

Welcome to Tableau Desktop

This online help system from Tableau Software® contains the complete product documentation for using Tableau Desktop.

Get Started

Take a tutorial that tells a data story, or follow steps through opening Tableau, building and enhancing a view, and saving your work. This section includes:

Build-It-Yourself Exercises on page 57 – Create popular view types based on sample data sources that come with Tableau.

The Tableau Environment on page 125 – Learn what elements of the Tableau UI are called.

Tableau Concepts on page 262 – Dig into the ideas behind the terms and behaviors you see in Tableau.

Connect to and Prepare Data

Connect to a wide variety of data sources, including files, SQL databases, web data, and cube (multidimensional) databases. Then configure and optimize your data and learn how to keep it up-to-date. This section includes:

Connect to Your Data on page 346 – Connect to a variety of data sources, including Microsoft Office files, text files, and web data.

Set Up Data Sources on page 353 – Configure and optimize your data source for analysis.

Manage Data Sources on page 458 – Edit, refresh, replace, export, and upgrade data sources.

Design Views and Analyze Data

Discover the various features at your disposal as you build views, and learn the basic skills you need to create elegant, insightful views, dashboards, and stories. This section includes:

Building Data Views on page 472 – Drag and drop fields to create views of your data, then refine the view.

Work with Time on page 612 – Identify trends and make forecasts by working with date and time fields.

Filter Data from Your Views on page 672 – Display only the data you choose.

Build and Use Maps on page 702 – Assign geographic roles, build basic map views, explore data in maps, and more.

Advanced Analysis on page 822 – Create custom fields, use the built-in statistical tools, and more.

Optimize Workbook Performance on page 1817 – Speed up your workbooks.

Present Your work on page 1040 – Work with formatting, dashboards, and stories.

Publish Data Sources and Workbooks on page 1225 – Share your work with others using Tableau Online, Tableau Server, the Tableau mobile app, or on Tableau Public.

Save, Export, and Print on page 1266 – Make your work available to other applications.

Reference

Upgrade Tableau Desktop, consult a Quick Start, find out how to connect to a particular data source, look up Tableau functions, keyboard shortcuts, connectors, glossary definitions, and more.

Copyright Notices

Version: 10.1 | Updated on 1/11/2017

What's New in Tableau Desktop

Browse summaries of new features in version 10.1 or in previous versions (9.0 through 10.0).

What's new in Tableau Desktop 10.1

- [Changes to the Installer](#) below
- [Join tables using right and full outer join types](#) below
- [Union data using wildcard search](#) on the next page
- [Geocode airports](#) on the next page
- [Map data updates](#) on the next page
- [New JSON file connector](#) on page 5
- [New Anaplan connector](#) on page 5
- [New Oracle Eloqua connector](#) on page 5
- [Added initial SQL parameter support](#) on page 5
- [Teradata support for encryption](#) on page 5
- [IBM PDA support for SSL](#) on page 5
- [SAP HANA SQL query-based prompts support](#) on page 5
- [Google BigQuery feature enhancements](#) on page 5
- [SAP NetWeaver Business Warehouse feature enhancements](#) on page 6

Changes to the Installer

When you install Tableau Desktop, Tableau Reader or Tableau Public, you might want to perform a quiet installation. When you perform a quiet installation, you must accept the End User License Agreement (EULA) by setting the `ACCEPTEULA` option to "1".

For example: `tableau.exe /quiet ACCEPTEULA="1"`.

For more information, see [Performing a Quiet Installation of Tableau Desktop, Tableau Reader, or Tableau Public](#) in the Knowledge Center.

When you install a new version of Tableau Public, the installer will overwrite or remove any previous versions that you have on your computer. You can have only one instance of Tableau Public on your computer at one time.

Join tables using right and full outer join types

Use right and full outer join types when joining Excel tables to Excel tables, text file tables to text file tables, or tables in a cross-database join. **Note:** For cross-database joins, right and full

outer joins are supported only if the underlying database or file supports right and full outer join types.

Union data using wildcard search

In addition to unioning tables manually, you can now union tables automatically by specifying a wildcard search. For more information, see [Union Your Data on page 392](#).

Geocode airports

You can now geocode airports in Tableau using the International Air Transport Association (IATA) and International Civil Aviation Organization (ICAO) airport codes.

To assign the Airport geographic role to a field:

- In the Data pane, click the data type icon next to the field, select **Geographic Role**, and then select the **Airport** geographic role.

For more information, see [Types of geographic roles on page 737](#) and [Assign a geographic role to a field on page 736](#).

Map data updates

Netherlands municipalities are now automatically recognized by Tableau and can be geocoded with the County geographic role. For more information, see [County-equivalents recognized by Tableau on page 740](#).

Postcodes for South Africa and Thailand are now automatically recognized by Tableau and can be geocoded with the Zip Code/Post Code geographic role. For more information, see [ZIP codes and postcodes recognized by Tableau on page 741](#).

Postcode data has been updated in the following countries:

- Belgium
- Czech Republic
- Denmark
- France
- Germany
- Italy
- Netherlands
- Spain
- Slovakia
- Switzerland

- Turkey
- United States

New JSON file connector

Use the JSON file connector to connect to a local JSON file. For more information, see [JSON File](#) on page 1308.

New Anaplan connector

Starting in 10.1.2, use the Anaplan connector to connect to Anaplan data. For more information, see [Anaplan](#) on page 1340.

New Oracle Eloqua connector

Starting in 10.1.2, use the Oracle Eloqua connector to connect to Oracle Eloqua data. For more information, see [Oracle Eloqua](#) on page 1447.

Added initial SQL parameter support

Initial SQL parameter support is added for a number of connectors, including Amazon EMR, MapR Hadoop Hive, Cloudera Hadoop, HP Vertica, and Spark SQL. For more information about initial SQL, see [Run Initial SQL](#) on page 349.

IBM PDA support for SSL

IBM PDA (Netezza) supports connecting to an SSL server. For more information, see [IBM PDA \(Netezza\)](#) on page 1404.

Teradata support for encryption

You can now use encryption with your connections to Teradata. For more information, see [Teradata](#) on page 1504.

SAP HANA SQL query-based prompts support

You can now use your SQL query-based SAP HANA prompts in Tableau. For more information, see [SAP HANA](#) on page 1478.

Google BigQuery feature enhancements

Support has changed from BigQuery legacy SQL (BQL) to standard SQL. Your workbooks will upgrade when you open them in Tableau. With standard SQL comes other benefits for BigQuery connections including:

- Support for level of detail calculations.
- Improved speed of creating an extract.
- Faster validation of custom SQL.
- The ability to select a billing project for each connection.

SAP NetWeaver Business Warehouse feature enhancements

- Tableau now provides a default schema selection.
- Search functionality is added for selecting a catalog or InfoProvider, and a cube or query.

Added in version 10.0

- [Retain connection details](#) on the next page
- [New connectors](#) on the next page
- [Size marks on an axis](#) on the next page
- [New table calculation experience](#) on page 8
- [Display titles in worksheets](#) on page 8
- [Combine data with cross-database joins](#) on page 8
- [Highlight data points in context](#) on page 9
- [LOD expressions: Dimension expressions supported in dimensionality declaration](#) on page 9
- [Find clusters in data](#) on page 10
- [Oracle support for table functions and for Kerberos](#) on page 10
- [SQL Server support for contained databases](#) on page 11
- [Use JDBC to connect to SAP HANA on a Mac](#) on page 11
- [Create device layouts for a dashboard](#) on page 11
- [Format at the workbook level](#) on page 12
- [Create territories on a map](#) on page 13
- [Measure distance in metric or imperial \(U.S.\) units](#) on page 14
- [Map data updates](#) on page 14
- [Update to the default map server address](#) on page 14
- [Filter data across multiple primary data sources](#) on page 15
- [On-demand connections](#) on page 15

Retain connection details

After you connect to a database, Tableau 10.0 stores every connection detail that you enter in the Connection dialog box except the password. The next time you connect to that database, the only thing you have to enter is the password. These settings are stored per user, so shared computers do not share connection settings.

New connectors

New Google Sheets connector

Use the Google Sheets connector to connect to a Google Sheets data source. For more information, see [Google Sheets](#) on page 1382.

New Presto connector

Use the Presto connector to connect to Presto 141t from Teradata. For more information, see [Presto](#) on page 1461.

New MemSQL connector

Use the MemSQL connector to connect to MemSQL data. MemSQL uses a MySQL driver, so if you already have a MySQL driver installed, you can connect to MemSQL. For more information, see [MemSQL](#) on page 1422.

New QuickBooks Online connector

Use the QuickBooks Online connector to connect to QuickBooks Online data. For more information, see [QuickBooks Online](#) on page 1467.

Size marks on an axis

For views where the mark type is Bar and there are continuous fields on both **Rows** and **Columns**, Tableau 10.0 adds options for sizing the bar marks on the axis where the bars are anchored.

- The bar marks in histograms are continuous by default (with no spaces between the marks), and are sized to match the size of the bins. See [Build a Histogram](#) on page 89 for an example.
- When there is a field on **Size**, you can determine the width of the bar marks on the axis where the bars are anchored using the field on **Size**. To do this, click the **Size** card and choose **Fixed**.
- When there is no field on **Size**, you can specify the width of the bar marks on the axis

where the bars are anchored in axis units. To do this, click the **Size** card, choose **Fixed**, and then type a number in the **Width in axis units** field.

- When there is a continuous date field on the axis where the bars are anchored, the width of the marks is set to match the level of the date field. For example, if the level of the continuous date field is MONTH, the bars are exactly one month wide—that is, slightly wider for 31-day months than for 30-day months. You can configure the width of the bars by clicking the **Size** card, choosing **Fixed**, and then typing a number in the **Width in days** field, but the resulting bar widths don't take account of the varying lengths of time units, such as months and years.

New table calculation experience

Tableau 10.0 enhances the experience of creating table calculations:

- When you work in the Table Calculation dialog box to configure a calculation, the view updates in real time to show the results of your choices.
- Highlighting** shows you the scope and direction of your calculation by setting off part of the view with a colored background. As you change the **Compute Using** option in the Table Calculation dialog box, the highlighting reflects your changes.

State	Order Date / Segment								
	2011			2012			2013		
	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..
Alabama	\$2,184	\$6,139	\$6,139	\$2,418	\$3,892	\$3,892	\$2,668	\$7,646	\$7,651
Arizona	\$4,658	\$7,846	\$8,295	\$5,632	\$9,232	\$9,611	\$2,184	\$4,077	\$6,242
Arkansas	\$5,196	\$6,303	\$6,303	\$437	\$437	\$444	\$1,989	\$1,989	\$2,224
California	\$53,365	\$75,971	\$91,304	\$53,741	\$74,158	\$88,444	\$64,507	\$109,659	\$131,211

See [Transform Values with Table Calculations](#) on page 843 for details.

Display titles in worksheets

Titles are now turned on automatically in worksheets and appear as part of the view. You can create polished presentations using your worksheet without creating a separate dashboard. For more information, see [Titles](#) on page 234.

Combine data with cross-database joins

Combine data from different databases using a cross-database join. For example, you can combine related data from tables stored in an Excel workbook and a SQL Server database. For more information, see [Join Your Data](#) on page 354.

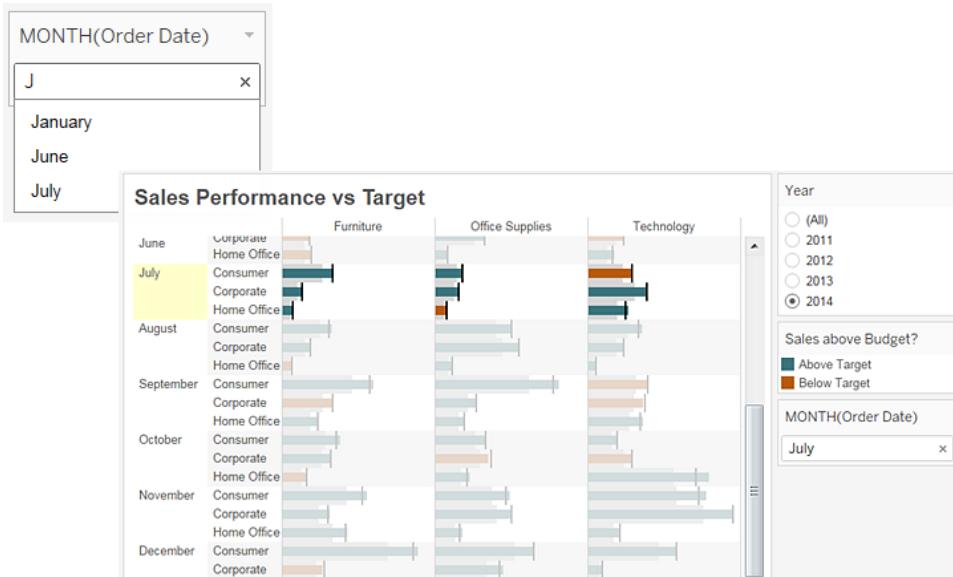
Highlight data points in context

The Highlighter is a new interactive exploration tool that you can use to quickly find and highlight specific data points while maintaining the context of the data. It is simple and powerful.

To use this new tool, do the following:

1. Select **Show Highlighter** from the context menu on a discrete field that is used in the view and that affects the level of detail in the view.
2. Search for the data you are interested in (a specific mark or group of marks) using a keyword search, or select a mark from the drop-down list.

The marks are highlighted in the view.



You can also perform ad hoc comparisons with instant highlighting when you hover over items in the drop-down list.

LOD expressions: Dimension expressions supported in dimensionality declaration

You can now use any expression that evaluates as a dimension, including Date expressions, in the dimensionality declaration for a level of detail expression.

This example excludes a discrete date bin in the view:

```
{EXCLUDE YEAR(Date) : SUM([Sales])}
```

This example excludes the YEAR(Date), which is a discrete date bin in the view:

```
{EXCLUDE DATEPART('year',Date) : SUM([Sales])}
```

This example excludes the YEAR(Date), which is a continuous date field in the view:

```
{EXCLUDE DATETRUNC('year', Date) : SUM([Sales])}
```

For information about level of detail expression syntax, see [Overview: Level of Detail Expressions](#) on page 899.

Find clusters in data

You can now find meaningful clusters in your data by dragging **Cluster** from the **Analytics** pane and dropping it in the view. The result is a statistical object that Tableau places on **Color**. If there is already a field on **Color**, Tableau moves that field to **Label**.



Cluster analysis partitions marks in the view into clusters, where the data values within each cluster are more similar to one another than they are to data values in other clusters.

You can drag a cluster from **Color** to the **Data** pane to make it a group dimension in which the individual members (Cluster 1, Cluster 2, etc.) contain the marks that the cluster algorithm has determined are more similar to each other than they are to other marks.

See [Find Clusters in Data](#) on page 920 for details.

Oracle support for table functions and for Kerberos

One way to simplify data access in Oracle is to create table functions, which return tables while hiding complex logic and exposing parameters. When you connect to Oracle, you can use table functions, which appear under **Stored Procedures** on the data source page.

Tableau supports Kerberos for Oracle. Enterprise environments that use Active Directory with Tableau can now have a seamless single sign-on experience with Oracle and Tableau.

For more information, see [Oracle](#) on page 1443.

SQL Server support for contained databases

Microsoft SQL Server and Azure SQL support a new feature called *contained databases*. When a database is *contained*, user information is stored in the database rather than in the database management system, which makes the database more portable. In Tableau 10.0, you can connect to contained databases by entering the database name in the connection dialog box. See [Microsoft SQL Server](#) on page 1430.

Use JDBC to connect to SAP HANA on a Mac

You can now connect to SAP HANA on a Mac using JDBC drivers. For more information, see [SAP HANA](#) on page 1478.

Create device layouts for a dashboard

After you've built a dashboard you can create layouts for it that are specific to particular devices.

When you publish the dashboard to Tableau Server or Tableau Online, people who interact with it experience a dashboard expressly designed for their screen display, whether it's a phone, tablet, or desktop. For more information, see [Create Dashboard Device Layouts](#) on page 1160.

Dashboard Layout Device Preview Device type ▲ Phone ▶

Layouts

- Default dashboard
- Desktop/Laptop
- Tablet
- Phone**

Hide Preview

Size - Phone

- Default
- Fit all
- Fit width

Height

Layout - Phone

Default **Custom**

The content and layout of the dashboard on all devices will match the Default dashboard.

Click "Custom" to change how the dashboard will look for the devices.

Chicago Bike Sharing

Divvy is a new bike sharing system featuring thousand stations throughout Chicago and Hyde Park, available 24/7, 365 days a year.

Date range: 7/9/2013 12:00:00 AM — 8/29/2013 11:59:59 PM

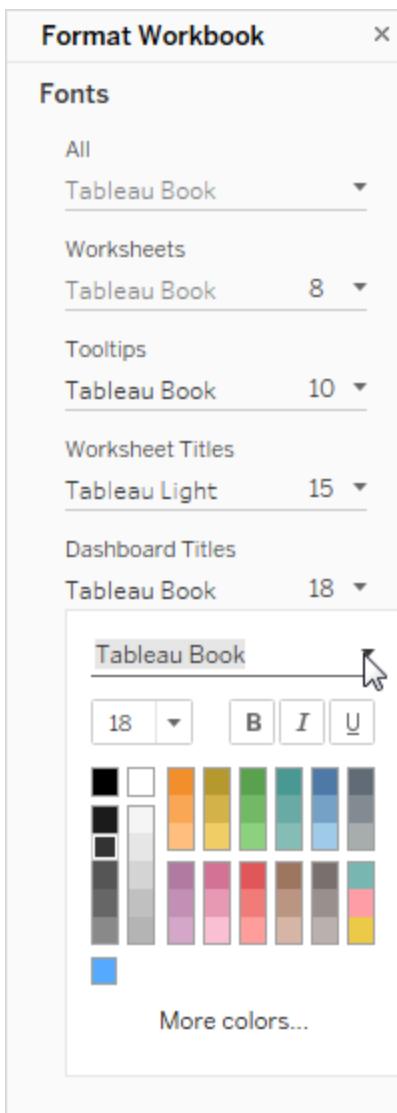
219,501 TRIPS 25.88 AVG RIDE (MIN) 76,805 MEMBERS

Station map



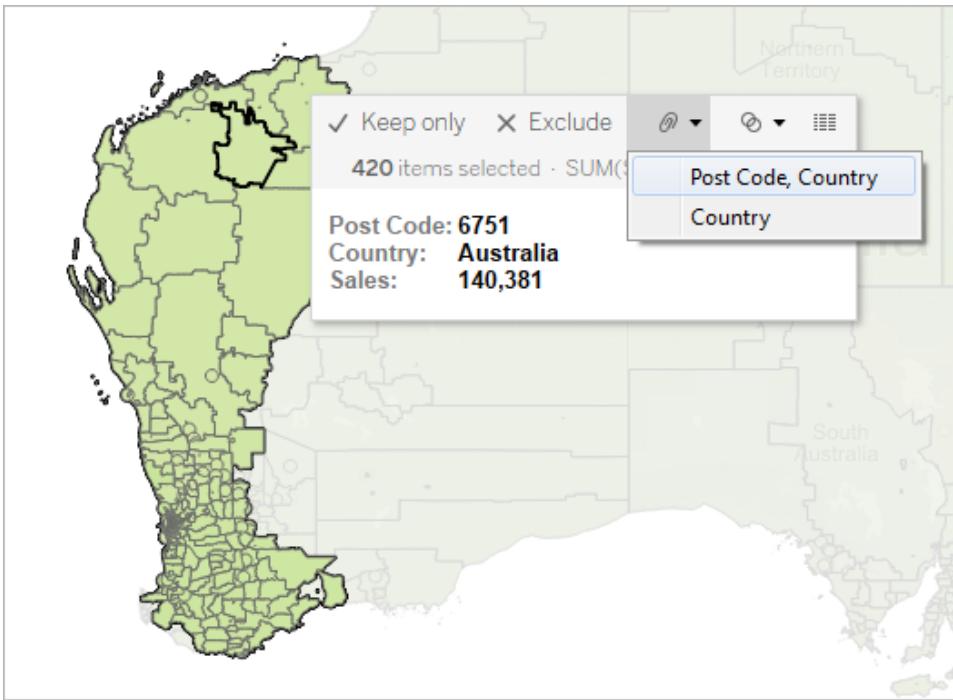
Format at the workbook level

You can quickly change how fonts and titles look in every view in a workbook by specifying format settings at the workbook level instead of at the worksheet level. See [Format at the Workbook Level on page 1059](#) for more information.



Create territories on a map

You can create territories on a map view by grouping existing locations (location points or polygons) together. For more information, see [Create Territories on a Map](#) on page 774.



Measure distance in metric or imperial (U.S.) units

You can now choose to measure in meters and kilometers or feet and miles when you measure distances on a map with the Radial tool. For more information, see [Change the units of measurement on page 798](#)

Map data updates

Hong Kong districts; Singapore URA subzones, South Korea Sigungu (Metropolitan Areas); and Taiwan cities and districts are now automatically recognized by Tableau and can be geocoded with the **County** geographic role. For more information, see [County-equivalents recognized by Tableau on page 740](#).

Postcodes for Argentina, Brazil, Costa Rica, India, Mexico, Singapore, and Taiwan are now automatically recognized by Tableau and can be geocoded with the **Zip Code/Post Code** geographic role. For more information, see [ZIP codes and postcodes recognized by Tableau on page 741](#).

Update to the default map server address

When configuring Tableau Desktop and Tableau Server proxy settings, use `maps.tableau.com` to connect to the default map server. For more information, see [Internet Access Requirements](#) in the Tableau Desktop Deployment Guide.

Filter data across multiple primary data sources

You can apply a filter across multiple primary data sources. For more information, see [Filter Data Across Multiple Data Sources](#) on page 689.

On-demand connections

When you load a workbook, Tableau connects only to the data sources that are required to display your data. This means that if there are multiple sheets in a workbook, Tableau only connects to the data sources for the current sheet so that you can view your data as soon as possible. For data sources that require authentication, now you only have to enter your credentials for the current sheet.

Added in version 9.3

- [Desktop installer improvements](#) below
- [Stay connected to Tableau Server or Tableau Online](#) on page 17
- [Forecasting improvements](#) on page 17
- [New data source support for Tableau functions and aggregations](#) on page 18
- [Add color to sheets in sorter view](#) on page 18
- [Workbook updates to use published data source](#) on page 18
- [Union your data](#) on page 19
- [Data grid enhancements](#) on page 19
- [Zooming improvements for maps](#) on page 19
- [WMS Support for Web Mercator](#) on page 19
- [New default tool for maps](#) on page 19
- [View tools remain active](#) on page 19
- [Totals are no longer included in color encoding](#) on page 20
- [New Snowflake data connector](#) on page 20
- [New Kognitio data connector](#) on page 20
- [New Cisco Information Server data connector](#) on page 21
- [Data preview for Web Data Connectors](#) on page 21
- [Kerberos support for PostgreSQL and Teradata](#) on page 21
- [OAuth support for Salesforce in Tableau Desktop](#) on page 21
- [Initial SQL support added to more data sources](#) on page 21
- [Initial SQL parameter support](#) on page 21

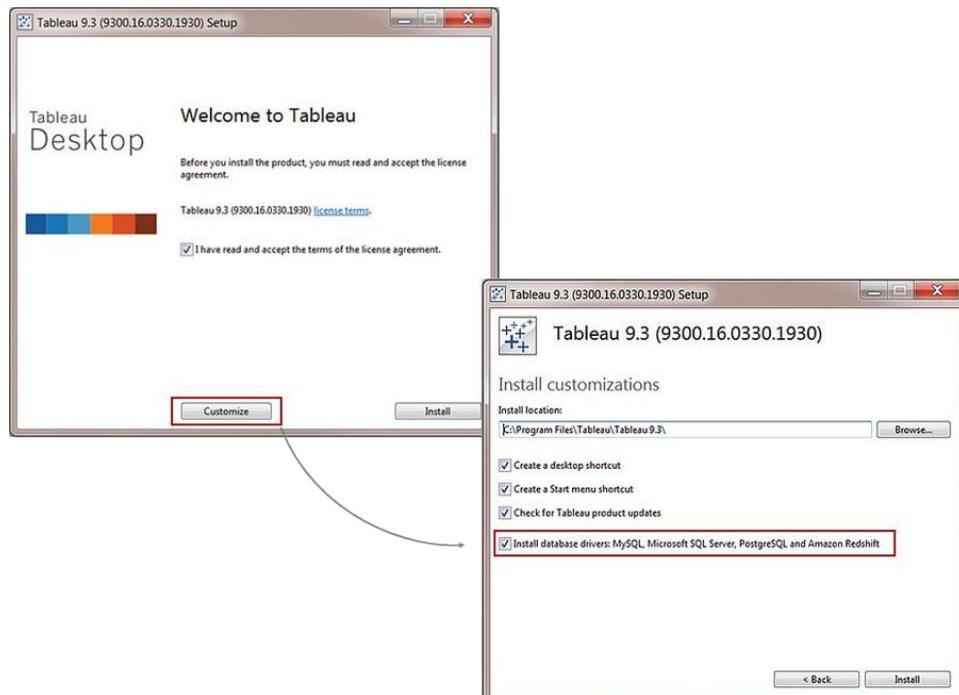
Desktop installer improvements

To streamline the install experience for Tableau Desktop for Windows, the four most common database drivers are now included as part of the setup process and are automatically installed by default. The following drivers are included:

- MySQL (version 5.3.6)
- Microsoft SQL Server (version 2008 R2 SP3)
- PostgreSQL (version 9.3.400)
- Amazon Redshift (version 1.2.1)

Note: Microsoft Visual C++ Redistributable is also automatically installed as a prerequisite for MySQL.

To view the list of drivers, or to clear the option to install these drivers, accept the license agreement on the installer page, and then click **Customize**.



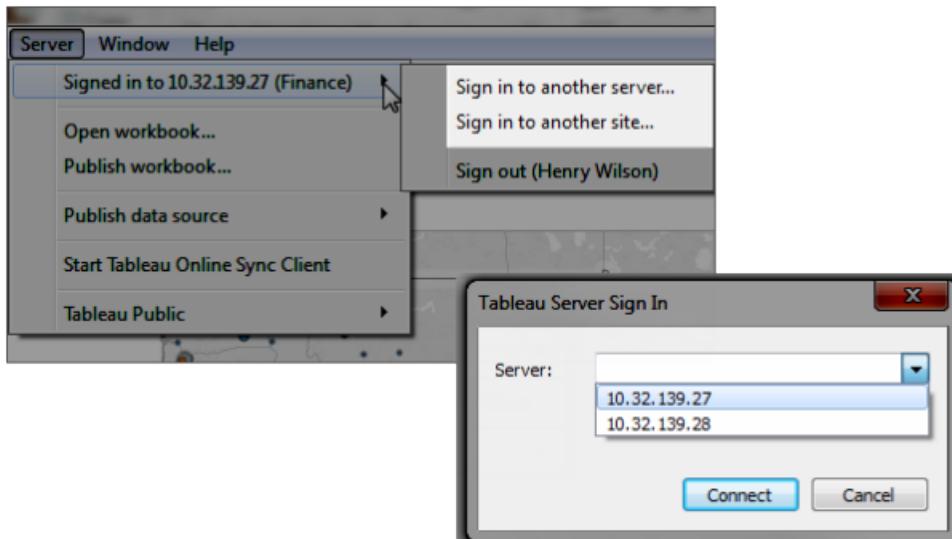
You can clear the option to install the drivers in either the user interface or from the command line.

If you want to perform a quiet install for Tableau Desktop 9.3, use the installer (.exe) file. The installer package (.msi) is included as part of the installer file and can no longer be extracted separately.

For more information about how to perform a quiet install of Tableau Desktop, see [Performing a Quiet Installation of Tableau desktop or Tableau Reader](#) in the Tableau Knowledge Base.

Stay connected to Tableau Server or Tableau Online

When you connect to Tableau Server or Tableau Online, Tableau Desktop saves each connection from session to session if you don't sign out. Next time you launch Tableau Desktop, you are signed in to your most recent server connection. You can also sign into different servers and easily switch between your available server and site connections.



