in the image above).



- Legend value labels in table cells.



- Values in table cells.



Hovering over the labels shows if an actions menu is available. Clicking the value, cell, or legend displays the menu, and the user can then select from the available actions or open a record.

Actions in the Actions Menu

The 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, accounts, groups, and so on. See the section "Actions Menu Support for Objects and Actions" below for more details about the Salesforce objects and actions supported in the current release.

Open Record Link in a Quick Actions Menu

Links can pass specific bits of data, such as Salesforce account IDs, from the 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, as well as 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.

Developing Action Menus

Your development team creates actions menus by editing a dataset's extended metadata (XMD) file. For more information, see Dimensions Section in the [Extended Metadata \(XMD\) Reference](#).

To create an open record link in an action menu for a dataset with records from multiple Salesforce org instances, you have to 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](#).

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.

Work with Wave Analytics Apps

Wave Analytics apps are folders that contain datasets, lenses, and dashboards.

You can create your own apps to organize your data projects and control Wave Analytics asset sharing.

You can also use the ready-made [Sales Wave Analytics app](#) on page 202, which includes prebuilt datasets, lenses, and dashboards to provide a ready-made path through your Sales Cloud data and help you get started fast with Wave.

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.

[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.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To view visualizations:

- “Use 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, and this app can't be shared. When users are ready to share an item from their private app, they can save the item to any shared app.

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	X	X	X
See who has access to the app	X	X	X
Save contents of the app to another app that the user has Editor or Manager access to	X	X	X
Save changes to existing dashboards, lenses, and datasets in the app (saving dashboards requires the appropriate permission set license and permission)		X	X
Change the app's sharing settings			X
Rename the app			X
Delete the app			X

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 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.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: [Developer Edition](#)

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

USER PERMISSIONS

To create an app:

- "Create Wave Analytics Apps"

- b. Click **Change Icon**.
- c. Click an icon from the gallery.

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**.

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.

12 asset(s) are linked to the "sales" app. Deleting this app deletes 11 of the asset(s).
When the datasets in this app are deleted, these 1 asset(s) will be in error:

Cancel **Delete 11 Asset(s)**

Dashboards (1)		Lenses (0)	
Name	Dataset	Application	
embed dash	opportunity	basic	

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To share an app:

- "Use Wave Analytics" and Manager access to the app

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To delete an app:

- "Use Wave Analytics" and Manager access to the app

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.

- Click **Delete Asset(s)**.

The Admin Analytics App (Pilot)

The Admin Analytics app integrates with event monitoring and setup audit trail data to give you insights into your user and org behavior.

The Admin Analytics 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.

Note: The Admin Analytics app is currently available through a pilot program. Any unreleased services or features referenced in this document, press releases, or public statements are not currently available and may not be delivered on time or at all. Customers who purchase our services should make their purchase decisions based upon features that are currently available. Contact Salesforce for more information about this pilot.

IN THIS SECTION:

[Set Up Salesforce Permissions for the Admin Analytics App](#)

Select licenses and create and assign permission sets to enable admins and users to access the Admin Analytics app.

[Create and Share the Admin Analytics App](#)

Once you've enabled the correct permissions, the setup for the Admin Analytics app is simple.

[Learn About Admin Analytics App Prebuilt Dashboards](#)

The Admin Analytics 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 Admin Analytics app datasets when it's generated.

[Get to Know Admin Analytics App Terminology](#)

To make the best use of the Admin Analytics app, it's helpful to understand the metrics and terms used in the app.

Set Up Salesforce Permissions for the Admin Analytics App

Select licenses and create and assign permission sets to enable admins and users to access the Admin Analytics app.

Permission	Function
"Use Wave Analytics Templated Apps"	Enables access to all Wave templated apps.
"Access Admin Analytics Templates and Apps"	Provides specific access to Admin Analytics as well as the data and prebuilt dashboards included with it.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

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

USER PERMISSIONS

To use the Admin Analytics App:

- "Use Wave Analytics Templated Apps"
- "Access Admin Analytics Templates and Apps"

To create and manage Wave apps:

- "Manage Wave Analytics Templated Apps"
- "Edit Wave Analytics Dataflows"

USER PERMISSIONS

To create and edit permission sets:

- "Manage Profiles and Permission Sets"


Permission	Function
"Manage Wave Analytics Templated Apps"	Gives admins the ability to create Wave apps for users in their orgs, in this case the Admin Analytics app.
"Edit Wave Analytics Dataflows"	Enables admins to upload and download JSON for existing data flows.

Create permission sets using these permissions the same way that you do for any Salesforce permission set. Select a permission set license, create permission sets based on the permissions enabled by that license, and then assign these permission sets to users or profiles. Typically you'll create two types of permission sets, one for users and one for admins.

- 1. Select the Admin Analytics App permission set license.** All Admin Analytics users in your org need the Admin Analytics App permission set license. Select and assign this license to users just as you would other Salesforce permission set licenses.
- 2. Create and assign a permission set that enables users to access the Admin Analytics app.** Most users in your org only need to view the Admin Analytics app. For those users, create a permission set that includes "Use Wave Analytics Templated Apps" and "Access Admin Analytics Templates and Apps". These permissions let users view anything in Admin Analytics and create and edit dashboards. Assign this permission set to the appropriate users or profiles.
- 3. Create and assign a permission set that enables admins and some users to access and manage the Admin Analytics app.** Admins and some users need to create the Admin Analytics app and share it with others. Create a permission set with the Admin Analytics license and assign it to these admins and users. With these permissions, users can view everything in Admin Analytics and create and edit dashboards. But they can also create, share, and delete the app, schedule and work with data flows, upload external data, and more.

Create and Share the Admin Analytics App

Once you've enabled the correct permissions, the setup for the Admin Analytics app is simple.

 **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.

1. Log in to Salesforce.
2. Select Wave Analytics from the App Picker.
3. Click **App**, select **Admin Analytics**, 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.
6. Once 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 Admin Analytics Templates and Apps" permissions can see the app. To share your app, mouse over the app tile on the Wave Analytics home screen and click **Share**. You can also share from the app itself.

USER PERMISSIONS

To use the Admin Analytics App:

- "Use Wave Analytics Templated Apps"
- "Access Admin Analytics Templates and Apps"

To create and manage Wave apps:

- "Manage Wave Analytics Templated Apps"
- "Edit Wave Analytics Dataflows"

Learn About Admin Analytics App Prebuilt Dashboards

The Admin Analytics 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 Admin Analytics app datasets when it's generated.

Most of the dashboards represent a single event type on the EventLogFile object. The [EventLogFile object reference](#) contains detailed information about each event type. 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 Salesforce1.

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.

Reports

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 Admin Analytics App Terminology

To make the best use of the Admin Analytics app, it's helpful to understand the metrics and terms used in the app.

Term	Description
APT	Average page time. An aggregate metric across multiple transactions.
EPT	Experienced page time. The timing of a page transaction as seen by the user.
Event	A user-initiated interaction in Salesforce. For example, a login, the firing of an API call, or the download of a report. Events are related to the EventLogFile sObject.

The Sales Wave Analytics App

The Sales Wave app—the first ready-made Wave app from Salesforce—gets you started fast with Wave Analytics and provides a clear path through your Sales Cloud data on any device. You get everything you need in one place to uncover key insights to help you grow your business.

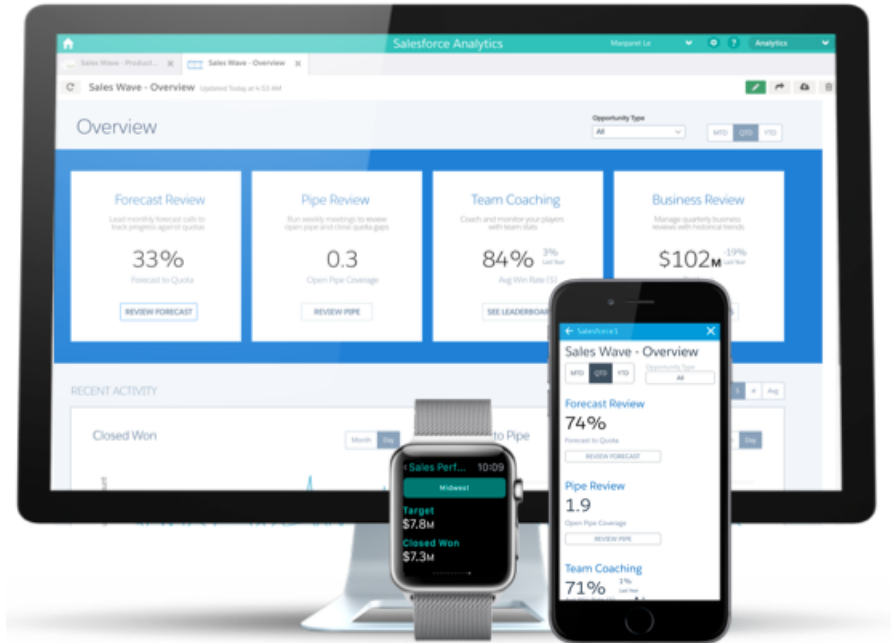
 **Tip:** Follow these steps in order to get started with Sales Wave. If you haven't used Wave before, learn more about it from the [Wave Analytics Library](#).

IN THIS SECTION:

1. [About the Sales Wave App](#)
Learn the benefits of Sales Wave before you create and use the app to explore your Sales Cloud data.
2. [Set Up Salesforce Permissions for the Sales Wave App](#)
Select the Sales Wave permission set license and create and assign permission sets to enable administrators and users to access the Sales Wave app.
3. [Create and Share the Sales Wave App](#)
Use the Sales Wave app's simple configurator to create the app and start uncovering the value of your Sales Wave data—fast.
4. [Update the Sales Wave App 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 Quota dataset so the app includes quotas data for your team.
5. [Schedule Sales Wave App Daily Dataflows](#)
Before you start exploring with Sales Wave, reschedule your daily dataflow to a time outside standard business hours.
6. [Integrate Sales Wave With Salesforce](#)
Make Sales Wave more usable by performing a variety of optional integrations and customizations.
7. [Understand Sales Wave App Limitations](#)
Sales Wave provides the following support for Salesforce objects and data.
8. [Learn About Sales Wave App Prebuilt Dashboards and Datasets](#)
The Sales Wave app includes prebuilt dashboards and datasets to accelerate your Sales Wave data exploration.
9. [Get to Know Sales Wave App Data Terminology](#)
To make the best use of the Sales Wave app, it's helpful to understand the metrics and terms used in the app.

About the Sales Wave App


Learn the benefits of Sales Wave before you create and use the app to explore your Sales Cloud data.



Salesforce created the Sales Wave app to make it easy for you to analyze your Sales Cloud data.

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, dashboards, and lenses 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 around your business needs.

 **Note:** Your organization can use the Sales Wave app with or without Salesforce Wave Analytics platform by purchasing a Sales Wave license.

Set Up Salesforce Permissions for the Sales Wave App

Select the Sales Wave permission set license and create and assign permission sets to enable administrators and users to access the Sales Wave app.

Your organization can use the Sales Wave app with or without the Salesforce Wave Analytics platform by purchasing a Sales Wave license. The Sales Wave license enables the following permissions:

- "Use Wave Analytics Templated Apps" enables access to all Wave templated apps. Sales Wave is the first such app.
- "Access Sales Cloud Analytics Templates and Apps" provides specific access to Sales Wave as well as the data and prebuilt dashboards included with it.
- "Manage Wave Analytics Templated Apps" gives administrators the ability to create Wave apps for users in their organizations, in this case the Sales Wave app.

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Available in: **Developer Edition**

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

USER PERMISSIONS

To use Wave templated apps:

- "Use Wave Analytics Templated Apps"

To use Sales Wave:

- "Access Sales Cloud Analytics Templates and Apps"

To create and manage Wave apps:

- "Manage Wave Analytics Templated Apps"
- "Edit Wave Analytics Dataflows"

USER PERMISSIONS

To create and manage Wave apps:

- "Manage Wave Analytics Templated Apps"
- "Edit Wave Analytics Dataflows"

- “Edit Wave Analytics Dataflows” enables administrators to upload and download JSON for existing dataflows.

You create permission sets using these permissions the same way that you work with any Salesforce permission set. Select a permission set license, create permission sets based on the permissions enabled by that license, and then assign permission sets to users. You can follow the same process you’d use to set up Wave Analytics platform permissions, except you use the permissions enabled by the Sales Wave Analytics App permission set license. See [Wave Platform Setup](#) on page 4. Typically, create two types of Sales Wave permission sets, one for users and another for administrators.

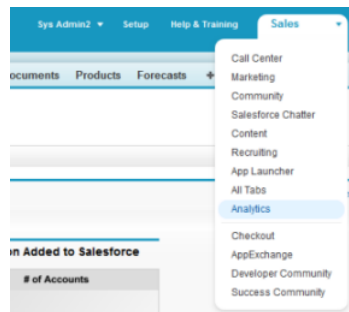
- 1. Select the Sales Wave Analytics App permission set license.** All Sales Wave users in your organization need the Sales Analytics Apps permission set license, which you select and assign to users just as you would other Salesforce permission set licenses.
- 2. Create and assign a permission set that gives access to the Sales Wave app.** Most users in your organization require only the ability to view the Sales Wave app. For those users, create a permission set that includes the first two permissions listed above, that is “Use Wave Analytics Templated Apps” and “Access Sales Cloud Analytics Templates and Apps”. With these permissions, users can view everything in Sales Wave, and create and edit dashboards. After you create the permission set, you can assign it to users.
- 3. Create and assign a permission set that enables administrators and some users to access and manage the Sales Wave app.** Administrators and some users in your organization require the ability to create the Sales Wave app and share it with others. Create a permission set that includes all the permissions included with the Sales Wave license and assign it to them. With these permissions, users can not only view everything in Sales Wave and create and edit dashboards, but they can also create, share, and delete the app, schedule and [work with dataflows](#), and upload external data (including the CSV file required to [update the Quotas dataset](#)). Users with these permissions can edit and create dashboards and lenses, edit and create datasets with any Salesforce data (including custom objects), and upload a CSV file for the Quota dataset.

Create and Share the Sales Wave App

Use the Sales Wave app’s simple configurator to create the app and start uncovering the value of your Sales Wave data—fast.

Follow these steps to create the Sales Wave app. 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](#), which covers the steps described here.

1. Log in to Salesforce.
2. Select **Wave Analytics** from the Force.com menu (top right of the Salesforce window).



3. Click **Create**, select **App**, select **Sales Wave**, and then click **Continue**
4. Salesforce opens the Sales Wave configuration wizard, which helps you personalize your app experience. The wizard presents a series of screens, each with a set of questions and their possible answers in pull-down menus. Answer the questions according to how your company prefers to view data. For more information about how to answer these questions, see [Personalize Sales Wave with the Configuration Wizard](#). Sales Wave saves the answers you select in the Personalize Sales Wave Analytics dashboard, which is created along with your app.

USER PERMISSIONS

To create and manage Wave apps:

- “Manage Wave Analytics Templated Apps”
- “Edit Wave Analytics Dataflows”

5. Give your app a name that's easily recognizable to others in your company.
6. Click **Create**. That starts a dataflow that creates the app and its assets (which include a [dataflow](#) definition file, datasets, and dashboards).
7. 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.
8. If you'd like to change your answers after looking at the dashboards, delete the app and repeat these steps. To delete, position your cursor over the icon for the app you just created and click Delete. Then click Yes to confirm deletion of all the assets associated with the app. Before you delete the app, take a screen shot of the Personalize Sales Wave Analytics dashboard so you can review your previous selections while you create the app again.
9. 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 App Quota \(Target\) Dataset](#).

When you first create the app, Sales Wave 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 With Salesforce](#) for details.

SEE ALSO:



[Personalize Sales Wave with the Configuration Wizard](#)

[Personalize Sales Wave with the Configuration Wizard](#)

Update the Sales Wave App 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 Quota dataset so the app includes quotas data for your team.

When you create the Sales Wave app, 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, 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](#), which covers the steps described here.

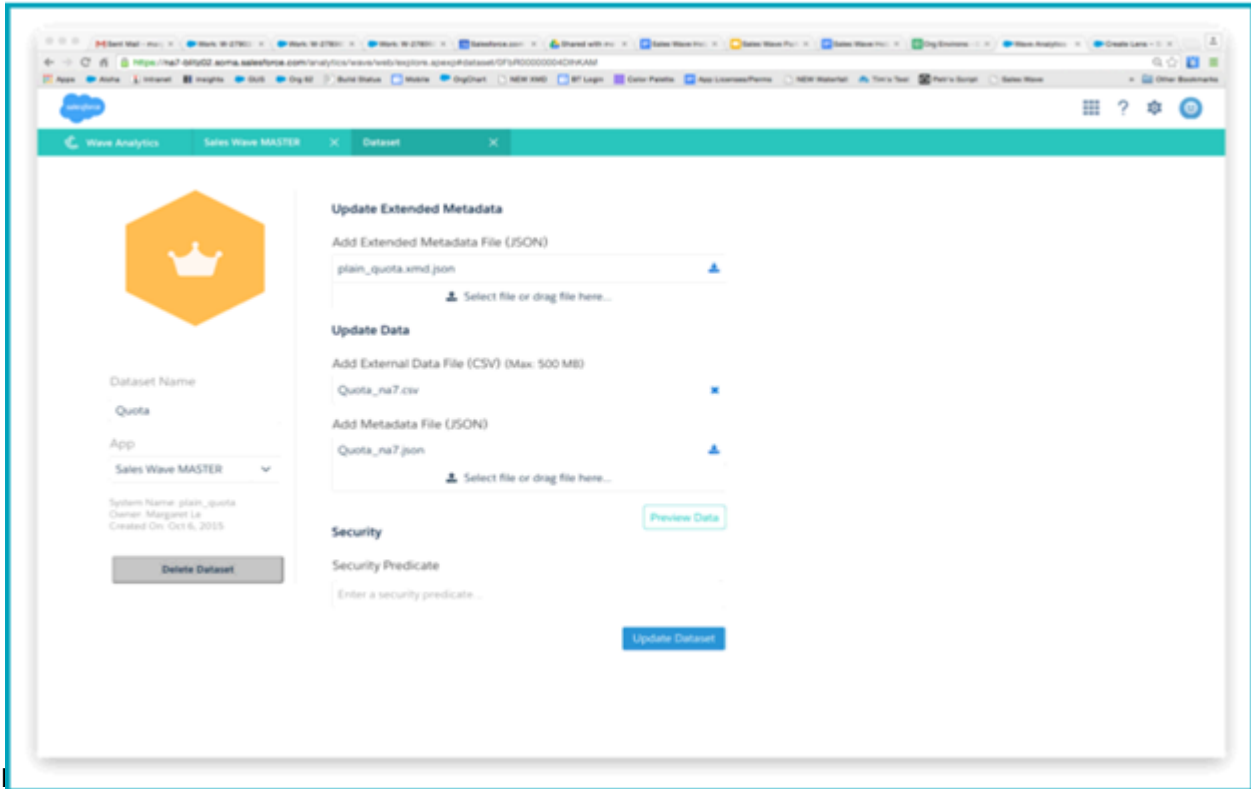
-  **Note:** When you work with .CSV files you want to import to Sales Wave, create and open them using only a UTF-8-compliant text editor. Opening them in Microsoft Excel or other spreadsheet software reformats .CSV files and makes them unusable in Sales Wave
 -  **Note:** If your business doesn't track quotas at the user level, you can skip this step. You only need to complete this if you answer "yes" to the [configuration wizard](#) question about whether your business tracks quotas at the user level.
1. Create a .CSV file to include the following fields, in this order: QuotaAmount, StartDate (in yyyy-mm-dd format), OwnerName, and Username. For an example, see [Sales Wave App 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.

USER PERMISSIONS

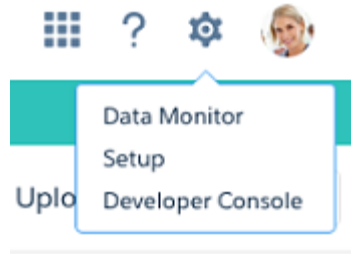
To create and manage Wave apps:

- "Manage Wave Analytics Templated Apps"
- "Edit Wave Analytics Dataflows"

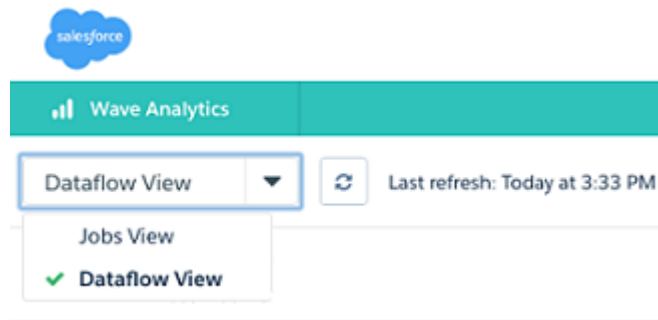
4. Hover over the dataset, then click **Edit**.
5. Salesforce displays the dataset editing screen for the Quota dataset. From the Update Data section, select **Select file or drag file**



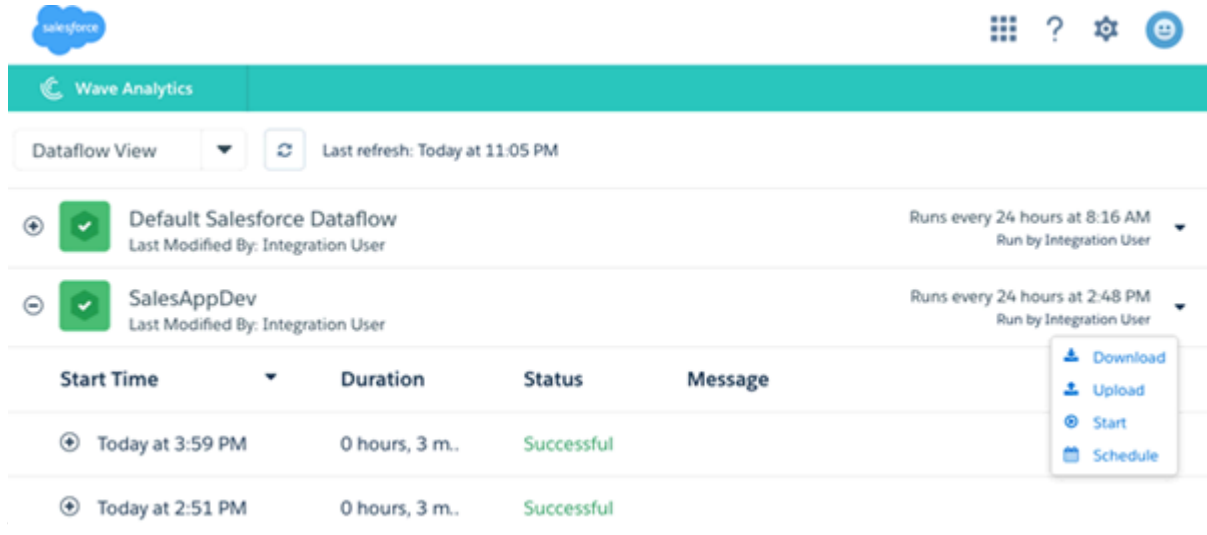
6. In the dialog box that opens, navigate to the .CSV file you created in Step 1, and double-click it.
7. Click **Update Dataset**.
8. If your fiscal period is different than calendar period, that is if it starts another date besides January 1, you also need to update the Quota Metadata file.
 - a. Copy the JSON from [Sales Wave 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 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 [Keep Sales Wave App Data Up-to-Date by Scheduling a Dataflow.](#)

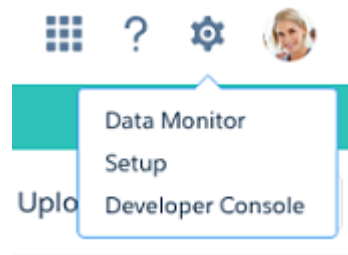
Schedule Sales Wave App Daily Dataflows

Before you start exploring with Sales Wave, reschedule your daily dataflow to a time outside standard business hours.

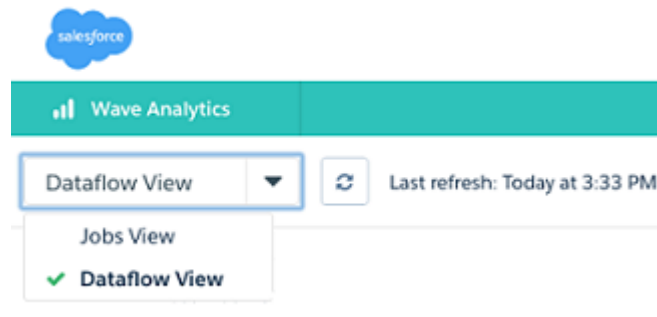
When you create Sales Wave, the creation process includes a dataflow, which is repeated every day at the time the app was created. This assures you that you're looking at the latest Salesforce data. Reschedule the dataflow to take place sometime outside normal business hours so the dataflow doesn't interrupt your use of the app.

You can also watch the video [Create the Sales Wave App, Part 2: Upload the Quota CSV File and Schedule a Dataflow](#), 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. Click the gear icon at the upper right of the screen to open the Data Monitor.



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 located 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

USER PERMISSIONS

To create and manage Wave apps:

- "Manage Wave Analytics Templated Apps"
- "Edit Wave Analytics Dataflows"

doesn't interrupt business activities. Click

The screenshot shows the Salesforce Wave Analytics interface. At the top, there's a 'Wave Analytics' header with a refresh button and 'Last refresh: Today at 11:05 PM'. Below this, two dataflow views are listed: 'Default Salesforce Dataflow' and 'SalesAppDev', both with a status of 'Success' and scheduled to run every 24 hours. A table below shows the last two successful runs for 'SalesAppDev' at 3:59 PM and 2:51 PM, both with a duration of 0 hours, 3 minutes. A 'Save.' button is visible at the bottom left of the screenshot.

Integrate Sales Wave With Salesforce

USER PERMISSIONS

To use Wave templated apps:	"Use Wave Analytics Templated Apps"
To use Sales Wave:	"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 in: **Developer Edition**

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

Make Sales Wave more usable by performing a variety of optional integrations and customizations.

Overriding role names in datasets, embedding the "Sales Wave - Account Overview" dashboard in your Salesforce Accounts page, and expose Sales Wave as a tab in Salesforce Classic can make it easier for your team to use the app.

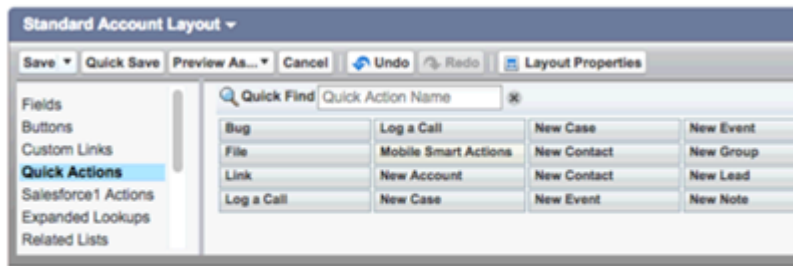
Add Actions to Sales Wave Dashboards

Sales Wave 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 will now be available from charts and tables in Sales Wave dashboards.

Override Role Name in Datasets

Sales Wave uses Salesforce Role Hierarchy to help you to understand your company's sales performance. When you first create the app, Sales Wave dashboards show the number associated with role name in the hierarchy—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 dashboards, update the Opportunity dataset XMD file with the following:

```
{"labels": {
  "keys": {
    "RoleName": {
      "CEO": "Paul",
      "Sales_WW": "Yves"
    },
  },
}
```

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 – Account Overview" dashboard 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 Available in a Salesforce Classic Tab

You can access Sales Wave 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 App Limitations

Sales Wave provides the following support for Salesforce objects and data.

Sales Wave App Support for Salesforce Objects and Fields

Sales 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.

Sales Wave creates a dataflow that exposes selected fields from the following standard Salesforce objects:

- Accounts
- Users
- Roles
- Opportunities
- Products (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](#).

 **Note:** Sales Wave limits custom objects support to no more than ten custom objects for each org that implements Sales Wave. This is a contractual, not a technical, limitation.

To add custom objects or additional fields not included when you first create the app, you need to update the Sales Wave dataflow. For details see [Create Datasets with a Dataflow](#).

Sales Wave App Support for Non-Salesforce Data

Sales Wave does not support external data sources [except a CSV file](#) that contains quota data at the user level. Importing other external data requires an additional license. See your Salesforce representative for details.

 **Note:** To include data about quotas in Sales Wave, you need to upload a CSV file with the details. For more information, see [Update the Sales Wave App Quota \(Target\) Dataset](#).

Learn About Sales Wave App Prebuilt Dashboards and Datasets

The Sales Wave app includes prebuilt dashboards and datasets to accelerate your Sales Wave data exploration.

Note: The dashboards and datasets included in your instance of Sales Wave may differ, depending on how you answer configuration wizard questions when you create the app.

Sales Wave Dashboards

The prebuilt dashboards in the Sales 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:

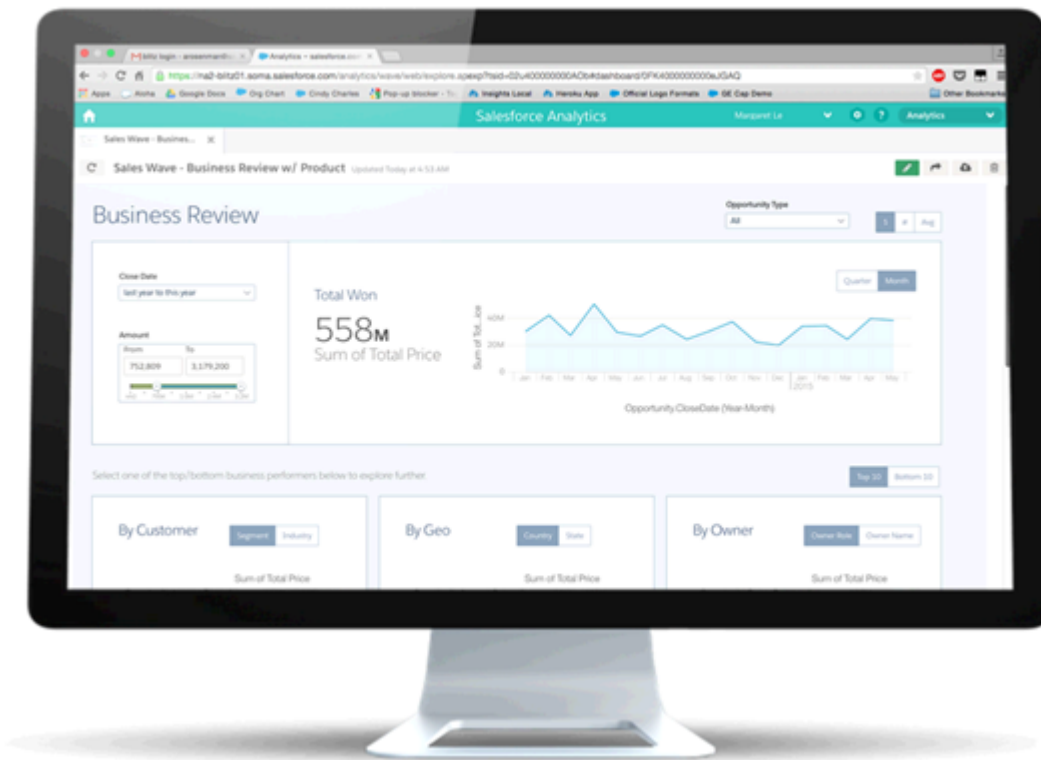
- “Use Wave Analytics Templated Apps”

To use Sales Wave:

- “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 dashboards, which can give you get 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](#) on page 147.

Note: Sales Wave uses the Salesforce role hierarchy to determine the security level of the data users can access. Users can see only data at their role level and all the roles below that level. The role hierarchy is also used to determine the manager roll-up structure on some Sales Wave dashboards. For example, if you want to drill down on your hierarchy to see how other teams and individuals are performing, use the “Owner Role View as” picklist.

Table 2: Sales Wave Dashboards

Dashboard Name	Contents	Effort Needed to Customize*	Has Manager Rollup?***
Sales Wave–Overview	Start here. Gives you a quick pulse of your business across forecast and pipe management, team performance, historical closed won business, and activity management.	High	No
Sales Wave–Pipeline Trending	Tracks opportunity changes in your pipeline such as Closed Won/Lost, Moved In/Out, Amount Reduced/Expanded.	High	No
Sales Wave–Forecast Review	Displays Closed Won Business and Expected to Close to let you know if you're going to hit your quota.	High	Yes
Sales Wave–Pipe Review	Shows all open deals in the pipeline and lets you organize by multiple key dimensions so you can quickly identify the top open deals. You can also perform actions such as Create Task or Event in Salesforce.	Low	No
Sales Wave–Productivity	Monitors tasks and events completed by each person on the sales team.	High	Yes
Sales Wave–Business Review	Shows historical Closed Won business across key dimensions, such as by owner, customer, geography, product, and source. Shown in above image.	Low	No
Sales Wave–By Customer	Shows visualizations of trends within your customer base.	Low	No
Sales Wave–By Geo	Shows visualizations of trends by geographical area.	Low	No
Sales Wave–By Source	Shows visualizations of trends by source.	Low	No
Sales Wave–By Product	Shows visualizations of trends by product.	Low	No
Sales Wave–Leaderboard	Shows top- and bottom-ranked players by time periods and	High	Yes

Dashboard Name	Contents	Effort Needed to Customize*	Has Manager Rollup?***
	across key performance categories.		
Sales Wave–Player Stats	Provides details for every person on your team.	High	Yes
Sales Wave - Pipe Activity	Benchmarks open vs. won deals to help you determine best practices to accelerate deals. Also helps you stay on top of top open deals by monitoring activity.	High	No
Sales Wave - Account Summary	Summarizes a selected account with average win rate, average sale cycle, open and won totals by geography, source, and/or product. Can be embedded in the Salesforce Account page layout.	Medium	No
Sales Wave - Personalized App Setup	Shows the answers you selected to the configuration wizard questions when you created the app.	Customization not recommended.	Not applicable

* Indicates the level of difficulty to edit the dashboard. If it's marked High, we recommend you work with a partner or Salesforce service representative. The complexity of the queries used to create the dashboard requires expertise in the Wave platform..

** Indicates if the dashboard lets you drill down by team structure. Currently, Sales Wave supports only role hierarchy.

Sales Wave Datasets

The next table lists all the possible datasets that can become part of Sales Wave when you create the app based on your responses to configuration wizard questions.

Table 3: Sales Wave Datasets

Dataset Name	Contents
Opportunities	Information about Opportunity, Accounts, and Users.
Opportunity Products	Product info for Opportunity, Accounts, and Users.
Tasks	Task information such as calls, emails, or tasks.
Events	Event information such as meetings.
Quota	Quota information; you need to upload CSV file on page 205 with Quota data to update this dataset before using Sales Wave.

Dataset Name	Contents
Opportunity History	Tracks changes on the Close Date, Stage, and Amount for an Opportunity.
Pipeline Trending	Creates the change type (Closed Won / Lost, Moved In / Out, Reduced / Expand) for the pipeline waterfall chart.
User Role	Basic information about Users.
User Manager	Role hierarchy information for Users used during manager roll-up.
Quota User Manager	Applies role hierarchy information to the Quota data used during manager roll-up.

Get to Know Sales Wave App Data Terminology

To make the best use of the Sales Wave app, it's helpful to understand the metrics and terms used in the app.

Table 4: 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.

Metric/Term	Formula	Description/Notes
Open Pipe Coverage	Open Pipe / Need to Close	How much in pipe to cover what is needed to close.
Bookings		Amount of Closed Won opportunities for new business.
Diff	This Year - Last Year	Difference between two items.
Y / Y %	(This Year / Last Year) - 1	Year-over-year growth rate.
Avg Selling Price	Total Amount (\$) / Total Count (#)	Average selling price for Opportunities, Products, and so on.
Avg Win Rate by \$	Closed Won / Closed Total	Success rate on closed opportunities, based on amount (\$).
Avg Win Rate by #	Closed Won / Closed Total	Success rate on closed opportunities, based on count (#).
Avg Sale Cycle	Closed Date - Created Date	How long it took to close the deal.
Avg Discount %	abs ((Selling Price / List Price)-1)	How much was reduced from what was listed.

Personalize Sales Wave with the Configuration Wizard

The Sales Wave configuration wizard lets you create the Sales Wave app so it reflects how your company views its data.

When you create the Sales Wave app, the built-in configuration wizard asks you a series of questions. Your answers determine how users view your sales data in Sales Wave'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.

For example, you're asked to specify both the primary and secondary fields you use to view customers according to their geographic locations. Perhaps you first want to see customers according to their country and then by their state. You'd select Country as the primary field to view by geography and State/Province for the secondary field. Or perhaps you want to sort by Region first, then by Country, so you'd select Region for the primary field and Country for secondary. Once you create Sales Wave, its dashboards provide a switch that toggles between the primary and secondary fields you select. You're asked to make similar choices about how to view data by customer, source, product, and other ways you track business with Salesforce.

USER PERMISSIONS

To create and manage Wave apps:

- "Manage Wave Analytics Templated Apps"
- "Edit Wave Analytics Dataflows"

Personalize Sales Wave - Accounts (1 of 3)

What's the primary field to view data by customer? *

Industry x ▾ High-level grouping to identify your customers

What's the secondary field to view data by customer?

Account Type x ▾ Detail-level grouping to identify your customers

What's the primary field to view data by geography? *

Billing Country x ▾ High-level grouping to identify your geography

What's the secondary field to view data by geography?

Billing State/Province x ▾ Detail-level grouping to identify your geography

What's the primary field on Accounts object to view data by source? *

Account Source x ▾ High-level grouping to identify your source for getting deals

What's the secondary field on Opportunities object to

[Back](#) [Continue](#)

Sales Wave saves your answers in the Personalize Sales Wave Analytics dashboard, which is created along with the other dashboards in the app. It's recommended that you save a screen shot of this dashboard for future reference if you create the app again.

Think through your answers to these questions. Your answers affect both the number and contents of the Sales Wave dashboards and how users view sales data in the app. You can further customize dashboards after you create the app. If you decide you didn't like your first answers, you can delete Sales Wave and quickly create it again.

SEE ALSO:

[Create and Share the Sales Wave App](#)

[Create and Share the Sales Wave App](#)

Sales Wave App Example .CSV File

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

See [Update the Sales Wave App Quota \(Target\) Dataset](#) on page 205.

 Example:

```

QuotaAmount,StartDate,OwnerName,Username
5000000,2014-01-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-02-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-03-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-04-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-05-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-06-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-07-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-08-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-09-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-10-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-11-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-12-01,Priya Anand,user9.kthjbnrxfifo@na2.com
5000000,2014-01-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-02-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-03-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-04-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-05-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-06-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-07-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-08-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-09-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-10-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-11-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com
5000000,2014-12-01,Nathan Sinha,user18.esm4nmdicvgc@na2.com