For more information, see [Quick Start: Stay Connected with Automatic Sign-In on page 1811](#).

Forecasting improvements

Forecasting in Tableau Desktop has been improved in the following ways:

- You can now forecast values ordered by an integer dimension. Formerly it was only possible to forecast measures that are ordered by time. For more information, see [Forecasting When No Date is in the View on page 645](#).
- Automatic selection of the best season length for a forecast has been improved. For more information, see the **Seasonality** section in [How Forecasting Works in Tableau on page 636](#).
- Tableau forecasting will now skip null or missing values when estimating model parameters. However, the nulls remain in the series so the indexes of the values remain the same. For example, for a daily time series that is missing weekend days, the algorithm computes the model, potentially with parameters for a weekly cycle, but without using the missing values. But because the missing values are in the series, the cycle length is still seven days.

Due to this change:

- Forecasting now supports irregular time series without zero fill. So no error will result if you do not select **Fill in missing values with zeros** in the Forecast Options dialog box.
- The **Fill in missing values with zeros option** is no longer selected by default in the Forecast Options dialog box.

New data source support for Tableau functions and aggregations

For Tableau 9.3, more functions and aggregations are available for more data sources:

- The PERCENTILE function and aggregation are now available for Cloudera Hive, Hortonworks Hadoop Hive, and EXASolution data sources.
- The MEDIAN function and aggregation are now available for Cloudera Hive, Hortonworks Hadoop Hive, and EXASolution data sources.
- Regular expression functions (beginning with REGEXP_) are now available for HP Vertica, Pivotal Greenplum, and Teradata (version 14.1 and above) data sources. See [Additional Functions on page 1603](#). In addition, REGEXP_MATCH is now supported for Impala 2.3.0 and higher in Cloudera Hadoop data sources.
- The SPLIT function is now supported for Impala 2.3.0 and higher in Cloudera Hadoop data sources. Split functionality in the Data source pane is also now supported for Impala 2.3.0 and higher in Cloudera Hadoop data sources. See [Quick Start: Split a Field into Multiple Fields on page 1688](#).

Add color to sheets in sorter view

To make it easier to identify or group worksheets, you can now add color to the sheet tabs in the sheet sorter.

Workbook updates to use published data source

When you publish a data source to Tableau Server or Tableau Online, the workbook you're publishing it from is updated to connect to the published data source. At the same time the local data source is closed. To continue using the local data source, clear the **Update workbook to use the published data source** check box in the **Publish data source to Tableau Server** dialog box.

If you click **Undo** after publishing a data source, Tableau will revert to using the local data source, but it will not "un-publish" the data source.

Tableau does not replace the local data source when you publish a cube (multidimensional) data source.

For more information, see [Publish a Data Source on page 1234](#).

Union your data

In your Tableau data source, you can now append rows of data from one table to another table thereby creating unions. If columns in the union do not align correctly, you can merge the columns whose names don't match. For more information, see [Union Your Data on page 392](#).

Data grid enhancements

From the grid on the Data Source page, you can now do the following:

- See extract data and extract-only calculations, including data from the Web Data Connector
- Review data after applying extract filters and aggregating extracts
- Create groups and bins
- Join on pivot columns
- Join on merged columns
- See sheet and table name information for unioned tables

Zooming improvements for maps

Zooming in map views just got easier. You can now scroll to zoom in and out of a point on a map.

WMS Support for Web Mercator

Tableau now supports WMS servers that use Web Mercator. For more information, see [Supported Spatial Reference Systems on page 785](#) in the Use WMS Servers topic.

New default tool for maps

When you click and drag in a map view, you can now automatically select marks with the rectangular tool. This is a change in behavior to Tableau Desktop version 9.2, where clicking and dragging in a map view would automatically allow you to pan across the view instead.

View tools remain active

When you select a tool from the view toolbar (the pan tool, zoom area tool, radial tool, rectangular tool, or lasso tool), it remains active until you select to use another tool. The same is true if you use keyboard shortcuts to switch between tools. This is a change in behavior to Tableau Desktop version 9.2, where a tool would revert back to the default after one use.

The last tool you select will be saved with the workbook and available when you reopen it or publish it to the web.

Totals are no longer included in color encoding

Totals are no longer color encoded by default when you add them to the view.

To include totals in color encoding, click **Color** on the Marks card and then select **Edit Colors**. In the Edit Colors dialog box, select **Include Totals**.

Default

Sub-Category	Region			
	Central	East	South	West
Accessories	\$7,252	\$11,196	\$7,005	\$16,485
Appliances	-\$2,639	\$8,391	\$4,124	\$8,261
Art	\$1,195	\$1,900	\$1,059	\$2,374
Binders	-\$1,044	\$11,268	\$3,901	\$16,097
Bookcases	-\$1,998	-\$1,168	\$1,339	-\$1,647
Chairs	\$6,593	\$9,358	\$6,612	\$4,028
Copiers	\$15,609	\$17,023	\$3,659	\$19,327
Envelopes	\$1,778	\$1,812	\$1,465	\$1,909
Fasteners	\$237	\$264	\$174	\$275
Furnishings	-\$3,906	\$5,881	\$3,443	\$7,641
Labels	\$1,073	\$1,129	\$1,041	\$2,303
Machines	-\$1,486	\$6,929	-\$1,439	-\$619
Paper	\$6,972	\$9,015	\$5,947	\$12,119
Phones	\$12,323	\$12,315	\$10,767	\$9,111
Storage	\$1,970	\$8,389	\$2,274	\$8,645
Supplies	-\$662	-\$1,155	\$2	\$626
Tables	-\$3,560	-\$11,025	-\$4,623	-\$17,725
Grand Total	\$39,706	\$91,523	\$46,749	\$108,418

Totals included in color encoding

Sub-Category	Region			
	Central	East	South	West
Accessories	\$7,252	\$11,196	\$7,005	\$16,485
Appliances	-\$2,639	\$8,391	\$4,124	\$8,261
Art	\$1,195	\$1,900	\$1,059	\$2,374
Binders	-\$1,044	\$11,268	\$3,901	\$16,097
Bookcases	-\$1,998	-\$1,168	\$1,339	-\$1,647
Chairs	\$6,593	\$9,358	\$6,612	\$4,028
Copiers	\$15,609	\$17,023	\$3,659	\$19,327
Envelopes	\$1,778	\$1,812	\$1,465	\$1,909
Fasteners	\$237	\$264	\$174	\$275
Furnishings	-\$3,906	\$5,881	\$3,443	\$7,641
Labels	\$1,073	\$1,129	\$1,041	\$2,303
Machines	-\$1,486	\$6,929	-\$1,439	-\$619
Paper	\$6,972	\$9,015	\$5,947	\$12,119
Phones	\$12,323	\$12,315	\$10,767	\$9,111
Storage	\$1,970	\$8,389	\$2,274	\$8,645
Supplies	-\$662	-\$1,155	\$2	\$626
Tables	-\$3,560	-\$11,025	-\$4,623	-\$17,725
Grand Total	\$39,706	\$91,523	\$46,749	\$108,418

New Snowflake data connector

Use the Snowflake connector to connect to a Snowflake data warehouse. For more information, see [Snowflake on page 1497](#).

New Kognitio data connector

Starting in 9.3.1, use the Kognitio connector to connect to a Kognitio database. For more information, see [Kognitio on page 1407](#).

New Cisco Information Server data connector

Starting in 9.3.1, use the Cisco Information Server connector to connect to a Cisco Information Server virtual database. For more information, see [Cisco Information Server](#) on page 1349.

Data preview for Web Data Connectors

When you connect using a Web Data Connector, Tableau now opens on the Data Source page so that you can prepare your data (for example, change data types or hide columns) before you start your analysis.

Kerberos support for PostgreSQL and Teradata

Kerberos support has been added for PostgreSQL and Teradata connections. For more information, see [PostgreSQL](#) on page 1458 or [Teradata](#) on page 1504. For information about configuring Tableau Server for Kerberos, see [Kerberos](#) in the Tableau Server Help.

OAuth support for Salesforce in Tableau Desktop

You can now use OAuth when you connect to Salesforce in Tableau Desktop. After you provide your Salesforce credentials and allow Tableau access to the data, Salesforce.com creates an OAuth access token that is used to connect to the data. Instead of having to embed your credentials in data sources or workbooks, you can use the access token.

Initial SQL support added to more data sources

Oracle, Pivotal Greenplum, and Microsoft SQL Server now support initial SQL statements.

Initial SQL parameter support

For data source connections that support initial SQL, you can now perform parameter substitution for a set of useful parameters, such as ServerUser and WorkbookName. For more information, see [Run Initial SQL](#) on page 349.

Added in Version 9.2

- [Google Analytics query enhancements](#) on the next page
- [Recent colors save to the application](#) on the next page
- [Data grid enhancements](#) on the next page
- [Data Interpreter enhancements](#) on the next page
- [Data pane enhancements](#) on page 23
- [Match mark label color to mark color](#) on page 24
- [Label most recent marks](#) on page 25
- [Fix one or both ends of an axis](#) on page 25

- [Move totals to the top or left of the view](#) on page 26
- [New Map Options dialog box](#) on page 27
- [Support for Mapbox maps](#) on page 27
- [SAP HANA column labels](#) on page 27
- [Query improvements](#) on page 28
- [Show Quick Filter has changed to Show Filter](#) on page 28

Google Analytics query enhancements

By default, Tableau returns all data in a Google Analytics query to avoid returning sampled data. See [Quick Start: Query All Data from Google Analytics](#) on page 1704.



Recent colors save to the application

From any Color drop-down control, you can select a recently used color. Up to eight recently used colors save to the application and are available in any workbook.



Data grid enhancements

The following functionality is now available from the grid on the Data Source page:

- Sort columns and rows
- Double-click the name of the column to rename the field
- Reset the name of a single column or select multiple columns to reset the column names at once
- Create calculations based on existing fields in the data source
- Copy values in the data source using Ctrl+C (Command-C on a Mac).

For more information, see [Quick Start: Data Grid Enhancements](#) on page 1664.

Data Interpreter enhancements

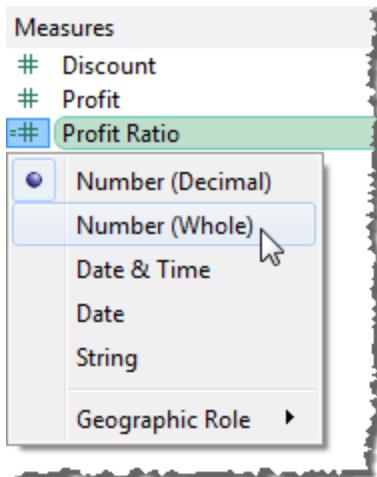
The Data Interpreter now detects subtables in your Excel data. For more information, see [Excel](#) on page 1294.



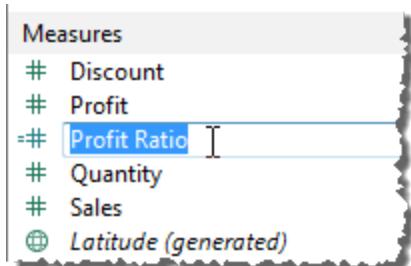
Data pane enhancements

The following functionality is now available in the Data pane:

- Click the icon to the left of a field in the Data pane to change the field's data type or geographic role:



- Click a field and hold down the mouse button to rename the field:



- Enhanced Search options in the Data pane. Click the search icon at the top of the Data pane or press Ctrl + F (Command-F on a Mac) to initiate a search. As you type in the

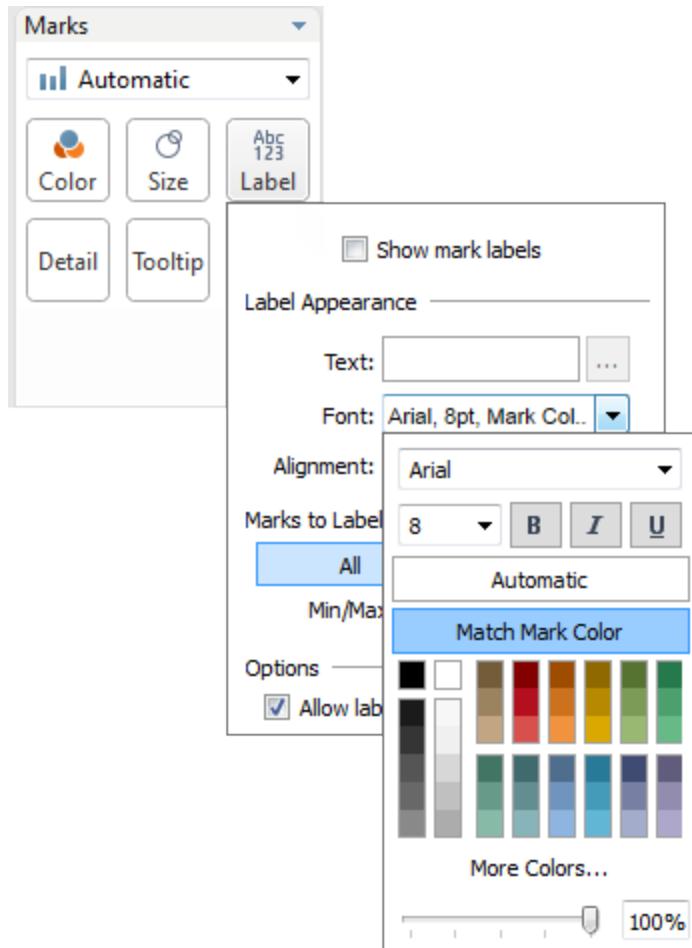
search box, search now filters the contents of the Data pane to show all fields that contain the typed string. Search remains open until you click the search icon or press Ctrl + F again.

- The zones apportioned for Dimensions, Measures, Parameters, and Sets in the Data pane automatically size so that scroll bars are not shown unless there are more fields than can be displayed in the available space.

See [Data Types](#) on page 262, [Rename Fields](#) on page 310, and [Find Fields](#) on page 309.

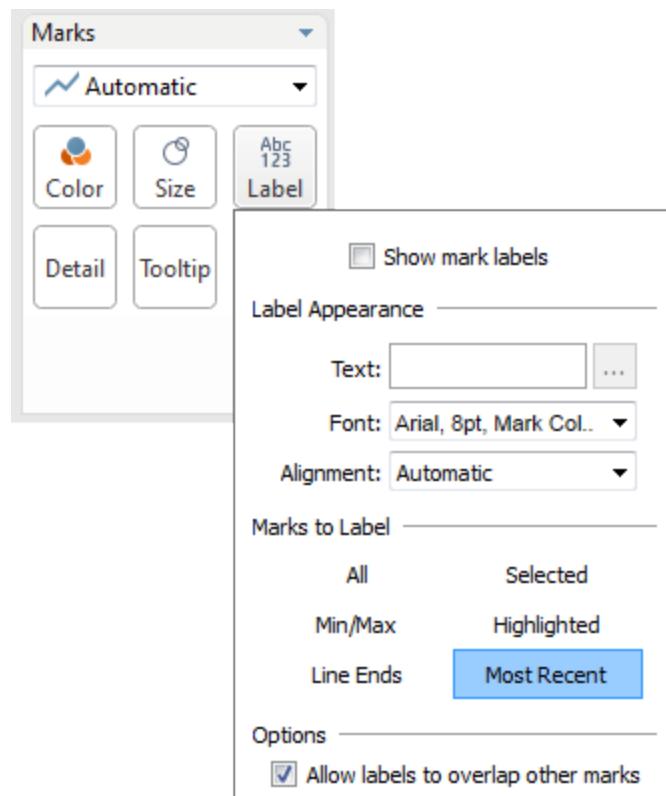
Match mark label color to mark color

You can select to closely match the color of each label to the color of its mark. See [Select label color](#) on page 560 in the Format Mark Labels topic.



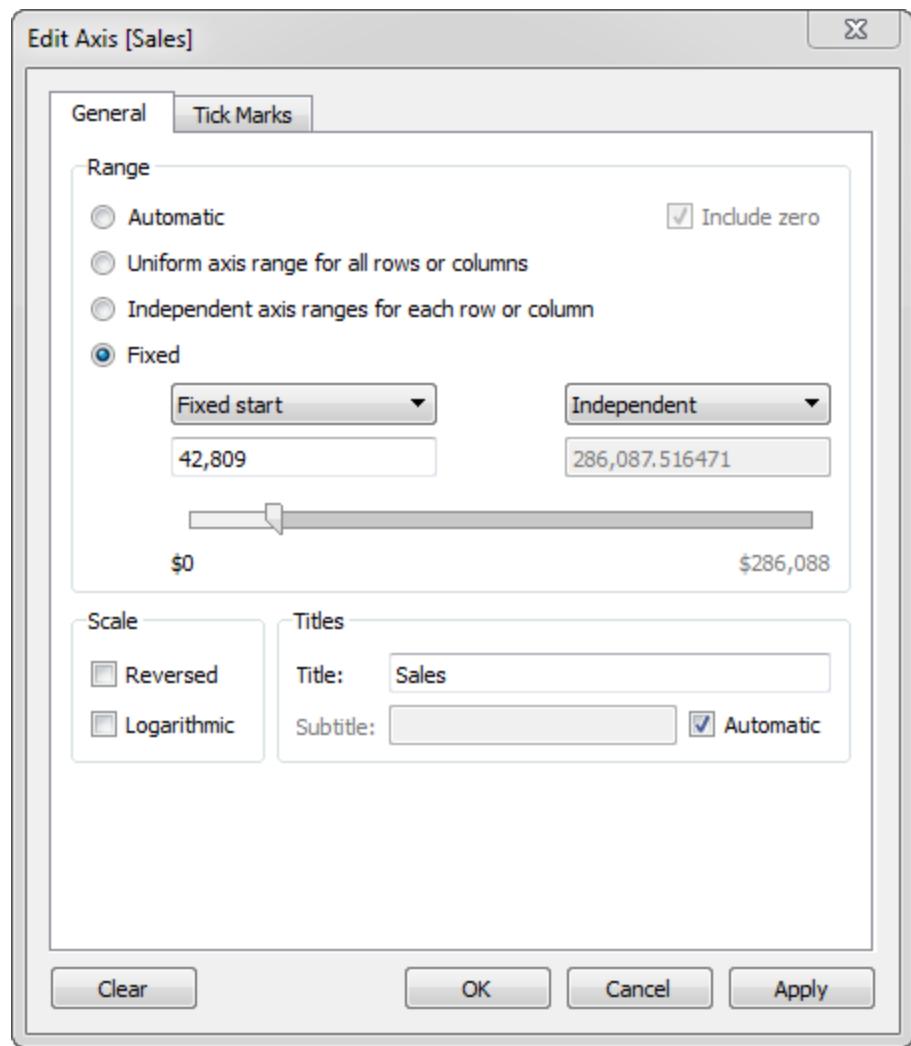
Label most recent marks

When there is a date field in the view, you can select to label the most recent marks in the view. Labels are based on the level of detail of fields in the view. See [Show and Hide Mark Labels](#) on page 549.



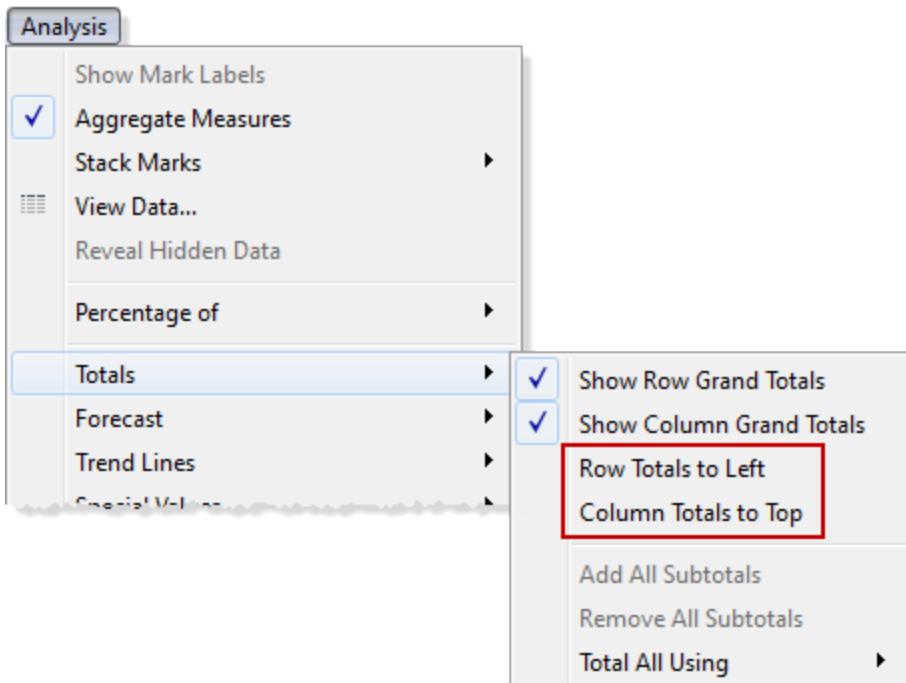
Fix one or both ends of an axis

You can specify the values for the start or end of an axis. See [Edit Axes](#) on page 1047 in the Edit Axes topic.



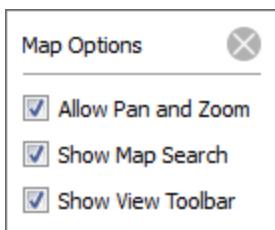
Move totals to the top or left of the view

You can select to show totals at the top or left of the view. See [Move Totals](#) on page 988.



New Map Options dialog box

Use the new Map Options dialog box to customize how your audience interacts with your map view. Use the Map Layers pane (previously the Map Options pane) to customize the appearance of your map. See [Customize How People Interact with your Map](#) on page 781.



Support for Mapbox maps

You can now connect to Mapbox maps in Tableau. See [Use Mapbox Maps](#) on page 788.

SAP HANA column labels

By default, the SAP HANA column names now display the column label, rather than the column name.

Query improvements

The following query improvements have been added for Tableau 9.2.

- **Filtering on categorical aggregates**

You can now create a filter that aggregates a measure and use it as a domain filter.

- **Numerical bins and combined fields in level of detail expressions**

You can now refer to a numerical bin or a combined field in a level of detail expression.

- **MIN and MAX supported for boolean data type**

These aggregations can now be used with boolean fields.

- **Additional optimizations**

Tableau can now detect “always false” predicates and avoid issuing queries for them.

For known domains such as boolean, we short circuit to return {true, false} or {true, false, null}. Other rewrites include folding case expressions with constant results to IN SET or NOT IN set.

Show Quick Filter has changed to Show Filter

In Tableau 9.2, the option to Show Quick Filter for a field has changed to Show Filter. For more information, see [Filter Data from Your Views on page 672](#)

Added in Version 9.1

The following new features and enhancements have been added to Tableau Desktop for version 9.1.

- [Product Updates below](#)
- [Single Sign-On for SAP HANA on the next page](#)
- [Web Data Connector on the next page](#)
- [SAP Prompts on the next page](#)
- [Bin Size Optimization on the next page](#)
- [Auto-complete Availability on the next page](#)
- [Confidence Intervals for Reference Lines on the next page](#)
- [Measure Distance with the Radial Tool on the next page](#)
- [Turn Off Pan and Zoom on page 30](#)

Product Updates

To ensure that you always have the most up-to-date features, security resolutions, and corrected issues, Tableau Desktop includes a product update feature. See [Turn Product Updates Off or On on page 1815](#).

Single Sign-On for SAP HANA

When SAP HANA is configured to support single sign-on (SSO), after you sign in to the SAP HANA server, you can access data, and publish data sources and workbooks to Tableau Server, without having to re-enter your user name and password. See [Quick Start: SAP HANA Single Sign-On](#) on page 1684. **Note:** Tableau Desktop requires SAP HANA driver version 1.00.85 and later to support SSO for SAP HANA.

Web Data Connector

You can use a web data connector to connect to data that is accessible over HTTP and that doesn't already have a connector. You can create your own web data connector or use one that has been created by someone else. See [Web Data Connector](#) on page 1513.

SAP Prompts

With SAP HANA and SAP Netweaver Business Warehouse, you can prompt for a variable when a workbook is opened. See [SAP HANA](#) on page 1478 or [SAP NetWeaver Business Warehouse](#) on page 1485.

Bin Size Optimization

When you create discrete bins from a continuous measure, Tableau can compute an optimal bin size. See [Create Bins from a Continuous Measure](#) on page 597.

Auto-complete Availability

You can now take advantage of auto-completion for formulas in additional locations where you can create formulas—for example, when you add conditions or limits to filters, or when you create a set based on a condition. See (for example) [Filter Data from Your Views](#) on page 672.

Confidence Intervals for Reference Lines

You can now add and configure confidence intervals (bands) for a reference line. See [Adding Reference Lines](#) on page 953. You can no longer add confidence intervals for reference distributions.

Measure Distance with the Radial Tool

Use the Radial tool to measure approximate distances between your data and the locations or landmarks in a map view. For more information see, [Measure Distances Between Data Points and Locations in a Map](#) on page 794.

Turn Off Pan and Zoom

You can turn off pan and zoom to control how your audience interacts with your map view or background image. For more information see, [Customize How People Interact with your Map](#) on page 781

Added in Version 9.0

The following new features and enhancements have been added to Tableau Desktop for version 9.0.

- [New start page experience below](#)
- [Pivot below](#)
- [Split fields into multiple fields below](#)
- [Metadata area on the next page](#)
- [Data Interpreter on the next page](#)
- [New data sources and enhancements to existing data sources on the next page](#)
- [Analytics pane on page 32](#)
- [Recalculated lines on page 33](#)
- [New calculation features on page 33](#)
- [Story point formatting on page 34](#)
- [Advanced selection tools on page 34](#)
- [Map search on page 34](#)
- [Configurable tooltip behavior on page 34](#)

New start page experience

The start page in Tableau Desktop is a central location from which you can connect to your data, access most recently used workbooks, and explore content produced by the Tableau community. For more information, see [Start Page](#) on page 125.

Pivot

Pivot your Microsoft Excel and text file data sources from crosstab format to columnar format. For more information, see [Pivot Data from Columns to Rows](#) on page 423.

Split fields into multiple fields

Split existing string fields into new fields. For more information, see [Split a Field into Multiple Fields](#) on page 429.

SPLIT is also available as a Tableau function, for use in calculated fields. For more information, see [String Functions](#) on page 1560.

Metadata area

Perform bulk management tasks, like hiding multiple fields at once, using the metadata area. For more information, see [Data Source Page](#) on page 130.

Data Interpreter

Detect and remove unique formatting and extraneous information in your Excel data sources. For more information, see [Excel](#) on page 1294.

New data sources and enhancements to existing data sources

New: Amazon Aurora

Use the Amazon Aurora connector to connect to an Amazon Aurora data source. For more information, see [Amazon Aurora](#) on page 1329.

New: Spark on Azure HDInsight

Use the Spark SQL connector to connect to a Spark on Azure HDInsight data source. For more information, see [Spark SQL](#) on page 1501.

New: Microsoft Azure SQL Database

Connect to Microsoft Azure SQL Database data using the Microsoft SQL Server connector. For more information, see [Microsoft SQL Server](#) on page 1430

New: Google Cloud SQL

Use the Google Cloud SQL connector to connect to a Google Cloud SQL database instance. For more information, see [Google Cloud SQL](#) on page 1379.

New: Microsoft Azure SQL Data Warehouse

Connect to Microsoft Azure SQL Data Warehouse data using the Microsoft SQL Server connector. For more information, see [Microsoft SQL Server](#) on page 1430.

New: Amazon EMR

Use the Amazon EMR connector to connect to an Amazon Elastic MapReduce (EMR) database. For more information, see [Amazon EMR](#) on page 1332.

New: Spark SQL

Use the Spark SQL connector to connect to a Spark SQL cluster (requires Apache Spark 1.2.1 or later). For more information, see [Spark SQL on page 1501](#).

New: Statistical File

Use the Statistical File connector to connect to SAS (*.sas7bdat), SPSS (*.sav), and R (*.rdata, *.rda) data files. For more information, see [Statistical File on page 1314](#).

Enhanced: MapR Hadoop Hive

MapR Hadoop Hive now supports connectivity for the Mac.

Enhanced: MySQL

SSL encryption is supported. Mac drivers are now included in the TableauDrivers.dmg file. For more information, see [MySQL on page 1438](#).