```

Sales Wave Quota Dataset JSON File

Use this file to update the Sales Wave 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 Sales Wave 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 following the instructions in [Update the Sales Wave App 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"
      }
    ]
  }
}

```

Wave Analytics Limits


This section describes Wave Analytics limits.

The following limits apply to all supported editions. Contact salesforce.com to extend the limits.

API Call Limits

Limit	Value
Maximum concurrent Wave Analytics API calls per organization	100
Maximum Wave Analytics API calls per user per hour	5,000

Wave Analytics License Data Storage Limits

 **Note:** The data limits shown in the following table apply *only* to Analytics Cloud - Wave Analytics Platform and Analytics Cloud - Sales Wave Analytics App licenses purchase on or after October 20, 2015.

License	Limit. Wave Analytics license data storage limits are contractual, not technical. Licensee agrees to strictly monitor its total number of data rows.
Analytics Cloud - Wave Analytics Platform	100 million rows


License	Limit. Wave Analytics license data storage limits are contractual, not technical. Licensee agrees to strictly monitor its total number of data rows.
Sales Wave Analytics App	25 million rows when used without Analytics Cloud - Wave Analytics Platform license. Use of Sales Wave app license does not increase data limit for platform license
Analytics Cloud - Additional Data Rows	100 million rows

Dataset Storage Limits

Limit	Value
Maximum number of rows for all registered datasets combined.	250 million per platform license <i>purchased before</i> October 20, 2015.
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

Limit	Value
Maximum number of dataflow runs in a rolling 24-hour period	24

 **Note:** Wave uses the bulk API, which counts towards your 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.

Sales Wave App Limits

Sales Wave limits custom object support to no more than ten 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
Maximum number of external data files that can be uploaded in a rolling 24-hour period	50

Limit	Value
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

Limit	Value
Maximum concurrent queries per organization	50 per platform license
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 <code>limit</code> statement.	
Query timeout	2 minutes

The following limits apply to the Wave-enabled Developer Edition.

Developer Edition Limits

Limit	Value
Maximum number of data rows	250,000
Maximum concurrent queries	5



Note: Wave is built on 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 sfcdigest transformation can extract data. If you exceed a force.com limit, an error occurs. See [force.com platform limits](#).

Wave Analytics Limitations

Wave Analytics differs from other Salesforce features in some ways.

Releases Aren't Staggered

All customers receive Wave Analytics updates on the same schedule, regardless of instance. Related core updates happen by instance on a staggered schedule.

Localization

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 may only be available in English.
- The mobile iOS app includes only English versions of sample datasets and dashboards.

Limited Internationalization Support

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 localized.
- Sorting data within datasets may not work as expected.
- Each dataset may have a single locale specified in the metadata. Everyone viewing that dataset will see the same date, time, and number formats, as well as dimension names, regardless of their own locale and language settings.
- Search in filters is both case-sensitive and accent-sensitive. For example, using the filter “Andre” does not return “André” 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.

Accessibility Features Aren’t Supported

Accessibility features haven’t been incorporated into Wave Analytics.

Packaging and Metadata API Are Not Supported

The use of the Metadata API isn’t supported for Wave Analytics. You can’t package or version Wave Analytics applications. Migrating Wave Analytics data, objects, and applications from sandbox to production orgs requires manual effort. The Default Org Template (DOT) isn’t supported for org duplication.

Limited Field-Level Security

Field-Level security isn’t available for external data that’s 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](#).


Sales Wave App Support for Salesforce Objects and Fields

Sales 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.

Sales Wave creates a dataflow that exposes selected fields from the following standard Salesforce objects:

- Accounts
- Users
- Roles
- Opportunities
- Products (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](#).

 **Note:** Sales Wave limits custom objects support to no more than ten custom objects for each org that implements Sales Wave. This is a contractual, not a technical, limitation.

To add custom objects or additional fields not included when you first create the app, you need to update the Sales Wave dataflow. For details see [Create Datasets with a Dataflow](#).

Sales Wave App Support for Non-Salesforce Data

Sales Wave does not support external data sources [except a CSV file](#) that contains quota data at the user level. Importing other external data requires an additional license. See your Salesforce representative for details.

 **Note:** To include data about quotas in Sales Wave, you need to upload a CSV file with the details. For more information, see [Update the Sales Wave App Quota \(Target\) Dataset](#).

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, as well as 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.

Reports and Dashboards

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

Gather Data with Reports

Reports give you access to the data your organization has accumulated over time, enabling you to make informed decisions. 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.

Share Insights with 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.

Reports and Dashboards

Reports

Learn about Reports

Reports give you access to your Salesforce data. Before building, reading, and sharing reports, review these reporting basics.

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

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

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

Report Builder

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

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

Fields

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

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

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

Filters

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

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

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

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

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](#)

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](#)
A report type is a set of rules that determine which records and fields appear in a report. You can start with one of the available standard report types or use a custom report type provided by your administrator.
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: **All** editions except **Database.com**

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

A report type is a set of rules that determine which records and fields appear in a report. You can start with one of the available standard report types or use a custom report type provided by your administrator.

Choosing the right report type is one of the most important steps in creating a report.

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.

For more information, review information about standard and custom report types.

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.

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”

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.

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.

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"

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`.

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 in activities related lists and reports represents the last time someone in your company acted on an account or a contact. This date is either the date and time of the most recent event or the due date of the most recently closed related task.

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All Editions** except **Database.com** (The edition determines which reports you see.)

- 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 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 organization'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.
 - 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All Editions** except **Database.com** (The edition determines which reports you see.)

- 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 create, edit, save, and delete document reports:	“Create and Customize Reports”
To run pending approval request reports:	“Run Reports” AND “View All Data”

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

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

- 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.

EDITIONS

Available in: Salesforce Classic

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

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.

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 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 <code>Title</code> 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.

EDITIONS


Available in: Salesforce Classic


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

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.

 **Warning:** The My Team view shows forecasts across multiple hierarchy levels. Consequently, values may be included in a subordinate's forecast total and a forecast manager's forecast total. This may lead to larger than expected values in the column total because subordinate amounts are included multiple times in the total.

 **Note:** Click **Show Hierarchy** to see your organization'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.

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

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

- 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

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

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.
- 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 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 territory management, the effect of this option depends on the value of the Hierarchy filter above. If you select **Role**, you see your opportunities and opportunities owned by all of your subordinates in the role hierarchy. If you select **Territory**, you see opportunities that you own and any opportunities owned by your territories' descendants.
 - **My team's team-selling and their opportunities**—For users who report to you in the role hierarchy, 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

- 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: <ul style="list-style-type: none"> • 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	<p>Opportunities by month, including split percentages for the opportunity team. To access this report, product scheduling must be enabled for your organization.</p> <p>You can also summarize information by these fields:</p> <ul style="list-style-type: none"> • <code>Split Total Price</code>—Split percentage multiplied by Total Price. • <code>Split Schedule Amount</code>—Split percentage multiplied by Schedule Amount. • <code>Split Expected Product Amount</code>—Split Total Price multiplied by Probability. • <code>Split Expected Schedule Amount</code>—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 team 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 organization'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	<ul style="list-style-type: none"> • Opportunity Name • Amount • Close Date • Stage • Age • Type • Probability (%) • Lead Source • Fiscal Period • Next Step • Created Date
Opportunity Owner Information	<ul style="list-style-type: none"> • Opportunity Owner • Owner Role

EDITIONS

Available in: Salesforce Classic

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

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

- Title
- First Name
- Last Name

Contact Role: Phone/Fax/Email

- Phone
- Email

Contact Role: Address

- Mailing Street
- Mailing City
- Mailing State/Province
- Mailing ZIP/Postal Code
- 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	<ul style="list-style-type: none"> Opportunity Name
Product Information	<ul style="list-style-type: none"> Product Name
Opportunity Owner Information	<ul style="list-style-type: none"> Opportunity Owner
Account: General Information	<ul style="list-style-type: none"> Account Name
Contact Role: General	<ul style="list-style-type: none"> Title First Name Last Name
Contact Role: Phone/Fax/Email	<ul style="list-style-type: none"> Phone Email
Contact Role: Address	<ul style="list-style-type: none"> 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 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	<ul style="list-style-type: none"> Opportunity Name Close Date Amount
Competitor Information	<ul style="list-style-type: none"> Competitor Name Strengths Weaknesses
Account: General Information	<ul style="list-style-type: none"> Account Name

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: Salesforce Classic

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

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

Object Information Type	Columns
History Data	<ul style="list-style-type: none"> Edit Date New Value Edited By Field/Event Old Value
Opportunity Fields	<ul style="list-style-type: none"> Opportunity Name
Opportunity Owner Information	<ul style="list-style-type: none"> Opportunity Owner

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](#)

Opportunity History Report

View information about the history of your opportunities, including stages and close date.

The default settings for this report are:

Format

Summary

Selected Columns

Object Information Type	Columns
Opportunity Information	<ul style="list-style-type: none"> Opportunity Name

EDITIONS

Available in: Salesforce Classic

Available in all editions

Object Information Type	Columns
Opportunity Owner Information	<ul style="list-style-type: none"> Owner
Opportunity History Information	<ul style="list-style-type: none"> 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 team and their roles, organized by opportunity.

 **Note:** You must enable and set up team selling to use the Opportunities with Opportunity Teams report.


The default settings for this report are:

Format

Tabular

Selected Columns

Object Information Type	Columns
Opportunity Information	<ul style="list-style-type: none"> Opportunity Name
Team Member Information	<ul style="list-style-type: none"> Team Member Name Team Role

 **Note:** If your administrator has created custom opportunity team fields, they can be included in the Opportunity with Opportunity Teams Report.

SEE ALSO:

[The Report Run Page](#)

[Limit Report Results](#)

[Opportunity Reports](#)

EDITIONS

Available in: Salesforce Classic

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

Opportunities with Opportunity Teams and Products Report

View information about your opportunity team members and their products, organized by opportunity. You must specify either a product or an opportunity team member to filter results by when you run the report.

The default settings for this report are:

Format

Tabular

Selected Columns

Object Information Type	Columns
Opportunity Information	<ul style="list-style-type: none"> Opportunity Name
Product Information	<ul style="list-style-type: none"> Product Name
Team Member Information	<ul style="list-style-type: none"> Team Member Name Team Role

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	<ul style="list-style-type: none"> Opportunity Name Close Date Amount
Opportunity Owner Information	<ul style="list-style-type: none"> Opportunity Owner
Partner Information	<ul style="list-style-type: none"> Partner Owner Partner Role Partner

EDITIONS

Available in: Salesforce Classic

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

EDITIONS

Available in: Salesforce Classic

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

Object Information Type

Columns

Account Information

- Account Owner
- 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

EDITIONS

Available in: Salesforce Classic

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

Object Information Type	Columns
Opportunity Owner Information	<ul style="list-style-type: none"> Opportunity Owner Owner Role
Account: General Information	<ul style="list-style-type: none"> 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 `Historical Stage` and columns grouped by `As of Date`.

Summary Fields

`Historical Amount (sum)`

Selected Columns

Object Information Type	Columns
Opportunity Information	<ul style="list-style-type: none"> Opportunity Name
Opportunity Trend Information	<ul style="list-style-type: none"> Historical Amount As of Date Historical Stage
Opportunity Owner Information	<ul style="list-style-type: none"> Opportunity Owner Alias

EDITIONS

Available in: both Salesforce Classic Lightning Experience

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

SEE ALSO:

[The Report Run Page](#)

[Limit Report Results](#)

[Opportunity Reports](#)

Opportunities with Quotes and Quote PDFs Report

View details about the quote PDFs created for each quote associated to an opportunity. The default settings show you the most commonly used information from each object, but you can customize the report to view other information, such as who created or last modified each listed quote PDF.

The default settings for this report are:

Format

Summary

Grouping

The default report shows quotes and quote PDFs grouped by `Opportunity Name`.

Selected Columns

Object Information Type	Columns
Opportunity Information	<ul style="list-style-type: none"> Opportunity Name
Quote Information	<ul style="list-style-type: none"> Quote Name Syncing
Quote PDF Information	<ul style="list-style-type: none"> Quote PDF: Created Date Quote PDF: Name Quote PDF: Discount Quote PDF: Grand Total

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 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

Available in: Salesforce Classic

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

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	<ul style="list-style-type: none"> • <code>Opportunity Name</code> • <code>Amount</code>
Quote Information	<ul style="list-style-type: none"> • <code>Quote Name</code> • <code>Discount</code> • <code>Syncing</code> • <code>Status</code>
Quote Line Item Information	<ul style="list-style-type: none"> • <code>Quote Line Item: Discount</code> • <code>Product: Product Name</code> • <code>Line Item Number</code> • <code>Sales Price</code> • <code>List Price</code> • <code>Quote Line Item: Subtotal</code> • <code>Quote Line Item: Total Price</code>

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.

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

- 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 organizations. Existing organizations 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	
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 Start Time	The time the case entered an entitlement process. This field displays if an entitlement process applies to the case.
Entitlement Process End Time	The time the case exited an entitlement process. This field displays if an entitlement process applies to the case.
Super User	A contact enabled to view case information, add comments, and upload attachments for all cases 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 <code>Account Name</code> field to see if the account on the case is different from the account on the contact.
Parent Case ID	The ID of a parent case, which can be used to access a parent case via the API.

Fields Available for Solution Reports

You can report on a number of key solution fields in addition to the standard and custom fields.

Field	Description
# Solution Searches	The number of solution searches performed by Self-Service users via the Self-Service portal. (Self-Service Usage Report)
Self-Service Access Count	The number of times a solution was viewed in the Self-Service portal.
Self-Service Answer Count	The number of times the survey question “Does this Solution help you answer your question?” is answered, either positively or negatively, on a solution in the Self-Service portal.
Self-Service Positive Count	The number of times the survey question “Does this Solution help you answer your question?” is answered positively on a solution in the Self-Service portal.
Author	The name of the user who originally created the solution.
Num Related Cases	The number of cases associated with the solution.
Reviewed	Checkbox that indicates whether the solution has a reviewed status.
Old Value	The value in a tracked case or solution field before it was changed. (Case History Report and Solution History Report)
New Value	The value in a tracked case or solution field after it was changed. (Case History Report and Solution History Report)
History ID	The unique identifier for each change tracked on a specified case or solution field. (Case History Report and Solution History Report)
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 Forecasting Custom Report Type](#)
8. [Forecasting Custom Report Types](#)
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.
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.

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"

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 what content is missing or needs to be re-named 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.

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](#)


EDITIONS

Available in: Salesforce Classic

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

Reports for Answers

Create custom report types so users can analyze questions, replies, and votes.

 **Note:** Starting with Summer '13, Answers isn't available for new organizations. Now you can use Chatter Answers, a self-service and support community where users can post questions and receive answers and comments from other users or your support agents. Chatter Answers brings together Case, Questions and Answers, and Salesforce Knowledge articles in a unified experience. Existing organizations will continue to have access to Answers if it was enabled prior to the Summer '13 release.

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.
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](#)

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"

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.

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: <i>Approval Process Instances</i>
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: Salesforce Classic

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

USER PERMISSIONS

To run reports:

- "Run Reports"
- AND
- "Read" on the records included in report

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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"

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.

- Click **Next**.
- Click the box under the primary object.
- 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.

- For the A to B relationship, select one of these options.

Option	Description
Each "A" record must have at least one related "B" record.	<p>The report includes only process instances that enter at least one approval step to create a process instance node.</p> <p>The report excludes process instances for records that were submitted for approval but that didn't meet any step criteria.</p>
"A" records may or may not have related "B" records.	The report includes all process instances.

- 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.

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 related to an activity	Activity reports. Add the <code>Relation Count</code> column to your reports. 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.

EDITIONS

Available in Lightning Experience in: **Contact Manager, Personal, Group, Professional, Enterprise, Performance, Unlimited,** and **Developer** Editions

Available in Salesforce Classic in: **All Editions Except Database.com**

To View This Information**Use This Report**

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.

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.

EDITIONS

Available in Lightning Experience in: **Contact Manager, Personal, Group, Professional, Enterprise, Performance, Unlimited,** and **Developer** Editions

Available in Salesforce Classic in: **All Editions Except Database.com**

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 Forecasting Custom Report Type

 **Note:** This information applies to Collaborative Forecasts and not Customizable Forecasts. You must enable Collaborative Forecasts to create a forecasting custom report type.


To make a forecasting report available to users, administrators must create a custom report type. A *report type* defines the set of records and fields available to a report based on the relationships between a primary object and its related objects. Reports display only records that meet the criteria defined in the report type.

1. Start creating a custom report type from Setup by entering *Report Types* in the *Quick Find* box, then selecting **Report Types** and **New Custom Report Type**.
2. For Primary Object, select a Forecasting object, such as Forecasting Items or Forecasting Quotas.
3. For Store in Category, select Forecasts.

After deploying the report types, let your users know their locations and names.

Forecasting Custom Report Types

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. You must enable Collaborative Forecasts to create a forecasting custom report type.

Use the table to understand the forecasting custom report types that you can create.

Primary Object	Use to create a report for...