Enhanced: Salesforce

The Salesforce connector now supports more flexible joins. You can create joins on any string fields in addition to fields that are constrained references between tables. For more information, see [Salesforce on page 1474](#).

Enhanced: SAP NetWeaver Business Warehouse (BW)

The SAP NetWeaver Business Warehouse connector has been updated to include support for variables. For more information, see [SAP NetWeaver Business Warehouse on page 1485](#).

Tableau Data Extracts

Tableau Data Extracts are no longer listed on the Connect pane. Use Other Files to connect to a Tableau Data Extract. For more information, see [Other Files on page 1319](#).

Windows Azure Marketplace

To connect to Windows Azure Marketplace, use the OData connector. Workbooks created in earlier versions of Tableau that used the Windows Azure Marketplace DataMarket connector will work as expected. For more information, see [OData on page 1441](#).

Analytics pane

The Analytics pane, on the left side of the Tableau workspace, provides quick and easy access to common analytic objects in Tableau. You can drag reference lines, forecasts, trend lines, and other objects into your view from the Analytics pane.

For more information, see [Analytics Pane](#) on page 135.

Recalculated lines

Recalculated lines help you gain insight into how a subset of your data compares to the overall data in your view. For more information, see [Marks and Data Analysis](#) on page 562.

New calculation features

The following features were added to Tableau Desktop to enhance the experience of creating and using calculated fields.

Non-modal calculation editor

The calculation editor has been redesigned to provide interactive editing, intelligent formula-completion, and drag-and-drop support. The calculation editor is now also available when you're editing a view in Tableau Server or Tableau Online.

For more information, see [Create or Edit a Calculated Field](#) on page 822.

Ad-hoc calculations

Ad-hoc calculations are calculations that you can create and update on a shelf in the view. Ad-hoc calculations can be useful for testing a hunch, trying a what-if scenario, or debugging a complex calculation.

For more information, see [Ad-Hoc Calculations](#) on page 831.

Auto-completion for formulas

As you type a formula, either in the calculation editor or in an ad-hoc calculation, Tableau displays a list of options for completing the formula.

For more information, see [Auto-Completion for Formulas](#) on page 829.

Level of detail expressions

Level of detail expressions support aggregation using dimensions other than those in the view. With level of detail expressions, you can attach one or more dimensions to any aggregate expression.

For more information, see [Level of Detail Expressions](#) on page 899.

Functions for regular expressions

Four new functions have been added to the calculation language to support regular expressions.

For more information, see functions beginning with REGEXP in [Additional Functions](#) on page 1603.

Functions for hexagonal binning

Hexagonal binning is a technique for clustering data in a two-dimensional plane. Two new functions have been added to the calculation language to support hexagonal binning. These functions support binning data in a single dimension using histograms, or by a geographic level of detail in a map.

For more information, see functions beginning with HEXBIN in [Number Functions](#) on page 1552.

Story point formatting

You can now re-size the captions in your story, change the shading and font of the navigator, and select to fit a dashboard to the exact size of a story in your workbook. For more information, see [Format a Story](#) on page 1209.

Advanced selection tools

Use the Radial, Rectangular, and Lasso tools on the view toolbar to select multiple marks in the view. For more information, see [View Toolbar](#) on page 565.

Map search

Use map search to find locations on a map view so you can quickly explore and inspect data. For more information, see [Search for Locations in Your Map](#) on page 793.

Configurable tooltip behavior

Show tooltips instantly as you move the mouse over the marks in a view or configure tooltips to display only after resting the mouse on a mark. For more information, see [Tooltip Properties](#) on page 218.

Get Started

This section shows you how to get started using Tableau Desktop and walks you through some exercises that introduce such concepts as connecting to data, creating a view, developing that view, and then saving it.

It also provides a set of build-it-yourself exercises to familiarize you with some of the basic chart types you can create with Tableau, and a set of conceptual topics that may help you better understand how Tableau works.

Tutorial: Get Started with Tableau Desktop

Tableau Desktop makes exploring your data and sharing your insights easy. If you're new to it, though, it can be difficult to get started.

If this is true for you, check out the [Get Started with Tableau Desktop](#) tutorial.

The Get Started with Tableau Desktop tutorial walks you through the features and functions of Tableau Desktop. You'll learn how to connect to data; build, present, and share some useful views; and apply key features along the way.

Open Tableau and Connect to Data

In this topic you'll start Tableau Desktop and connect to a Microsoft Excel sample data source that comes with Tableau, which is called **Sample - Superstore**.

Open Tableau

On a Windows computer, open Tableau by selecting **All Programs > Tableau 10.1** on the Windows Start menu, or by double-clicking the desktop shortcut.

On a Mac computer, open Tableau by selecting the Tableau icon in the Applications folder in Finder, by typing **Tableau** into Spotlight, or by clicking the Dock shortcut.

Tableau opens to *the start page*.

The start page contains sample workbooks and data sources. After you've used Tableau to connect to data or create a view, your recently used workbooks and saved data connections appear on the start page as well.

You can always return to the start page after you have started working in the data source page, or in a workbook or worksheet, by clicking the Tableau logo in the upper-left corner.

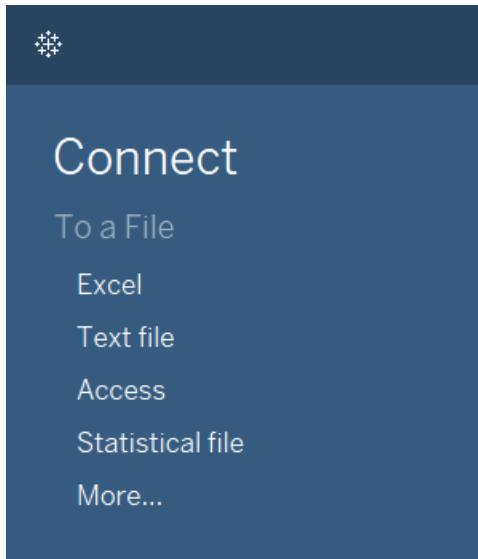
Connect to Data

The first step in Tableau Desktop is to connect to the data you want to explore.

This example shows how to connect to Microsoft Excel data from the start page.

1. On the start page, under **Connect**, click **Excel**.

The **Connect** pane lists the different types of data you can connect to:



2. In the Open dialog box, navigate to the Sample – Superstore Excel file on your computer at /My Documents/My Tableau Repository/Datasources and open it.

After you have connected to the Excel data, the *data source page* shows the sheets in your data. You drag one or more sheets to the *canvas*. You can combine sheets on the canvas to form the data source from which you will build your view.

The screenshot shows the Tableau Data Source page for a connection named 'Orders (Sample - ...)'. The connection type is set to 'Live'. The interface includes a sidebar with 'Connections' (Sample - Superstore Excel), 'Sheets' (Orders, People, Returns), and 'New Union'. The main area displays a preview grid of data with columns: Row ID, Order ID, Order Date, Ship Date, Ship Mode, and Customer ID. The grid shows 8 rows of sample data. At the bottom, there are buttons for 'Data Source' and 'Sheet 1'.

#	Abc	Abc	Abc	Abc	Abc
Orders	Orders	Orders	Orders	Orders	Orders
Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID
1	CA-2013-1521...	11/9/2014	11/12/2014	Second Class	CG-12520
2	CA-2013-1521...	11/9/2014	11/12/2014	Second Class	CG-12520
3	CA-2013-1386...	6/13/2014	6/17/2014	Second Class	DV-13045
4	US-2012-1089...	10/11/2013	10/18/2013	Standard Class	SO-20335
5	US-2012-1089...	10/11/2013	10/18/2013	Standard Class	SO-20335
6	CA-2011-1158...	6/9/2012	6/14/2012	Standard Class	BH-11710
7	CA-2011-1158...	6/9/2012	6/14/2012	Standard Class	BH-11710
8	CA-2011-1158...	6/9/2012	6/14/2012	Standard Class	BH-11710

3. At the top of the data source page, select how you want to connect to the data. You can select from the following options:
 - **Live** – Creates a direct connection to your data. The speed of your data source will determine performance.
 - **Extract** – By default, this option imports the entire data source into Tableau fast data engine as an extract. The extract is saved with the workbook. If you prefer to import a subset of the data, click the **Edit** link. This option requires you specify what data you want to extract using filters.
4. At the bottom of the data source page, preview the data in your data source in the *grid*. You can make the following changes in the grid:
 - Hide or rename a column by clicking the column header drop-down arrow.
 - Change the data type by clicking the data type icon in the column header.
5. Click the sheet tab to go to the new worksheet and begin your analysis.

Create a Scatter Plot with Show Me

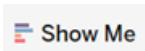
In this example you use **Show Me** to build a scatter plot to show sales versus profit for each product and customer.

1. Select **Customer Name** and **Product Name** (from the **Dimensions** area in the **Data** pane) and **Profit** and **Sales** from the **Measures** area. To select multiple fields, hold down the Ctrl key (or the command key on a Mac) as you click fields.

The screenshot shows the Tableau Data pane with the following structure:

- Data** tab selected.
- Dimensions** section:
 - Customer
 - Customer Name (highlighted with a blue rounded rectangle)
 - Segment
 - Order
 - Order Date
 - Order ID
 - Ship Date
 - Ship Mode
 - Location
 - Product
 - Category
 - Sub-Category
 - Manufacturer
 - Product Name (highlighted with a blue rounded rectangle)
 - Profit (bin)
 - Region
 - Measure Names
- Measures** section:
 - Discount
 - Profit (highlighted with a green rounded rectangle)
 - Profit Ratio
 - Quantity
 - Sales (highlighted with a green rounded rectangle)
 - Latitude (generated)
 - Longitude (generated)

2. Click **Show Me** on the toolbar.



3. Select the scatter plot chart type from **Show Me**.

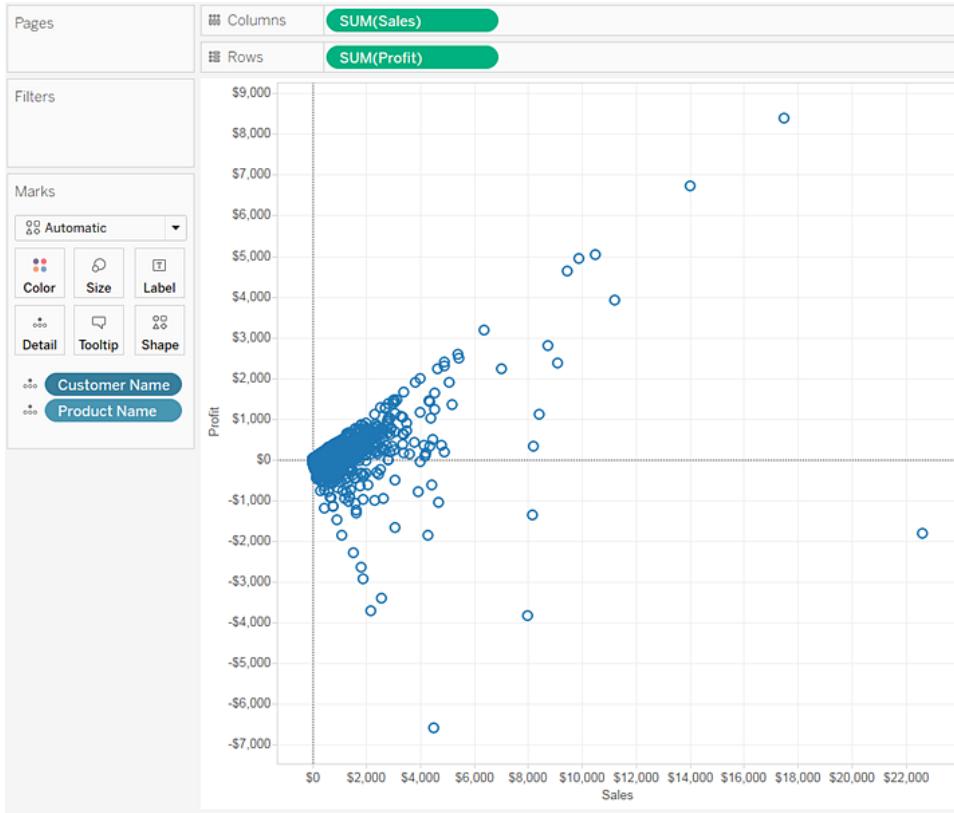


For **scatter plots** try

0 or more **Dimensions**

2 to 4 **Measures**

Tableau creates a scatter plot using the fields you selected. You can drag fields to further refine the view.



Create a Line Chart with Show Me

In this example, you use **Show Me** to create a line chart that displays profit as a function of time.

1. Select **[Order Date]** and **Profit** in the **Data** pane. Hold down the Ctrl key (or the command key on a Mac) as you select the fields.

The screenshot shows the Tableau Data pane with the following structure:

- Data** (selected) / Analytics
- Sample - Superstore
- Dimensions**
 - Customer
 - Customer Name
 - Segment
 - Order
 - Order Date (highlighted)
 - Order ID
 - Ship Date
 - Ship Mode
 - Location
 - Country
 - State
 - City
 - Postal Code
 - Product
 - Category
- Measures**
 - Discount
 - Profit (highlighted)
 - Profit Ratio
 - Quantity
 - Sales
 - Latitude (generated)
 - Longitude (generated)
- Sets
- Parameters
- Profit Bin Size

2. Click **Show Me** on the toolbar to display the **Show Me** card, if it isn't already displayed.



3. In the **Show Me** pane, see what type of view Tableau suggests.



Because a date dimension and a measure are selected, Tableau suggests you build a line view (as indicated by the check mark). A line is often the optimal way to look at how measure values change over time.

The view below shows the resulting chart. Each point on the line represents the sum of profit for the corresponding year.



You can more easily see the points by clicking the **Mark Labels** button on the toolbar.



Save Your Work

After you have created a set of views, you should save the results in a Tableau Workbook. You can open the workbook later to continue working on your views, or share your workbook with other users, either by sending them the file or publishing the workbook to the web. Follow the steps below to save your workbook.

1. Select **File > Save** or press **Ctrl + S** on your keyboard.
On a Mac, select **File > Save** or press **Command-S** on your keyboard.
2. Browse to a file location to save the workbook.
By default, Tableau saves workbooks in the **Workbooks** folder in **My Tableau Repository**.
3. Specify a file name for the workbook.
4. Specify a file type. You can select from the following options:
 - Tableau Workbook (.twb)** – Saves the all the sheets and their connection information in a workbook file. The data is not included.
 - Tableau Packaged Workbook (.twbx)** – Saves all the sheets, their connection information and any local resources (e.g., local file data sources, background images,

custom geocoding, etc.).

5. When finished, click **Save**.

Step-by-Step: Explore Data with Tableau

You are now invited to follow along through a series of topics to see how a view can evolve in Tableau through the process of exploration. First you create a basic view that shows year-to-year profit. Then you diversify the view to include more data, filter the view to drill into the most important data, and finally add color to make the results stand out.

In the worksheet, the columns from your data source are shown as fields on the left side in the **Data** pane. The **Data** pane contains a variety of different kinds of fields; for now the most important ones are *dimensions* and *measures*.

The screenshot shows the Tableau Data pane interface. At the top, there are tabs for 'Data' (which is selected) and 'Analytics'. Below the tabs, there's a section for 'Orders (Sample - Superstore)' with a dropdown arrow. The main area is divided into two sections: 'Dimensions' and 'Measures'. The 'Dimensions' section lists the following fields: Category, City, Country, Customer ID, Customer Name, Order Date, Order ID, Postal Code, and Product ID. The 'Measures' section lists the following fields: Discount, Profit, Quantity, and Sales. There are also icons for search and refresh, and a vertical scroll bar on the right side of the pane.

Dimensions typically hold categorical data such as product types and dates, while measures hold numeric data such as sales and profit. (If you want to know more, see [Dimensions and Measures on page 268](#).)

When you build a view, you add fields from the **Data** pane. You can do this in different ways. For example:

- You can drag fields from the **Data** pane and drop them onto the cards and shelves that are part of every Tableau worksheet.

- You can double-click one or more fields in the **Data** pane.
- You can select one or more fields in the **Data** pane and then choose a chart type from **Show Me**, which identifies the chart types that are appropriate for the fields you selected. For details, see [Show Me on page 595](#).
- You can drop a field on the **Drop field here** grid, to start creating a view from a tabular perspective.



Proceed to [Step 1 - Create a Basic View below](#) to create a view to show year-by-year profit.

Step 1 - Create a Basic View

Here you build a basic view that shows year-by-year profit in the form of a line chart.

1. From the **Dimensions** area in the **Data** pane, drag the **Order Date** field to the **Columns** shelf.

When you drag a field over a shelf, a plus sign indicates that the shelf can accept the field.

The screenshot shows the Tableau Data pane. On the left, under Dimensions, there are four categories: Customer, Order, Location, and Product. Under Order, the fields Customer Name, Segment, Order Date, Order ID, Ship Date, Ship Mode, Country, State, City, Postal Code, Category, Sub-Category, Manufacturer, and Product Name are listed. A cursor is hovering over the 'Order Date' field. In the top right, the 'Columns' shelf has 'Order Date' selected. Below it, the 'Rows' shelf is empty. The 'Marks' shelf shows options for Color, Size, Text, Detail, and Tooltip, with 'Text' being the current mark type.

The resulting table has four columns and one row. Each column header represents a member of the Order Date field (2011, 2012, 2013, and 2014). Each cell contains an “Abc” label, which indicates that the current mark type for this view is text.

Columns	YEAR(Order Date)		
ROWS			
Order Date			
2011	2012	2013	2014
Abc	Abc	Abc	Abc

Notice that the field is colored blue, which indicates that it is *discrete*. Also, the field name changed to **YEAR(Order Date)** because year is the default date level for this field.

The default date level is determined by the highest level that contains more than one distinct value (for example, multiple years and multiple months). That means that if **[Order Date]** contained data for only one year but had multiple months, the default level

would be month. You can change the date level using the field menu.

The screenshot shows the Power BI interface with the 'Data' pane open. A context menu is displayed for the field 'YEAR(Order Date...)'. The menu items are:

- Filter...
- Show filter
- Sort...
- Format...
- Show header
- Include in tooltip
- Show missing values

A red box highlights the following section of the menu:

Year	2015
Quarter	Q2
Month	May
Day	8
More	▶

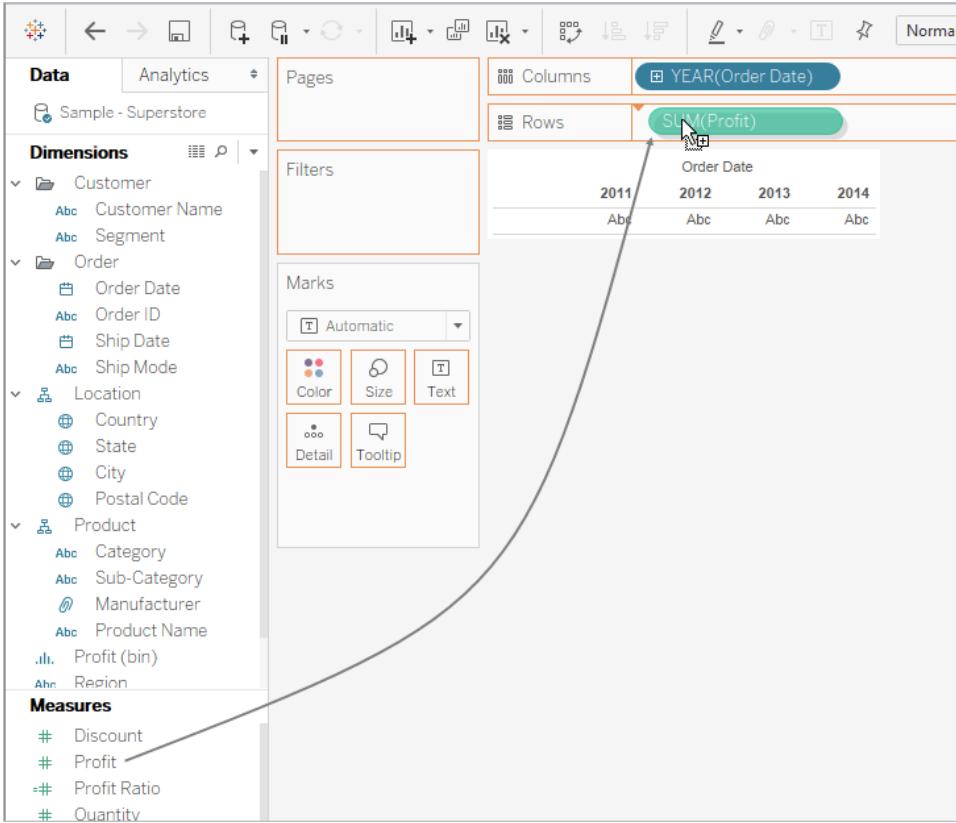
Below this section, another red box highlights the following section:

Year	2015
Quarter	Q2 2015
Month	May 2015
Week Number	Week 5, 2015
Day	May 8, 2015
More	▶

At the bottom of the menu, there are two more items:

- Exact Date
- Attribute

2. From **Measures** in the **Data** pane, drag the **Profit** field to the **Rows** shelf.

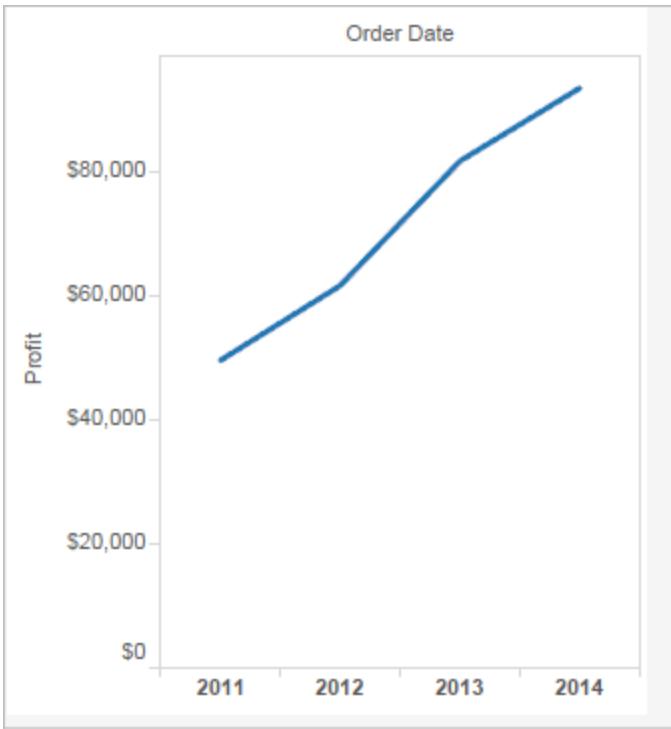


The **Profit** field is colored green on the **Rows** shelf, indicating that it is continuous. Also, the field name changes to **SUM(Profit)**. That's because measures are automatically aggregated when you add them to the view, and the default aggregation for this measure is SUM. For more information on what happens when you add a measure to a view—and why—see [Dimensions and Measures on page 268](#).

Tableau transforms the table into a line chart and creates a vertical axis (along the left side) for the measure.

A line chart is a great way to compare data over time and identify trends effectively.

This line chart shows profit over time. Each point along the line shows the sum of profit for the corresponding year.



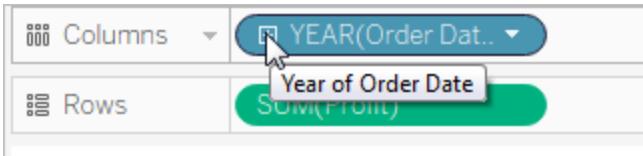
The next step is to drill a little deeper into the time dimension. See [Step 2 - Create a Nested Table View](#) below.

Step 2 - Create a Nested Table View

In this example you modify the view from [Step 1 - Create a Basic View](#) on page 46 to show quarters in addition to years.

You can show **[Order Date]** by quarters using either of the following methods:

- Click the plus button on the left side of the field **YEAR(Order Date)** field on **Columns**.



- Drag the **Order Date** field (again) from the **Data** pane and drop it on the **Columns** shelf to the right of **Year(Order Date)**.

The new dimension divides the view into separate panes for each year. Each pane, in turn, has columns for quarters. This view is called a nested table because it displays multiple headers, with quarters nested within years. The word "headers" might be a bit misleading because while

the year headers remain at the "head" of the view (that is, the top), the quarter headers are at the foot (that is, the bottom). So it might be more accurate to call them *footers*.



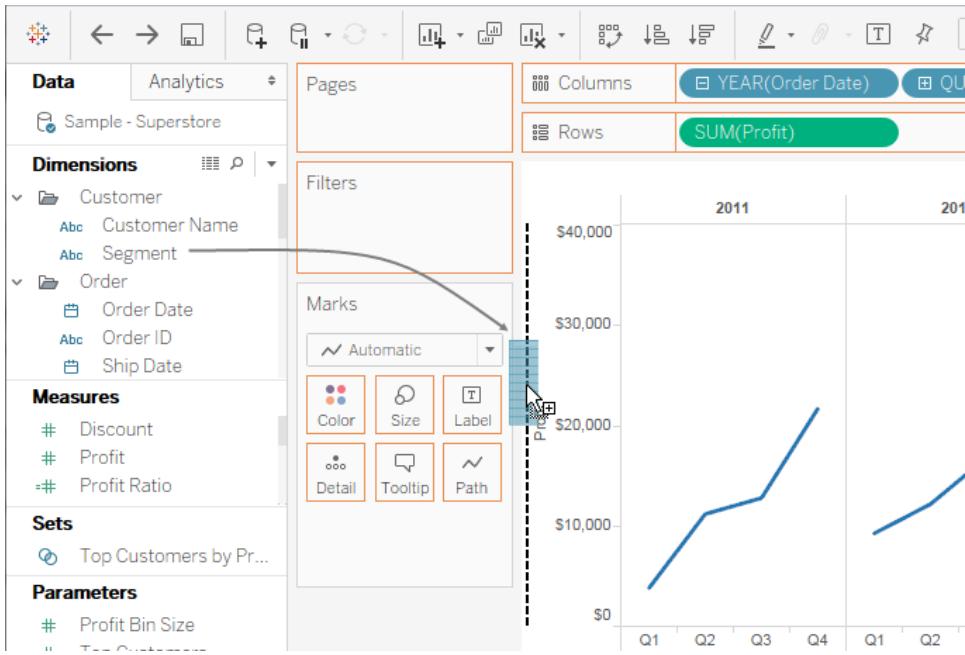
Continue to refine the view in [Step 3 - Create a Small Multiples View](#) below.

Step 3 - Create a Small Multiples View

In this step you modify the view from [Step 2 - Create a Nested Table View](#) on the previous page to add customer segment.

Drag the **Segment** dimension from the **Data** pane and drop it just to the left of the Profit axis in the view.

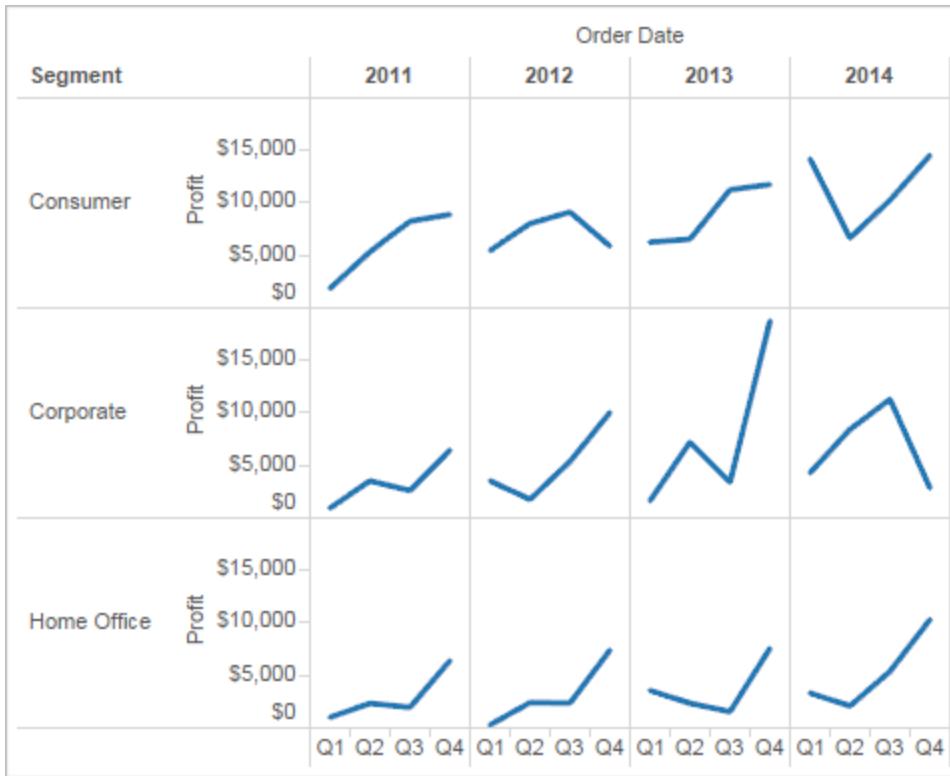
The field is added to the **Rows** shelf and row headers are created. Each header represents a member of the **Segment** field.



You could achieve the same result by dropping **Segment** on the **Rows** shelf, to the left of **SUM(Profit)**. Tableau typically supports multiple ways to drag-and-drop fields.

Note: Tableau does not allow you to place a dimension to the right of a measure on either the Rows or Columns shelves.

The new dimension divides the view into 12 panes, one for each combination of year and segment. This view is a more complex example of a nested table. Any view that contains this sort of grid of individual charts is referred to as a small multiples view.

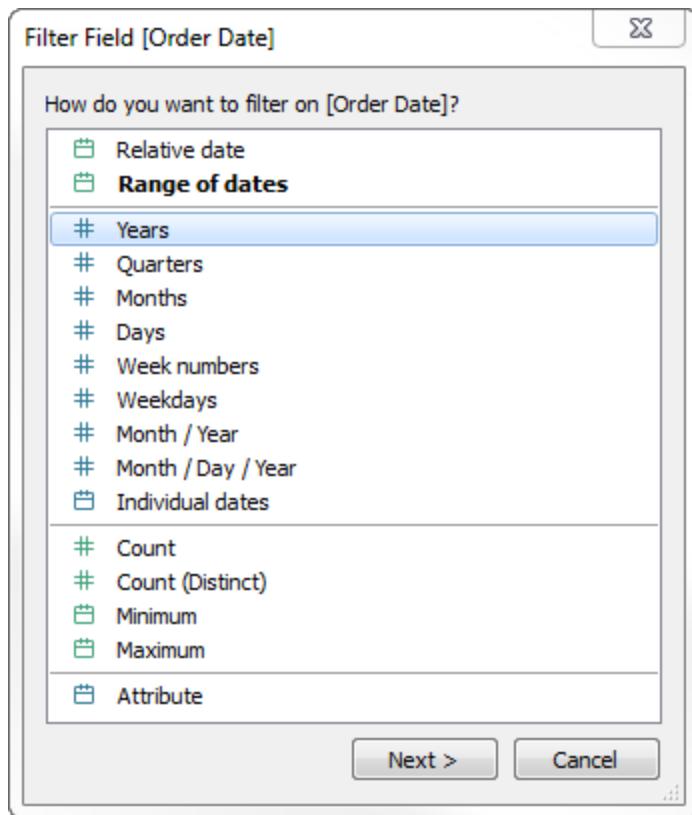


At this point you probably don't want to make the view any more granular; in fact, you may want to simplify the view by removing (that is, filtering) some data. See [Step 4 - Filter to Show a Subset of Data below](#).

Step 4 - Filter to Show a Subset of Data

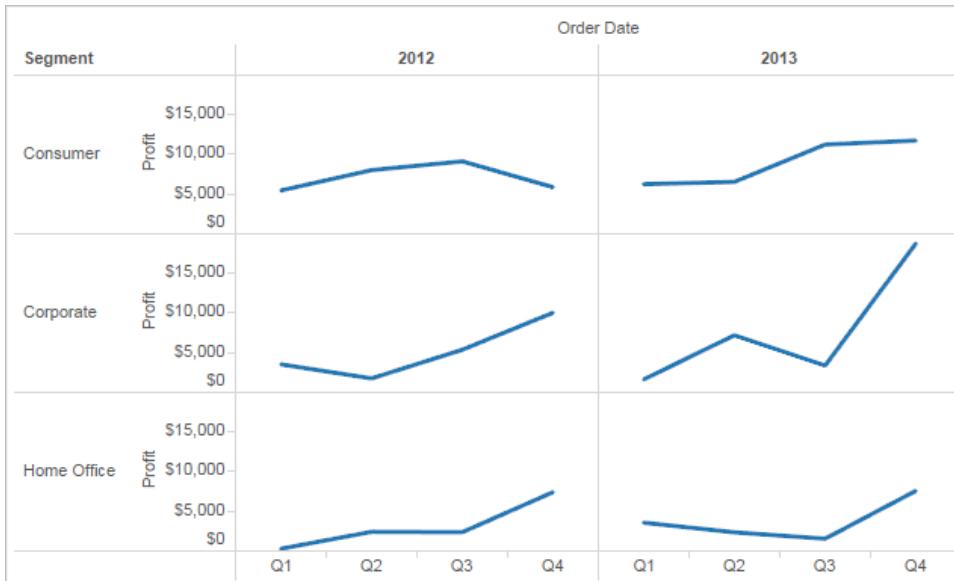
In this step you modify the view from [Step 3 - Create a Small Multiples View](#) on page 51 to show only data for orders in 2012 and 2013.

1. Drag the **Order Date** measure from the **Data** pane and drop it on the **Filters** shelf.



2. In the Filter Field dialog box, choose the date level you want to filter on—**Years**. Then click **Next**.
3. In the next pane, clear the years that you do not want to include in the view—2011 and 2014.
4. When you are finished, click **OK**.

The view updates to only show data rows where Order Date is 2012 or 2013. Tableau can now allocate more space to the data that interests you.



In [Step 5 - Color Marks by Region](#) below, you complete your data exploration by adding color to the view.

Step 5 - Color Marks by Region

In this step, you modify the view from [Step 4 - Filter to Show a Subset of Data](#) on page 53 to color the marks by region.

1. Drag the **Region** dimension from the **Data** pane and drop it on **Color**.

Placing a dimension on **Color** separates the marks according to the members in the dimension, and assigns a unique color to each member. The color legend displays each member name and its associated color.

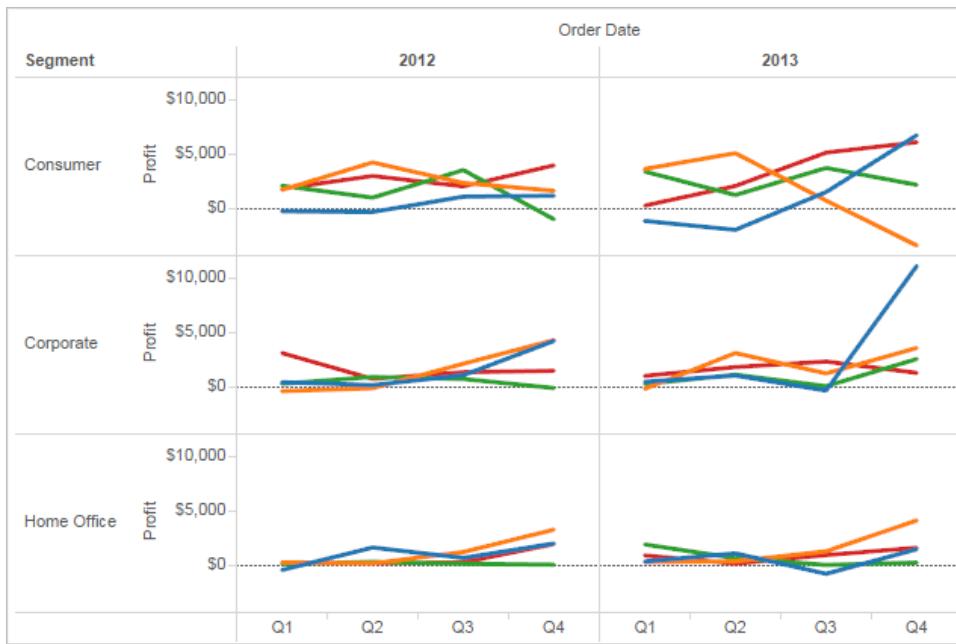
The screenshot shows the Tableau Data pane and Marks pane. In the Data pane, under Dimensions, 'Region' is selected. In the Marks pane, 'Region' is highlighted in blue, indicating it is the current mark type.

Dimensions
Postal Code
Product
Category
Sub-Category
Manufacturer
Product Name
Profit (bin)
Region
Measure Names

Measures
Discount
Profit
Profit Ratio

Marks
Automatic
Region
Detail
Tooltip
Path

Each pane now has four lines, one for each region. This view now shows profit for each customer segment and region for 2012 and 2013.



Build-It-Yourself Exercises

This section includes detailed exercises that guide you through the steps involved in building some common types of data views. All exercises use the **Sample - Superstore** data source, which is included with Tableau Desktop. This collection of exercises is just a sample of the many types of data views that you can create in Tableau.

Build a Bar Chart

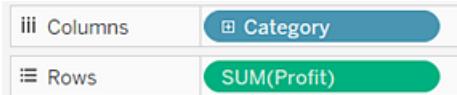
Use bar charts to compare data across categories. You create a bar chart by placing a dimension on the **Rows** shelf and a measure on the **Columns** shelf, or vice versa.

A bar chart uses the **Bar** mark type. Tableau selects this mark type when the data view matches one of the two field arrangements shown below. You can add additional fields to these shelves.

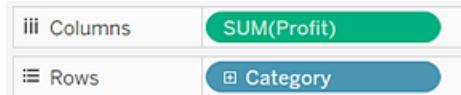
For more information about the **Bar** mark type, see [Bar Mark on page 174](#).

Note: At the end of the procedure is an extra step you can take to display totals at the tops of the bars.

Creates Vertical Bars



Creates Horizontal Bars



To create a bar chart that displays total sales over a four-year period, follow these steps:

1. Connect to the **Sample - Superstore** data source.

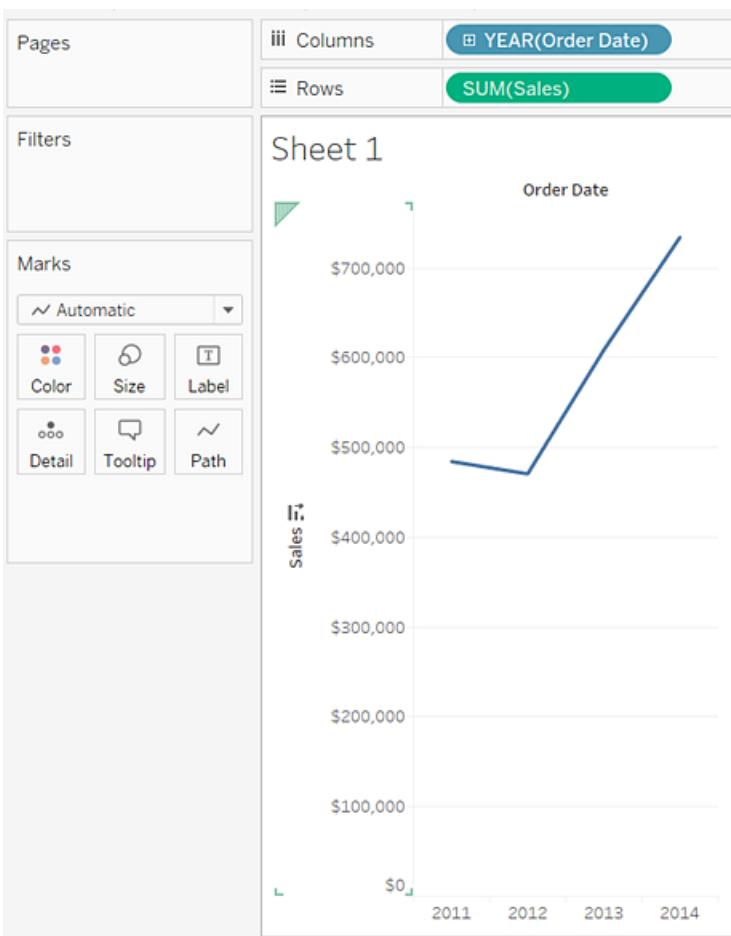
2. Drag the **Order Date** dimension to **Columns**.

The data is aggregated by year and column headers appear.

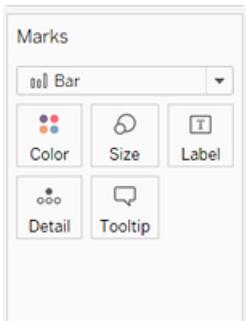
3. Drag the **Sales** measure to **Rows**.

The measure is aggregated as a sum and an axis is created. The column headers move to the bottom of the view.

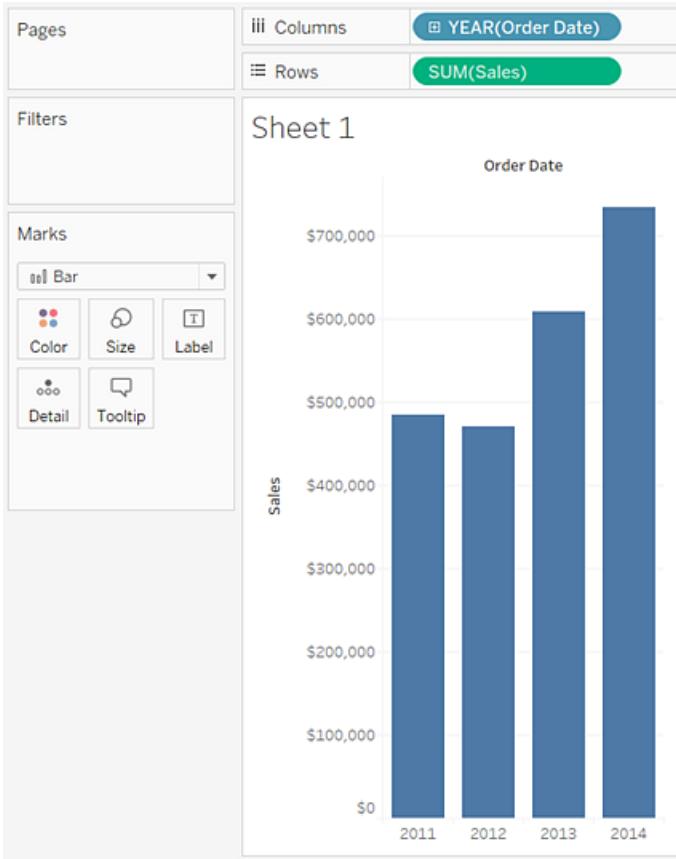
Tableau uses **Line** as the mark type because you added the date dimension.



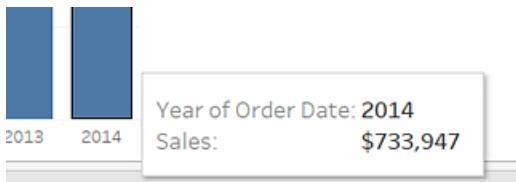
4. On the **Marks** card, select **Bar** from the drop-down list.



The view changes to a bar chart.

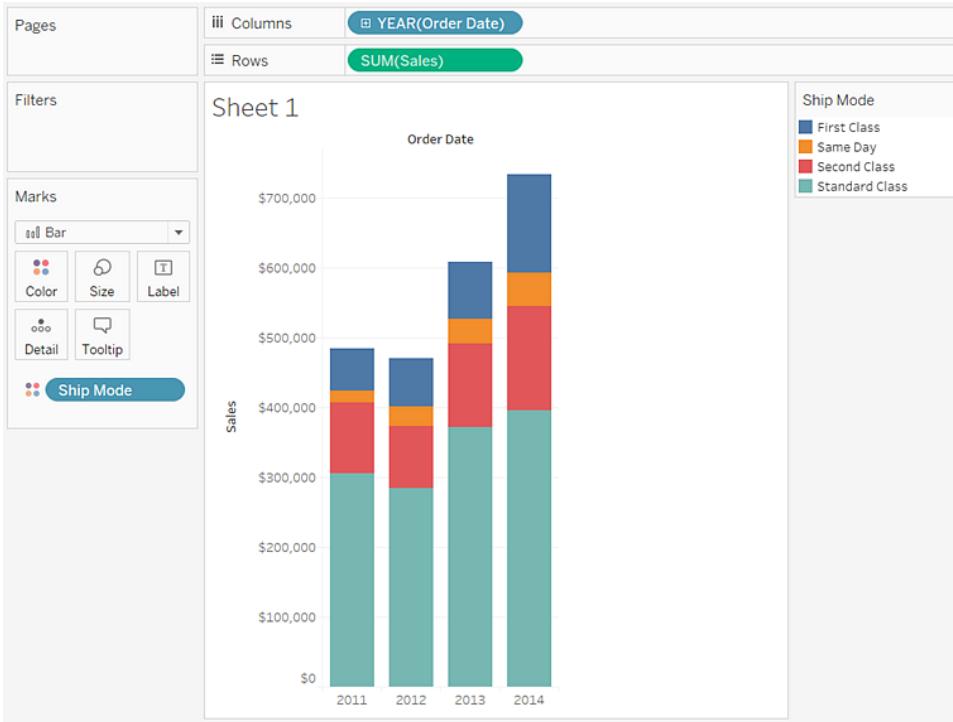


The marks (which are bars in this case) are vertical because the axis is vertical. The length of each mark represents the sum of the sales for that year. For example, the sum of the sales in 2014 is \$733,947, which you can verify by hovering over that column.

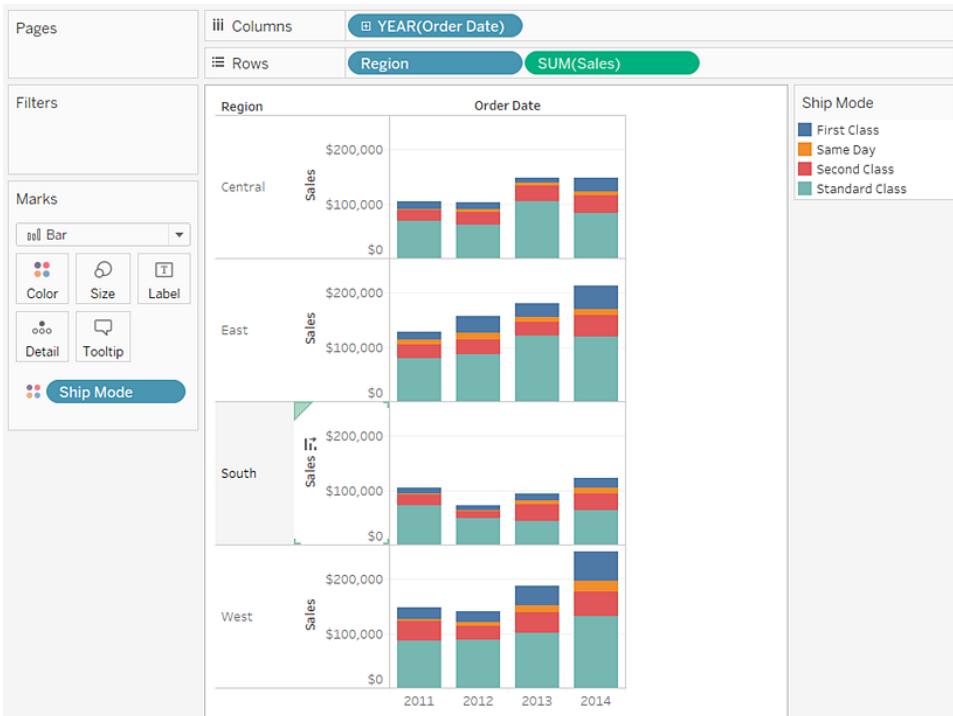


7. Drag the **Ship Mode** dimension to **Color** on the **Marks** card.

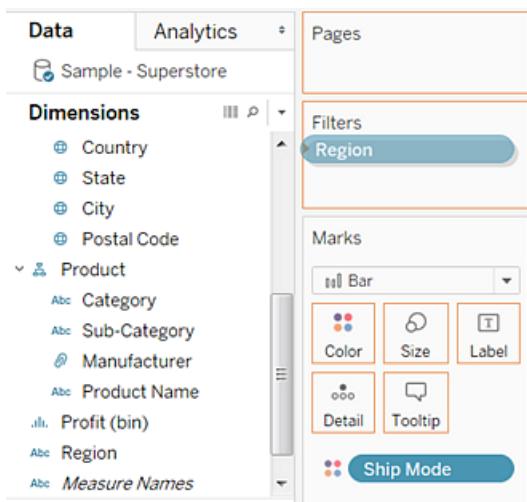
The view shows how different shipping modes have contributed to total sales over time. The ratios look consistent from year to year.



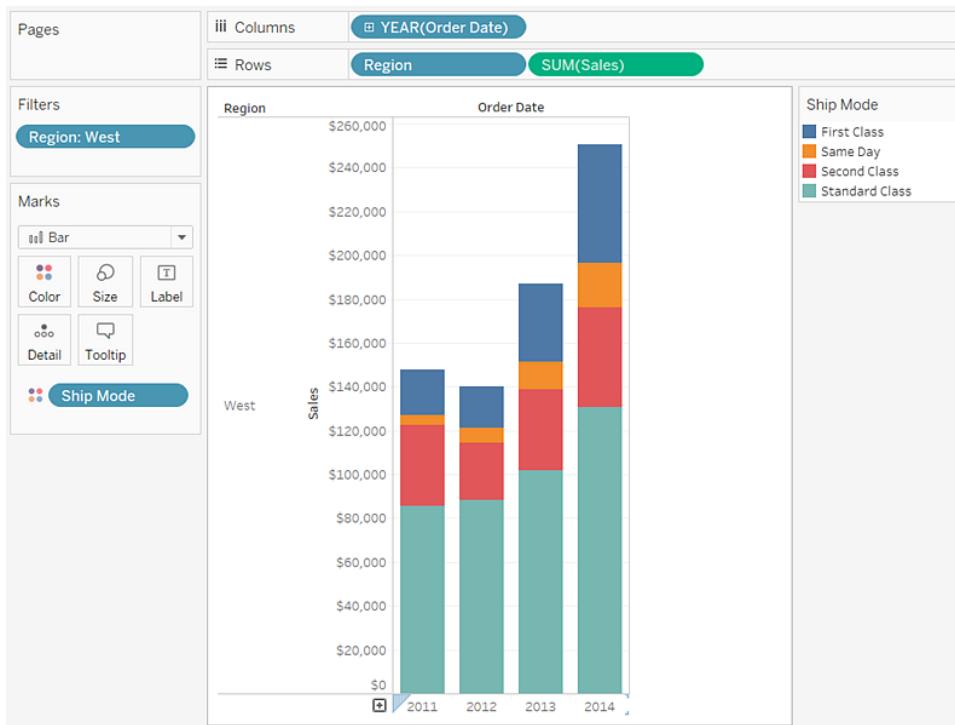
- Drag the **Region** dimension to **Rows**, and drop it to the left of **Sales** to produce multiple axes for sales by region.



9. To view data in the West region only, you can filter out the other regions. To do this, drag the **Region** dimension again, this time from the **Data** pane to the **Filters** shelf.



10. In the Filter [Region] dialog box, clear the **Central**, **East**, and **South** check boxes, and then click **OK**.

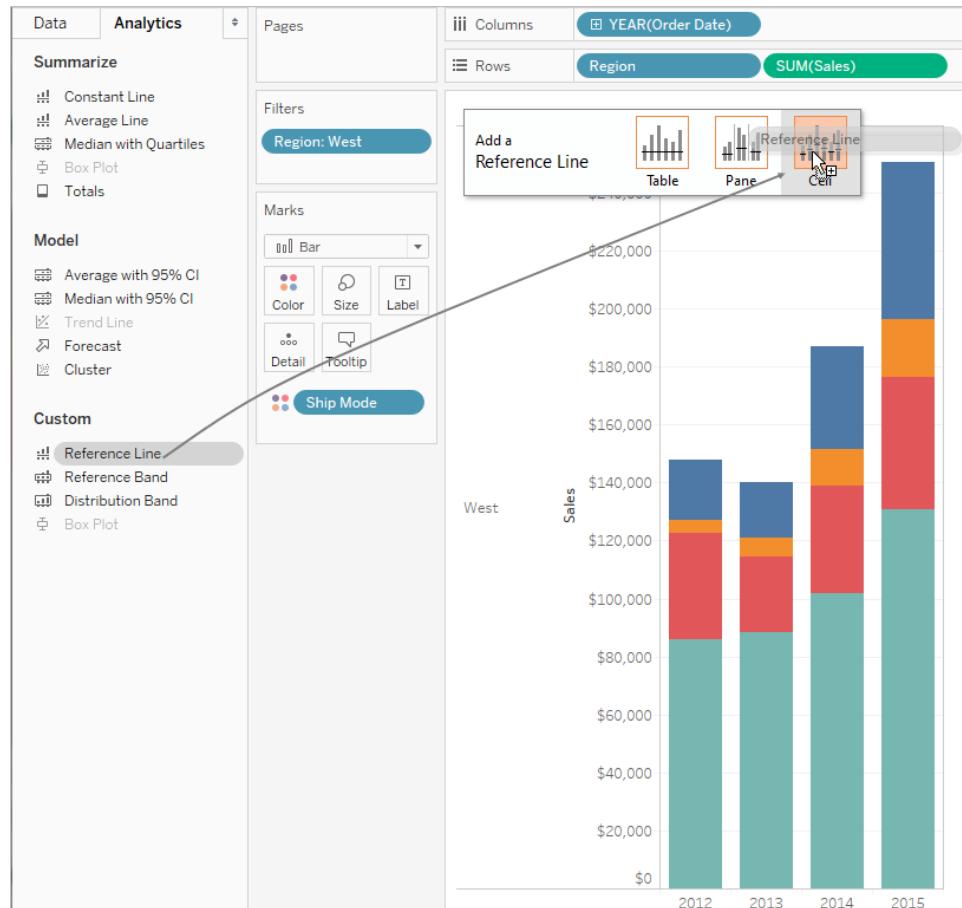


This view gives you insight into your data—for example, how the ship mode changed in the West over the four-year period.

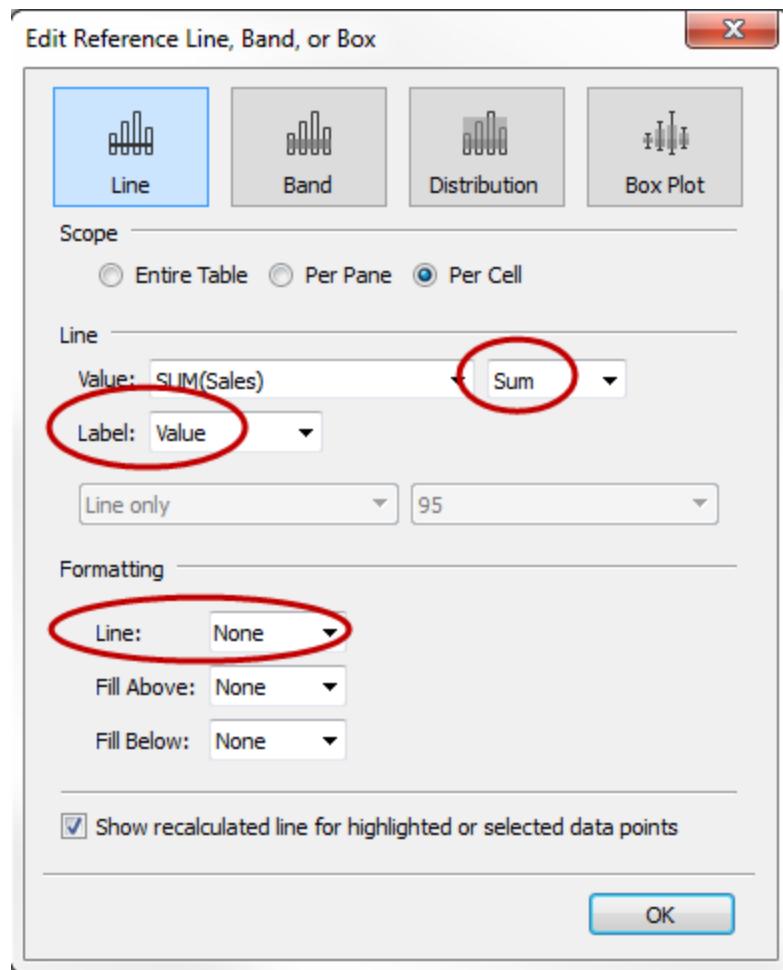
One Step Further: Add Totals To Stacked Bars

Adding totals to the tops of bars in a chart is sometimes as simple as clicking the **Show Mark Labels** icon in the toolbar. But when the bars are broken down by color or size, each individual segment would be labeled, rather than the total for the bar. With a few steps, you can add a total label at the top of every bar even when the bars are subdivided as in the view you just created. In the following procedure you will technically be adding a reference line. But by configuring that "line" in a certain way, you end up with the labels you want.