Forecasting Items	<p>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:</p> <ul style="list-style-type: none"> • Owner Only Amount—The sum of all 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 all of a person’s owned revenue opportunities and also his or her subordinates’ opportunities, without adjustments. Subordinates include everyone reporting up to a person in the forecast hierarchy. 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

EDITIONS

Available in: Salesforce Classic

Available in: **Professional** (no Opportunity Splits or Custom Field forecasts), **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”

EDITIONS

Available in: Salesforce Classic

Available in: **Professional** (no Opportunity Splits or Custom Field forecasts), **Enterprise, Performance, Unlimited,** and **Developer** Editions

Primary Object**Use to create a report for...**

opportunities and his or her subordinates' opportunities, including adjustments made by the forecast owner on her or her own or subordinates' forecasts. *It doesn't include adjustments made by forecast managers above the owner in the forecast hierarchy.* For example, say 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 her subordinate Ben. Additionally, she adjusted Ben's amount to \$65,000 for a total of \$85,000. If you adjust Anne's number from \$85,000 to \$100,000, then you see \$85,000 in **Amount Without Manager Adjustments** because this is what Anne sees (and Anne can't see your adjustments as 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, say 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 instead:

- **Owner Only Quantity, Quantity Without Adjustments, Quantity Without Manager Adjustments, and Forecast Quantity**

No matter whether you forecast based on revenue or quantity, add these fields:


- **Has Adjustment**—A checkbox that indicates if a manager adjustment has been made on a forecast owner's amount.
- **Has Owner Adjustment**—A checkbox that indicates if a forecast user has made an adjustment to their own forecast amount.

If your organization uses 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 your organization has changed the forecast category names, those changes also appear in the **ForecastingItemCategory** values.

Forecasting Items with Opportunities as a related object

Viewing opportunity information for specific forecasting line items. For example, you might want to create a summary report for each of your subordinates that includes the opportunity names and last activity dates for their forecasting items, alongside adjustment information and final forecast amounts.

-  **Note:** For opportunities with no opportunity products specified, this report type includes 2 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 might want to create a summary report for each of your subordinates that includes the opportunity split amounts and percentages for their forecasting items, alongside adjustment information and final forecast amounts.

Primary Object**Use to create a report for...**

Forecasting Items with Opportunity Product as a related object

Viewing product family information for specific forecasting line items. For example, you might want to create a summary report for each of your subordinates that includes the product families and total price for their forecasting items, alongside adjustment information and final forecast amounts.



Note: This report type only shows forecasting items for the Product Family—Revenue and Product Family—Quantity forecast types. It includes opportunities both with and without opportunity products specified.

Forecasting Quotas

Viewing data about individual or team quotas. As a best practice, include all the default fields in the report type. For example, you can include lookup fields, such the full name of the owner. When running the report, you can filter by your own name to see quotas you created and their related accounts and owners.

Forecasting Quotas with Forecasting Items as a related object

Viewing quota attainment. For example, you can use both Forecasting Quotas and Forecasting Items to create the custom report type. Then, when you create the report, include a team's quotas and forecasted revenue for closed forecasts and create a formula field to display the quota percentage attained.

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

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.

EDITIONS

Available in: Salesforce Classic

Salesforce Knowledge is available in **Performance** and **Developer** editions.

Salesforce Knowledge is available for an additional cost in **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"

EDITIONS

Available in: Salesforce Classic

Salesforce Knowledge is available in **Performance** and **Developer** editions.

Salesforce Knowledge is available for an additional cost in **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"

Article 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 and Article Version History	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.
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	<ul style="list-style-type: none"> • Analyze the number of searches per day, month, or year for each channel and language. • For each search, see the: <ul style="list-style-type: none"> – Date – ID and title of the article that was clicked • See which keywords users are looking for in your knowledge base. • For each keyword, see the average number of:

EDITIONS

Available in: Salesforce Classic

Salesforce Knowledge is available in **Performance** and **Developer** editions.

Salesforce Knowledge is available for an additional cost in **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"

- 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 and Article Version History	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
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 273.

Create an Article Report

Run your custom report on your Salesforce Knowledge articles and save them to your article reports folder.

Using your [custom report types](#), create article reports and save them to your [article reports folder](#).

1. On the Reports tab, click **New Report**.
2. Choose the category where your custom report types are stored; for example, **Customer Support Reports** or **Other Reports**.
The Cases with Articles report is available by default in the Customer Support Reports folder.
3. Find your report type and click **Create**.
4. When viewing your report, click **Save As** and save it in the new Article Reports folder to make the report available to other users.

 **Note:**

- In reports using the Knowledge Article custom report type, there are at least five rows per article (one for each channel, including “All Channels”).
- Daily values are reported independently for the last 90 days and monthly values for the last 18 months. After those time periods, use the monthly aggregate and yearly aggregate values, respectively.
- In reports using the Knowledge Article Votes, Knowledge Article Views, or Knowledge Article Searches custom report types, each row represents a day, channel, and role combination. For example, if a user with the Kingmaker role views articles in the internal app and the next day the same reader views more articles in the internal app, the Article Views report has two rows: one for each unique date.
- In reports using Knowledge Article Version (KAV), except Knowledge Article Version History, you can filter by data category. You can add up to four filters and set their logic to AT, ABOVE, BELOW, or ABOVE OR BELOW. The logic between filters is OR. You can use the same category group multiple times, however, you must use the same operator each time.
- To report on Approval Processes for Knowledge Articles, use **Process Instance** and **Process Instance Node** when creating a custom report type. Then filter the report on object type, which is the article type.

Fields Available on Salesforce Knowledge Reports

EDITIONS

Available in: Salesforce Classic

Salesforce Knowledge is available in **Performance** and **Developer** editions.

Salesforce Knowledge is available for an additional cost in **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 and Article Version History Reports

The Article Type primary object allows you to add a secondary object of the article type version.

Table 5: Article Type Report Primary Object Fields

Field	Description
Archived By	User who archived the article.
Archived Date	Date the article was archived.
Article Number	Unique number automatically assigned to the article.
Article Type ID	The ID associated with the article type.
Case Association Count	Number of cases attached to the article.
Created By	User who created the article.
Created Date	Date the current article version was created. If the article has been published more than once, this is the latest draft date. To create reports that use the original article creation date, first create a Custom Report Type that joins the Knowledge Article (<code>__ka</code>) and Knowledge Article Version (<code>__kav</code>) 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

Salesforce Knowledge is available in **Performance** and **Developer** editions.

Salesforce Knowledge is available for an additional cost in **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 6: 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 Daily, Monthly, and Yearly. For example, a record where the Count is 70 and the Duration is Monthly indicates that 70 searches took place over the past month. Totals are aggregated daily for the current month, monthly from the past full month through the past full year, and yearly beyond that.
Related Role	Name of the role that applies to the record. Each row in the report represents searches per channel per role.

Fields Available on Knowledge Article 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 six of the eight fields 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 <code>Daily</code> , <code>Monthly</code> , and <code>Yearly</code> . For example, a record where the Count is 70 and the Duration is <code>Monthly</code> 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. Note that scores take 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 <code>All Channels</code> , <code>Internal App</code> , <code>Customer</code> , <code>Partner</code> , and <code>Public Knowledge Base</code> .
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 <code>Daily</code> , <code>Monthly</code> , and <code>Yearly</code> . For example, a record where the Count is 70 and the Duration is <code>Monthly</code> 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 <code>All Channels</code> , <code>Internal App</code> , <code>Customer</code> , <code>Partner</code> , and <code>Public Knowledge Base</code> .
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 <code>Daily</code> , <code>Monthly</code> , and <code>Yearly</code> . For example, a record where the Count is 70 and the Duration is <code>Monthly</code> 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 <code>All Channels</code> , <code>Internal App</code> , <code>Customer</code> , <code>Partner</code> , and <code>Public Knowledge Base</code> .
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 <code>Daily</code> , <code>Monthly</code> , and <code>Yearly</code> . For example, a record where the Count is 70 and the Duration is <code>Monthly</code> 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 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.

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 **Developer** Edition organizations that were created after June 14, 2012

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.


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: **Professional, Enterprise, Performance, Unlimited, and Developer** Editions

USER PERMISSIONS

To report on partners:

- "Read" on accounts or opportunities
- AND
- "Run Reports"

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:

1. From Setup, enter *Account Owner Report* in the **Quick Find** box, then select **Account Owner Report**. This option is available only in organizations that have a private account sharing model.
2. Select the checkbox to allow all users to run this report. If you leave the box unchecked, only administrators and users with the “View All Data” permission can run this report.
3. Click **Save**.

Report on File Search Activity

Use a report to determine the top 300 search term queries for file searches and to identify what content is missing or needs to be re-named based on users’ search terms and the click-rank.

User Permissions Needed	
To create or update custom report types:	“Manage Custom Report Types”
To create a public reports folder:	“Manage Public Reports”
To run the File Search Activity report:	“Run Reports” AND “View All Data” OR “Content Administrator”. Users with “Manage Library” for the library don’t need the “View All Data” or “Content Administrator” permissions to run the report.

Custom report types are the only way to make reports about file search activity available for your users—Salesforce does not provide sample file search activity reports or a standard report folder.



[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.

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”

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Salesforce Files available in: **All** editions, **except** Customer Portals

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 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.

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"

Format	Description
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 <code>Stage</code> and <code>Owner</code> . 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 <i>and</i> 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
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

Changing the Report Format

Changing the format affects filters and groupings, as follows:

When you change...	What Happens?
Tabular to Summary or Matrix	The <code>Rows to Display</code> filter is removed.
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

When you change...**What Happens?**

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: **All** editions except **Database.com**

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.

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.

3. [Choose the report format.](#)
4. Follow the steps of the wizard using the **Next** button. For each report, customize the following:
 - **Specify Row and Column Headers:** On the Select Grouping page for summary and matrix reports, choose the fields by which you want to group and subtotal the data. In a summary report, choosing more than one sort field allows you to subsort your data. For matrix reports, select summary fields for the row labels and column headings. When grouping by a date field, you can further group the data by a specific time period such as days, weeks, or months.

 **Note:** On the Select Grouping page, if you set `Group Dates By` to "Calendar Month in Year" or "Calendar Day in Month," you won't be able to drill down to those date groupings in reports or dashboards. Users are taken to the unfiltered report instead.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

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”

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”

- **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 296.
- **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.

Tip: To use a tabular report on a dashboard, first limit the row count, by setting the `Rows to Display` option, the sort column, and the order on the Select Criteria page of the report. You can't use gauge or metric components on dashboards using tabular reports.

- **Chart Settings:** On the Select the Chart Type and Report Highlights step of the report wizard, [set chart properties](#) to display your report data in a chart. Charts are available only for summary and matrix reports.

5. Click **Run Report** to view the report, or click **Export Details** to save the report as an Excel file or other format.

Tip: Customizing your reports can require running them a few times as you adjust the report criteria and options. We recommend using a filter that gives you a smaller sampling of data until you are finished customizing the report and ready to save.

Group Your Report Data

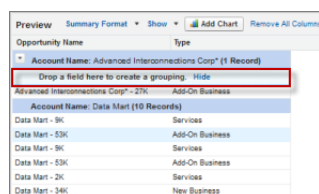
Group data in columns or rows in summary, matrix, and joined reports to display meaningful information. For example, group opportunities by Close Date to see closed opportunities or group cases by product to see the number of cases for each product. You can have groupings inside groupings.

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: **All** editions except **Database.com**

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.

With report builder, 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.



Note: Asynchronous loading isn't available for matrix and joined reports.

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.

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

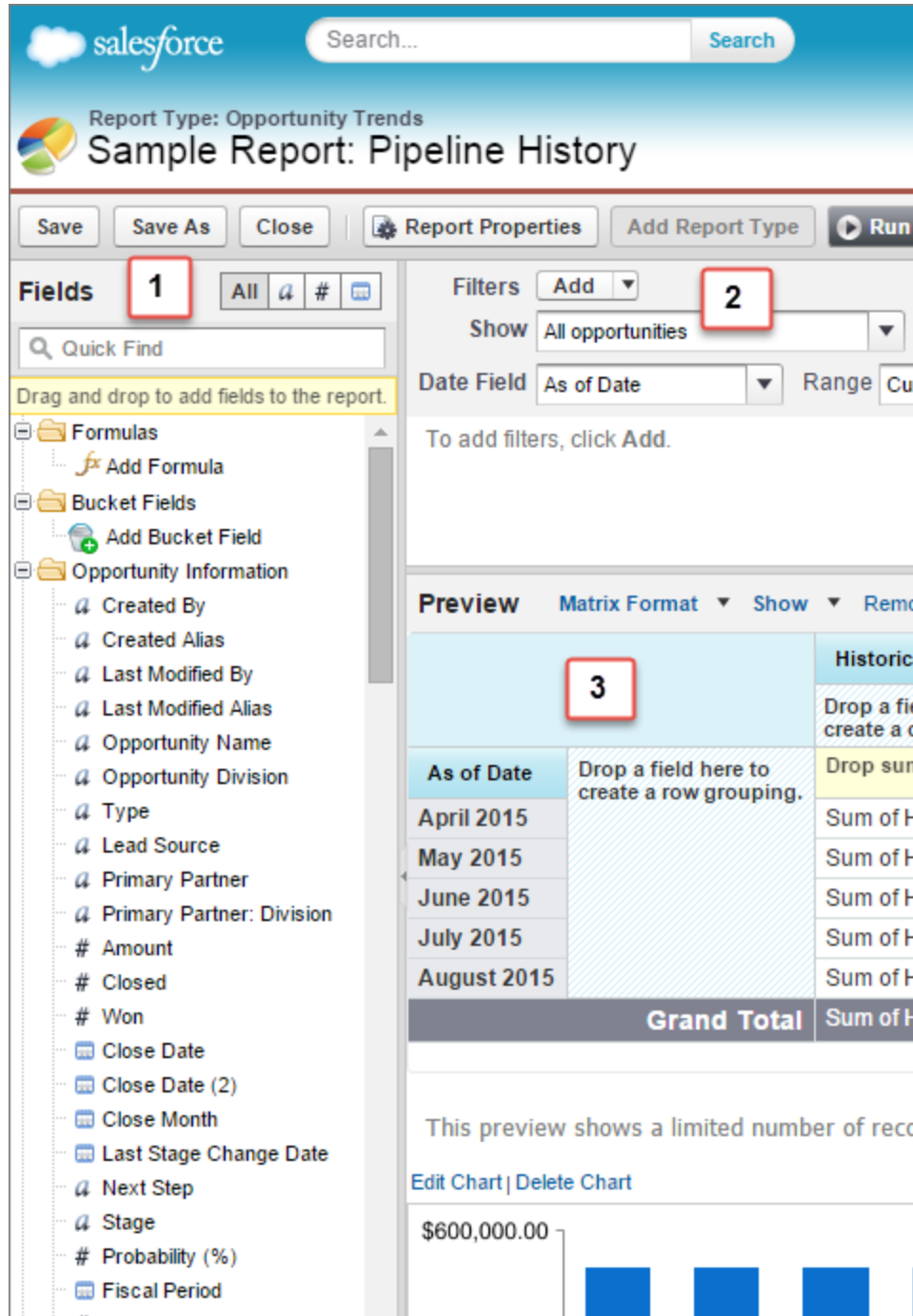
Report builder is a visual editor for reports. The report builder screen lets you work with report fields and filters, and shows you a preview of your report with just some of the data.

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: **All** editions except **Database.com**



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](#)

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

Adding and Removing Fields

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

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

Working with More than One Field

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

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

Note:

- When you add multiple fields, they appear in the report in the order selected.
- For summary and tabular formats, disable **Show > Details** when adding multiple summarizable fields to a report. The Summarize dialog automatically appears, letting you select summaries for all the fields at once.
- Enable **Show > Details** when adding non-summary fields, such as text fields to a report.

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"

- 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: **All** editions except **Database.com**

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](#)

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](#)

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

USER PERMISSIONS


To subtotal report results:


- "Create and Customize Reports"

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 .

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 , 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.

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](#)

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"

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
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.

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"

- 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`.

- Double-click **Add Formula** in the Fields pane.
- In the Custom Summary Formula dialog, under Functions, select *Summary*.
- Select `PARENTGROUPVAL` or `PREVGROUPVAL`.
- Select the grouping level and click **Insert**.
- Define the formula, including where to display the formula.
- 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.

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)`

Matrix: `PARENTGROUPVAL(summary_field, parent_row_grouping, parent_column_grouping)`

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:

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

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”

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:

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

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

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

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](#)


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:



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 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"

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.

Opportunity Name	Amount (Historical)		Close Date (Historical)	
	Yesterday	Today	Yesterday	Today
Acme	\$26,000	\$16,000	2/15/2013	2/28/2013
Advanced Interconnections Corp* - 32K	\$18,000	\$20,000	2/28/2013	2/28/2013
Advanced Interconnections Corp* - 8K	\$12,000	\$8,000	2/15/2013	2/28/2013
Allied Technologies - 40K	\$30,000	\$39,000	2/15/2013	2/15/2013
Data Solutions	\$10,000	\$3,000	2/28/2013	2/28/2013
DFC Inc. - 50K	\$55,000	\$45,700	2/15/2013	2/28/2013
Eastern Shipping	\$21,500	\$21,500	2/28/2013	2/28/2013

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.

EDITIONS

Available in: both Salesforce Classic Lightning Experience


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

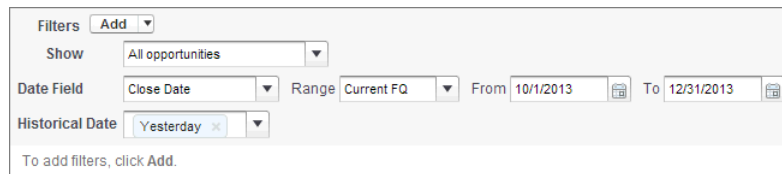
Watch Your Pipeline Change Over Time

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*.
 - 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.



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.

 **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](#)

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 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*.
 - 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**.

EDITIONS

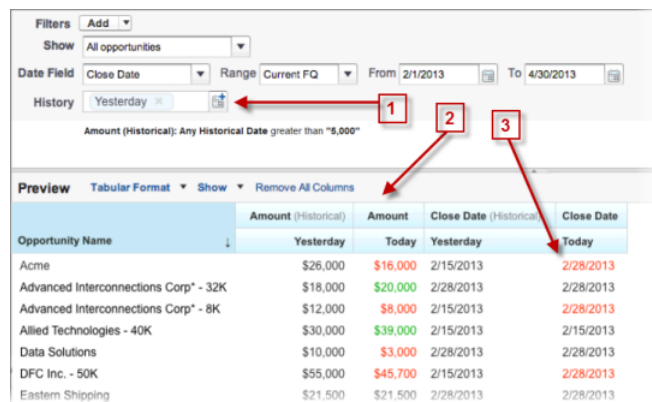
Available in: both Salesforce Classic Lightning Experience

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

USER PERMISSIONS

To create, edit, and delete reports:

- "Create and Customize Reports"
- AND
- "Report Builder"



4. Click **Run Report**.
5. In the finished report, observe that all the deals in the *Amount — Historical* column are worth more than \$5,000.

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



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

SEE ALSO:

[Opportunities with Historical Trending Report](#)

Find Deals that Have Been Pushed Out

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

1. Create an opportunity history report.
 - a. On the Reports tab, click **New Report**.
 - b. Under **Select Report Type > Opportunities**, select *Opportunities with Historical Trending*.
 - c. Click **Create**.
2. For *Show*, select *All Opportunities*.
3. Filter for close dates that are later now than they were in the past.
 - a. Click **Add** and select *Historical Field Filter*.
 - b. Set `Close Date (Historical)` to Feb. 1, 2013.
 -  **Tip:** Use the calendar under **Fixed Days** to select the date.
 - c. Click **OK**.
4. Click the down arrow in the header of the `Close Date` column and select *Show Changes*.
5. Click **Run Report**.
6. For each deal in the report, compare the date in the `Close Date - Historical` column with the date in the `Close Date - Today` column.
 -  **Tip:** You can see changes at a glance by looking for values that are colored green or red.

SEE ALSO:

[Opportunities with Historical Trending Report](#)

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"

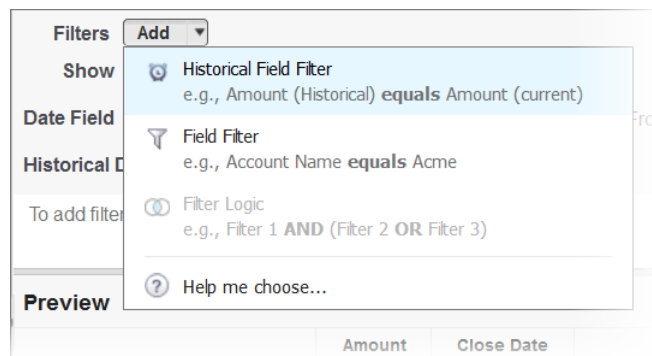
AND

"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*.
 - c. Click **Create**.
2. For *Show*, select *All Opportunities*.
3. Filter out any deal that wasn't worth more yesterday than today.
 - a. Click **Add** and select *Historical Field Filter*.
 - b. Set *Amount (Historical)* to *greater than*.
 - c. Set the last field to *Field*.
Here we're choosing to compare the historical amount with whatever amount is in the *Amount — Today* column, and not with a specific amount.
 - d. Click **OK**.



4. Click **Run Report**.

You've created a simple report that flags deals that have shrunk since yesterday. Note that all the results in the *Amount — Today* column are shown in red, to indicate decreases in value.

Tip: To change the color-coding for amount changes, click the down arrow above the *Change* column and click *Reverse Colors*.

SEE ALSO:

[Opportunities with Historical Trending Report](#)

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"

See How Forecast Amounts Have Changed

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

For this example, we'll assume you're about a month into the current fiscal quarter.

1. Create a forecast history report.
 - a. On the Reports tab, click **New Report**.
 - b. 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

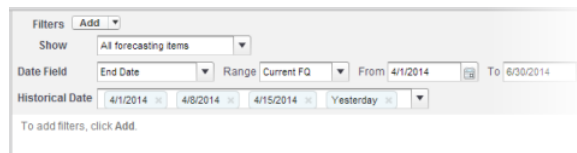
To create, edit, and delete reports:

- "Create and Customize Reports"

AND

"Report Builder"

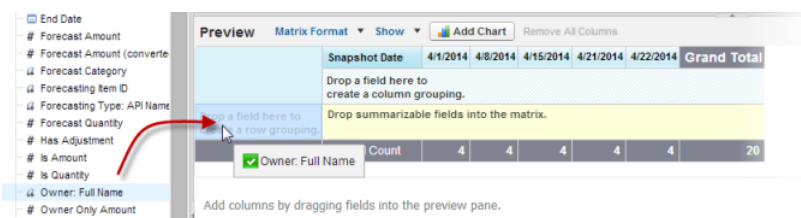
Filter for the historical data you want.



Tip: To move *Yesterday* to the right side of the field, delete it and add it again from the dropdown menu.

3. Choose the data you want to monitor for historical changes. In this case, we're interested in the category in which the changes were made and how the forecast amount changed.
 - a. In the **Preview** pane, change **Tabular Format** to **Matrix Format**.
 - b. Drag the **Owner: Full Name** field from the Field area to the Preview pane to create a row grouping.
 - c. Drag the **Forecast Category** field over to the right of **Owner: Full Name** to create another row grouping.
 - d. Drag the **Forecast Amount (Historical)** field to the matrix area of the Preview pane, below the yellow bar. In the **Summarize** dialog, select **Sum**.

Select the fields that contain the data you want to track.



Columns appear for each of the historical dates you chose in the filters area that contain records.

4. Click **Run Report** to see what data you've gathered so far. Then click **Customize** to keep adjusting your filters and fields if necessary.
5. Once you've defined the data you want to track, create a chart to show it graphically.
 - a. In the Preview area, click **Add Chart** and select the line chart icon.
 - b. On the Chart Data tab, choose `Sum of Forecast Amount (historical)` for the **Y-Axis** and `Snapshot Date` for the **X-Axis**.
 - c. For **Group By**, choose `Forecast Category`, and select **Cumulative**.
 - d. Click **OK**, then click **Run Report** again.
6. Click **Save As**. Choose a name and folder for the report and click **Save**.

Track History for Cases

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

1. One way to optimize contact center operations is to observe the `Status` field over time, watching for cases that move backward to a previous status. This can help reveal ways to resolve cases more effectively.
2. For another example, try analyzing historical values of the `Priority` field to identify cases that may have been incorrectly classified when they were opened. Watching for frequent changes in priority may also lead to ways to improve the handling of complex cases.

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"

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.

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.

EDITIONS


Available in: both Salesforce Classic Lightning Experience

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

 **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.
 - 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.
 - 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.
-  **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*.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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:

[Create a Salesforce Classic Dashboard](#)

[Create a Custom Report in Accessibility Mode](#)

Prepare Reporting Snapshots

To set up a reporting snapshot, you need a source report and a target object with fields to contain the data in the source report.

To set up a reporting snapshot:

1. Create a new custom report that includes the fields you want to load as records into a target object.
2. Create a new custom object in which to store the records loaded from the source report.
3. Create fields on the target object that will receive the source report's results when the reporting snapshot runs.

Tips on Source Reports for Reporting Snapshots

- If you save a tabular source report with its details hidden, the report will not be available to include in a reporting snapshot. Furthermore, if you hide the details of a tabular source report included in a reporting snapshot, the reporting snapshot will fail when it runs. To verify that the details of the tabular source report are not hidden, view the report, click **Show Details**, and save the report. The **Show Details** button only displays if the report's details are hidden.
- When creating the source report for your reporting snapshot, note the names of the fields you added to the report, as those field names may be useful to you when you create fields on the target object in which to store the report results.
- You can choose any custom tabular or summary report as the source report, except legacy forecast reports, Quota vs Actual reports, and Leads by Source reports. The `Source Report` drop-down list does not display standard reports..
- You can include up to 100 fields in your source report.
- You can delete the schedule of when a reporting snapshot runs. You can't stop or pause a reporting snapshot when it is running, nor can you delete its source report. To delete the source report, you must first remove the report from the reporting snapshot by changing the report in the `Source Report` drop-down list..
- If you select Load No Data in the Fields from Source Report column, no data will load into the corresponding field in the Fields in Target Object column when the reporting snapshot runs. .
- The (No fields with compatible data type) field displays in the Fields from Source Report column when a field on the target object does not match the data type of a field on the source report..

Tips on Target Objects for Reporting Snapshots

Consider the following when setting up target objects for reporting snapshots:

- In 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..

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

- 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.

1. From Setup, enter "Reporting Snapshots" in the `Quick Find` box, then select **Reporting Snapshots**.
2. Click **New Reporting Snapshot**.
3. Enter a name, unique name, and description for your reporting snapshot.
4. Choose a user in the `Running User` field by clicking the lookup icon.

The user in the `Running User` field determines the source report's level of access to data. This bypasses all security settings, giving all users who can view the results of the source report in the target object access to data they might not be able to see otherwise.

Only users with the "Modify All Data" permission can choose running users other than themselves.

5. Select a report from the `Source Report` drop-down list.

The report you choose determines the report results that will load as records into the target object when the reporting snapshot runs.

You can choose any custom tabular or summary report as the source report, except legacy forecast reports, Quota vs Actual reports, and Leads by Source reports. The `Source Report` drop-down list does not display standard reports.

6. Select a custom object from the `Target Object` drop-down list.

The custom object you choose will receive the source report's results as records when the reporting snapshot runs.

If a record used for an reporting snapshot has no record type associated with it, the record type of the running user is associated with the reporting snapshot by default.

7. Click **Save** to save the definition of your reporting snapshot, or click **Save & Edit Field Mappings** to save your reporting snapshot and map its fields.
8. [Map the fields](#) on the source report to the fields on the target object.

SEE ALSO:

[Standard Report Types](#)

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

Map Reporting Snapshot Fields

After you create a source report, target object, target object fields, and define your reporting snapshot, you can map the fields on your source report to the fields on your target object. You map source report fields to target object fields so that when the report runs, it automatically loads specific target object fields with data from specific source report fields.

1. From Setup, enter *Reporting Snapshots* in the *Quick Find* box, then select **Reporting Snapshots**.
2. Select the name of the reporting snapshot whose fields you want to map.
3. Click **Edit** on the Field Mappings section.
4. For summary reports, select the *Grouping Level* at which summary data is extracted from the source report. Data loaded into the target object is taken from summary fields at the grouping level you specify. The *Grand Summary* summarizes on the total for all grouping levels.
5. In the Fields from Source Report column, click a *Load No Data* drop-down list and select a field from the source report to map to a custom object field in the Fields in Target Object column. Only summary fields can be mapped for reporting snapshots based on summary reports. Note that the fields for summary reports may vary depending on the grouping level selected.
6. Click **Quick Save** to save field mappings and continue mapping fields, or click **Save** to save field mappings and return to the reporting snapshot's detail page.
7. Next, [schedule](#) the reporting snapshot to run.

Considerations for Mapping Reporting Snapshot Fields

- You must map at least one field from the source report to one field on the target object or data will not load from the source report to the target object when the reporting snapshot runs.
- You can only map fields with compatible data types. For example, you can map a currency field to a number field.
- A custom summary formula can be mapped only if the grouping level in the reporting snapshot and the grouping level in the custom summary formula match.
- If you select *Load No Data* in the Fields from Source Report column, no data will load into the corresponding field in the Fields in Target Object column when the reporting snapshot runs..
- The *(No fields with compatible data type)* field displays in the Fields from Source Report column when a field on the target object does not match the data type of a field on the source report.
- The fields on the target object determine field mapping availability. For example, your source report may include ten fields, but if your target object includes one field, then you can only map one field in your reporting snapshot.
- You cannot map fields from the source report to the following fields on the target object: *Created By*, *Last Modified By*, *Created Date*, and *Last Modified Date*.
- When you map fields from the source report to the target object, some data may lose its context when loaded to the target object. For example, if you map a date and time field from the source report to a text field on the target object, the date and time load to the target object without the time zone.
- When executing a reporting snapshot, if the running user does not have "read" or "write" access to a mapped field in the target object, that field is dropped from the mapping, but does not cause the execution to fail. If a required field in the target object is not mapped, the execution fails. To ensure that fields are always mapped, make them required or set default values for them.

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**.

- 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.



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.

- 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.

- 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.

- 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.

! **Important:** When you delete a reporting snapshot, the source report and target object aren’t deleted; however, when the source report runs, it won’t load the target object with data.

You can delete the schedule of when a reporting snapshot runs. You can’t stop or pause a reporting snapshot when it is running, nor can you delete its source report. To delete the source report, you must first remove the report from the reporting snapshot by changing the report in the **Source Report** drop-down list.

- Display detailed information about a reporting snapshot and customize it further by clicking its name. Then you can:
 - Click links in the Identification section that redirect you to the reporting snapshot running user, source report, and target object. In addition, you can view the preferred date and time at which the reporting snapshot will approximately run next in the **Next Run** field, and view the date and time at which it last ran in the **Last Run** field.
 - Click **Edit** in the Field Mappings section to further customize the fields mapped from the source report to the target object.

The Reporting Snapshot Field Mappings section displays which source report fields are mapped to the target object’s fields. You can view the number of fields in the source report available for mapping to the target object in the **Columns in Source Report** field. Also, you can view the number of fields available for mapping in the target object in the **Fields in Target Object** field.
 - Click **Edit** in the Schedule Reporting Snapshot section to schedule when to run the reporting snapshot.

The Schedule Reporting Snapshot section displays details about when the reporting snapshot is currently scheduled to run.
 - The Run History section displays details about when the reporting snapshot ran. Details include:
 - The date and time at which the reporting snapshot ran
 - The name of the source report, target object, and running user
 - The time it took for the reporting snapshot to run

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

- 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.

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.

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

Source report definition is obsolete.

The report in the `Source Report` field references a custom 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 an 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](#)

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.

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
- "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
- "Report Builder"

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 Classic

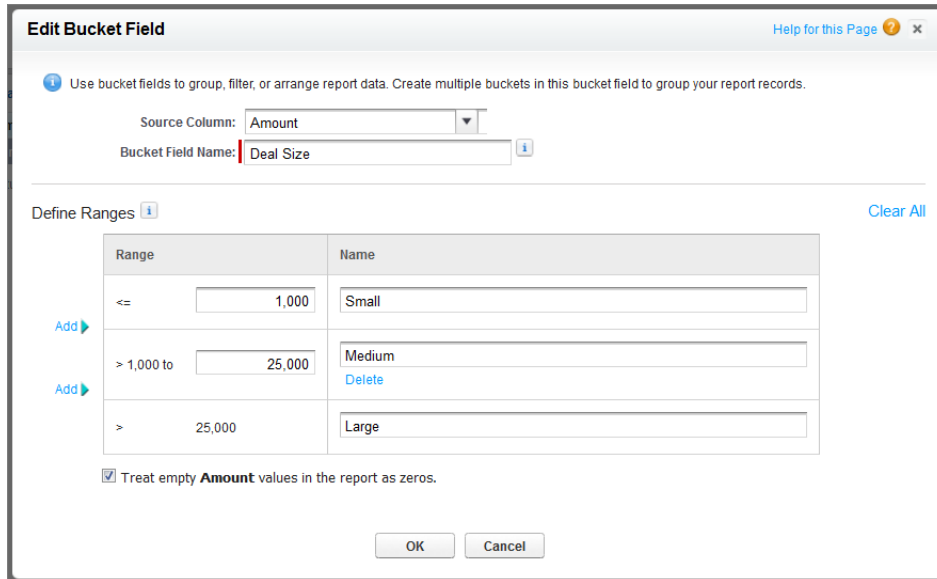
Available in: **Enterprise, Performance, Unlimited,** and **Developer** Editions

USER PERMISSIONS

To create, edit, and delete reports:

- “Create and Customize Reports”
AND
“Report Builder”

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
- "Report Builder"

 **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

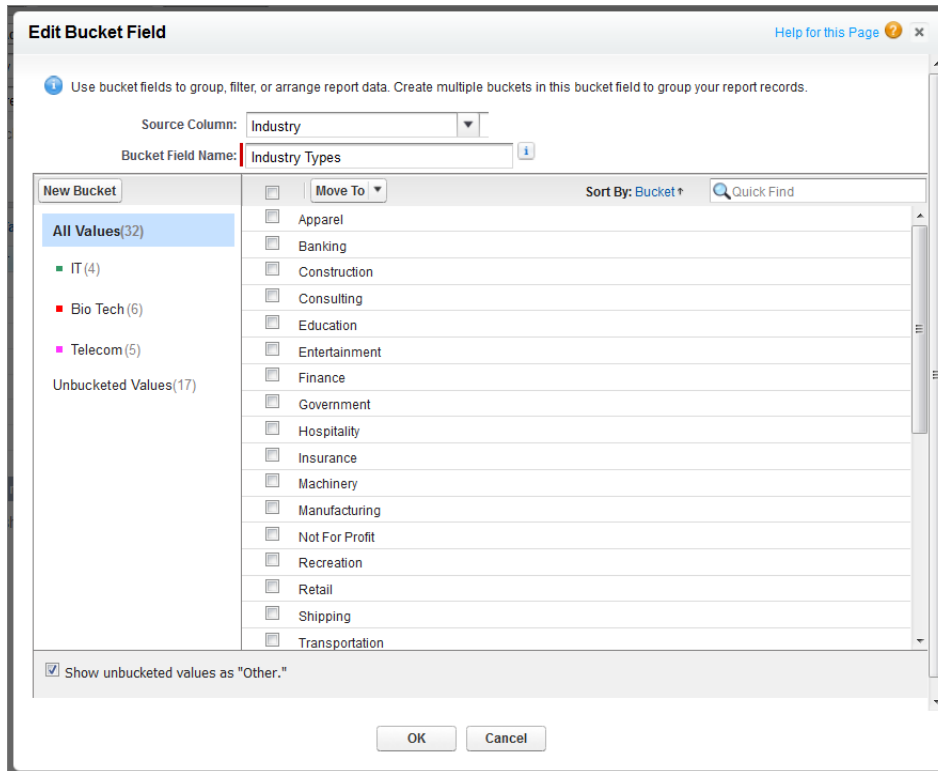
Available in: Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete reports:

- "Create and Customize Reports"
- AND
- "Report Builder"



SEE ALSO:

[Edit a Picklist Bucket Field](#)

Edit a Text Bucket Field

A text bucket helps you sort out values that consist of words or phrases.

 **Note:** The following text types can't be bucketed.

- Text area
- Text area—long
- Text area—rich
- Text area—encrypted
- URL
- Date
- Date/Time

In the Edit Bucket Field overlay for a text field:

1. For **Source Column**, select the field you want to bucket.
2. Enter a bucket field name. This appears as the column name in the report.
Since a bucket field is intended to have multiple buckets within it, a good name for a bucket field describes the scope of the buckets. For example, a bucket field named "Region" could have "East," "West," and "Central" buckets.
3. To create a bucket, click **New Bucket** and enter a bucket name. Create multiple buckets to group your report records.

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"

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
- "Report Builder"

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"

Edit Bucket Field Help for this Page ⓘ x

Use bucket fields to group, filter, or arrange report data. Create multiple buckets in this bucket field to group your report records.

Source Column:

Bucket Field Name: ⓘ

New Bucket	Move To	Sort By: Bucket ↑	Quick Find
Search for Values	<input type="checkbox"/>	Dell Barton Tech	
Enter Values	<input type="checkbox"/>	Dell Fast Buy	
Bucketed Values(3)	<input type="checkbox"/>	Dell Pacific Partners	
■ IBM (3)	<input type="checkbox"/>	HP Media Oats	
■ Dell (3)	<input type="checkbox"/>	HP Wilson & Associates	
■ HP (2)	<input type="checkbox"/>	IBM Ace Fasteners	
	<input type="checkbox"/>	IBM Orange Designs	
	<input type="checkbox"/>	IBM Wrasse Systems	

Show unbucketed values as "Other."

OK Cancel

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](#)

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

“Report Builder”

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
- "Report Builder"

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
- "Report Builder"

- 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

"Report Builder"

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

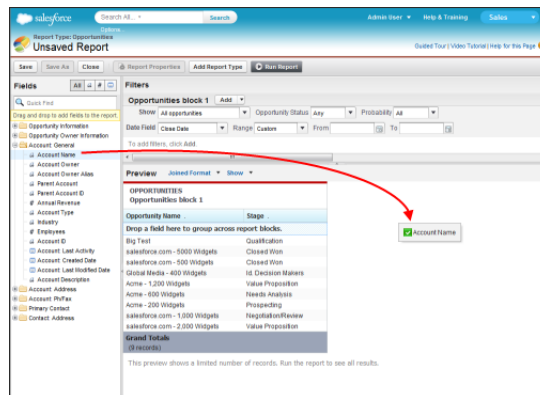
Available in: Salesforce Classic

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

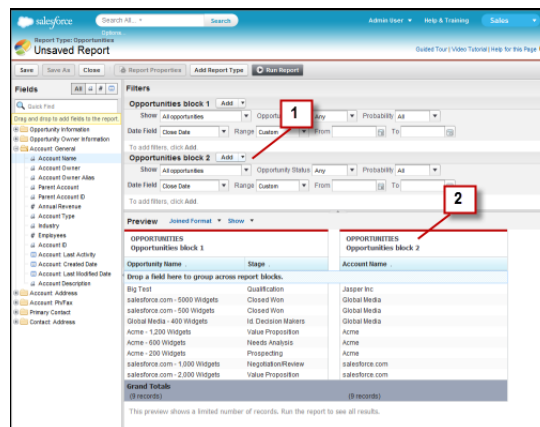
USER PERMISSIONS

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:

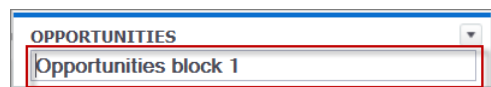
- "Create and Customize Reports"
- AND
- "Report Builder"

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.



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](#)

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"

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**.
-  **Note:** If you haven't summarized any rows in your report blocks and have also hidden both details for the report and row counts for all blocks, your blocks will be hidden on the run reports page. To display the blocks, choose **Show Details** from the run reports page or the report builder.


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
- "Report Builder"

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.

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  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](#)

EDITIONS

Available in: **Salesforce Classic**

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

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**.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

	USER AND OPPORTUNITY OWNER CUSTOM REPORT TYPE Opportunities - Owned by Rep	USER AND OPPORTUNITY CREATOR CUSTOM REPORT TYPE Opportunities - Created by Rep	USER AND ACTIVITY OWNER CUSTOM REPORT TYPE Activities - Owned by Rep
	Record Count .	Record Count .	Record Count .
Full Name: Ely East	25	0	23
Full Name: Joe Seller	8	0	18
Full Name: John Seller	15	0	27
Full Name: Mary Seller	20	13	37
Full Name: Ricky East	25	0	26
Full Name: Wendy West	21	0	37
Grand Totals	114	13	168

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**.

CASES Closed Cases			CASES New Cases			CASES Cases In-Progress		
Case Number	Status	Case Owner	Case Number	Status	Case Owner	Case Number	Status	Case Owner
Priority: Critical								
(2 records)			(2 records)			(8 records)		
00001196	Closed	Jake Borland	00001244	New	Linda Steinberg	00001217	New	Tier 2 Queue
00001258	Closed	Jake Borland	00001217	New	Tier 2 Queue	00001251	Escalated	Jake Borland
						00001244	New	Linda Steinberg
						00001175	Researching	Luke Williams
						00001203	Researching	Luke Williams
						00001182	Escalated	Luke Williams
						00001393	Waiting on customer	Luke Williams
						00001168	Researching	Luke Williams
Priority: High								
(2 records)			(5 records)			(10 records)		
00001195	Closed	Matt Buchanan	00001167	New	Amy Argenta	00001216	New	Tier 3 Queue
00001250	Closed	Tier 1 Queue	00001188	New	Linda Steinberg	00001296	Escalated	Tier 3 Queue
			00001181	New	Luke Williams	00001257	Escalated	Tier 3 Queue
			00001202	New	Matt Buchanan	00001167	New	Amy Argenta
			00001216	New	Tier 3 Queue	00001202	New	Matt Buchanan
						00001188	New	Linda Steinberg
						00001243	On Hold	Linda Steinberg
						00001400	Waiting on customer	Luke Williams
						00001321	Contacting PM	Luke Williams
						00001181	New	Luke Williams
Priority: Medium								
(19 records)			(11 records)			(25 records)		
00001361	Closed	Luke Williams	00001248	New	Linda Steinberg	00001218	New	Tier 2 Queue
00001371	Closed	Luke Williams	00001185	New	Linda Steinberg	00001214	New	Tier 3 Queue
00001382	Closed	Luke Williams	00001169	New	Luke Williams	00001165	New	Tier 1 Queue

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	<code>Opportunity Status equals Closed</code> <code>Date Field equals Close Date</code> <code>Range equals Current and Previous FY</code> <code>Stage equals Closed Lost</code>
Block 2	<code>Opportunity Status equals Closed Won</code> <code>Date Field equals Close Date</code> <code>Range equals Current and Previous FY</code>
Block 3	<code>Date Field equals Close Date</code> <code>Range equals Next Month</code> <code>Opportunity Status equals Open</code>

5. Group the blocks by `Opportunity Owner`.
6. Rename the blocks. For example, "Closed — Won", "Close — Lost", and "Closing Next Month".
7. Create a cross-block custom summary formula that predicts upcoming revenue based on past sales rep performance:

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

8. Add the formula to one or more of the blocks.
9. Optionally, add a cross-block custom summary formula that calculates the win ratio of each sales rep:

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

10. Click **Save** or **Run Report**.

Report Help for this Page ?

Opportunity Pipeline Predictor

« Go to Report List

Show Chatter Follow

Run Report Show Details Report Properties Save Save As Customize Delete Report generation complete.

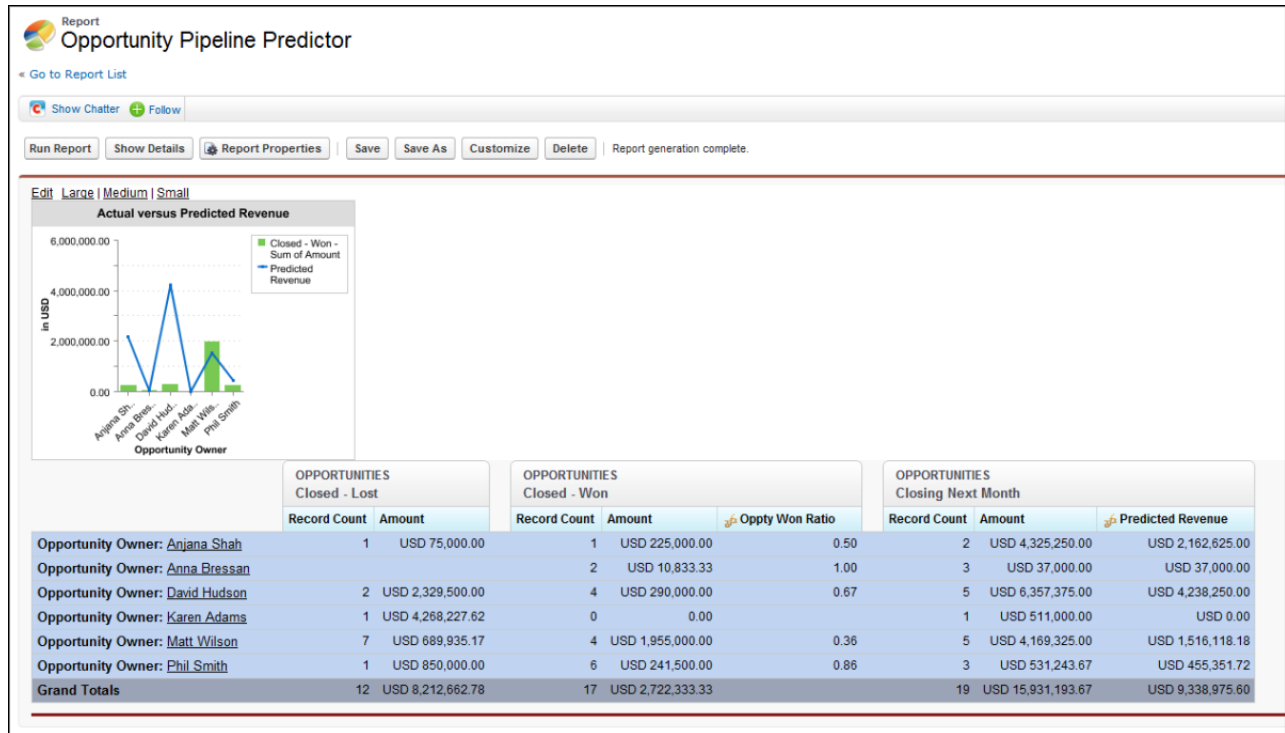
	OPPORTUNITIES Closed - Lost		OPPORTUNITIES Closed - Won			OPPORTUNITIES Closing Next Month		
	Record Count	Amount	Record Count	Amount	Oppty Won Ratio	Record Count	Amount	Predicted Revenue
Opportunity Owner: Anjana Shah	1	USD 75,000.00	1	USD 225,000.00	0.50	2	USD 4,325,250.00	USD 2,162,625.00
Opportunity Owner: Anna Bressan			2	USD 10,833.33	1.00	3	USD 37,000.00	USD 37,000.00
Opportunity Owner: David Hudson	2	USD 2,329,500.00	4	USD 290,000.00	0.67	5	USD 6,357,375.00	USD 4,238,250.00
Opportunity Owner: Karen Adams	1	USD 4,268,227.62	0	0.00		1	USD 511,000.00	USD 0.00
Opportunity Owner: Matt Wilson	7	USD 689,935.17	4	USD 1,955,000.00	0.36	5	USD 4,169,325.00	USD 1,516,118.18
Opportunity Owner: Phil Smith	1	USD 850,000.00	6	USD 241,500.00	0.86	3	USD 531,243.67	USD 455,351.72
Grand Totals	12	USD 8,212,662.78	17	USD 2,722,333.33		19	USD 15,931,193.67	USD 9,338,975.60

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.

- 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.

EDITIONS

Available in: Salesforce Classic

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

- 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.

EDITIONS

Available in: **Salesforce Classic**

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

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

- Accounts and Contacts
 - Account History
 - Account Owners*
 - Contact History
- Activities
 - My Delegated Approval Requests*
- Administrative
 - All Pending Approval Requests*
 - API Usage Last 7 Days*
- Campaign
 - Campaign Call Down*
 - Campaign Member
 - Campaign Member Analysis*
 - Campaigns with Influenced Opportunities
- Contract
 - Contract History
 - Order History
- Customer Support

- Case History
- Self Service Usage
- Solution History
- File and Content
 - Content Authors
 - Content Publication Time Frame
 - File and Content Downloads
 - File and Content Engagement
 - File and Content Links
 - Library Administrators
 - Library Content
 - Most Content Downloads
 - Most Content Subscriptions
 - Stale Content
- 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.

To add or edit a chart:

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:

[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 349 for limits, considerations, and tips.

Chart Data Settings

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

Field	Description
Chart Type	Select the type of chart to use when representing the data in your report. The chart type that you choose determines which chart properties are available to set.
X-Axis and Y-Axis	Choose what values to display on the axes of your chart. Depending on the chart type, axis values can be record count, summary fields, or groupings defined in the report.  Note: If the Y-axis corresponds to a custom summary formula that has the <code>Where Will this Formula Be Displayed?</code> option set to a grouping level other than <code>All summary levels</code> , then the X-axis and Groupings

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"

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"

Field	Description
	selection must correspond to that custom summary formula's grouping level.
Combination Chart	Select this option to plot additional values on this chart. The chart type you chose must allow combination charts.
Groupings	Choose how to group information on your chart. You can only pick from groupings defined in the report. For bar and column charts, click an icon to select the grouping display: side-by-side, stacked, or stacked to 100%. For either single or grouped line charts, you can select Cumulative .
Values	Choose what to display as values for your pie, donut, or funnel chart.
Wedges	Choose what to display as wedges for your pie or donut chart.
Segments	Choose what to display as segments for your funnel chart.

Chart Presentation

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

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

Field	Description
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:

[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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

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 joined	The existing report becomes the first block and the chart remains in the report. If the matrix report used the second column grouping, that grouping is replaced by the first available summary field that's not already used in the chart. If the summary or matrix report included a grouping from a bucket field, the grouping is replaced with the next available grouping not already used in the chart.

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
- “Report Builder”

- 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**.



Note:

- 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:

[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.

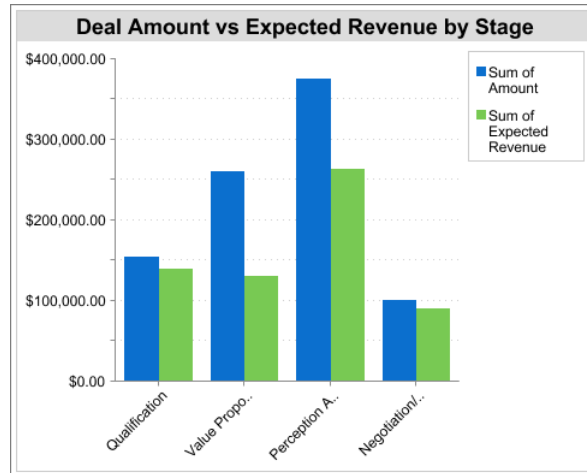
A *combination chart* plots multiple sets of data on a single chart.

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.

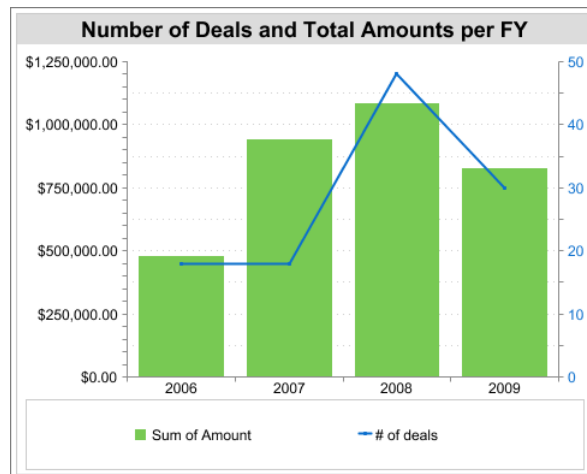


Line-on-Column

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.

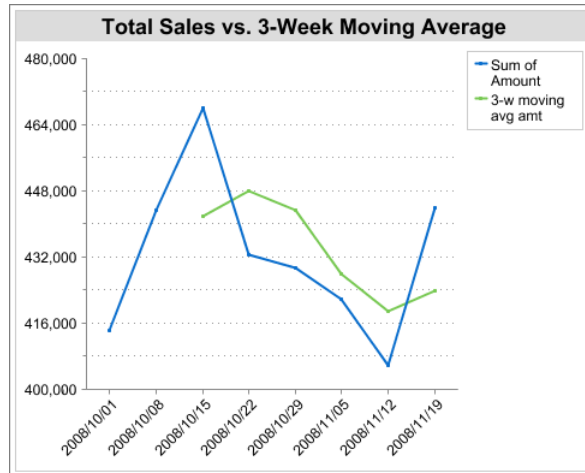


Line-on-Line

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

"Report Builder"

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 High Breakpoint value.

SEE ALSO:

[Build a New Report](#)

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”

Use a Tabular Report in a Dashboard

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

1. Click **Add > Row Limit**.
2. Set the `Row Limit` to `10`, `25`, or `Custom`. If you choose custom enter a number between one and 99.
3. Set the `Sort By` and sort order options. If you chose **Limit Rows by this Field** for a column, these options are already set.
4. Click **OK**.
5. Click **Dashboard Settings** in the toolbar.
6. Choose a `Name` and `Value` to use in dashboard tables and charts. Tables show both name and value. Charts are grouped by name.
7. Click **OK**. You can now use this tabular report as the source report for a dashboard component.

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

Limit Report Results

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

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

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

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

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

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”

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”

with contacts with or without cases report, only fields from accounts and contacts are available to use. If you change the report format, **Row Limit** settings are lost.

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


SEE ALSO:

[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.