1. From the **Analytics** pane, drag a **Reference Line** into the view and drop it on **Cell**.

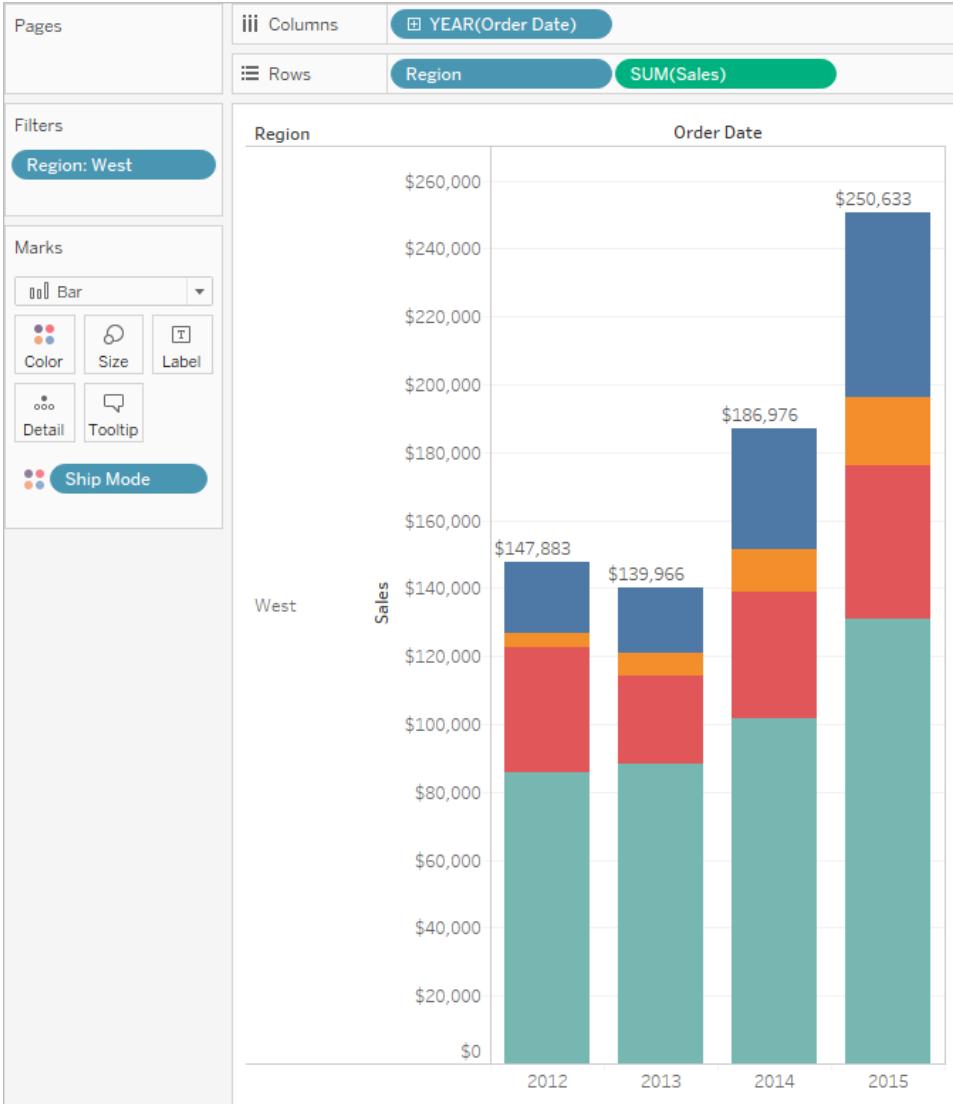


2. In the Edit Line, Band, or Box dialog box, set the aggregation for **SUM(Sales)** to **Sum**, set **Label** to **Value**, and set **Line** under **Formatting** to **None**:



Then click **OK** to close the Edit Reference Line, Band, or Box dialog box.

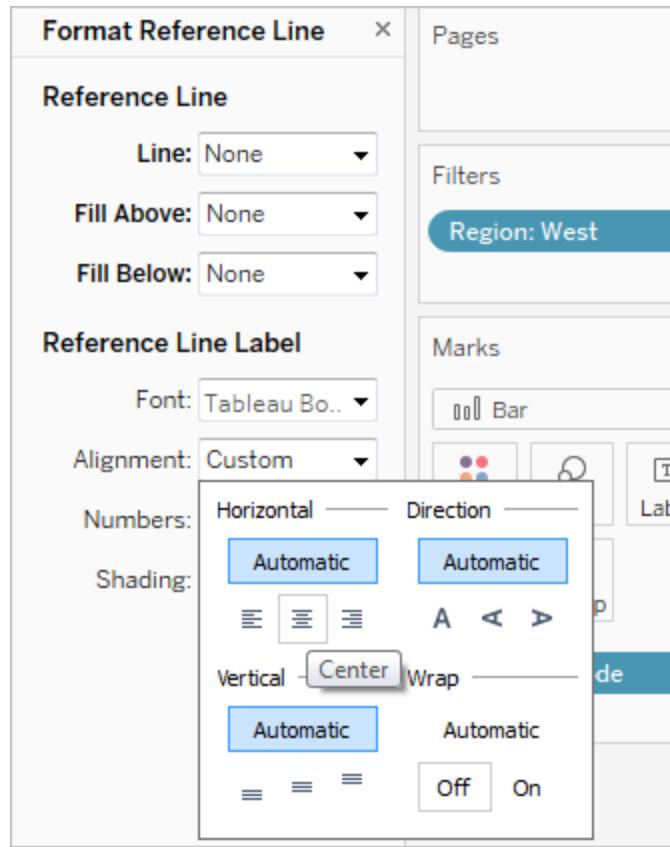
Your view now has currency totals at the top of each bar:



You may need to adjust the view to make it look just right. If the bars are too narrow, the numbers are truncated; to fix this, press Ctrl + Right on the keyboard to make the bars wider. Or if you want to center the totals over the bars—by default, they are left-aligned. Do the following:

1. Right-click any of the totals on the bar chart and select **Format**.
2. In the Format window, in the **Reference Line Label** area, open the **Alignment**

control and select the Center option for Horizontal alignment:



Build a Text Table

In Tableau, you typically create text tables (also called cross-tabs or pivot tables) by placing one dimension on the **Rows** shelf and another dimension on the **Columns** shelf. You then complete the view by dragging one or more measures to **Text** on the **Marks** card.

A text table uses the text mark type. Tableau uses this mark type automatically if the view is constructed using only dimensions (assuming the mark type is set to **Automatic**). For more information about the text mark type, see [Text Mark on page 183](#).

To create a text table that shows sales totals by year and category, follow these steps:

1. Connect to the **Sample - Superstore** data source.
2. Drag the **Order Date** dimension to **Columns**.

Tableau aggregates the date by year and creates column headers.

3. Drag the **Sub-Category** dimension to **Rows**.

Tableau creates row headers. Columns with headers plus rows with headers means that a valid table structure now exists. Now you can add a measure to the view to see actual data.

4. Drag the **Sales** measure to **Text** on the **Marks** card.

Tableau aggregates the measure as a sum.

The screenshot shows the Tableau Data pane. On the left, there are sections for 'Pages' (with 'YEAR(Order Date)' selected), 'Filters', and 'Marks'. Under 'Marks', 'Automatic' is selected, and 'Text' is chosen as the mark type. A green button labeled 'SUM(Sales)' is highlighted. The main area displays a table with the following data:

Sub-Categ..	Order Date			
	2011	2012	2013	2014
Accessories	\$25,014	\$40,524	\$41,896	\$59,946
Appliances	\$15,314	\$23,241	\$26,050	\$42,927
Art	\$6,058	\$6,237	\$5,910	\$8,914
Binders	\$43,488	\$37,453	\$49,485	\$72,986
Bookcases	\$20,037	\$38,544	\$26,275	\$30,024
Chairs	\$77,242	\$71,735	\$83,919	\$95,554
Copiers	\$10,850	\$26,179	\$49,599	\$62,899
Envelopes	\$3,856	\$4,512	\$4,730	\$3,379
Fasteners	\$661	\$545	\$960	\$858
Furnishings	\$13,826	\$21,090	\$27,874	\$28,915
Labels	\$2,841	\$2,956	\$2,827	\$3,861
Machines	\$62,023	\$27,764	\$55,907	\$43,545
Paper	\$14,835	\$15,288	\$20,638	\$27,718
Phones	\$77,391	\$68,314	\$78,660	\$105,643
Storage	\$50,329	\$45,048	\$58,632	\$69,834
Supplies	\$14,394	\$1,952	\$14,278	\$16,049
Tables	\$46,088	\$39,150	\$60,833	\$60,894

Tableau uses text as the mark type. Each cell in the table displays the sum of sales for a particular year and sub-category.

We can see that the chairs and phones sub-categories had the highest sales in every year.

5. Drag the **Region** dimension to **Rows** and drop it to the left of **Sub-Category**. A small triangle will appear to indicate that the new field will be inserted to the left of the existing field.

The screenshot shows the Tableau Data Prep interface. On the left, the 'Dimensions' pane lists City, Postal Code, Product (Category, Sub-Category, Manufacturer, Product Name), Profit (bin), and Region. The 'Measures' pane lists Sales. The main area displays a table with columns: Sub-Categ.., 2011, 2012, and Order Da. The data rows include Accessories, Appliances, Art, Binders, Bookcases, Chairs, Copiers, Envelopes, Fasteners, Furnishings, Labels, Machines, Paper, Phones, Storage, Supplies, and Tables.

Sub-Categ..	2011	2012	Order Da
Accessories	\$25,014	\$40,524	
Appliances	\$15,314	\$23,241	
Art	\$6,058	\$6,237	
Binders	\$43,488	\$37,453	
Bookcases	\$20,037	\$38,544	
Chairs	\$77,242	\$71,735	
Copiers	\$10,850	\$26,179	
Envelopes	\$3,856	\$4,512	
Fasteners	\$661	\$545	
Furnishings	\$13,826	\$21,090	
Labels	\$2,841	\$2,956	
Machines	\$62,023	\$27,764	

The view now breaks out sales by region, in addition to year and sub-category.

The screenshot shows the Tableau Data Prep interface with a different layout. The 'Dimensions' pane is not visible, but the 'Measures' pane is present. The main area displays a table with columns: Region, Sub-Categ.., and Order Date (2011, 2012, 2013, 2014). The data rows are grouped by Region (Central and East) and Sub-Category (Accessories, Appliances, Art, Binders, Bookcases, Chairs, Copiers, Envelopes, Fasteners, Furnishings, Labels, Machines, Paper, Phones, Storage, Supplies, Tables).

Region	Sub-Categ..	Order Date			
		2011	2012	2013	2014
Central	Accessories	\$4,439	\$7,795	\$10,802	\$10,920
	Appliances	\$3,659	\$4,975	\$6,015	\$8,933
	Art	\$822	\$1,132	\$1,520	\$2,291
	Binders	\$15,871	\$5,891	\$14,056	\$21,105
	Bookcases	\$1,834	\$8,298	\$8,385	\$5,640
	Chairs	\$20,754	\$17,909	\$23,350	\$23,218
	Copiers	\$3,270	\$12,810	\$17,500	\$3,680
	Envelopes	\$1,599	\$871	\$971	\$1,197
	Fasteners	\$122	\$89	\$247	\$320
	Furnishings	\$2,536	\$2,529	\$5,116	\$5,074
	Labels	\$1,048	\$305	\$511	\$587
	Machines	\$16,292	\$1,852	\$2,659	\$5,995
	Paper	\$2,347	\$3,544	\$5,366	\$6,235
	Phones	\$9,926	\$19,364	\$19,902	\$23,211
	Storage	\$11,093	\$8,331	\$12,812	\$13,694
	Supplies	\$440	\$324	\$4,295	\$4,408
Tables	\$7,785	\$6,857	\$13,923	\$10,589	
East	Accessories	\$6,054	\$17,911	\$6,231	\$14,837
	Appliances	\$5,779	\$6,691	\$9,427	\$12,291

Regions are listed alphabetically. You can drop **Region** to the right of **Sub-Category** to organize the view first by sub-category, and then by region.

The screenshot displays a Tableau data visualization with the following structure:

- Columns:** Sub-Categ., Region, YEAR(Order Date)
- Rows:** Sub-Category, Region
- Data Headers:** Order Date (2011, 2012, 2013, 2014)
- Data Rows:** The data is organized by Sub-Category (e.g., Accessories, Appliances, Art, Binders, Bookcases) and Region (Central, East, South, West). Each row represents a year from 2011 to 2014.

		Order Date			
Sub-Categ..	Region	2011	2012	2013	2014
Accessories	Central	\$4,439	\$7,795	\$10,802	\$10,920
	East	\$6,054	\$17,911	\$6,231	\$14,837
	South	\$5,595	\$4,142	\$9,380	\$8,160
	West	\$8,926	\$10,676	\$15,482	\$26,030
Appliances	Central	\$3,659	\$4,975	\$6,015	\$8,933
	East	\$5,779	\$6,691	\$9,427	\$12,291
	South	\$2,120	\$3,850	\$5,607	\$7,948
	West	\$3,755	\$7,725	\$5,001	\$13,754
Art	Central	\$822	\$1,132	\$1,520	\$2,291
	East	\$1,290	\$1,707	\$1,883	\$2,606
	South	\$566	\$1,362	\$1,391	\$1,337
	West	\$3,380	\$2,035	\$1,116	\$2,681
Binders	Central	\$15,871	\$5,891	\$14,056	\$21,105
	East	\$6,347	\$14,207	\$18,956	\$13,989
	South	\$8,307	\$13,467	\$4,112	\$11,143
	West	\$12,963	\$3,889	\$12,361	\$26,748
Bookcases	Central	\$1,834	\$8,298	\$8,385	\$5,640
	East	\$10,863	\$19,653	\$5,964	\$7,338
	South	\$794	\$1,239	\$3,709	\$5,157
	West	\$6,545	\$9,354	\$8,217	\$11,888

You can use a table calculation to show percentages of total instead of raw dollar values. First, you must determine how to frame the calculation.

In this case, there are three dimensions in the view: **Order Date**, **Sub-Category**, and **Region**.

You could show percentages of total for a single dimension, but that can be unwieldy. For example, if you show percentages just by region, the percentages would be calculated across the two remaining dimensions: **Sub-Category** (there are 17 sub-categories) and **Year(Order Date)** (there are 4 years). So you would be dividing the total $17 \times 4 = 68$ ways. That would make for some tiny percentages.

Instead, show percentages using two dimensions: **Year(Order Date)** and **Region**. Then the percentages are calculated on the remaining dimension, **Sub-Category**, that is, you calculate percent of total within each highlighted area shown below.

III Columns YEAR(Order Date)

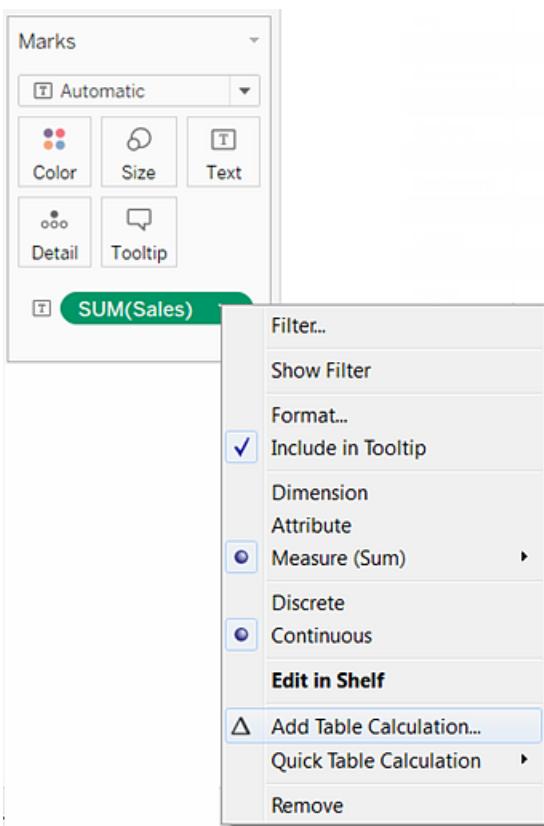
Rows Region Sub-Category

Region	Sub-Categ..	Order Date			
		2011	2012	2013	2014
Central	Accessories	\$4,439	\$7,795	\$10,802	\$10,920
	Appliances	\$3,659	\$4,975	\$6,015	\$8,933
	Art	\$922	\$1,132	\$1,520	\$2,291
	Binders	\$15,871	\$5,891	\$14,056	\$21,105
	Bookcases	\$1,834	\$8,298	\$8,385	\$5,640
	Chairs	\$20,754	\$17,909	\$23,350	\$23,218
	Copiers	\$3,270	\$12,810	\$17,500	\$3,680
	Envelopes	\$1,599	\$871	\$971	\$1,197
	Fasteners	\$122	\$89	\$247	\$320
	Furnishings	\$2,536	\$2,529	\$5,116	\$5,074
	Labels	\$1,048	\$305	\$511	\$587
	Machines	\$16,292	\$1,852	\$2,659	\$5,995
	Paper	\$2,347	\$3,544	\$5,366	\$6,235
	Phones	\$9,926	\$19,364	\$19,902	\$23,211
	Storage	\$11,093	\$8,331	\$12,812	\$13,694
East	Supplies	\$440	\$324	\$4,295	\$4,408
	Tables	\$7,785	\$6,857	\$13,923	\$10,589
East	Accessories	\$6,054	\$17,911	\$6,231	\$14,837
	Appliances	\$5,779	\$6,691	\$9,427	\$12,291

The dimensions that you use to frame your calculation are called the *addressing fields*, and the fields in which you run your calculation are the *partition fields*.

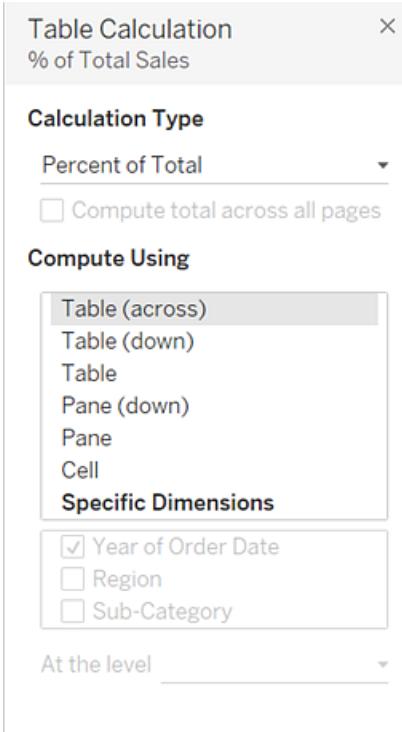
For more information about these concepts, see [Table Calculations: Addressing and Partitioning on page 855](#).

6. To create a table calculation to show percentages, right-click (control-click on Mac) the **SUM(Sales)** field on the **Marks** card, and then select **Add Table Calculation**.



7. In the Table Calculation dialog box, set **Calculation Type** to **Percent of Total**.

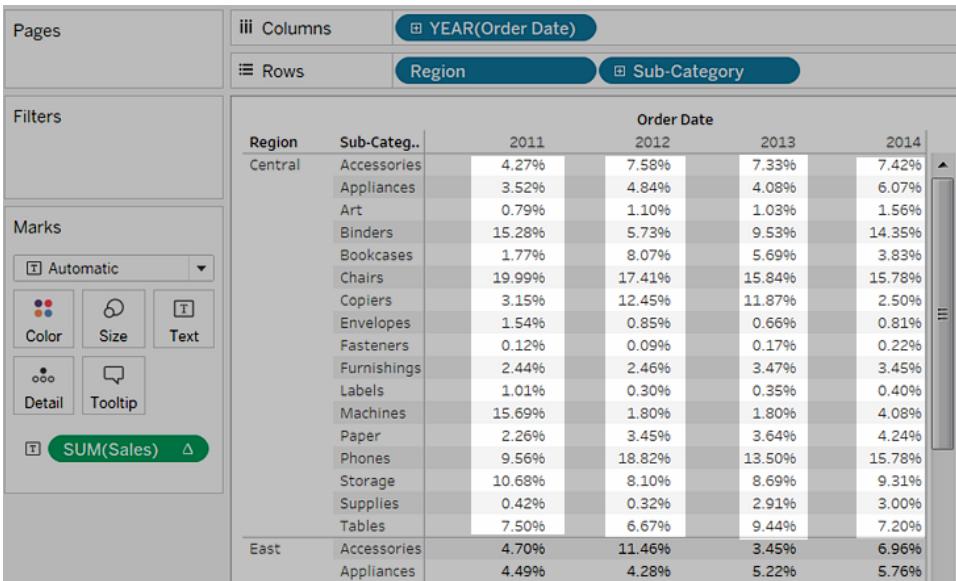
The options in the dialog box change depending on the type of calculation you choose.



For more information about using table calculations, see [Transform Values with Table Calculations on page 843](#).

8. For the Calculation definition, select **Pane (Down)**, and then close the Table Calculation dialog box.

Now we see percentages calculated within each sub-category, duplicated for each year within each region. The numbers within each highlighted area add up to 100%.



Pane (Down) is the appropriate choice because it specifies that the calculation should be performed from top to bottom within each pane of the table. The table has two vertical dimensions, so **Table (Down)** would have calculated the percent of total from top to bottom for the entire table, ignoring the **Region** dimension.

The pane is always the finest level of detail for the relevant direction (across or down). If you had three dimensions on the vertical axis, you might have had to use field names to define the calculation, because only the dimension furthest to the left on the **Rows** shelf (defined as Table) and the dimension furthest to the right (defined as Pane) could be captured with the structural options.

These concepts can be daunting. One solution is to use trial-and-error to see the results of different definitions for table calculations. But you should have a clear idea of what result you want, so that you can recognize it when you see it.

Building Line Charts

Line charts connect individual data points in a view. They provide a simple way to visualize a sequence of values and are useful when you want to see trends over time, or to forecast future values. For more information about the line mark type, see [Line Mark on page 176](#).

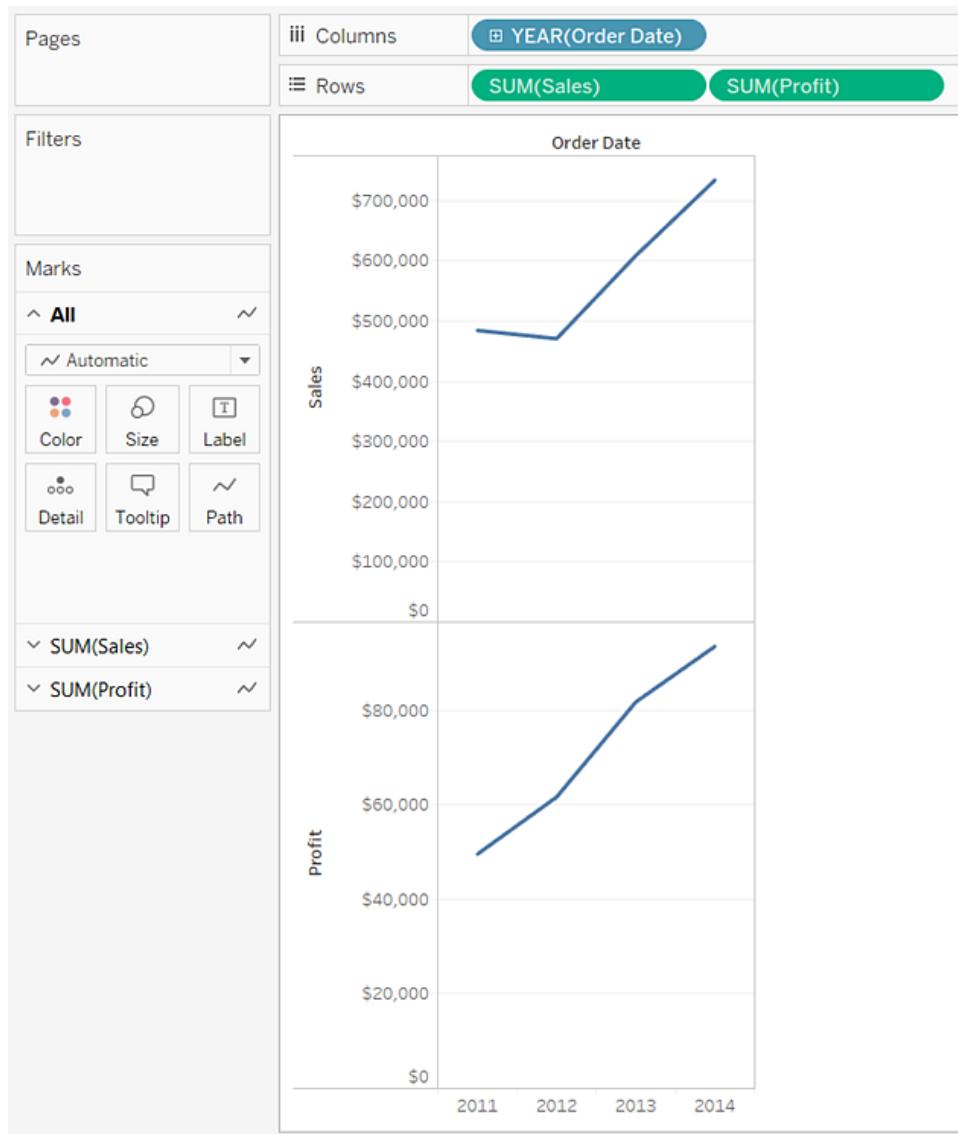
To create a view that displays the sum of sales and the sum of profit for all years, and then uses forecasting to determine a trend, follow these steps:

1. Connect to the **Sample - Superstore** data source.
2. Drag the **Order Date** dimension to **Columns**.
Tableau aggregates the date by year, and creates column headers.
3. Drag the **Sales** measure to **Rows**.

Tableau aggregates **Sales** as SUM and displays a simple line chart.

4. Drag the **Profit** measure to **Rows** and drop it to the right of the **Sales** measure.

Tableau creates separate axes along the left margin for **Sales** and **Profit**.



Notice that the scale of the two axes is different—the **Sales** axis scales from \$0 to \$700,000, whereas the **Profit** axis scales from \$0 to \$100,000. This can make it hard to see that sales values are much greater than profit values.

When you are displaying multiple measures in a line chart, you can align or merge axes to make it easier for users to compare values.