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.

EDITIONS


Available in: **Salesforce Classic**

Available in all editions



USER PERMISSIONS

To access Salesforce reports from Excel:

- “Run Reports”
- AND
- “Export Reports”


 **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.

 **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. 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 *XXXXXXXXXX*, you must enter *mypasswordXXXXXXXXXX* to log in.


EDITIONS

Available in: Salesforce Classic


Available in all editions

Import Reports Into Excel with Connect for Office

Import your custom or standard Salesforce reports into Excel so you can further analyze the data using Excel's formulas, charts, and pivot tables. When Salesforce disables TLS 1.0, we're ending support for Connect for Office.


 **Important:** 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"

- 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.

- 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.

EDITIONS

Available in: Salesforce Classic

Available in: **All Editions** except **Database.com**

USER PERMISSIONS

To modify report and dashboard settings:

- "Customize Application"


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.

 **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: **All** editions except **Database.com**

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"

 **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

 **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:

[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.

[Add Filter Logic](#)

Filter logic governs how and when filters apply to your report.

[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.

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 <code>Show</code> and <code>Date Field</code> . <code>Show</code> filters the object around coming groupings (like "My accounts" or "All accounts"). <code>Date Field</code> filters by a field (such as <code>Created Date</code> or <code>Last Activity</code>) 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](#)

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 Name equals 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
- “Report Builder”

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](#)

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
"Report Builder"

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:

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 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
- "Report Builder"

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](#)

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: 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: 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"

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>."
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	<p>Eliminates records that don't contain the value you enter. For example, "Mailing Address Line 2 does not contain P. O. Box."</p> <p> Note: When specifying filter criteria on roll-up summary fields, <code>does not contain</code> uses "or" logic on comma-separated values. On list views, reports, and dashboards, <code>does not contain</code> uses "and" logic.</p> <p>This operator is not case-sensitive.</p>
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	<p>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.</p> <p>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.</p>
excludes	<p>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.</p> <p>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.</p>

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.

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.

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](#)

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 is used as a filter criterion, the picklist value is automatically removed from the filter criteria. For example, if your report contains a filter where *Lead Source equals Email or Web* and you change the picklist value Web to Referral, your report filter changes to *Lead Source equals Email*. If the changed picklist value was the only value specified for a particular filter, it continues to show up in your filters, but an error appears.

Filtering on Blank or Null Values

- When you use the “less than,” “greater than,” “less or equal,” or “greater or equal” operators on fields that have numeric values, records with blank or “null” values are returned as if their value is zero (0). For example, if you create a workflow rule or a lead assignment rule for accounts with the criteria *Annual Revenue less than 100000*, account records match if their Annual Revenue is blank.

However, records with blank field values are not considered matches in report filters, custom list views, and account assignment rules (which assign accounts to territories).

- To limit results to records that are blank or contain “null” values for a particular field, choose the field and the “equals” or “not equal to” operators, leaving the third field blank. For example, *Amount equals* returns records with blank amount fields. You can search for blank values or other specified values at the same time. For example, *Amount equals 1,,2* returns records where the Amount is blank or contains the value “1” or “2”.

Tips for Filtering on Multiple Currencies

Tips for filtering on currency fields when your organization uses multiple currencies.

If your organization uses multiple currencies, follow these tips to create more effective filters:

- Use the *Currency* field to find items with a particular currency. For example, *Opportunity Currency equals AUD* finds opportunities with amounts in Australian dollars.
- Prefix currency amounts with a currency code, such as *Annual Revenue greater than USD 50000000*.

Without the currency code, all amounts are assumed to be in the user's currency. For example, if the user's currency is U.S. dollars, *Annual Revenue greater than 50000000* means 50 million U.S. dollars.

If the user's currency is invalid, the corporate currency is used.

From Setup, enter *Manage Currencies* in the Quick Find box, then select **Manage Currencies** to obtain the currency codes.

- All amounts are converted to the corporate currency for comparison. For example, *Annual Revenue greater than USD 50000000* finds accounts with revenue greater than 50 million U.S. dollars. This would include an account with revenue of 114 million Australian dollars, which is the equivalent of 60 million U.S. dollars, assuming a conversion rate of 1.9.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

- 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](#)

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.

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

Find a Report

USER PERMISSIONS

To search for reports:	“Run Reports”
To search for dashboards:	“Run Reports” AND access to dashboard folder

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

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:

To refine results by	Do this
	<ul style="list-style-type: none"> • <i>Reports</i> • <i>Dashboards</i> • <i>All Types</i> <p>Type filters don't apply to report templates in standard report folders.</p>
View	<p>Select a filter:</p> <ul style="list-style-type: none"> • <i>All Items</i> • <i>Recently Viewed</i> • <i>Items I Created</i> • <i>Items I'm Following</i> <p>View filters.</p>
A selected folder	<p>Select a folder, then type in the search box.</p> <p>Filters are set to <i>All Items</i> and <i>All Types</i> to display all of the folder's contents.</p>
Sorting	<p>Click <input type="button" value="▼"/> 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.</p>

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 <input type="button" value="▼"/> > 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 <input type="button" value="▼"/> 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.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

USER PERMISSIONS

To view the Reports tab:

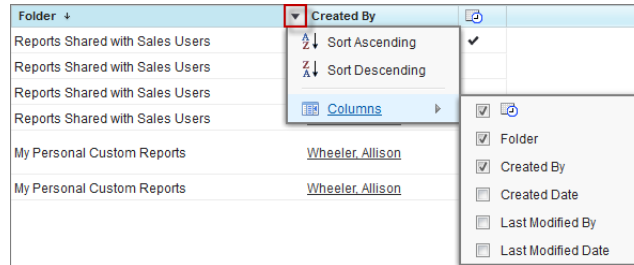
- "Run Reports"

Option**Description**

To change the number of records displayed per page

Click ▼ in the lower left corner of the list and select the desired setting. You can view 10, 25, 50, 100, or 200 records at a time. When you change this setting, you return to the first page of list results.

Once set, however, you view the same number of records throughout the list. The record display setting on the Reports tab doesn't affect list views elsewhere in Salesforce.

Select Column Header for Sorting and Hiding Columns

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](#)

Share a Report

Share your reports with colleagues so everyone's working with the same info.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

USER PERMISSIONS


To print reports:

- "Run Reports" and "Export Reports"

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

 **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 and Dashboards between Folders	
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.

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. [Overview of 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, just embed report charts in detail pages for standard or custom objects. When users see charts on pages, they are empowered to make decisions based on data they see in the context of the page without going elsewhere to look for it. 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.

 **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.

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

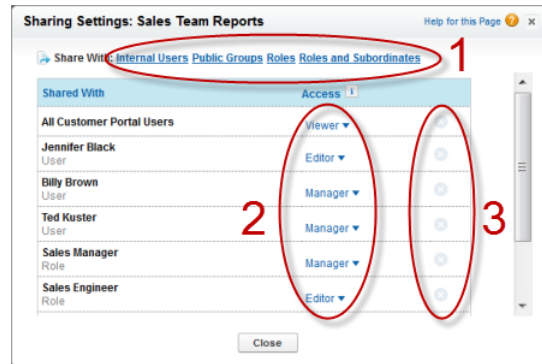
To share a report folder:

- “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](#)

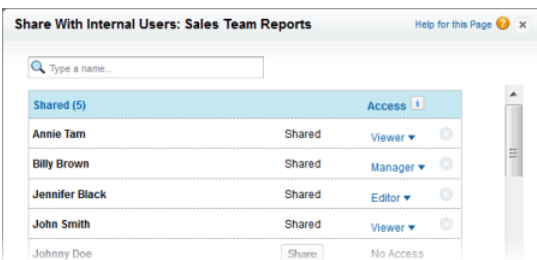
Share a Report or Dashboard with an Individual User

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.



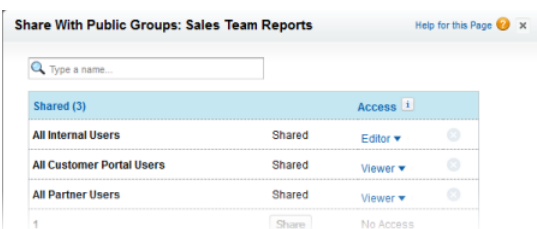
4. Click **Done**, review your changes, and click **Close**.

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**.

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"

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"

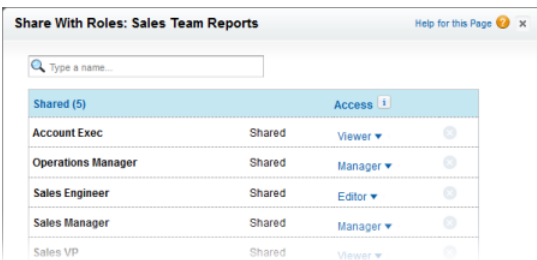
Share a Report or Dashboard by Role

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.



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.)

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"

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

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](#)

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.

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.

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: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**


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	✓	✓	✓
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		✓	✓
<p> Note: If Enhanced Folder Sharing is turned on for your organization, then you also need the “View All Data” permission to change these dashboards.</p>			
Change the folder’s name			✓
Change the folder’s sharing settings			✓

EDITIONS

Available in: Salesforce Classic

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

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

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	<p>Create, edit, delete dashboards¹, 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:</p> <ul style="list-style-type: none"> • Create and Customize Dashboards • Create Dashboard Folders • Edit My Dashboards • View Dashboards in Public Folders <p>(¹To edit a dynamic dashboard, users also need Manage Dynamic Dashboards and View My Team's Dashboards.)</p>
Manage Reports in Public Folders	<p>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:</p> <ul style="list-style-type: none"> • 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. [Export Report Data](#)

You can export your report data to a reusable format when you need to work with it in a dedicated tool, such as Microsoft Excel.

2. [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.

3. [Schedule Reports](#)

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.

4. [Subscribe to Get Report Notifications](#)

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.

SEE ALSO:

[Export a Report](#)

[Schedule a Report for Refresh](#)

Export Report Data

You can export your report data to a reusable format when you need to work with it in a dedicated tool, such as Microsoft Excel.



Note: For security purposes, Salesforce may require users to pass a CAPTCHA user verification test to export data from their organization. This simple text-entry test prevents malicious programs from accessing your organization's data. To pass the test, users must correctly type the two words displayed on the overlay into the overlay's text box field. Note that the words entered into the text box field must be separated by a space.

IN THIS SECTION:

1. [Export a Report](#)

From the Reports tab, you can export data for a selected report to Excel or in comma-delimited format.

2. [Locale Settings for Exporting Report Data](#)

Some tips on using the comma-delimited format and working with Excel when you export report data.

EDITIONS

Available in: **Salesforce Classic**

Available in: **All** editions except **Database.com**


USER PERMISSIONS

To export reports:

- "Export Reports"

Export a Report

From the Reports tab, you can export data for a selected report to Excel or in comma-delimited format.

1. On the Reports tab for a selected report, click  > **Export**.
 - If you set the “Do not save encrypted pages to disk” option in Internet Explorer, you will not be able to open your report online in Excel. You must 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 the **Export File Encoding** for your language.
3. Set the **Export File Format** to Excel or comma delimited.
4. Click **Export**.
5. In your browser’s dialog, choose to save the file.
You can export up to 256 columns and 65,536 rows of data in one report.

SEE ALSO:

- [Run Reports in the Background](#)
- [Locale Settings for Exporting Report Data](#)
- [Print a Report](#)

Locale Settings for Exporting Report Data

Some tips on using the comma-delimited format and working with Excel when you export report data.

Exporting in Comma Separated Value Format

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), the decimal separator is a period (“.”). If your locale setting is French (France), the decimal separator is a comma (“,”). You can override the default separator for your locale by choosing Comma Delimited (non-locale) .csv from the **Export File Format** drop-down list.

Note that Excel does not display the field separator in .csv format. If you want to export reports to Excel in .csv format, we recommend that your locale setting in Salesforce match your Regional Options setting in Windows.

SEE ALSO:

- [Export a Report](#)
- [Print a Report](#)

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

USER PERMISSIONS

To export reports:

- “Export Reports”

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

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.

Contact Salesforce to activate this feature for your organization.

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 are not appropriate for your locale, use a tab or other delimiter.

In the unlikely event that a background export of a report fails, an email notification is sent to you. Reports exported to the background can fail for a number of reasons. For example, between the time you exported the report and the export process began, fields may have been deleted from the report, the report may have been deleted, or the status of the custom report type from which the report was created may have been changed to "In Development."

View and Manage Background Reports

Once you've set up a report to run in the background, you can view and manage it from the background report exports list.

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 may require users to pass a CAPTCHA user verification test to export data from their organization. This simple text-entry test prevents malicious programs from accessing your organization's data. To pass the test, users must correctly type the two words displayed on the overlay into the overlay's text box field. Note that the words entered into the text box field 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.

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"

 **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.

Schedule Reports

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](#)

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](#)

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To schedule reports:

- "Schedule Reports"

Schedule a Report for Refresh


Schedule a report to run daily, weekly, or monthly. An HTML version of the report can be sent by email to users in your organization.

Watch a Demo: [▶ Tips for Scheduling Reports](#)

1. On the Reports tab, click a report name.
2. Click **Schedule Future Runs** from the **Run Report** drop-down menu.
If you're creating a new report, you are asked to save the report in a folder before scheduling. If the report folder is shared with a group, you can schedule the report only for the entire group. To schedule the report for an individual group member, share the report folder with that member.

3. On the Schedule Report page, specify a running user who has access to the folder where the report is stored.


The access level of the running user determines what other users, including portal users, see when they receive the scheduled report run results. This means that not all users can see every field, and the running user can see fields that others may not be able to see. When it's sent, the report indicates which fields the running user can see that can't be viewed by others. You need the "View All Data" permission to specify a running user other than yourself.

 **Note:** If the running user becomes inactive, the report doesn't run. Salesforce sends an email notification to either activate the user, delete the report schedule, or change the running user to an active one. Salesforce sends the notification to users with the "Manage Users," "Modify All Data," and "Manage Billing" permissions. If no user has all these user permissions, Salesforce sends the notification to users with the "Manage Users" and "Modify All Data" user permissions.

4. Select an email setting.

Select	To
To me	Send the report to your email address specified on your user profile.
To me and/or others	Email the report to additional users.

You can send reports only to email addresses included on Salesforce user records. When portal users receive emailed reports, they see the same data as the running user set in the report schedule. If you have information you'd rather not share, schedule the report to run with a portal user as the running user.

 **Note:** Portal users receive report and dashboard refresh email notifications when the `Allow Reports and Dashboards to Be Sent to Portal Users` option is enabled.

5. Set the frequency, duration, and time for running the report:
 - In the `Frequency` field, select `Daily`, `Weekly`, or `Monthly` and then refine the frequency criteria.
 - Using the `Start` and `End` fields, specify the dates during which you want to schedule the report. To enter the current date, click the link showing the date.
 - Next to `Preferred Start Time`, click **Find available options** to choose a start time.

Your preferred start time might not be available if other users have already selected that time to schedule a report.

6. Click **Save Report Schedule**. You can choose:

EDITIONS

Available in: **Salesforce Classic**

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

USER PERMISSIONS

To schedule reports:

- "Schedule Reports"

Choose**To****Save report modifications with this schedule**

Save both the report schedule and changes you made to the report.

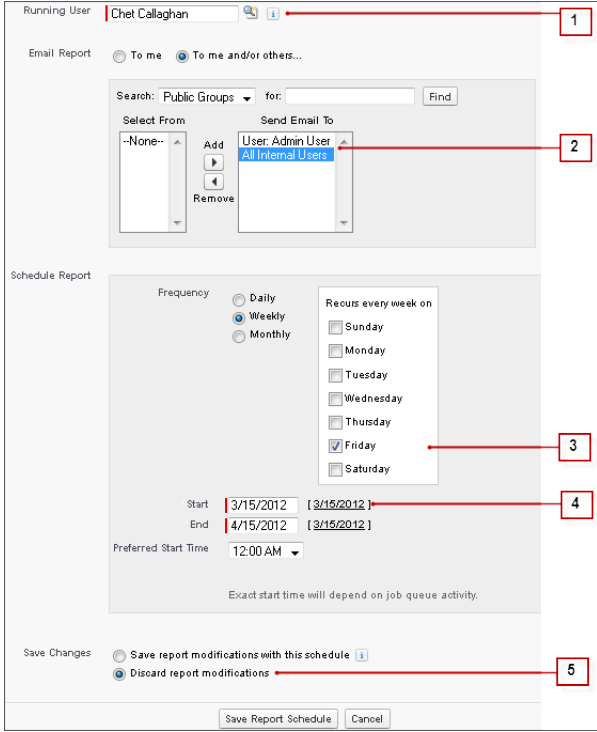
Discard report modifications

Save the schedule only. Changes you made to the report are discarded.

Report recipients can click the report name in emailed reports to log in to Salesforce and view the report directly.

 **Example: Example**

A report is scheduled to run every Friday at midnight, and its results are emailed to a selected group and user.



The screenshot shows the 'Schedule Report' configuration window. At the top, the 'Running User' is set to 'Chet Callaghan' (callout 1). Below this, the 'Email Report' section has 'To me and/or others...' selected. The 'Send Email To' list includes 'User: Admin User' and 'All Internal Users' (callout 2). The 'Schedule Report' section shows 'Weekly' frequency with 'Friday' selected under 'Recurs every week on' (callout 3). The 'Start' date is '3/15/2012' (callout 4) and the 'End' date is '4/15/2012'. The 'Preferred Start Time' is '12:00 AM'. At the bottom, 'Discard report modifications' is selected under 'Save Changes' (callout 5). Buttons for 'Save Report Schedule' and 'Cancel' are at the bottom.

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](#)

- To see the schedule for a report on the Reports tab, hover over  in the schedule  column. This shows the frequency and the date of the next run. Users without the "Schedule Reports" permission can't see the icon and information.
- To see a report's run schedule on the Schedule Report page:
 1. Click a scheduled report name on the Reports tab.
 2. Click **Schedule Future Runs** from the **Run Report** drop-down menu..
- To see all scheduled reports for your organization, from Setup, enter *Scheduled Jobs* in the **Quick Find** box, then select **Scheduled Jobs**. Only users with the "View Setup and Configuration" permission can view this information.

SEE ALSO:

[Manage a Report's Schedule](#)

Manage a Report's Schedule

Create, change, view or delete a scheduled report from the Schedule Report page.

[Tips for Scheduling Reports](#)

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](#)

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"

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"

Change a Report's Schedule

You can make changes to an already scheduled report on the Schedule Report page.

[Tips for Scheduling Reports](#)

1. On the Reports tab, click the name of the scheduled report.
2. Click **Schedule Future Runs** from the **Run Report** drop-down menu.
3. Make the required changes on the Schedule Report page.
4. Click **Save Report Schedule**.

SEE ALSO:

[Manage a Report's Schedule](#)

Delete a Report's Schedule

Select a scheduled report and unschedule it to delete its scheduled run.

[Tips for Scheduling Reports](#)

1. On the Reports tab, click the name of the scheduled report.
2. Click **Schedule Future Runs** from the **Run Report** drop-down menu.
3. Click **Unschedule Report**.
The run schedule for the report is canceled and *not* sent to the Recycle Bin.



SEE ALSO:

[Manage a Report's Schedule](#)

Tips on Scheduling Reports

Some tips to keep in mind about timings, limits, and email notifications when scheduling a report.

Scheduling Report Runs

- On the Reports tab, hover over  in the schedule column () to [view](#) a report's schedule. Note that users without the "Schedule Reports" permission can't see this icon and information.
- You can't create schedules for [joined reports](#).
- Your organization is limited to no more than 200 scheduled reports. Daily limits differ by edition. Additional scheduled reports may be available for purchase.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To schedule reports:

- "Schedule Reports"

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

- 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).
- 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](#)

Subscribe to Get Report Notifications

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](#)

 **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**.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

USER PERMISSIONS

To subscribe to reports:

- "Run Reports"

To enable or disable report notification subscriptions:

- "Customize Application"

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.

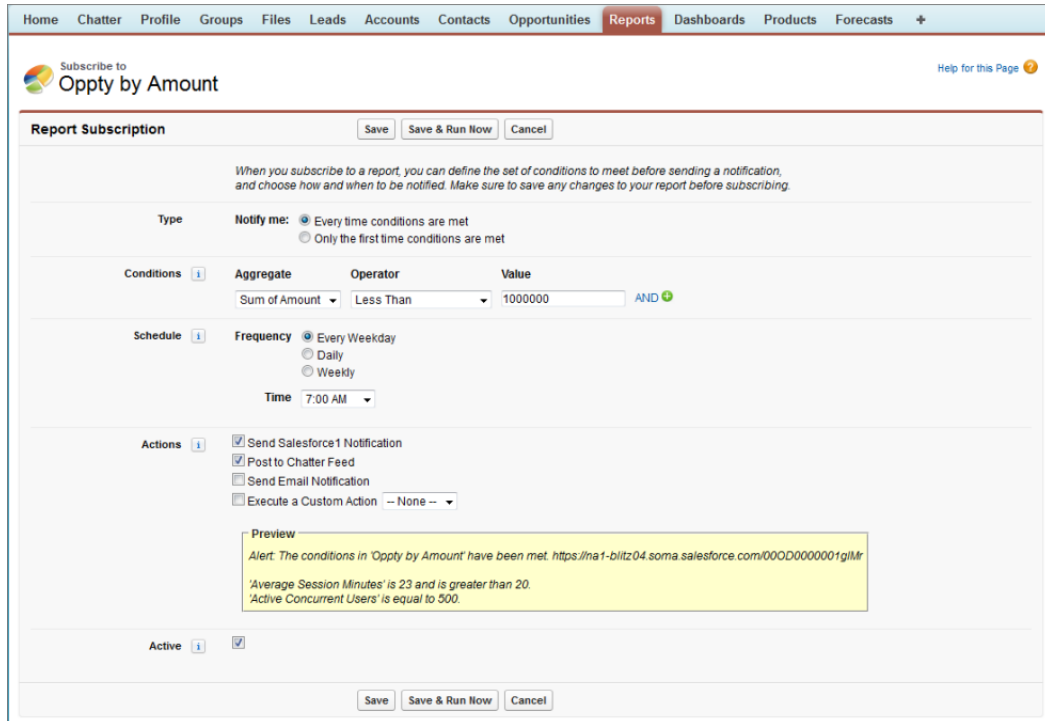
5. Select one or more notification types.

- 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.

 **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.



The screenshot shows the 'Report Subscription' configuration interface in Salesforce. At the top, there is a navigation bar with tabs for Home, Chatter, Profile, Groups, Files, Leads, Accounts, Contacts, Opportunities, Reports (selected), Dashboards, Products, and Forecasts. Below the navigation bar, the page title is 'Subscribe to Oppty by Amount' with a 'Help for this Page' link. The main section is titled 'Report Subscription' and includes 'Save', 'Save & Run Now', and 'Cancel' buttons. A descriptive text states: 'When you subscribe to a report, you can define the set of conditions to meet before sending a notification, and choose how and when to be notified. Make sure to save any changes to your report before subscribing.'

The configuration is divided into several sections:

- Type:** 'Notify me:' with radio buttons for 'Every time conditions are met' (selected) and 'Only the first time conditions are met'.
- Conditions:** A table with columns 'Aggregate', 'Operator', and 'Value'. The first row shows 'Sum of Amount' as the aggregate, 'Less Than' as the operator, and '1000000' as the value. An 'AND' button is visible to the right.
- Schedule:** 'Frequency' with radio buttons for 'Every Weekday' (selected), 'Daily', and 'Weekly'. Below it, 'Time' is set to '7:00 AM'.
- Actions:** Checkboxes for 'Send Salesforce1 Notification' (checked), 'Post to Chatter Feed' (checked), and 'Send Email Notification' (unchecked). There is also an option for 'Execute a Custom Action' set to 'None'.
- Preview:** A yellow box showing a sample alert: 'Alert: The conditions in 'Oppty by Amount' have been met. https://na1-blitz04.soma.salesforce.com/000D0000001gllMr. 'Average Session Minutes' is 23 and is greater than 20. 'Active Concurrent Users' is equal to 500.'
- Active:** A checkbox that is checked.

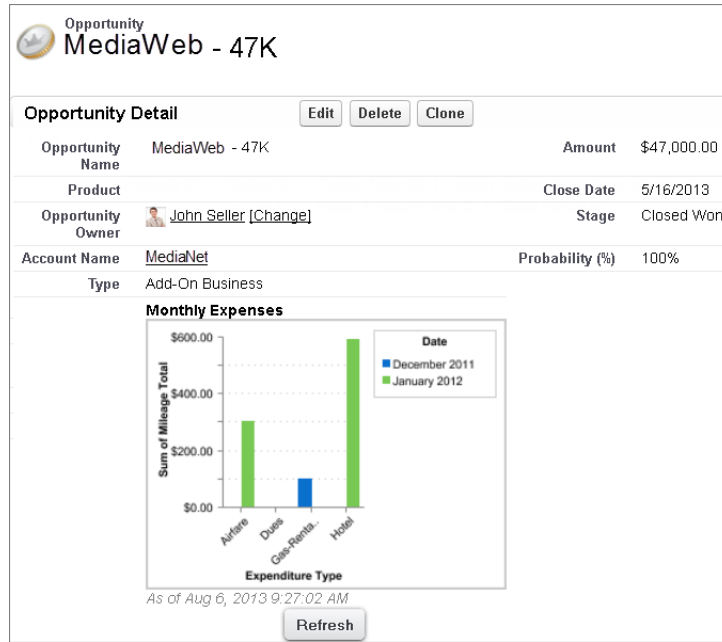
At the bottom, there are 'Save', 'Save & Run Now', and 'Cancel' buttons.