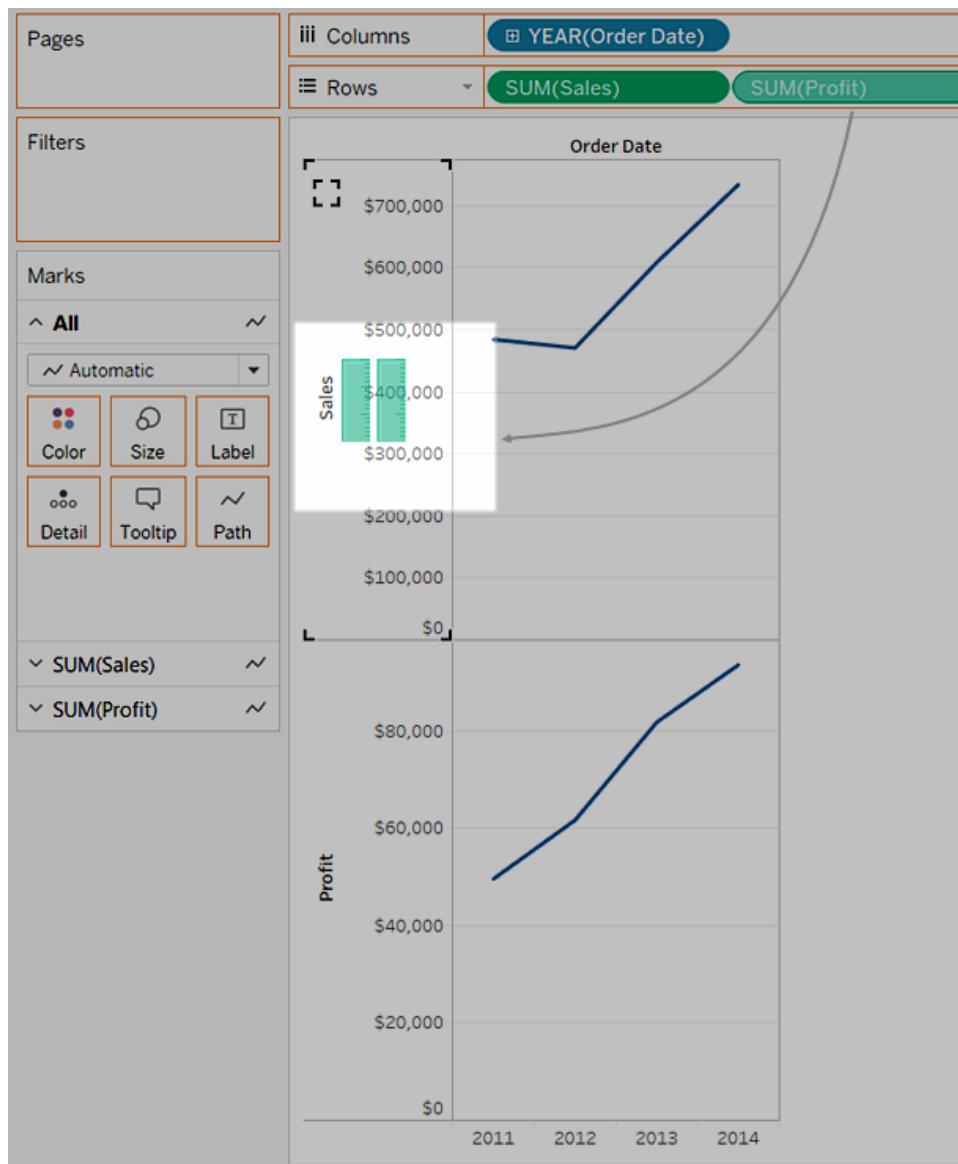
For more information about aligning the axes, see [Dual Axes on page 607](#).

For more information about enforcing a single axis across multiple measures, see [Blended Axes](#) on page 606.

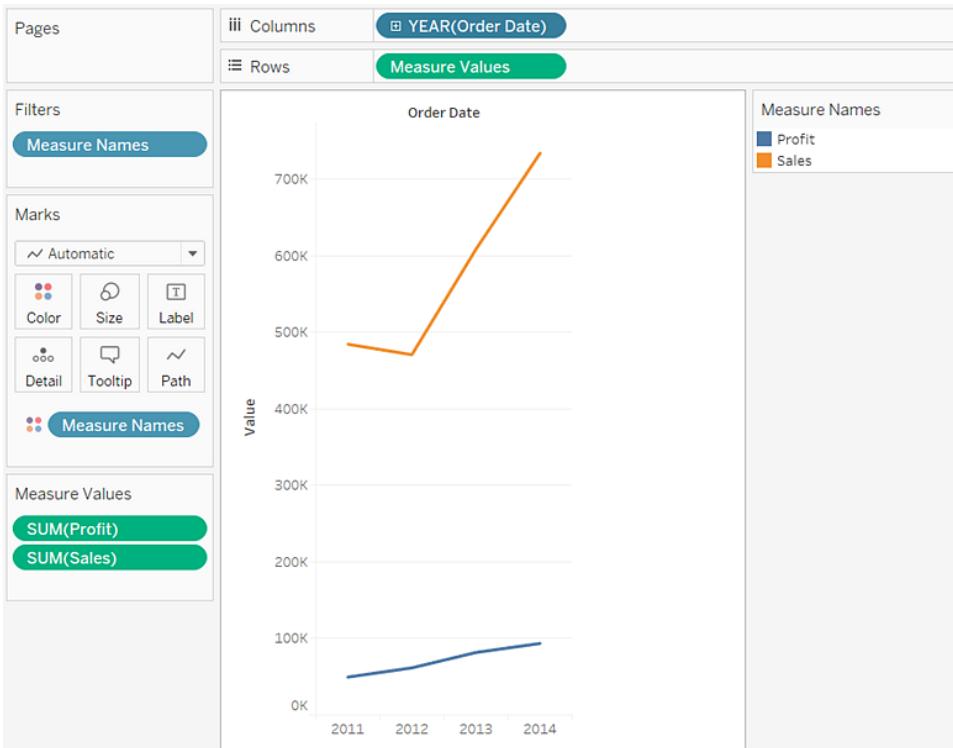
With either of these options, you can create a combination chart to change the mark type for one of your measures.

For more information, see [Combination Charts](#) on page 610.

5. Drag the **SUM(Profit)** field from **Rows** to the **Sales** axis to create a blended axis. The two pale green parallel bars indicate that **Profit** and **Sales** will use a blended axis when you release the mouse button.

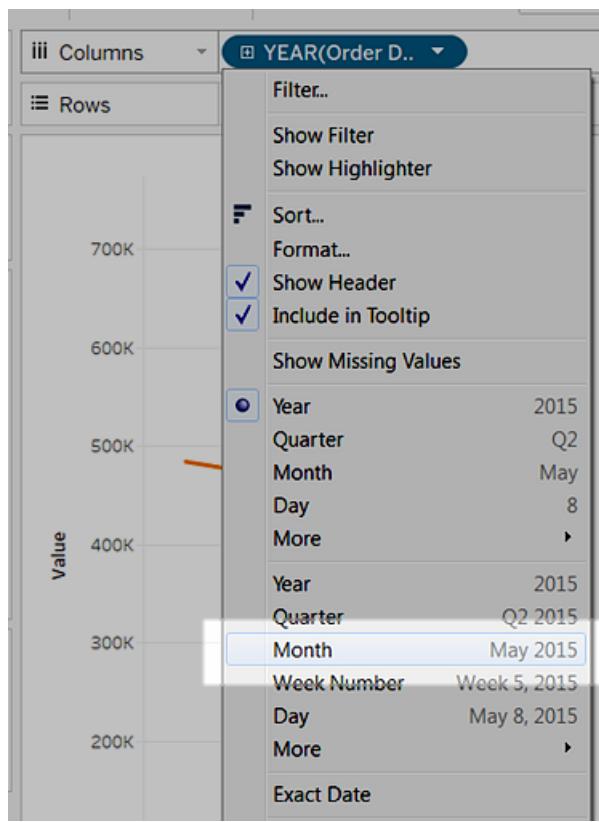


The view updates to look like this:

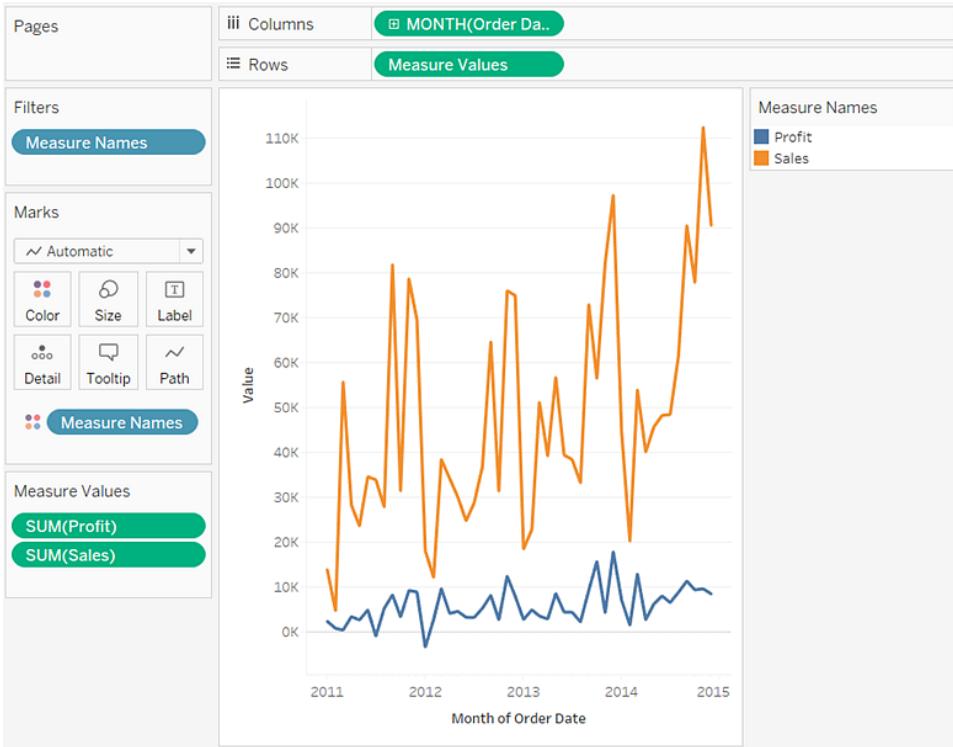


The view is rather sparse because we are looking at a summation of values on a per-year basis.

6. Click the drop-down arrow in the **Year(Order Date)** field on the **Columns** shelf and select **Month** in the lower part of the context menu to see a continuous range of values over the four-year period.

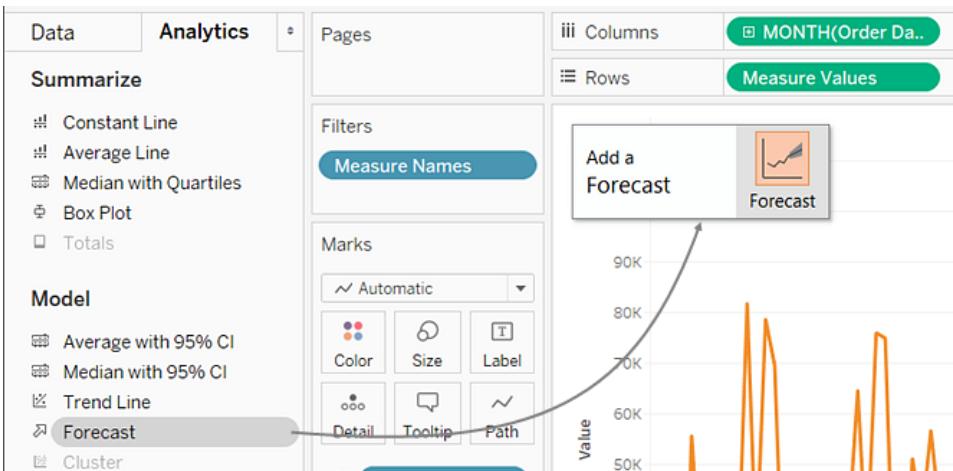


The resulting view is a lot more detailed than the original view:

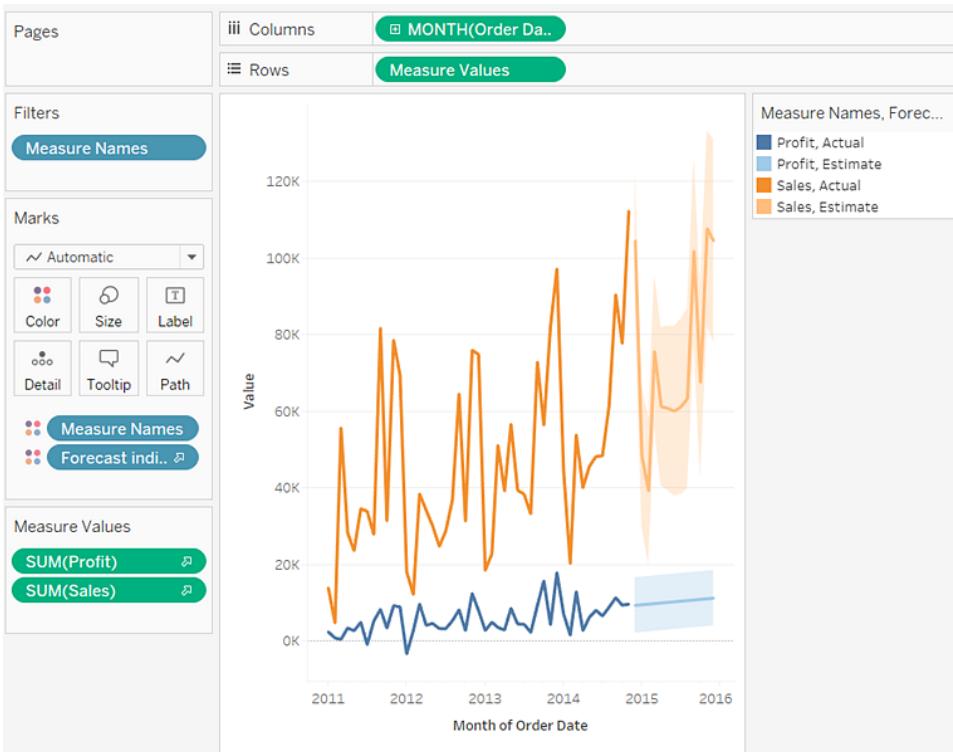


Notice that the values seem to go much higher just before the end of each year. A pattern like that is known as *seasonality*. If we turn on the forecasting feature in the view, we can see whether we should expect that the apparent seasonal trend will continue in the future.

- To add a forecast, in the **Analytics** pane, drag the **Forecast** model to the view, and then drop it on **Forecast**.



We then see that, according to Tableau forecasting, the seasonal trend does continue into the future:



Build a Scatter Plot

Use scatter plots to visualize relationships between numerical variables.

In Tableau, you create a scatter plot by placing at least one measure on the **Columns** shelf and at least one measure on the **Rows** shelf. If these shelves contain both dimensions and measures, Tableau places the measures as the innermost fields, which means that measures are always to the right of any dimensions that you have also placed on these shelves. The word "innermost" in this case refers to the table structure.

Creates Simple Scatter Plot

iii Columns	SUM(Sales)
≡ Rows	SUM(Profit)

Creates Matrix of Scatter Plots

iii Columns	Region	SUM(Sales)
≡ Rows	Category	SUM(Profit)

A scatter plot can use several mark types. By default, Tableau uses the shape mark type. Depending on your data, you might want to use another mark type, such as a circle or a square. For more information, see [Mark Types on page 168](#).

To use scatter plots and trend lines to compare sales to profit, follow these steps:

1. Open the **Sample - Superstore** data source.
2. Drag the **Profit** measure to **Columns**.

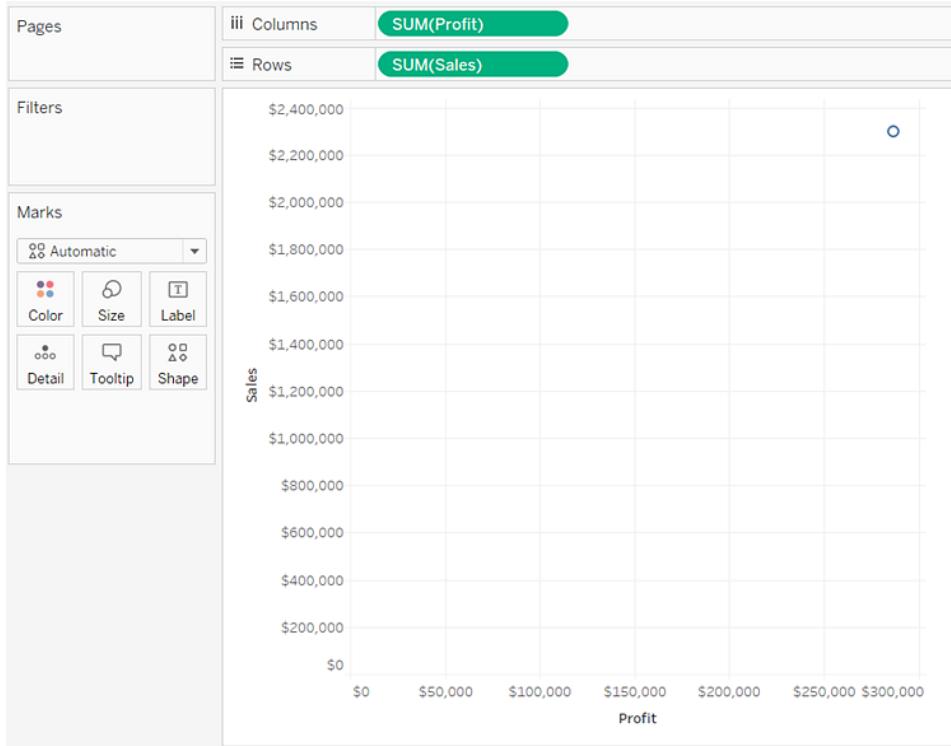
Tableau aggregates the measure as a sum and creates a horizontal axis.

3. Drag the **Sales** measure to **Rows**.

Tableau aggregates the measure as a sum and creates a vertical axis.

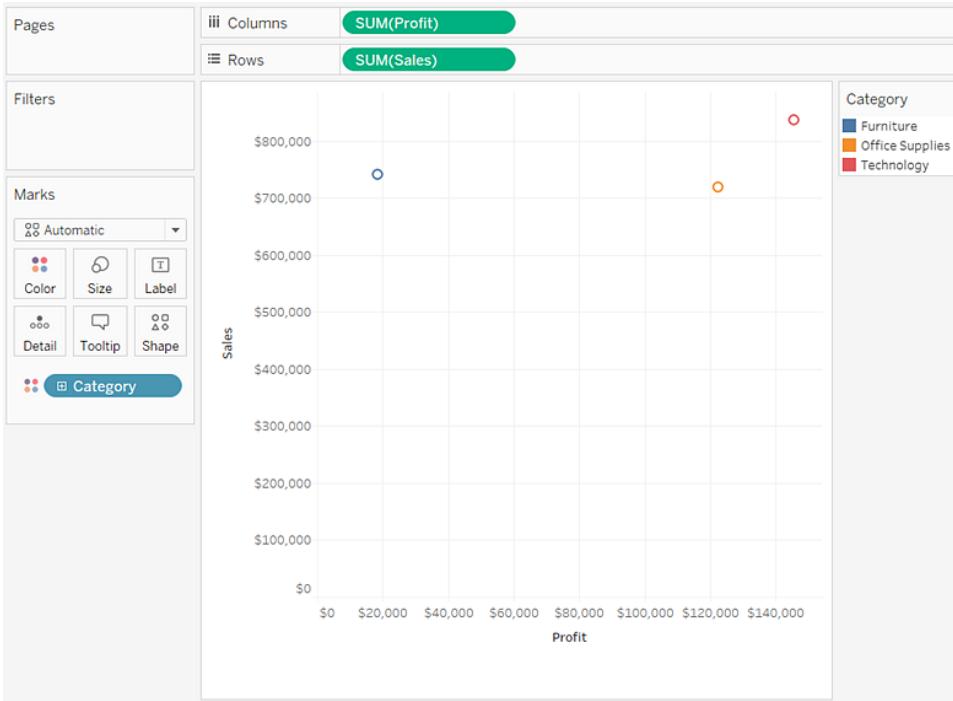
Measures contain continuous numerical data. When you plot one number against another, you are comparing two numbers; the resulting chart is analogous to a Cartesian chart, with x and y coordinates.

Now you have a one-mark scatter plot:



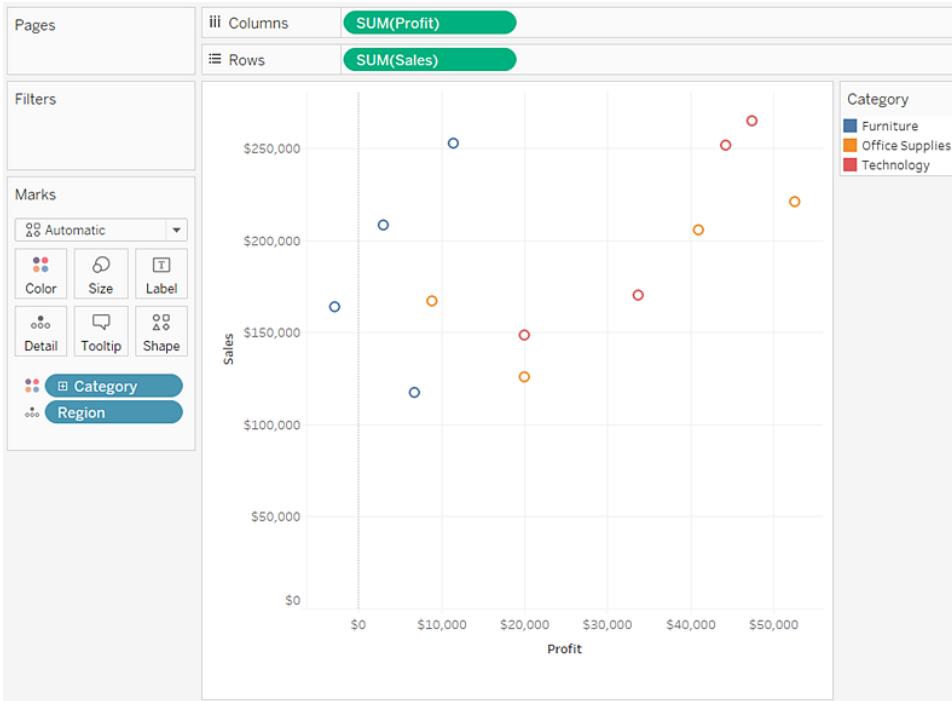
4. Drag the **Category** dimension to **Color** on the Marks card.

This separates the data into three marks—one for each dimension member—and encodes the marks using color.

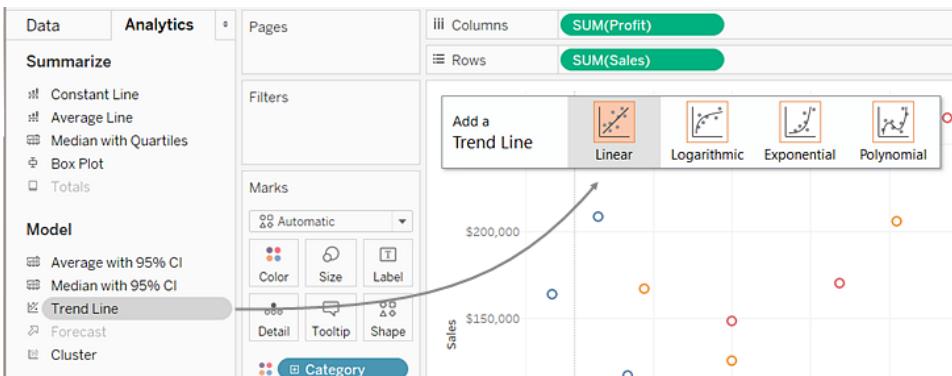


5. Drag the **Region** dimension to **Detail** on the **Marks** card.

Now there are many more marks in the view. The number of marks is equal to the number of distinct regions in the data source multiplied by the number of departments. (If you're curious, use the **Undo** button on the toolbar to see what would have happened if you'd dropped the **Region** dimension on **Shape** instead of **Detail**.)



- To add trend lines, from the **Analytics** pane, drag the **Trend Line** model to the view, and then drop it on the model type.

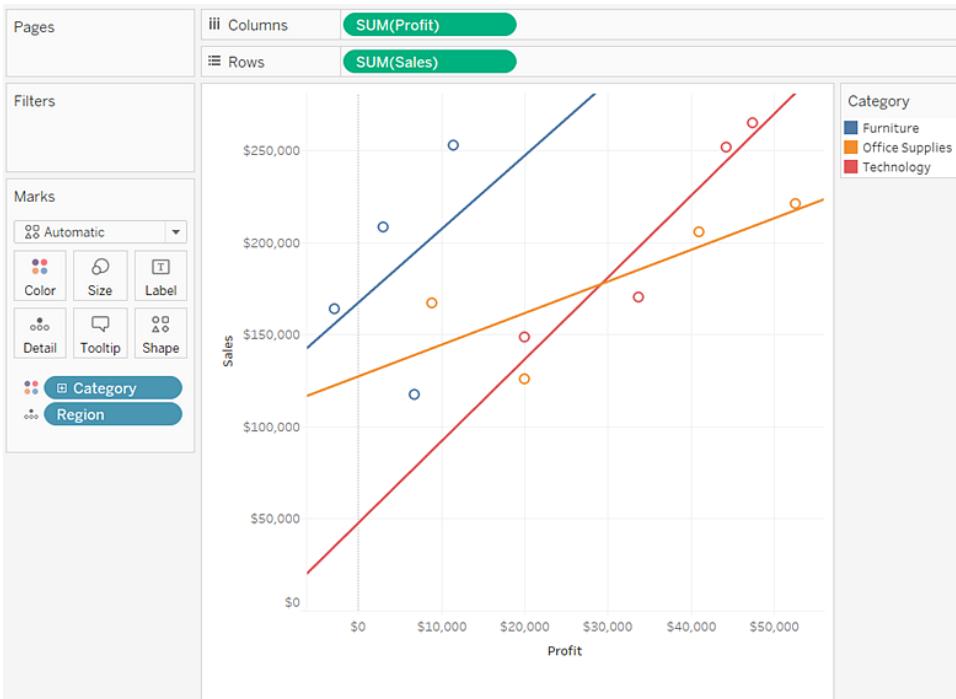


A trend line can provide a statistical definition of the relationship between two numerical values. To add trend lines to a view, both axes must contain a field that can be interpreted as a number—by definition, that is always the case with a scatter plot.

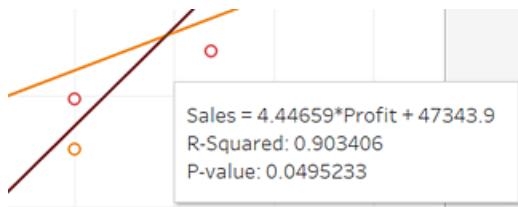
Tableau adds three linear trend lines—one for each color that you are using to distinguish the three categories. Tableau also shows the confidence bands for each trend line.

- To declutter the view, remove the confidence bands: right-click (control-click on Mac) in the view and choose **Trend Lines > Edit Trend Lines**. In the Trend Line Options dialog

box, in the **Options** section, clear the **Show Confidence Bands** check box, and then click **OK**.



8. Hover the cursor over the trend lines to see statistical information about the model that was used to create the line:



For more information, see [Assessing Trend Line Significance on page 664](#). You can also customize the trend line to use a different model type or to include confidence bands. For more information, see [Adding Trend Lines on page 658](#).

See Also

[Example – Scatter Plots and Aggregation on page 339](#)

Build a Heat Map

Use heat maps to compare categorical data using color.

In Tableau, you create a heat map by placing one or more dimensions on the **Columns** shelf and one or more dimensions on the **Rows** shelf. You then select **Square** as the mark type and place a measure of interest on the **Color** shelf.

You can enhance this basic heat map by setting the size and shape of the table cells.

To create a heat map to explore how profit varies across regions, product sub-categories, and customer segments, follow these steps:

1. Connect to the **Sample - Superstore** data source.
2. Drag the **Segment** dimension to **Columns**.

Tableau creates headers with labels derived from the dimension member names.

3. Drag the **Region** and **Sub-Category** dimensions to **Rows**, dropping **Sub-Category** to the right of **Region**.

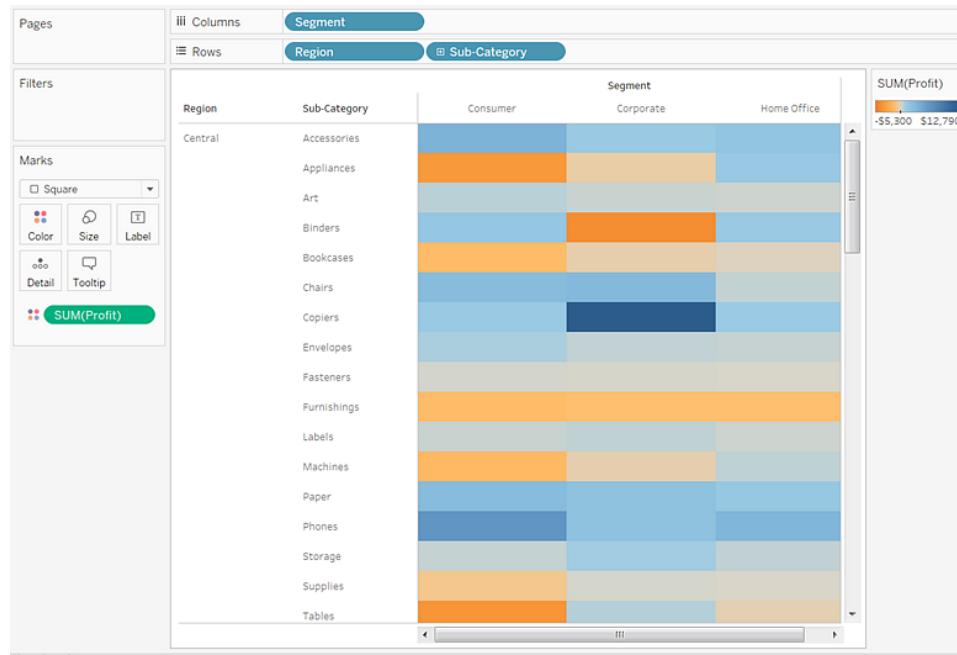
Now you have a nested table of categorical data (that is, the **Sub-Category** dimension is nested within the **Region** dimension).

4. Drag the **Profit** measure to **Color** on the **Marks** card.

Tableau aggregates the measure as a sum. The color legend reflects the continuous data range.

5. Optimize the view format:

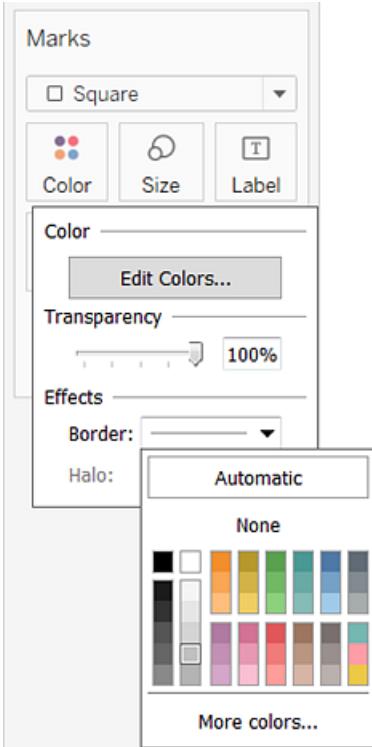
- On the **Marks** card, select **Square** as the mark type.
- Make the columns wider by pressing Ctrl + Right arrow (on a Mac, the combination is ⌘→). Hold down Ctrl (or ⌘) and continue pressing the Right arrow key until the headings for **Segment** are displayed in full:
- Increase the mark size by pressing Ctrl + Shift + B (on a Mac: ⌘B). Hold down Ctrl + Shift (⌘) and continue to press B until the squares are large enough.



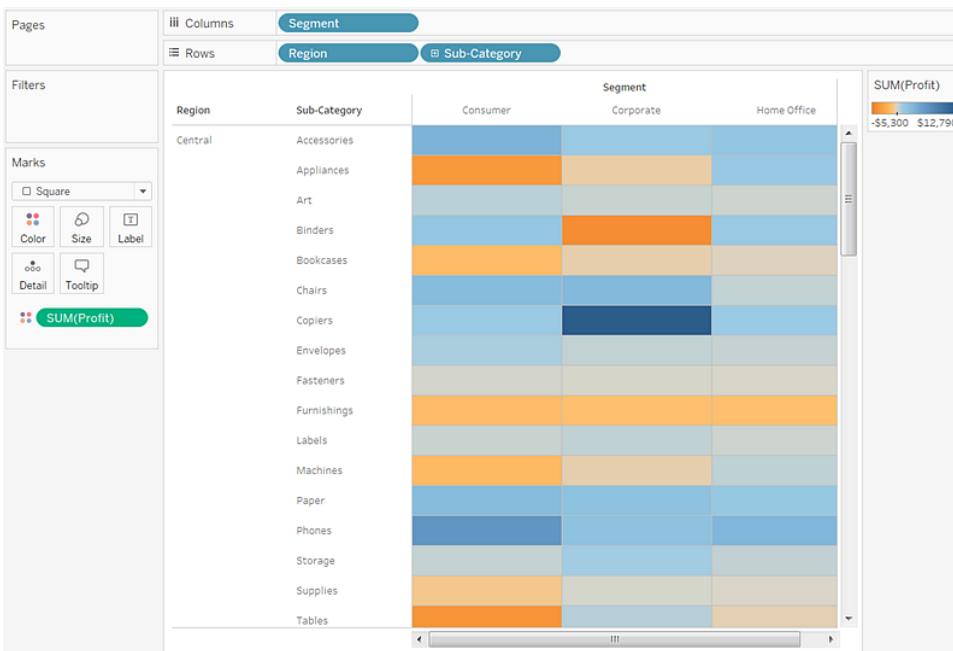
In this view, you can see data for only the Central region. Scroll down to see data for other regions.

In the Central region, copiers are shown to be the most profitable sub-category, and binders and appliances the least profitable.

6. Click **Color** on the **Marks** card to display configuration options. In the **Border** dropdown list, select a medium gray color for cell borders, as in the following image:



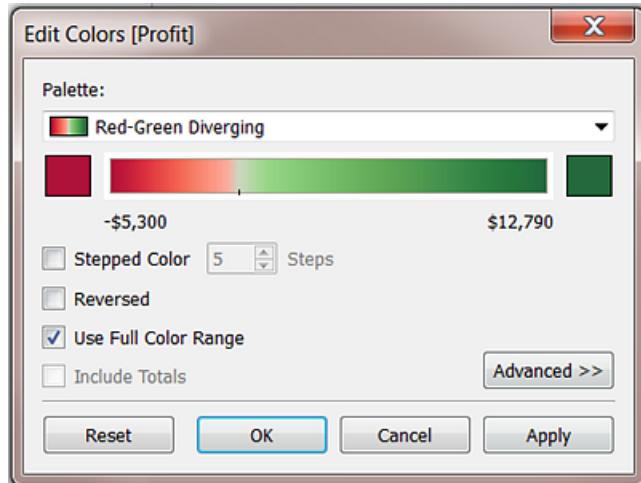
Now it's easier to see the individual cells in the view:



7. The default color palette is Orange-Blue Diverging. A Red-Green Diverging palette might be more appropriate for profit. To change the color palette and to make the colors more

distinct, do the following:

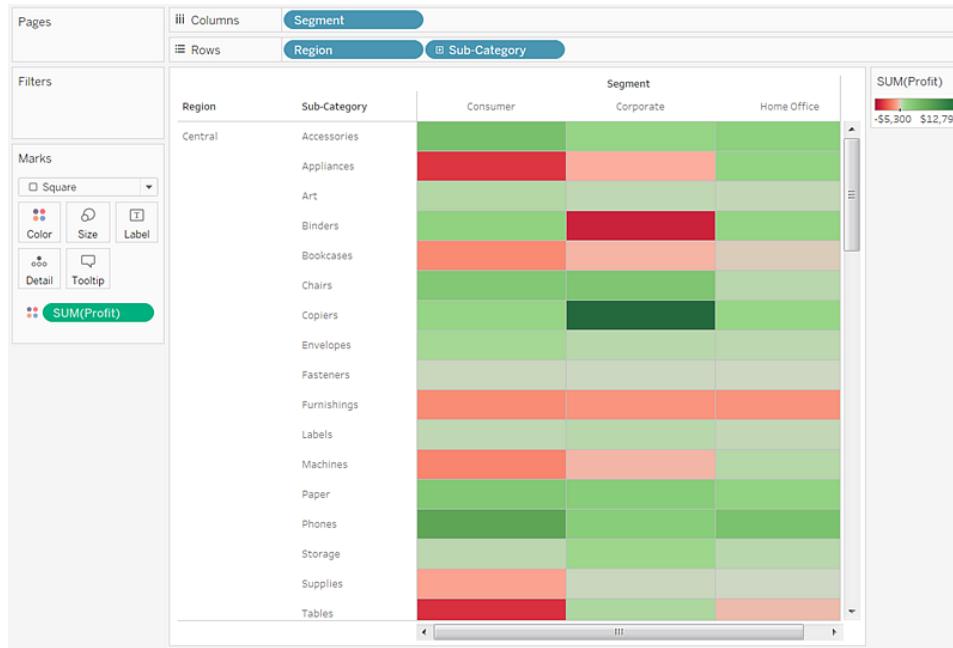
- Hover over the **SUM(Profit)** color legend, then click the drop-down arrow that appears and select **Edit Colors**.
- In the **Edit Colors** dialog box, in the **Palette** field, select **Red-Green Diverging** from the drop-down list.
- Select the **Use Full Color Range** check box and click **Apply** and then click **OK**.



When you select this option, Tableau assigns the starting number a full intensity and the ending number a full intensity. If the range is from -10 to 100, the color representing negative numbers changes in shade much more quickly than the color representing positive numbers.

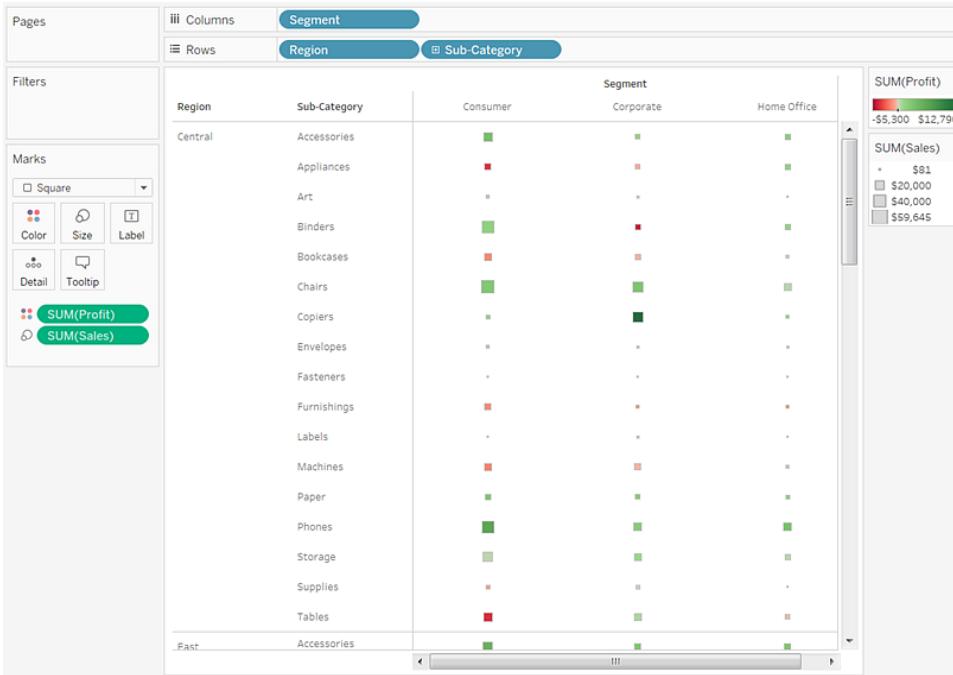
When you do not select **Use Full Color Range**, Tableau assigns the color intensity as if the range was from -100 to 100, so that the change in shade is the same on both sides of zero. The effect is to make the color contrasts in your view much more distinct.

For more information about color options, see [Color Properties](#) on page 192.

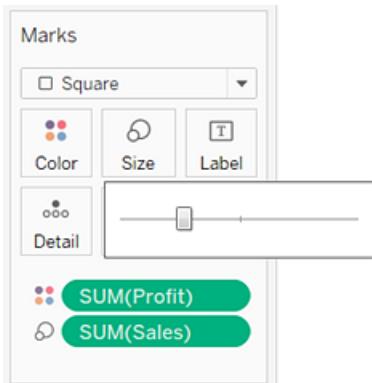


8. Drag the **Sales** measure to **Size** on the **Marks** card to control the size of the boxes by the Sales measure. You can compare absolute sales numbers (by size of the boxes) and profit (by color).

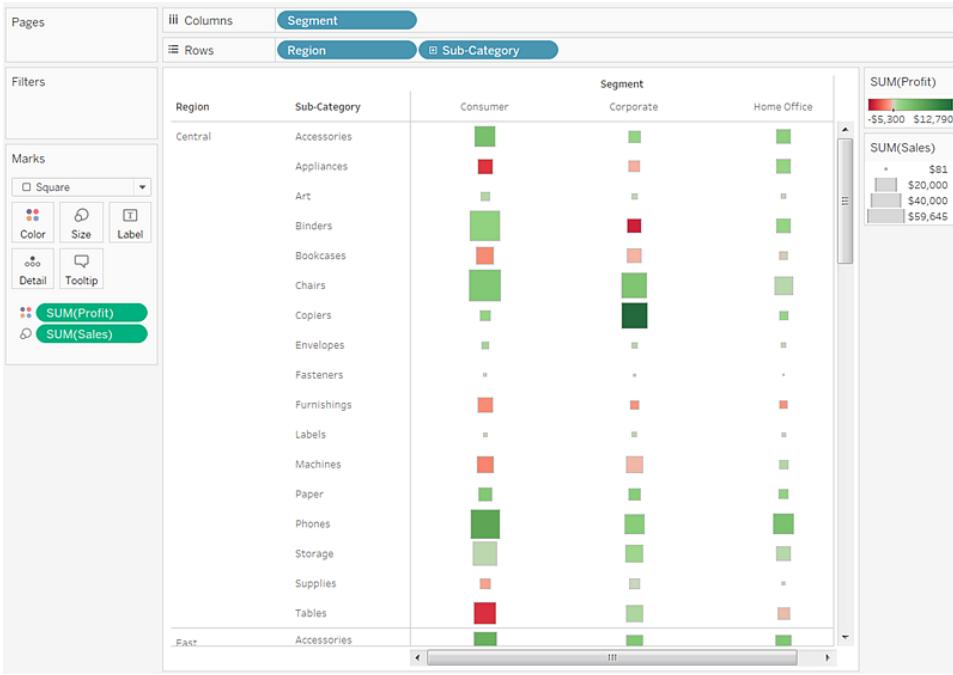
Initially, the marks look like this:



- To enlarge the marks, click **Size** on the **Marks** card to display a size slider:



- Drag the slider to the right until the boxes in the view are the optimal size. Now your view is complete:



11. Use the scroll bar along the right side of the view to examine the data for different regions.

Build a Histogram

A histogram is a chart that displays the shape of a distribution. A histogram looks like a bar chart but groups values for a continuous measure into ranges, or bins.

In Tableau you can create a histogram using **Show Me**.

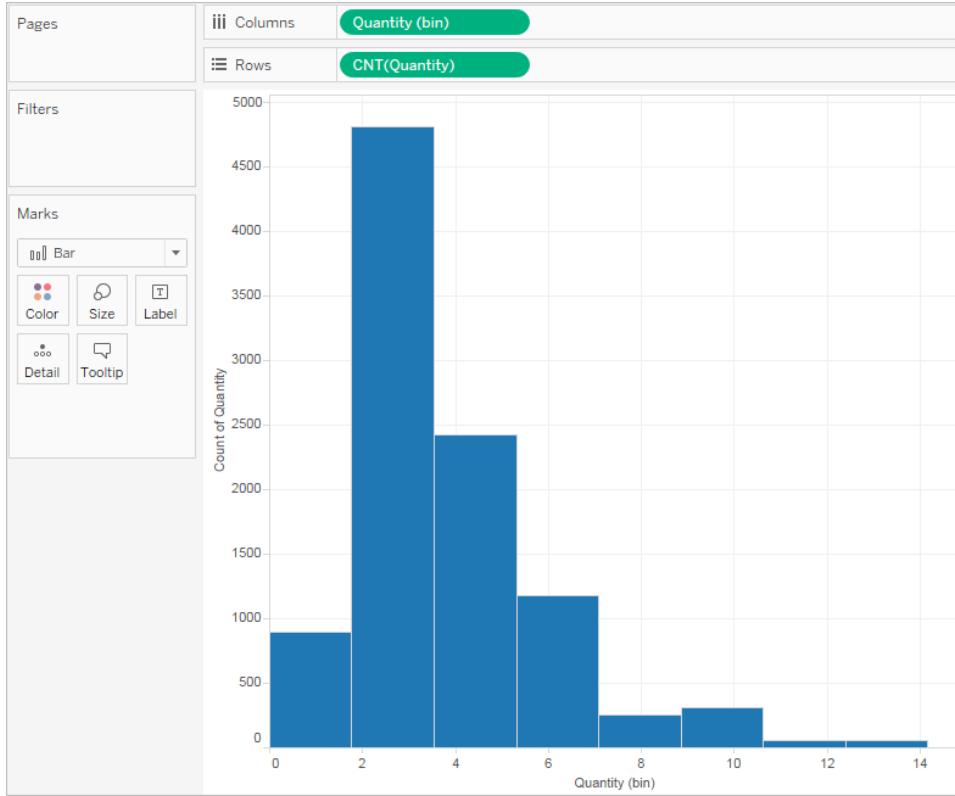
1. Connect to the **Sample - Superstore** data source.
2. Drag **Quantity** to **Columns**.
3. Click **Show Me** on the toolbar, then select the histogram chart type.



The histogram chart type is available in **Show Me** when the view contains a single measure and no dimensions.

Three things happen after you click the histogram icon in **Show Me**:

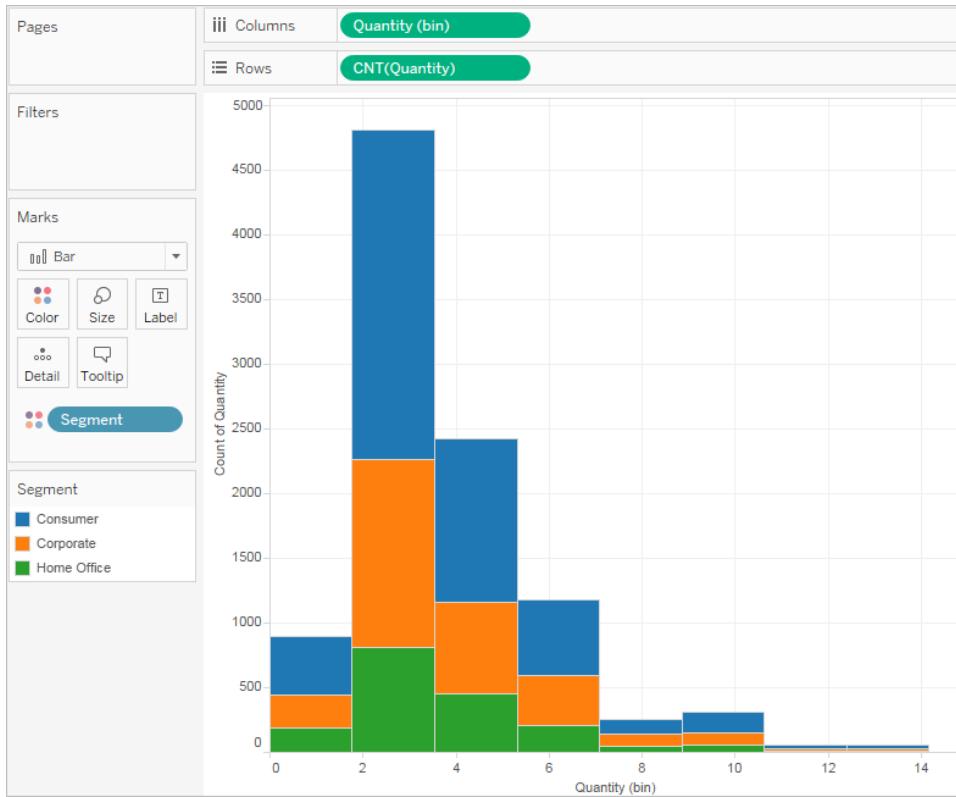
- The view changes to show vertical bars, with a continuous x-axis (1 – 14) and a continuous y-axis (0 – 5,000).
- The **Quantity** measure you placed on the **Columns** shelf, which had been aggregated as SUM, is replaced by a continuous **Quantity (bin)** dimension. (The green color of the field on the **Columns** shelf indicates that the field is continuous.)
- The **Quantity** measure moves to the **Rows** shelf and the aggregation changes from SUM to CNT (Count).



The **Quantity** measure captures the number of items in a particular order. The histogram shows that about 4,800 orders contained two items (the second bar), about 2,400 orders contained 4 items (the third bar), and so on.

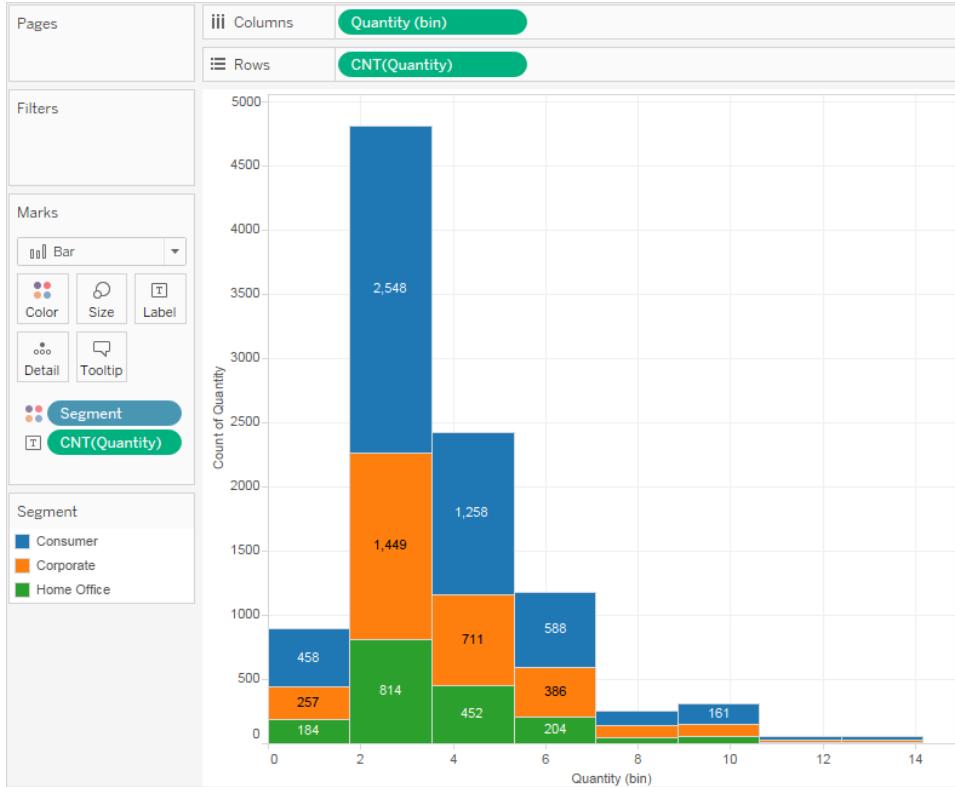
Let's take this view one step further and add **Segment** to **Color** to see if we can detect a relationship between the customer segment (consumer, corporate, or home office) and the quantity of items per order.

4. Drag **Segment** to **Color**.



The colors don't show a clear trend. Let's show the percentage of each bar that belongs to each segment.

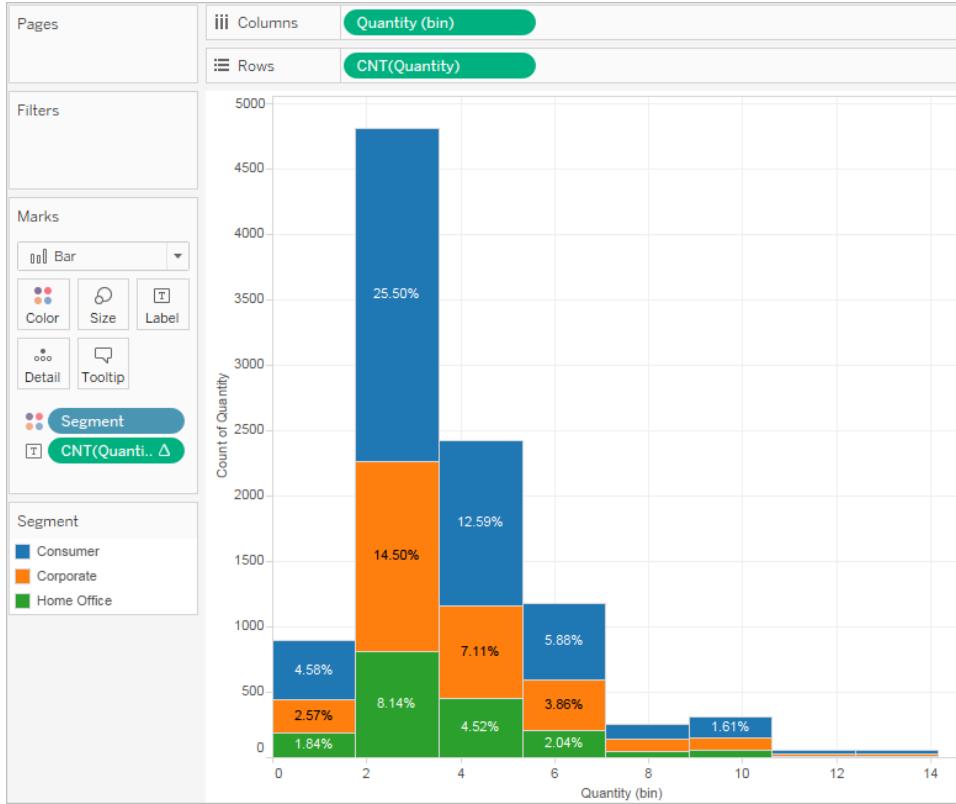
5. Hold down the Ctrl key and drag the **CNT(Quantity)** field from the **Rows** shelf to **Label**.



Holding down the Ctrl key copies the field to the new location without removing it from the original location.

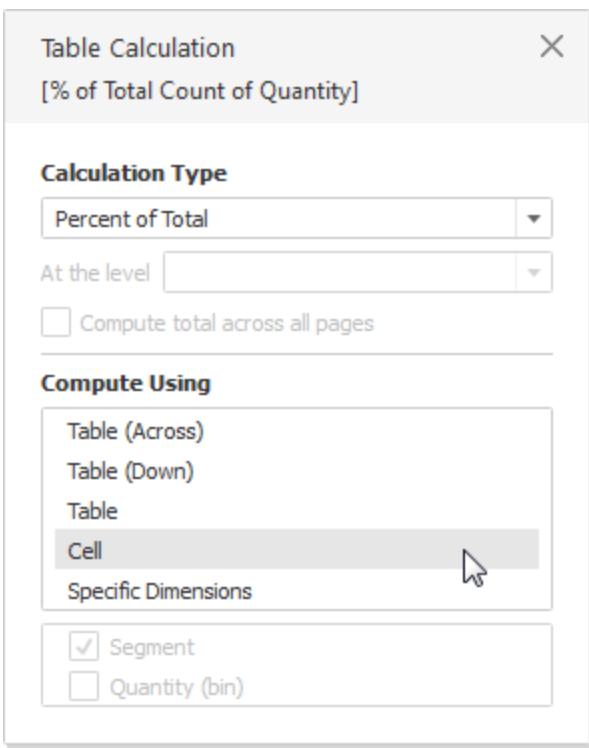
- Right-click (Control-click on a Mac) the **CNT(Quantity)** field on the **Marks** card and select **Quick Table Calculation > Percent of Total**.