SEE ALSO:

[Schedule Reports](#)

Overview of 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, just embed report charts in detail pages for standard or custom objects. When users see charts on pages, they are empowered to make decisions based on data they see in the context of the page without going elsewhere to look for it. 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 standard or custom 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

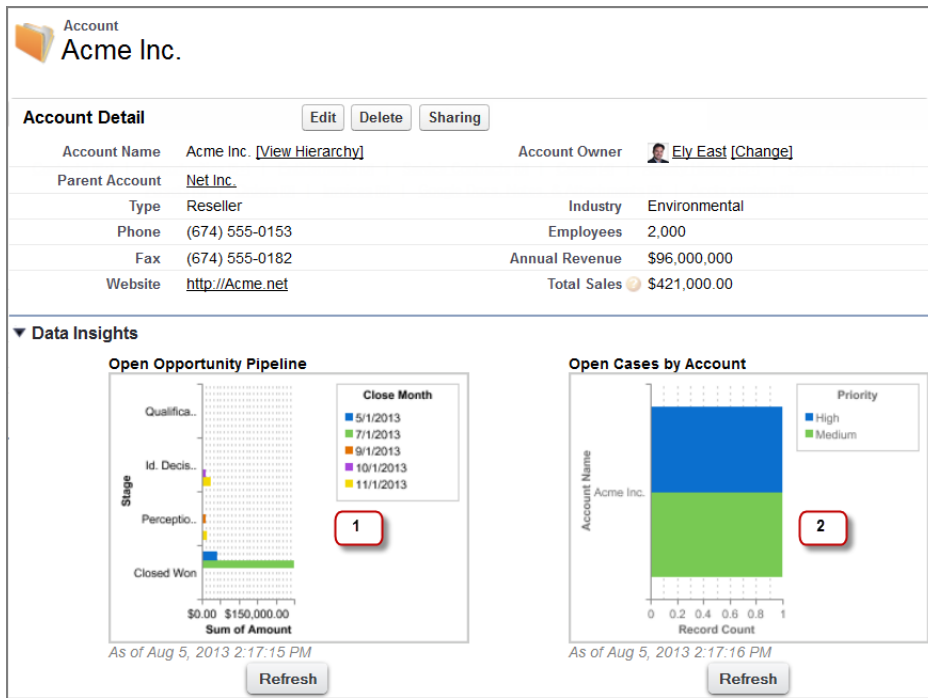
In this example, we've embedded two report charts on an important account page that show deals in the pipeline and open support cases for the account. From looking at the charts, the account executive can quickly gauge the account's activity and health.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

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 standard or custom 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**.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

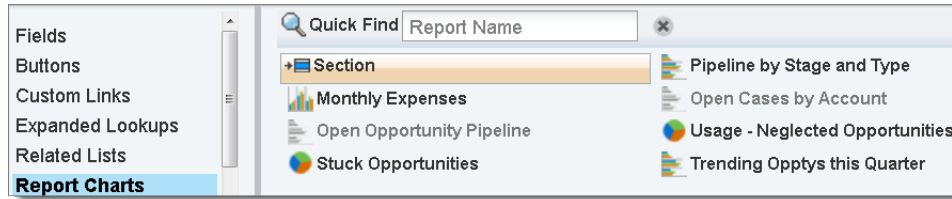
USER PERMISSIONS



To view page layouts:

- “View Setup”

To edit page layouts:

- “Customize Application”



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.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

USER PERMISSIONS

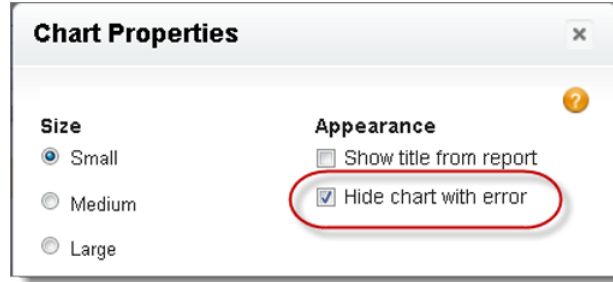
To view page layouts:

- "View Setup"

To edit page layouts:

- "Customize Application"

To hide such a chart, select **Hide chart with error** from the Chart Properties dialog box in the page layout editor.



A chart can show an error on a page for one or more of these reasons.

- The viewer doesn't have access to the field that's used for filtering.
- The viewer doesn't have access to the report folder.
- The report chart has been deleted from the report.
- The report definition has changed.
- The report itself is no longer available.

Filtering Report Charts to Show Data Relevant to the Page

Charts usually filter to show relevant data when the chart's report type has a matching ID field for the record, such as Account ID for an account record. Occasionally, charts show data unfiltered. But you can set up a chart to be filterable if its report type has a lookup to the matching ID field.

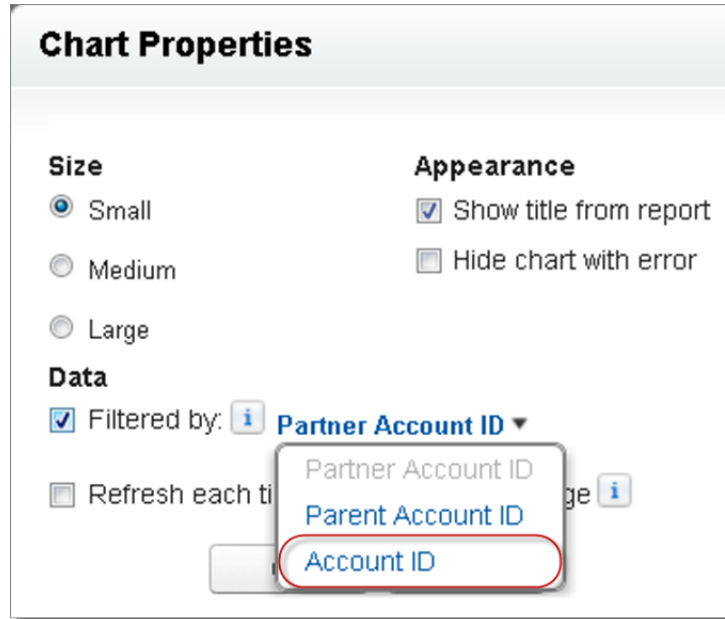
Make a Chart Filterable

When the chart's report type doesn't have a matching ID field for the record, the chart is not filtered. Instead, it shows all data.

A report chart on open cases shows information for all accounts on an account record. To make the chart filterable, edit the layout of the chart's cases custom report type, and add the `Account ID` field via lookup. Now when you view an account record, the chart is filtered.

Pick a Relevant Filter

Sometimes when there are multiple ID fields available to filter, the chart may not show any data. In such situations, select the most relevant ID field from the **Filtered by** drop-down in the Chart Properties dialog box. A chart on open opportunities on account records is filtered by `Partner Account ID`, which isn't that useful. But we can edit the chart properties and choose `Account ID` from the drop-down to filter by the account record.



 **Note:** Charts can't filter for relevant data if the source report has reached the limit of 20 field filters.

Refreshing Report Chart Data

Normally, charts refresh data once every 24 hours. But you can change a chart to refresh each time someone opens the page it's on. The refresh option is under the Chart Properties dialog box of the page layout editor. However, we recommend daily refresh over selecting the option, because users will soon reach the refresh limit or will wait for chart data to show until refresh is complete.

Daily Refresh

Charts refresh data once every 24 hours. If within that time users want the latest, they can click **Refresh** on the chart.

Refresh When User Opens the Page

To change a chart's normal refresh, select **Refresh each time a user opens the page** in the Chart Properties dialog box of the page layout editor. This option triggers a chart refresh each time someone opens the page the chart is on. Selecting the option is not recommended for two reasons.

- There's a risk of reaching the chart refresh limit faster. Refreshes count towards the hourly limit for each user and organization.
- For reports that take longer to run, selecting this option can make users wait to see chart data.

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.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

- Your organization 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:

- [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.

- [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.

- [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.

- [Delete a Report](#)

You can delete a selected report from the Reports tab or from its run page.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

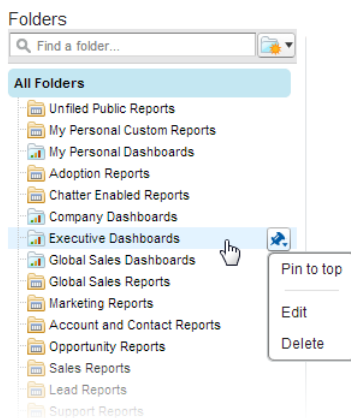
Available in: **All** editions except **Database.com**

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.

- In the list of report and dashboard folders, hover over any folder, then click .

Reports & Dashboards



- Select **Pin to top**.

Your folder moves to the top of the folders list.

EDITIONS

Available in: Salesforce Classic

Available in all editions

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.

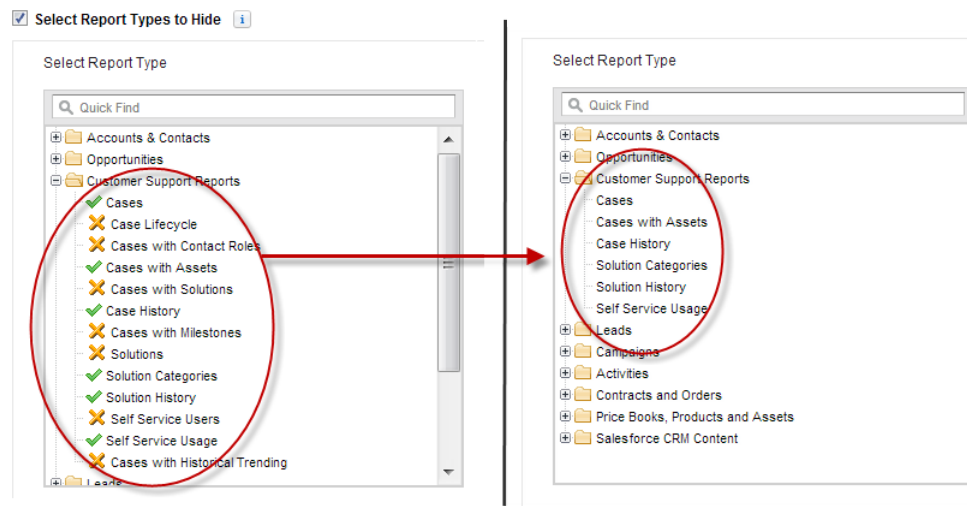
1. On the Reports tab, click **New Report**.
2. Select **Select Report Types to Hide**.
The green check mark next to a report means it's visible to everyone.
3. To hide the report type, click the check mark to change it to an X.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

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.

EDITIONS

Available in: Salesforce Classic

Available in: **All** editions except **Database.com**

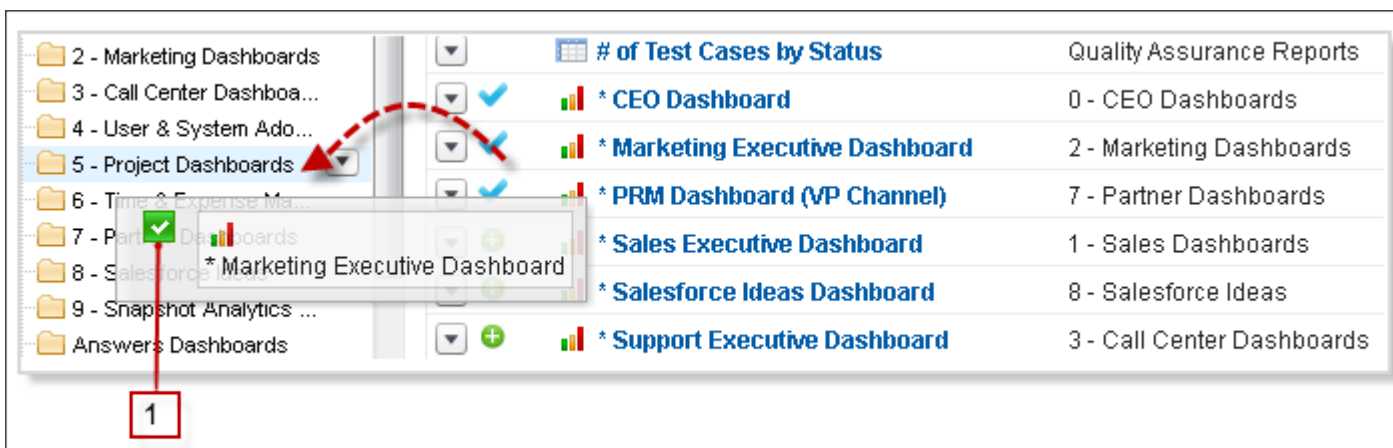
USER PERMISSIONS

To view the Reports tab:

- "Run Reports"



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](#)

Report Limits

As you report on your Salesforce data, be aware of these limits, restrictions, and differences between reporting in Lightning Experience and Salesforce Classic.

Salesforce Reports and Dashboards Limits

Feature	Personal Edition	Contact Manager	Group Edition	Professional Edition	Enterprise Edition	Unlimited and Reference Edition	Developer Edition
Field Filters per Report ¹	20	20	20	20	20	20	20
Formulas per Report	5	5	5	5	5	5	5
Scheduled Reports (Emailed reports can be up to 10 MB.)	N/A	N/A	N/A	One per hour ² (off-peak hours only ³) (Limited to three preferred start	One per hour ²	Two per hour ²	One per hour ² (off-peak hours only ³) (Limited to three preferred start

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

USER PERMISSIONS

To delete reports in My Personal Custom Reports folder:

- "Create and Customize Reports"

To delete reports in public folders:

- "Manage Public Reports"

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
				times, which can't be changed)			times, which can't be changed)
Dashboard Filters	N/A	N/A	N/A	3 per dashboard	3 per dashboard	3 per dashboard	3 per dashboard
Dynamic Dashboards	N/A	N/A	N/A	N/A	Up to five per organization	Up to 10 per organization	Up to three per organization
Scheduled Dashboard Refreshes	N/A	N/A	N/A	N/A	One per hour ²	Two per hour ²	N/A
Custom Report Types (Limits apply to all custom report types regardless of development status.)	N/A	N/A	N/A	50	200	2,000	400
Reporting Snapshots	N/A	N/A	N/A	One per hour ² (off-peak hours only ³) (Limited to one preferred start time per day, which can't be changed)	One per hour ²	Two per hour ²	One per hour ² (off-peak hours only ³) (Limited to one preferred start time per day, which can't be changed)

¹ These limits apply to the report builder. If you're using the report wizard, the limit is 10.

² Up to a possible maximum of 200, total.

³ Off-peak hours are between 6 PM and 3 AM local time.

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.
- 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 organization 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.

Acceptable range for values: The maximum value allowed is 9999999999999999. The minimum value allowed is -9999999999999999.

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 organization 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 organization can request up to 200 dashboard refreshes per hour.
- Your organization 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:

1. [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.

2. [Why doesn't my report return any data?](#)

Check with your administrator to make sure 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.

3. [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.

4. [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 you are asking the right questions of your data.

5. [What are some common report limits?](#)

Here are some common report limits. To see a full list of analytics limits, see the Salesforce Limits guide.

6. [Why am I getting an "obsolete report" error message?](#)

You may be trying to report on data that isn't available.

7. [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.

8. [Why is an old role name appearing in the "role hierarchy" trail of my report?](#)

Changes to role names aren't automatically updated in reports. To see the latest role name, change the `Role name as displayed on reports` field for the role record.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

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 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.

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.

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 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.

What are some common report limits?

Here are some common report limits. To see a full list of analytics limits, see the Salesforce Limits guide.

Feature	Limit	Editions Info
Bucket fields per report	Up to 5 (each bucket field can have 20 buckets)	Available in Enterprise, Unlimited, Performance, and Developer 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.

1 2 3

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](#).

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:

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

- Querying too many objects
- Dealing with intricate lookups
- Has too many fields

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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

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"

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.

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

Option	Description
	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. Select --Current-- 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 <code>Product Name</code> 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® 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 <code>From</code> and <code>To</code> 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.
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.

Option	Description
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: **All** editions except **Database.com**

Dashboards

Learn about Dashboards

Dashboards give over-arching views of your Salesforce data. 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 the 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.

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. [Create a Salesforce Classic Dashboard](#)
Create a dashboard to provide a graphical view of the data in your reports.
3. [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.

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: **All** editions except **Database.com**

Create a Salesforce Classic Dashboard

Create 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 Wave Analytics, review the article [Create a Dashboard in Wave Analytics](#) on page 175.

 **Tip:** You can clone a dashboard to quickly create a new 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 create a 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: 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:

- "Run Reports" AND "Manage Dynamic Dashboards"

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.

 **Note:** There are some things you can't do with dashboard filters.

- 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)

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.

1. 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.

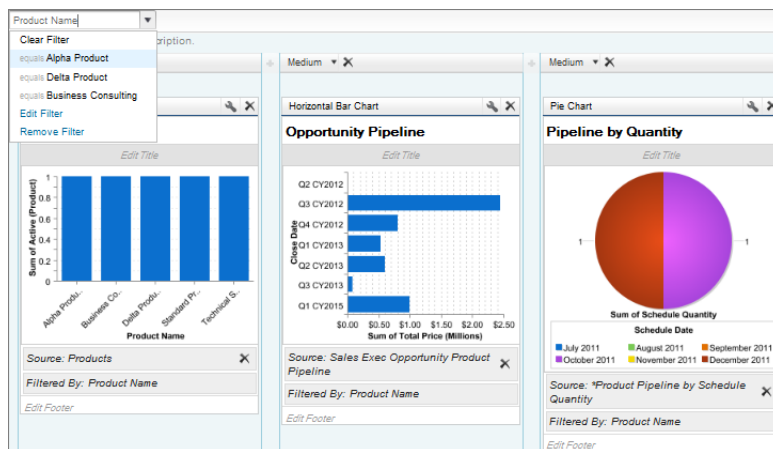
EDITIONS

Available in: Salesforce Classic

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

Sales Performance by Products

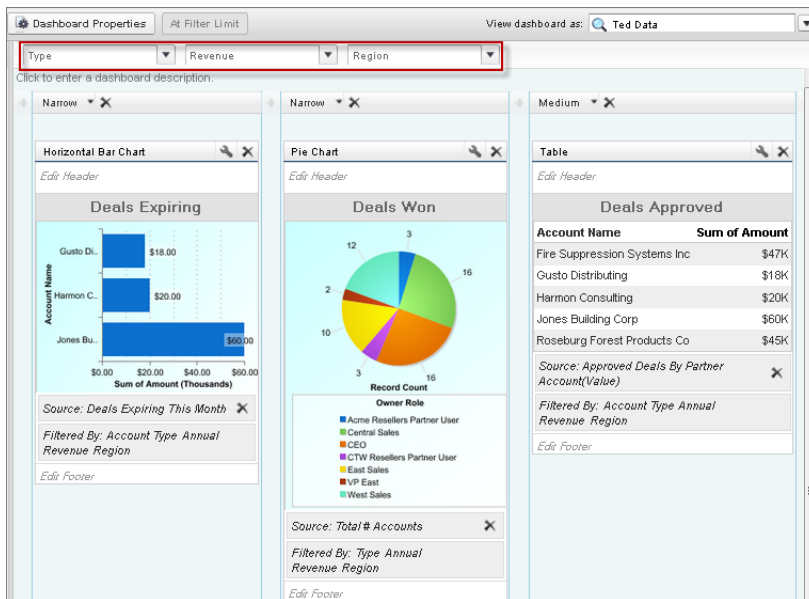
Create a sales dashboard that lets viewers track sales performance by products. To do so, create a dashboard with key performance indicators, like closed revenue. Add a filter on the **Product Name** field so viewers can see performance by product.



Deals for Different Account Segments

Show how different types of deals are performing across different account segments by creating a dashboard with two filters. Create a filter for **Annual Revenue** that contains several ranges that reflect how your organization segments accounts, then create a filter

on opportunity Type, and finally add a filter to show deals by Region. Create this as a dynamic dashboard to let the entire sales organization use it: managers can use it to view the performance of their teams, while reps can use it to monitor their own performance.



Provide Individualized Views of a Dashboard

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:

- **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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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

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 five dynamic dashboards for Enterprise Edition, 10 for Unlimited and Performance Edition, and three for Developer Edition. Additional dynamic dashboards may be available for purchase.

 **Note:**

- Dynamic dashboards are viewable in both Salesforce Classic and Lightning Experience, but you must switch to Salesforce Classic to set them up.
- Following components isn't supported for dynamic dashboards.
- 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!

IN THIS SECTION:

1. [Set Up Dynamic Dashboards](#)

To set up a dynamic dashboard, switch to Salesforce Classic. Create a folder to hold the dashboard and its underlying reports, then create the dashboard.

SEE ALSO:

[Filter Your Dashboard](#)

[Create a Salesforce Classic Dashboard](#)

[Set Up Dynamic Dashboards](#)


Set Up Dynamic Dashboards


To set up a dynamic dashboard, switch to Salesforce Classic. Create a folder to hold the dashboard and its underlying reports, then create the dashboard.

Your organization can have up to five dynamic dashboards for Enterprise Edition, 10 for Unlimited and Performance Edition, and three for Developer Edition.

Note:

- Dynamic dashboards are viewable in both Salesforce Classic and Lightning Experience, but you must switch to Salesforce Classic to set them up.
- 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.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

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"

 **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](#)

[Choose a Dashboard Running User](#)


Choose a Dashboard Running User

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. Consider creating separate dashboards for users with different license types.

To select a dashboard's running user, switch to Salesforce Classic and then:

1. Edit a dashboard.
2. Click the  button next to the `View dashboard as` field.

 **Note:** If you don't have “Manage Dynamic Dashboards” permission, just enter a running user and skip to the final step. Enter “*” to see all available users.

3. Choose a running user setting.
 - **Run as specified user.** The dashboard runs using the security settings of that single, specific user. All users with access to the dashboard see the same data, regardless of their own personal security settings. This approach is perfect for sharing the big picture across a hierarchy or motivating team members by showing peer performance within a team. If you don't have “View All Data,” you can only choose yourself. If you have “View My Team's Dashboards,” you can choose any user below you in the role hierarchy.
 - **Run as logged-in user.** A *dynamic dashboard* runs using the security settings of the user viewing the dashboard. Each user sees the dashboard according to his or her own access level. This approach helps administrators share one common set of dashboard components to users with different levels of access.

EDITIONS

Available in: both Salesforce Classic and 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 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”

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**.
 6. In the `View dashboard as` field, enter a running user.
 7. Save your dashboard.

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.

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 *Scheduled Jobs* in the Quick Find box, then select **Scheduled Jobs**. Users with “Modify all Data” permission can click **Del** next to a specific scheduled dashboard refresh to permanently delete all instances of the scheduled refresh.

You can also set up Salesforce to send an email with an HTML version of the dashboard when the refresh completes. For email applications that don't support HTML, the email includes text and a link to the dashboard.

1. On the Dashboards tab, select a dashboard using the `View Dashboard` field.
2. Click **Refresh** and choose **Schedule Refresh**


Filters aren't applied when you schedule or email a dashboard.

Each dashboard has a *running user*, whose security settings determine which data to display in a dashboard.

 **Note:** If the running user becomes inactive, the dashboard doesn't run.

3. Select notification settings.

- Click `To me` to send an email to your user's address.
- Click `To others...` to send an email to additional Salesforce users.

 **Note:** Portal users receive report and dashboard refresh email notifications when the `Allow Reports and Dashboards to Be Sent to Portal Users` option is enabled.

- Dashboard refresh notifications may not display properly in Outlook 2007.
- In HTML-formatted dashboard refresh notifications, users can click the name of the dashboard to log in to Salesforce and view the dashboard.
- To send a dashboard refresh notification to other users, store the dashboard in a public folder with access granted to others. Other users can't access dashboards in your personal folders. To add a dashboard to a public folder, edit the dashboard properties.
- Users can click components in a dashboard refresh notification to view the source report in Salesforce.
- Dashboard components that include Visualforce pages and s-controls may not display in dashboard refresh notifications. Users must view them in Salesforce.
- Dashboard refresh notifications can be viewed offline in email clients.
- If a dashboard has filters, only the unfiltered version is emailed.
- By default, Salesforce sends images in dashboard emails as `.png` (Portable Network Graphic) files, which are not supported in Lotus Notes. When you enable the `Use Images Compatible with Lotus Notes in Dashboard Emails` > option, Salesforce uses `.jpg` images, which Lotus Notes supports, when sending dashboard emails. The “Schedule Dashboard” permission is required to view this option.

 **Note:** Dashboard emails that contain images compatible with Lotus Notes are substantially larger and the image quality may 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](#)

Share a Dashboard

A dashboard is the best way to share a curated, interactive set of data with colleagues.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **All** editions except **Database.com**

 **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 and Dashboards between Folders	
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.

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. [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.

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:

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"

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**

- This folder is accessible by all users, including portal users
- This folder is accessible by all users, except for portal users


Access Level	Permissions to Access Read-Only Folders	Permissions to Access Read/Write Folders
Read	Any of the following: <ul style="list-style-type: none"> • "View Reports in Public Folders" • "Run Reports" • "Manage Public Reports" • "View All Data" 	Any of the following: <ul style="list-style-type: none"> • "Run Reports" • "Manage Public Reports" • "View All Data"
Write New	"Manage Public Reports"	"Create and Customize Reports"
Modify/Delete	"Manage Public Reports"	"Manage Public Reports"

Hidden Folders

The following permissions apply to folders that have this visibility setting:

- This folder is hidden from all users

Access Level	Permissions to Access Read-Only Folders	Permissions to Access Read/Write Folders
Read	Any of the following: <ul style="list-style-type: none"> • "View Reports in Public Folders" • "View All Data" 	"View All Data"
Write New	"Manage Public Reports"	"Manage Public Reports"
Modify/Delete	"Manage Public Reports"	"Manage Public Reports"

 **Note:** The Hidden Folders setting is available only in organizations without enhanced sharing for reports and dashboards. The equivalent for organizations with enhanced sharing for reports and dashboards is a folder with no sharing settings set, that is, a folder that's not explicitly shared with anyone.

Shared Folders

Access Level	Permissions to Access Read-Only Folders	Permissions to Access Read/Write Folders
Read	Any of the following: <ul style="list-style-type: none"> • "View Reports in Public Folders" • "Run Reports" (for shared users) • "Manage Public Reports" • "View All Data" 	Any of the following: <ul style="list-style-type: none"> • "Run Reports" (for shared users) • "Manage Public Reports" (for shared users) • "View All Data"
Write New	"Manage Public Reports"	"Create and Customize Reports" (for shared users)
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: <ul style="list-style-type: none"> • "Run Reports" • "Manage Dashboards" • "View All Data" 	Both of the following: <ul style="list-style-type: none"> • "Run Reports" • "Manage Dashboards"
Modify/Delete	All of the following: <ul style="list-style-type: none"> • "Run Reports" • "Manage Dashboards" • "View All Data" 	All of the following: <ul style="list-style-type: none"> • "Run Reports" • "Manage Dashboards" • "View All Data"

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**

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: <ul style="list-style-type: none"> • "Run Reports" • "View All Data" 	Both of the following: <ul style="list-style-type: none"> • "Run Reports" • "View All Data"
Write New	All of the following: <ul style="list-style-type: none"> • "Run Reports" • "Manage Dashboards" • "View All Data" 	All of the following: <ul style="list-style-type: none"> • "Run Reports" • "Manage Dashboards" • "View All Data"
Modify/Delete	All of the following: <ul style="list-style-type: none"> • "Run Reports" 	All of the following: <ul style="list-style-type: none"> • "Run Reports"

Access Level	Permissions to Access Read-Only Folders	Permissions to Access Read/Write Folders
	<ul style="list-style-type: none"> • "Manage Dashboards" • "View All Data" 	<ul style="list-style-type: none"> • "Manage Dashboards" • "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	<i>All of the following:</i> <ul style="list-style-type: none"> • "Run Reports" • "Manage Dashboards" • "View All Data" 	<i>Both of the following:</i> <ul style="list-style-type: none"> • "Run Reports" • "Manage Dashboards"
Modify/Delete	<i>All of the following:</i> <ul style="list-style-type: none"> • "Run Reports" • "Manage Dashboards" • "View All Data" 	<i>All of the following:</i> <ul style="list-style-type: none"> • "Run Reports" • "Manage Dashboards" • "View All Data"

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.

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.

Dashboard Limits

As you build dashboards of your Salesforce data, be aware of these limits, restrictions, and differences between dashboards in Lightning Experience and Salesforce Classic.

Salesforce Reports and Dashboards Limits

Feature	Personal Edition	Contact Manager Edition	Group Edition	Professional Edition	Enterprise Edition	Unlimited and Performance Edition	Developer Edition
Field Filters per Report ¹	20	20	20	20	20	20	20
Formulas per Report	5	5	5	5	5	5	5

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: **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"

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
Scheduled Reports (Emailed reports can be up to 10 MB.)	N/A	N/A	N/A	One per hour ² (off-peak hours only ³) (Limited to three preferred start times, which can't be changed)	One per hour ²	Two per hour ²	One per hour ² (off-peak hours only ³) (Limited to three preferred start times, which can't be changed)
Dashboard Filters	N/A	N/A	N/A	3 per dashboard	3 per dashboard	3 per dashboard	3 per dashboard
Dynamic Dashboards	N/A	N/A	N/A	N/A	Up to five per organization	Up to 10 per organization	Up to three per organization
Scheduled Dashboard Refreshes	N/A	N/A	N/A	N/A	One per hour ²	Two per hour ²	N/A
Custom Report Types (Limits apply to all custom report types regardless of development status.)	N/A	N/A	N/A	50	200	2,000	400
Reporting Snapshots	N/A	N/A	N/A	One per hour ² (off-peak hours only ³) (Limited to one preferred start time per day, which can't be changed)	One per hour ²	Two per hour ²	One per hour ² (off-peak hours only ³) (Limited to one preferred start time per day, which can't be changed)

¹ These limits apply to the report builder. If you're using the report wizard, the limit is 10.

² Up to a possible maximum of 200, total.

³ Off-peak hours are between 6 PM and 3 AM local time.

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.
- 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 organization 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.

Acceptable range for values: The maximum value allowed is 9999999999999999. The minimum value allowed is -9999999999999999.

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 organization 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 organization can request up to 200 dashboard refreshes per hour.

- Your organization 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 424 and [Refresh Dashboard Data](#) on page 425.

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](#)

 **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. [Use Dashboards on the iPad](#)
Salesforce Mobile Dashboards for the iPad, available from the Apple App Store, lets you access dashboards you recently viewed or are following when you're on the go.
3. [View Dashboard Lists](#)
The dashboard list contains all the dashboards you can view.

SEE ALSO:

[Filter Your Dashboard](#)

[Create a Salesforce Classic Dashboard](#)

[Get Started with Dashboards](#)

[Provide Individualized Views of a Dashboard](#)

EDITIONS

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. 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](#)

[Create a Salesforce Classic Dashboard](#)

[Refresh Dashboard Data](#)

Use Dashboards on the iPad

Salesforce Mobile Dashboards for the iPad, available from the Apple App Store, lets you access dashboards you recently viewed or are following when you're on the go.

 **Important:** As of Summer '15, the Mobile Dashboards for iPad app is no longer supported. You can continue to use the app, but Salesforce no longer provides support in the form of bug fixes or enhancements for any issues you may encounter. Talk to your Salesforce administrator about migrating to the Salesforce1 app, the new Salesforce mobile experience.

The mobile dashboards app keeps you in touch with your organization's data, performance, and trends, whether you're in the office or on the road. Use this app to:

- Browse recently viewed dashboards, or ones you're following
- Search for dashboards
- View individual dashboard components, highlight their values, and drill into reports for each
- Modify your report view

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"

EDITIONS

Available in: Salesforce Classic


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

- Email a dashboard or an individual component to others
- View, post, and comment on a dashboard's Chatter feed
- When offline, access some dashboards and reports you recently viewed in the app

 **Note:** The app doesn't support dynamic dashboards or dashboard filters.

You can download the app for free from the [Apple App Store](#) or from the [AppExchange](#). You can use the mobile dashboards app on all iPad models with iOS 5 or higher. The app is available in Salesforce Enterprise, Unlimited, Performance, and Developer Editions, and additionally in any organization that has enabled REST API.


If you can access your Salesforce organization from your iPad, you can use the mobile dashboards app. Open the app and log in using your Salesforce email and password. If you don't have a Salesforce account, you can still explore the app by tapping **Demo**.

 **Note:** Dashboard access for your iPad is available by default for your organization. If it isn't, an administrator must enable it before you can log in to the app.


In the Salesforce Classic Mobile client application, the Dashboards tab displays if your Salesforce administrator has added the tab to your mobile configuration.

View Dashboard Lists

The dashboard list contains all the dashboards you can view.

1. 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"

Let Users View Dashboards on the iPad

The Mobile Dashboards for iPad app is automatically enabled for your organization so your users can access the app without any configuration on your part.

Important: As of Summer '15, the Mobile Dashboards for iPad app is no longer supported. You can continue to use the app, but Salesforce no longer provides support in the form of bug fixes or enhancements for any issues you may encounter. Talk to your Salesforce administrator about migrating to the Salesforce1 app, the new Salesforce mobile experience.

You can disable the app if you don't want users accessing Salesforce data from mobile devices and you can easily re-enable it if you change your mind later.

To configure access to Mobile Dashboards for iPad:

1. From Setup, enter *Mobile Dashboards* in the Quick Find box, then select **Mobile Dashboard Settings**.
2. Select or deselect *Enable the Mobile Dashboards iPad app for all users*.
3. Click **Save**.

Users can download and install the Mobile Dashboards for iPad app from the [Apple App Store](#) or [AppExchange Mobile](#).

Aside from editions noted, the app is available to organizations enabled with REST API.

EDITIONS

Available in: Salesforce Classic

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

USER PERMISSIONS

To view Mobile Dashboards for iPad settings:

- "View Setup and Configuration"

To modify Mobile Dashboards for iPad settings:

- "Customize Application"

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	✓	✓
Getting Started with Wave Replication Tipsheet	✓	✓
Videos		
▶ Create the Sales Wave App, Part 1: Using the Configuration Wizard		✓
▶ Create the Sales Wave App, Part 2: Upload the Quota CSV File and Schedule a Dataflow		✓

Reports and Dashboards

	For End Users	For Admins
Guides and Tip Sheets		
Getting Started with Salesforce Reports and Dashboards	✓	
Analytics Workbook		✓
Reports Tab	✓	
Using the Drag-and-Drop Report Builder	✓	
Do your big reports take a long time to return data?	✓	
Report Formula Summary Functions	✓	
FAQ: Scheduling Reports	✓	
Tips for Creating Dashboards	✓	
One Dashboard For Many Viewers		✓
Taking Advantage of Dynamic Dashboards		✓

	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		✓
▶ Make Data-Driven Decisions with Reports and Dashboards		✓
▶ An Overview of Dashboards	✓	
▶ Getting Dashboards to Display the Right Data	✓	
▶ Analytics Folder Sharing	✓	
▶ Building Matrix Reports	✓	
▶ Using Cross Filters in Reports	✓	
▶ Getting Started with Buckets	✓	
▶ Getting Started with Report Builder	✓	
▶ Introducing Joined Reports in Salesforce	✓	
▶ Making Your Reports Run Faster	✓	
▶ Report Notifications	✓	
▶ Embedding Charts Anywhere	✓	

INDEX

A

- access, controlling [15](#)
- Accounts
 - reports [229](#)
- Activities
 - reports [230](#)
- activities reports [266](#)
- admin analytics app [199–200](#), [202](#)
- admin analytics dashboards [200](#)
- Admin Analytics Permissions [199](#)
- administrator [104](#)
- advanced filter in sfcdcDigest transformation [78](#)
- Age
 - cases [256–258](#)
- aggregation
 - definition [149](#)
- aggregation change [157](#)
- Analytics Cloud Builder permission set license [15](#), [17](#)
- Analytics Cloud Explorer permission set license [15](#), [17](#)
- Analytics Cloud Wave Platform license [4](#), [6](#), [15](#), [17](#)
- Analytics Cloud Wave Platform permission set license [7](#)
- Analytics Cloud Wave Platform permission sets [7–8](#)
- Analytics Cloud Wave Platform setup [9](#)
- Answers
 - reports [262](#)
- API limits [219](#)
- app [197–198](#)
- append transformation
 - description [36](#)
 - example [36](#)
 - JSON syntax [36](#)
 - parameters [38](#)
- AppExchange
 - Sample CRM Dashboards [417](#)
- Approval history reports
 - creating [263](#)
- Article Types
 - report [270](#)
- Articles
 - report [270](#), [272](#)
 - reporting [270](#)
- Assets
 - reports [255](#)
- augment transformation
 - description [38](#)

- augment transformation (*continued*)
 - example [38](#)
 - JSON syntax [38](#)

B

- binding [177](#)
- box widget [178](#)
- Builder permission set license [15](#), [17](#)

C

- calculating deltas [55](#)
- Call Center
 - reports [263](#)
- Campaigns
 - reports [233](#)
- Cases
 - reports [256–258](#)
- change chart scale [160](#)
- change chart type [156](#), [160](#), [163](#)
- change measures [157](#)
- chart [180–181](#)
- chart widget [175](#)
- Charts
 - chart formatting tab [353](#)
 - combination [350](#)
 - combination chart examples [351](#)
 - combining small groups [353](#)
 - customizing properties [346](#)
 - hover details [353](#)
 - See Reports [414](#)
- clipped lens [175](#)
- clone [165](#)
- clone visualization [165](#)
- Cloning
 - dashboards [438](#)
- column types
 - Date [21](#)
 - Numeric [21](#)
 - Text [21](#)
- Compare Table [182](#)
- computeExpression transformation description [45](#)
- computeExpression transformation example [45](#)
- computeExpression transformation JSON syntax [45](#)
- computeExpression transformation parameters [47](#)
- computeRelative transformation description [49](#)
- computeRelative transformation example [49](#)

- computeRelative transformation JSON syntax [49](#)
 - computeRelative transformation parameters [52](#)
 - configurator [216](#)
 - configuring [30](#)
 - Connect for Office
 - Excel integration [358](#)
 - installing [358](#)
 - Connected App [12](#)
 - considerations for datasets [116](#)
 - Contacts
 - contacts related to a task, viewing in reports [264](#)
 - contacts related to an event, viewing in reports [264](#)
 - reports [229](#)
 - Shared Activities, viewing in reports [264](#)
 - container widget [178](#)
 - create app [197–198](#)
 - create measures from dimensions [58](#)
 - creating [27](#), [102](#)
 - creating a dataset [28](#), [99](#)
 - creating datasets [106](#)
 - creating with dataflow [28](#)
 - creating with external data [106](#)
 - creating with the dataset builder [99](#)
 - cross-block custom summary formula
 - example [337](#)
 - Cross-block custom summary formula [341](#)
 - Custom Report Type
 - Salesforce Knowledge [271](#)
 - custom report type activities [266](#)
 - Custom Report Types
 - approval history [263](#)
 - forecasting [267](#)
 - forecasting [267](#)
 - process instance [263](#)
 - Custom summary formulas
 - joined reports [330](#)
 - reports [296](#)
 - Custom views
 - field filters [370–371](#)
 - Customizable forecasts
 - reports [238](#)
 - customizations [209](#)
 - Customizing
 - reports [287](#)
- D**
- dashboard [168–169](#), [171–172](#), [181–182](#), [185–186](#), [188](#)
 - Dashboard
 - overview [433](#)
 - Dashboard builder
 - editing filters [418](#)
 - filters overview [417](#), [420](#)
 - Dashboards
 - applying filters [419](#)
 - build [415](#)
 - classic [416](#)
 - cloning [438](#)
 - creating [416](#)
 - deleting [433](#)
 - drag and drop [403](#)
 - dynamic dashboards [421](#)
 - dynamic dashboards, setting up [423](#)
 - editing filters [418](#)
 - email [426](#)
 - enable mobile access [440](#)
 - FAQs: Mobile Dashboards for the iPad [438](#)
 - filters overview [417](#), [420](#)
 - move [403](#)
 - overview [437](#)
 - printing [432](#)
 - read [425](#)
 - refreshing data [425](#)
 - run [425](#)
 - running user, selecting [424](#)
 - Sample CRM Dashboards on AppExchange [417](#)
 - schedule refresh [426](#)
 - scheduled jobs queue [426](#)
 - searching [373](#)
 - share [427](#)
 - viewing [438](#)
 - viewing dashboard lists [439](#)
 - data [20](#), [217–218](#)
 - data exploration
 - concepts [149](#)
 - data integration
 - description [21](#)
 - external data [21](#)
 - Salesforce data [21](#)
 - data integration dataflow [26](#)
 - data monitor [33](#), [110](#)
 - data storage [219](#)
 - data storage limits [219](#)
 - dataflow [28](#), [30](#), [32](#), [34](#), [145](#)
 - dataflow definition file
 - sample with append [36](#)
 - sample with augment [38](#)
 - sample with filter [61](#)
 - sample with flatten [63](#)

- dataflow description [26](#)
- dataflow designing [29](#)
- dataflow jobs [33](#), [145](#), [219](#)
- dataflow jobs limits [219](#)
- dataset builder [99](#)
- dataset builder description [27](#), [102](#)
- Dataset definition [148](#)
- datasets
 - column types [21](#)
 - creating [21](#)
 - description [21](#)
 - filtering datasets [61](#)
- datasets delete [115](#)
- datasets description [23](#), [27](#), [102](#)
- datasets edit [112](#)
- datasets row-level security example [117](#), [121](#), [129](#)
- date columns [24](#)
- dates description [24](#)
- Deleting
 - dashboards [433](#)
 - reports [404](#)
- delta transformation [55](#), [57](#)
- delta transformation description [55](#)
- description [33](#), [35](#), [59](#), [67](#), [89](#), [92](#)
- designer [2](#), [168–169](#), [171–172](#), [182](#), [188](#)
- dim2mea transformation [58–59](#)
- dim2mea transformation description [58](#)
- dimension
 - definition [149](#)
- dimensions
 - description [21](#)
- Dynamic dashboards
 - setting up [423](#)

E

- edgemart transformation [59](#), [61](#)
- Email
 - dashboards [426](#)
 - report [388–389](#), [391–392](#)
- embed [188](#)
- empty keys [40](#)
- Enable
 - Mobile Dashboards for iPad app [440](#)
- Enabling [10–12](#)
- Error
 - obsolete report [410](#)
- Events
 - multiple contacts related to an event, viewing [264](#)
 - reports, viewing multiple contacts related to an event [264](#)

- example [58–59](#), [67](#), [89](#), [92](#)
- example JSON syntax [55](#)
- excel data [27](#), [102](#), [104](#)
- explorer [150–151](#), [164](#), [166](#), [193](#), [196](#)
- Explorer [148](#)
- Explorer data collections within home page [148](#)
- Explorer permission set license [15](#), [17](#)
- export as image [166](#)
- Exporting
 - reports [384–385](#)
- external data [27](#), [102](#), [104](#), [106](#), [109–110](#), [146](#), [219](#)
- External Data API [28](#)
- external data limits [219](#)
- external data row-level security example [117](#)
- external data upload [28](#)
- external data uploads [110](#)
- extracting data from datasets [59](#)

F

- faceting [177](#)
- FAQ
 - control access, reports [429](#)
 - error, obsolete report [410](#)
 - limits, reports [409](#)
 - report data [409](#)
 - reports, control access [429](#)
 - reports, limits [409](#)
 - reports, obsolete report error [410](#)
 - reports, old name in role hierarchy [411](#)
 - role hierarchy, old role name in report [411](#)
- FAQs
 - Mobile Dashboards for the iPad [438](#)
- field metadata overrides [79](#)
- field-level security [14](#)
- fields [23](#)
- fields overriding attributes [95](#)
- File [283](#)
- Files
 - reports [235](#)
- filter [154](#)
- filter expressions
 - syntax [62](#)
- filter transformation
 - description [61](#)
 - example [61](#)
 - filter expressions syntax [62](#)
 - JSON syntax [61](#)
 - parameters [62](#)
- filters [71–72](#), [78](#)

Index

- Filters
 - reports [362](#)
- flatten transformation
 - description [63](#)
 - example [63](#)
 - JSON syntax [63](#)
 - parameters [66](#)
- Flatten transformation [129](#)
- Flex [182](#)
- flex designer [181](#)
- Folder
 - analytics [377](#), [379–382](#)
 - dashboard [377](#), [379–382](#)
 - reports [377](#), [379–382](#)
 - sharing [377](#), [379–382](#)
- Folders
 - dashboards [431](#)
 - Dashboards [376](#)
 - permissions [429](#), [431](#)
 - reports [429](#)
 - Reports [376](#)
 - Sharing [376](#)
- Forecasts
 - custom report type [267](#)
 - reports [267](#)
- Formulas
 - joined reports [330](#)
 - PARENTGROUPVAL [297](#)
 - PREVGROUPVAL [297](#)
 - reports [296](#)
 - summary functions, overview [297](#)
 - summary functions, using [297](#)
- G**
 - generating metadata file [109](#)
 - get oriented [148](#)
 - get URL to share [166](#), [196](#)
 - getting started [17](#), [20](#), [199–200](#), [202–205](#), [208](#), [211–212](#), [215–218](#)
 - group by [152](#)
- H**
 - handling [24](#)
 - Hiding
 - report types [402](#)
- I**
 - Ideas
 - reports [269](#)
 - incorporate [188](#)
 - incremental extract [69](#)
 - integrate with Salesforce [209](#)
 - Integration User description [14](#)
 - integrations [209](#)
- J**
 - job management [145](#)
 - joined reports
 - examples [337](#)
 - Joined reports
 - cross-block custom summary formulas [330](#)
 - custom summary formulas [330](#)
 - JSON syntax [58–59](#), [67](#), [89](#), [92](#)
- K**
 - keyboard [185–186](#)
- L**
 - Leads
 - reports [239](#)
 - lens [151](#), [164–165](#)
 - lens definition [148](#)
 - licenses [4](#), [6](#), [15](#), [17](#)
 - Lightning Pages
 - standard components [191](#)
 - limitations [80](#)
 - limits [23](#)
 - limits API calls [219](#)
 - link widget [179](#)
 - links to Salesforce records [193](#)
 - List View
 - Customize [374](#)
 - Dashboards [374](#)
 - Reports [374](#)
 - Live Agent
 - reports [281](#)
 - sessions [281](#)
 - Live Agent session
 - reports [281](#)
- M**
 - matching records [40](#)
 - measure
 - definition [149](#)
 - measure change [157](#)
 - measures
 - description [21](#)
 - Mobile
 - FAQs: Mobile Dashboards for the iPad [438](#)

Mobile Dashboards for iPad

settings [440](#)

monitoring [33](#)

monitoring uploads [110](#)

Move

dashboards [403](#)

reports [403](#)

multiple matches [40](#)

N

null keys [40](#)

number widget [173](#)

numeric values [24](#)

O

object-level security [14](#)

objects overriding attributes [95](#)

operations [40](#)

Opportunities

creating custom reports [243](#)

creating reports [243](#)

custom reports [240](#)

Opportunities report [245](#)

Opportunities with Competitors report [247](#)

Opportunities with Contact Roles and Products report [247](#)

Opportunities with Contact Roles report [246](#)

Opportunities with Opportunity Teams and Products report [251](#)

Opportunities with Opportunity Teams report [250](#)

Opportunities with Partners report [251](#)

Opportunities with Products report [252](#)

Opportunities with Quotes and Quote Line Items report [254](#)

Opportunities with Quotes and Quote PDFs report [253](#)

Opportunity Field History report [249](#)

Opportunity History report [249](#)

Opportunity Trends report [253](#)

reports [240](#)

standard reports [240](#)

using reports [242](#)

using standard reports [242](#)

Overview

reports [414](#)

Salesforce Reports and Dashboards [223](#)

P

Page layout

report charts [395–400](#)

parameters

append transformation [38](#)

filter transformation [62](#)

flatten transformation [66](#)

Partners

reports [282](#)

permission set licenses for Wave Analytics [4, 15, 17](#)

permission sets for Wave Analytics [6](#)

predicate considerations [116](#)

predicates [116](#)

principal report type

changing [329](#)

Printing

reports [375](#)

process [148](#)

Process Instance Node object [263](#)

Process Instance object [263](#)

Products

reports [255](#)

Q

queries [219](#)

queries limits [219](#)

quick actions menu [193](#)

Quotes

Opportunities with Quotes and Quote Line Items report [254](#)

Opportunities with Quotes and Quote PDFs report [253](#)

R

range [24](#)

record ownership example [117](#)

registering [89](#)

Relationship groups

reports [282](#)

Report [283](#)

Report builder

adding and removing fields [292](#)

asynchronous loading [289](#)

changing currencies [292](#)

charting tips [349](#)

charts [346](#)

conditional highlighting [356](#)

customizing reports [226](#)

editing reports [226](#)

filter [362](#)

formulas [295](#)

grouping fields [288](#)

preview loading [289](#)

reordering columns [292](#)

- Report builder (*continued*)
 - sorting columns 292
 - summary fields 293
 - working with fields 292
- Report Builder
 - upgrading 361
- Report schedule
 - report 392
- Report type specific custom summary formula 341
- Report types 223
- Report Types
 - hiding 402
- Reporting
 - adding blocks 332
 - blocks 331, 334–335
 - custom summary formula tips 298
 - custom summary formulas and joined reports 341
 - deleting blocks 335
 - hide details 355
 - improving report performance 410
 - moving blocks 333
 - naming blocks 334
 - record count 335
 - renaming blocks 334
 - reordering blocks 333
 - show details 355
 - working with blocks 331
- Reporting snapshot
 - about 308
 - creating 310
 - mapping fields 313
 - running 314
 - scheduling 314
 - setting up 310
 - terminology 308
 - troubleshooting 317
- Reporting snapshots
 - defining 312
 - managing 316
- reports
 - control access 429
- Reports
 - about 414
 - adding and removing fields 292
 - adding report types 329
 - analytics 307
 - approval history 263
 - bucket fields 320
 - bucketing, numeric 322
- Reports (*continued*)
 - bucketing, picklist 323–324
 - bucketing, text 325, 327
 - build 225
 - Case 307
 - changing currencies 292
 - chart formatting tab 353
 - chart properties 346
 - charts 395–400
 - choosing a report format 284
 - combination chart examples 351
 - combination charts 350
 - conditional highlighting 356
 - creating 286
 - creating charts 346
 - creating cross filters 365
 - creating with the report wizard 287
 - cross filters 364, 368
 - cross filters example 366–367
 - custom options 412
 - customizing 287
 - customizing reports 226
 - data 409
 - deleting 300, 404
 - drag and drop 403
 - editing reports 226
 - email 388–389, 391–392
 - error, obsolete report 410
 - events, viewing multiple contacts related to 264
 - exception reporting 364, 368
 - exporting 384–385
 - exporting to the background 386–387
 - field filters 370–371
 - files 235
 - filter definitions 363
 - filter logic 363
 - filter types 363
 - filters 362
 - formulas 295–296
 - grouping 328, 336
 - grouping fields 288
 - historical 248, 300–308
 - joined format 329
 - joined report format 328, 336
 - joined reports 284
 - limiting results 357
 - limits 409
 - matrix reports 284
 - move 403

Reports (*continued*)

- Notifications [393](#)
- Opportunity [248](#), [300–306](#), [308](#)
- organize [401](#), [433](#)
- outer joins [364](#), [368](#)
- overview [224](#), [404](#), [414–415](#)
- PARENTGROUPVAL [297](#)
- partners [282](#)
- pinning folders [401](#)
- PREVGROUPVAL [297](#)
- principal report type [328–329](#), [336](#)
- printing [375](#)
- publicly shared [300](#)
- read [373](#)
- record page layout [395–400](#)
- reordering columns [292](#)
- report blocks [328](#), [336](#)
- report builder [290](#)
- role name, old [411](#)
- row limit example [368](#)
- run [373](#)
- Salesforce CRM Content [235](#)
- Salesforce Knowledge [270](#), [272](#)
- Salesforce Knowledge fields [273](#)
- saving [300](#)
- scheduled jobs queue [391–392](#)
- scheduling [388–389](#), [391–392](#)
- searching [373](#)
- selecting a report type [227](#)
- share [375](#)
- Shared Activities, viewing in [264](#)
- sorting columns [292](#)
- standard report types [343](#)
- subtotaling results [294](#)
- summary fields [293](#)
- summary functions, overview [297](#)
- summary functions, using [297](#)
- summary reports [284](#)
- tab [414](#)
- tabular reports [284](#)
- tabular reports in dashboards [357](#)
- tasks, viewing multiple contacts related to [264](#)
- trending [248](#), [300–308](#)
- troubleshoot [408](#)
- unscheduling [392](#)
- upgrading report builder [361](#)
- working with [412](#)
- working with fields [292](#)

- Reports schedule
 - delete [392](#)
 - report [392](#)
- Reports Tab
 - list view [374](#)
- reschedule [34](#)
- rescheduling [145](#)
- Role hierarchy
 - role name in report, old [411](#)
- role hierarchy example [129](#)
- row-level security
 - example, determining [118](#)
- row-level security on datasets [115](#)
- row-level security record ownership example [129](#)
- run-time monitoring [33](#)

S

- sales wave app [17](#), [20](#), [202–205](#), [208–209](#), [211–212](#), [215–218](#)
- Sales Wave Permissions [203](#)
- Salesforce CRM Content
 - reports [235](#)
- Salesforce data [14–15](#), [27](#), [102](#), [104](#)
- Salesforce data row-level security example [121](#), [129](#)
- Salesforce Knowledge
 - report [270](#), [272](#)
 - reporting [270](#), [273](#)
- Salesforce Knowledge Article Types
 - report [270](#)
- Salesforce Knowledge Articles
 - report [270](#), [272](#)
 - reporting [270](#)
- Salesforce Reports and Dashboards
 - Overview [223](#)
- sample with delta [55](#)
- sample with dim2mea [58](#)
- sample with edgemart [59](#)
- sample with sfdcDigest [67](#)
- sample with sfdcDigest incremental extract [69](#)
- sample with sfdcRegister [89](#)
- sample with update [92](#)
- SAQL editor [182](#)
- SAQL viewer [164](#)
- save [165](#)
- save visualization [165](#)
- Schedule
 - change [391](#)
 - dashboard refresh [426](#)
 - delete [391](#)
 - report [391–392](#)

Index

- Schedule (*continued*)
 - view [391](#)
 - Scheduled jobs queue
 - viewing [391–392](#), [426](#)
 - screenshot [166](#)
 - Search [283](#)
 - security [13](#)
 - security on datasets [115](#)
 - security on Salesforce data [14](#)
 - security policy examples [115](#)
 - Security User description [14](#)
 - Self-Service
 - reports [256](#)
 - set up Admin Analytics [199](#)
 - Setup process [104](#)
 - setup Sales Wave [203](#)
 - sfcdigest transformation [67](#), [69](#), [71](#), [79–80](#), [84](#)
 - sfcdigest transformation example [69](#)
 - sfcdregister transformation [89–90](#)
 - share [188](#)
 - share dashboard [166](#), [196](#)
 - share lens [166](#), [196](#)
 - Shared Activities
 - reports, viewing details in [264](#)
 - Sharing
 - dashboards [377](#), [379–382](#)
 - Dashboards [376](#)
 - folders [377](#), [379–382](#)
 - Folders [376](#)
 - reports [377](#), [379–382](#)
 - Reports [376](#)
 - SoftPhone calls
 - reports [263](#)
 - Solutions
 - reports [256–258](#)
 - sort order [156](#)
 - start [32](#)
 - starting [145](#)
 - stopping [145](#)
 - structured filter in sfcdigest transformation [72](#)
 - swap chart position [157](#)
- ## T
- Tabs
 - reports [414](#)
 - Tasks
 - multiple contacts related to a task, viewing [264](#)
 - reports, viewing multiple contacts related to a task [264](#)
 - team membership example [121](#)
 - Territories
 - reports [261](#)
 - transformations
 - append transformation [36](#)
 - augment transformation [38](#)
 - filter transformation [61](#)
 - flatten transformation [63](#)
 - troubleshooting [33](#), [109](#)
- ## U
- update transformation [92](#), [94](#)
 - use history [163](#)
 - use in row-level security [129](#)
 - user permissions list [6](#)
 - user profile [14](#)
 - Users
 - reports [232](#)
- ## V
- Videos [441](#)
 - view data [151](#), [164](#)
 - visualization [149](#)
- ## W
- Wave Analytics Trailhead [2](#)
 - Wave Analytics tutorial [2](#)
 - Wave Connector [104](#)
 - Wave dashboard keyboard shortcuts [186](#)
 - Wave for Communities [10](#)
 - Wave Mobile [12](#)
 - widget [180–181](#)