Now each colored section of each bar shows its respective percentage of the total quantity:

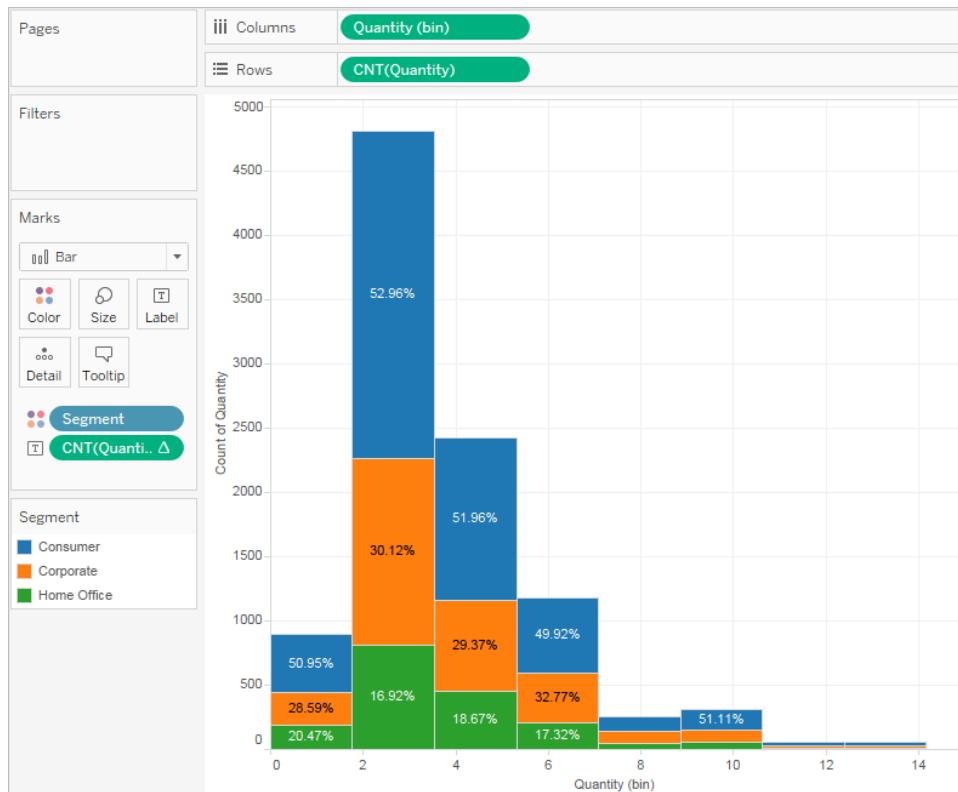


But we want the percentages to be on a per-bar basis.

7. Right-click the **CNT(Quantity)** field on the **Marks** card again and select **Edit Table Calculation**.
8. In the Table Calculation dialog box, change the value of the **Compute Using** field to **Cell**.



Now we have the view that we want:



There is still no evidence that the percentages by customer segment show any trend as the number of items in an order increases.

For information on how to create a binned dimension from a continuous measure, see [Create Bins from a Continuous Measure on page 597](#). If you create a binned dimension, place it on **Columns**, and then place the initial measure that you used as the basis for the binned dimension on **Rows**. Set the aggregation to **Count (Distinct)** to create a histogram.

Build a Gantt Chart

Use Gantt charts to show the duration of events or activities.

In a Gantt chart, each separate mark (usually a bar) shows a duration. For example, you might use a Gantt chart to display average delivery time for a range of products.

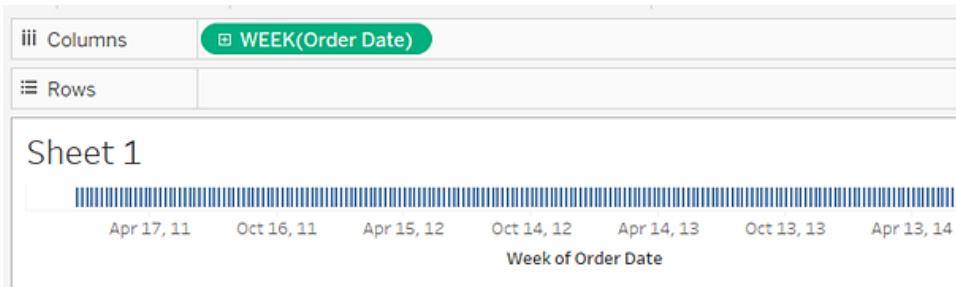
For more information about the Gantt bar mark type, see [Gantt Bar Mark on page 189](#).

To create a Gantt chart that shows how many days elapse on average between order date and ship date, follow these steps:

1. Connect to the **Sample - Superstore** data source.
2. Drag the **Order Date** dimension to **Columns**.
Tableau aggregates the dates by year and creates column headers with labels for the years.
3. On the **Columns** shelf, click the **Year (Order Date)** drop-down arrow, and then select **Week Number**.

The screenshot shows the context menu for the 'Order Date' dimension. The 'Year' option is selected, and the 'Week Number' and 'Day' options are also visible in the list.

The column headers change. Individual weeks are indicated by tick marks because there are 208 weeks in a four-year span—too many to show as labels in the view.



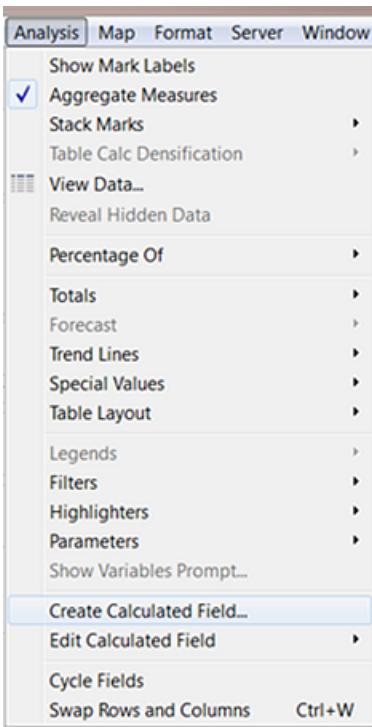
4. Drag the **Sub-Category** and **Ship Mode** dimensions to the **Rows** shelf. Drop **Ship Mode** to the right of **Sub-Category**.

This builds a two-level nested hierarchy of dimensions along the left axis.



Next, we'll size the marks according to the length of the interval between the order date and the ship date. To do this, create a calculated field to capture that interval.

5. In the toolbar menu, click **Analysis > Create Calculated Field**. You can also right-click (Control-click on Mac) any field in the **Data** pane and select **Create > Calculated Field**.



6. In the calculation dialog box, name your calculated field **OrderUntilShip**.
7. Clear any content that's in the **Formula** box by default.

8. In the **Formula** box, enter the following formula and then click **OK**:

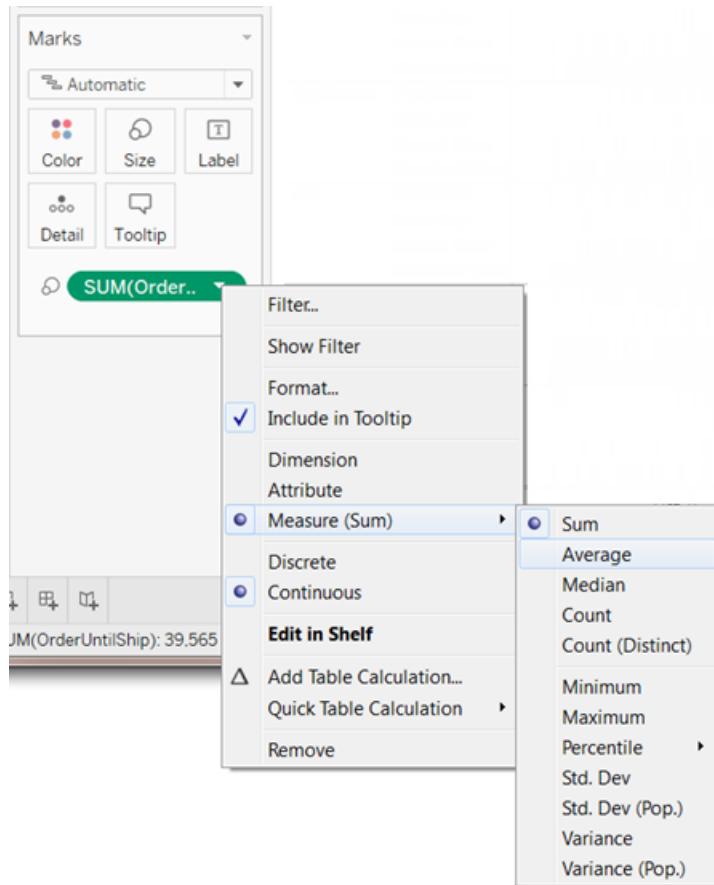
```
DATEDIFF('day',[Order Date],[Ship Date])
```

The formula creates a custom measure that captures the difference between the **Order Date** and **Ship Date** values, in days.

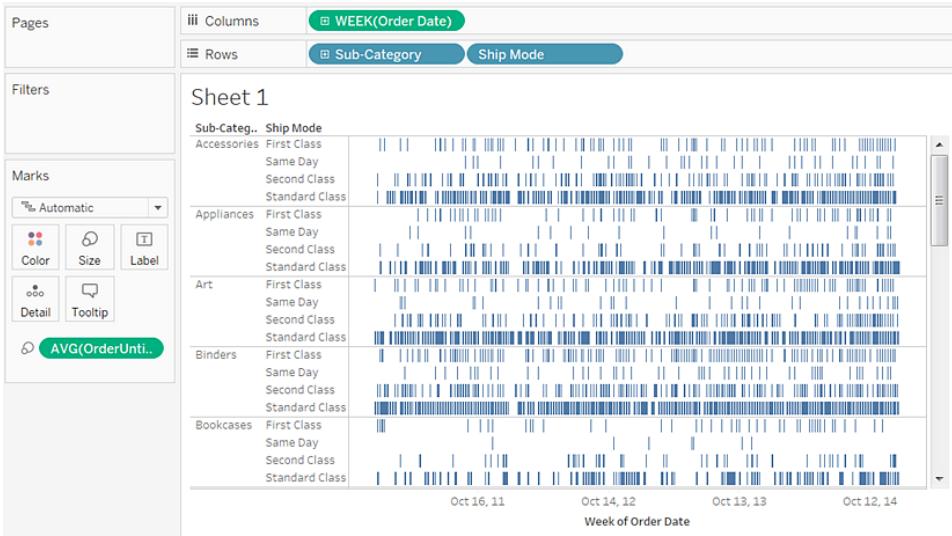
9. Drag the **OrderUntilShip** measure to **Size** on the **Marks** card.

The default aggregation for **OrderUntilShip** is **Sum**, but in this case it makes more sense to average the values.

10. Right-click (Control-click on Mac) the **SUM(OrderUntilShip)** field on the **Marks** card, and then select **Measure (Sum) > Average**.

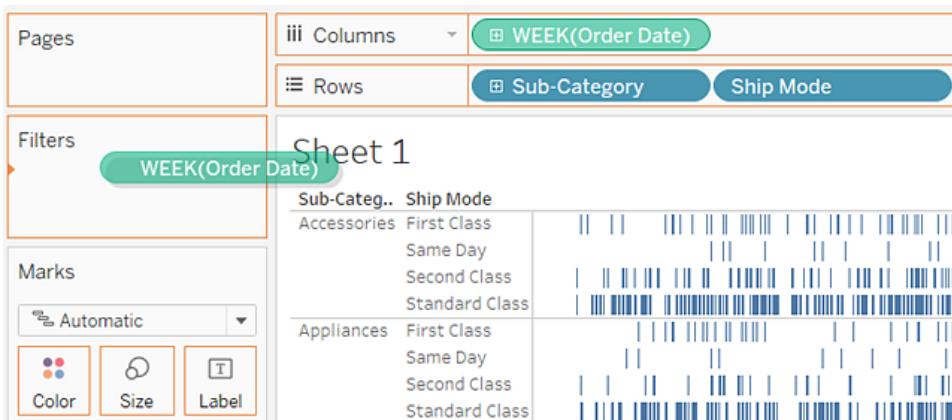


The view is coming along. But there are too many marks squeezed into the view.



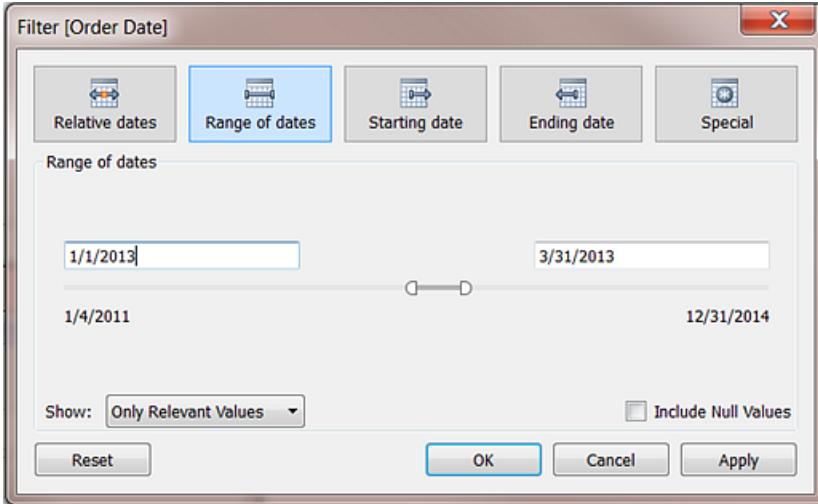
We can make our data more readable by filtering down to a smaller time window.

11. Hold down the Ctrl key (Option key on the Mac) and drag the **Week(Order Date)** field from the **Columns** shelf to the **Filter** shelf.



By holding down the Ctrl key (or the Option key), you tell Tableau that you want to copy the field to the new location, with whatever customizations you have added, without removing it from the old location.

12. In the Filter Field dialog box, select **Range of Dates** and then click **Next**.

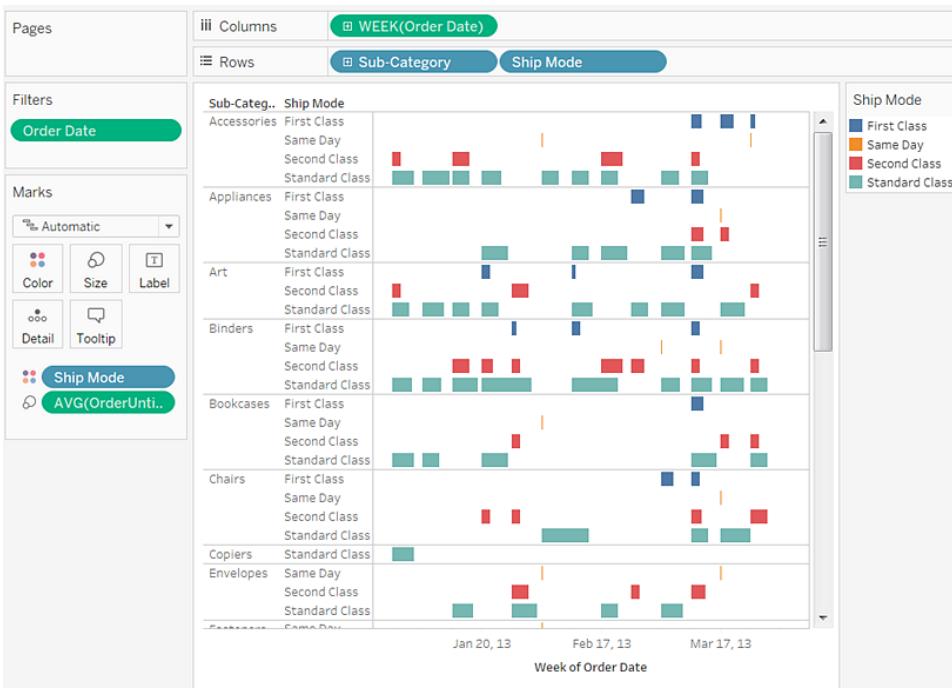


- Set the range to a three-month time interval, such as 1/1/2013 to 3/31/2013, and then click **OK**.

It can be difficult to get the exact date using the sliders—it's easier just to enter the numbers you want directly into the date boxes or use the calendar to select the dates.

- Drag the **Ship Mode** dimension to **Color** on the **Marks** card.

Now your view shows you all sorts of information about the lag between order times and ship times.



For example, you can see which ship modes are more prone to longer lag times, whether lag times vary by category, and whether lag times are consistent over time.

Note: If you publish this view to Tableau Server, you can include filters that let users interact with the view by varying the time window, or filtering out various sub-categories or ship modes. For more information, see [Publish Data Sources and Workbooks on page 1225](#).

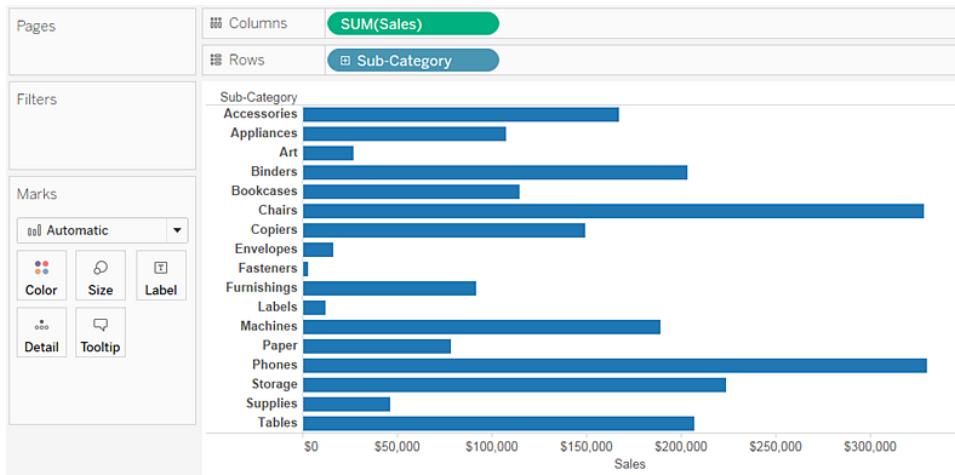
Build a Pie Chart

Use pie charts to show proportions.

To create a pie chart view that shows how different product categories contribute to total sales, follow these steps:

1. Connect to the **Sample - Superstore** data source.
2. Drag the **Sales** measure to **Columns**.
Tableau aggregates the measure as a sum.
3. Drag the **Sub-Category** dimension to **Rows**.

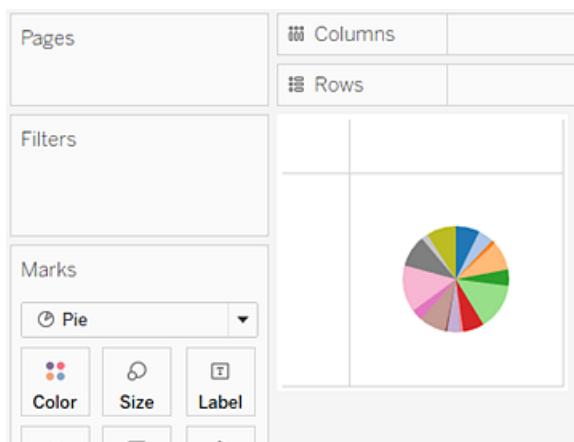
The default chart type is a bar chart.



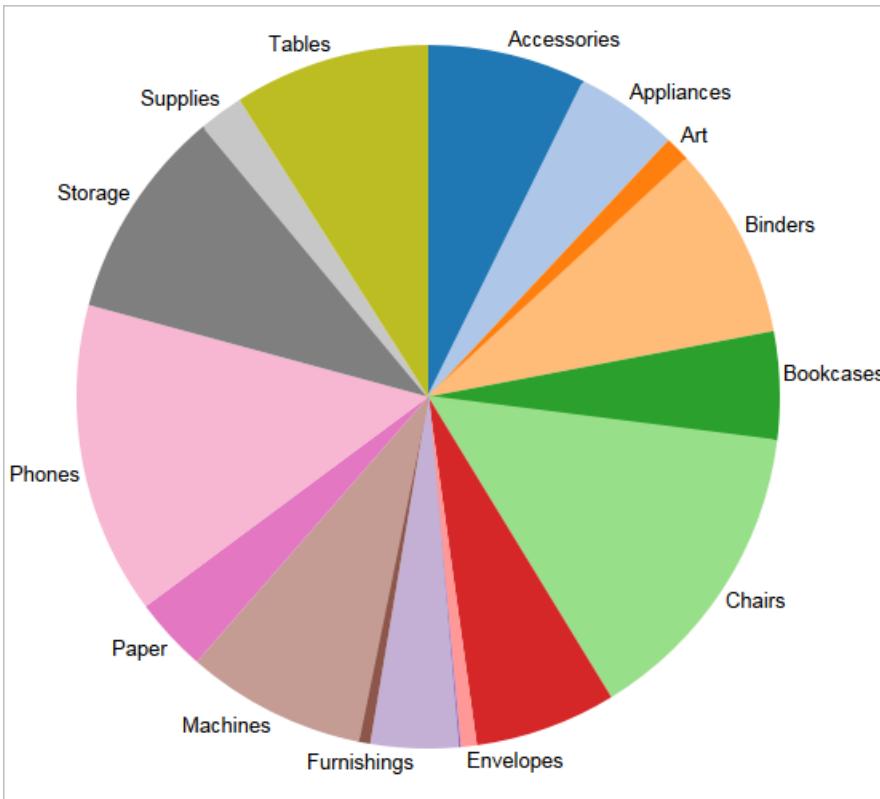
4. Click **Show Me** on the toolbar, then select the pie chart type.



The result is a rather small pie:



5. To make the chart bigger, hold down Ctrl + Shift (hold down ⌘ + ⌥ on a Mac) and press B several times.
6. To add labels, drag the **Sub-Category** dimension from the **Data** pane to **Label** on the **Marks** card.



7. If you don't see labels, press Ctrl + Shift + B (press ⌘ + ⌥ + B on a Mac) to make sure most of the individual labels are visible.

You can make a pie chart interactive in a dashboard. For more information, see [Actions and Dashboards on page 1139](#).

Note: Pie charts can also be used as a mark type in a visualization. For more information, see [Pie Mark on page 188](#).

Build a Treemap

Use treemaps to display data in nested rectangles. You use dimensions to define the structure of the treemap, and measures to define the size or color of the individual rectangles. Treemaps are a relatively simple data visualization that can provide insight in a visually attractive format.

To create a treemap that shows aggregated sales totals across a range of product categories, follow the steps below.

1. Connect to the **Sample - Superstore** data source.
2. Drag the **Sub-Category** dimension to **Columns**.

A horizontal axis appears, which shows product categories.

3. Drag the **Sales** measure to **Rows**.

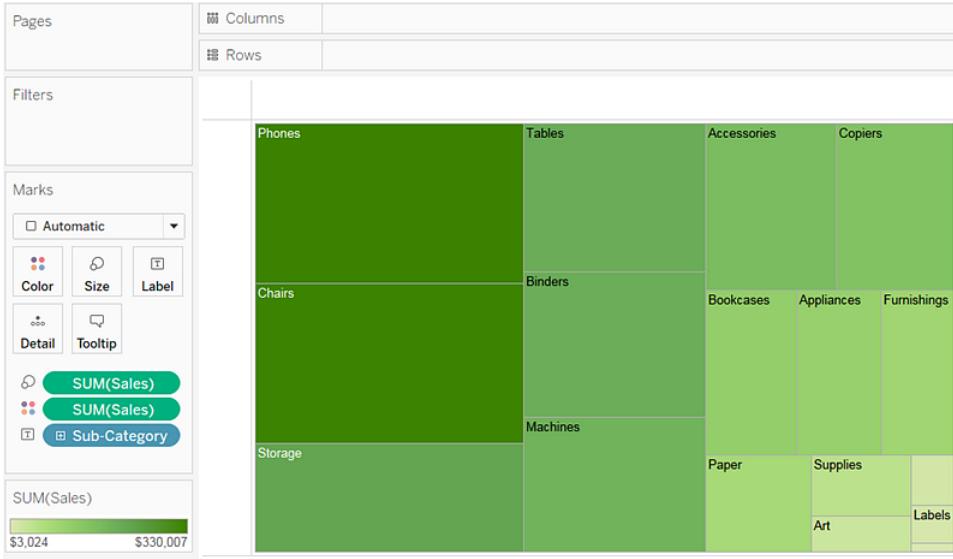
Tableau aggregates the measure as a sum and creates a vertical axis.

Tableau displays a bar chart—the default chart type when there is a dimension on the **Columns** shelf and a measure on the **Rows** shelf.

4. Click **Show Me** on the toolbar, then select the treemap chart type.



Tableau displays the following treemap:



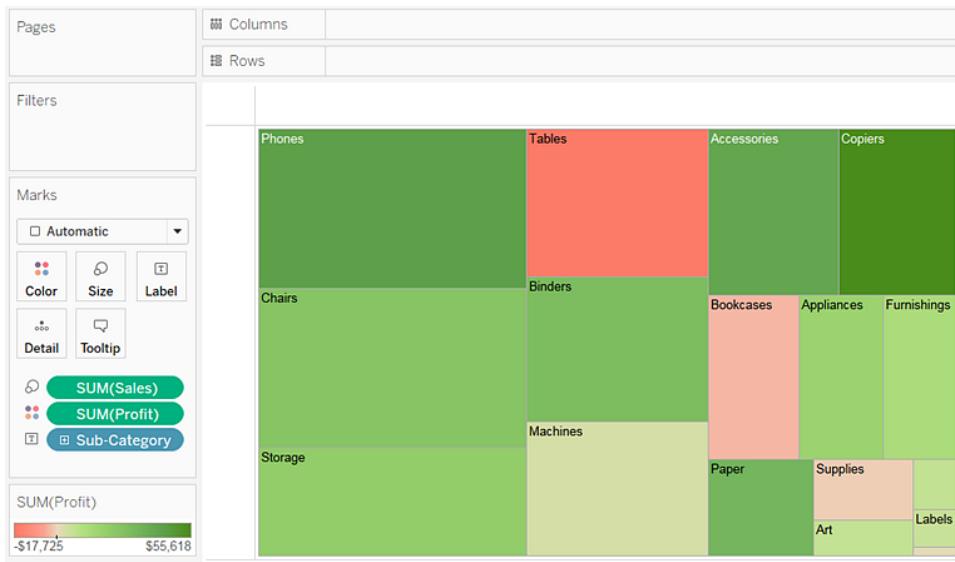
In this treemap, both the size of the rectangles and their color are determined by the value of **Sales**—the greater the sum of sales for each category, the darker and larger its box.

5. Drag the **Ship Mode** dimension to **Color** on the **Marks** card. In the resulting view, **Ship Mode** determines the color of the rectangles—and sorts them into four separate areas accordingly. **Sales** determines the size of the rectangles:



6. Try another option to modify the treemap: click the **Undo** button to remove **Ship Mode** from view.

7. Drag the **Profit** measure to **Color** on the **Marks** card. Now **Profit** determines the color of the rectangles, and **Sales** determines their size:



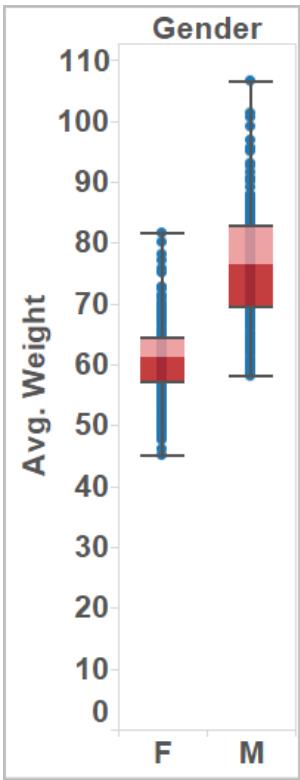
With treemaps, **Size** and **Color** are the crucial elements. You can place measures on **Size** and **Color**, but placing a measure anywhere else has no effect. Treemaps can accommodate any number of dimensions, including one or even two on **Color**. But beyond that, adding dimensions only breaks the map into an ever greater number of smaller rectangles. For information on how to place two dimensions on **Color** in a treemap, see [Example – Multiple Fields on Color on page 202](#).

Build a Box Plot

Use box plots, also known as box-and-whisker plots, to show the distribution of values along an axis.

Boxes indicate the middle 50 percent of the data (that is, the middle two quartiles of the data's distribution).

You can configure lines, called *whiskers*, to display all points within 1.5 times the interquartile range (in other words, all points within 1.5 times the width of the adjoining box), or all points at the maximum extent of the data, as shown in the following image:



To create a box plot that shows discounts by region and customer segment, follow these steps:

1. Connect to the **Sample - Superstore** data source.
2. Drag the **Segment** dimension to **Columns**.

The measure is aggregated as a sum and row headers appear, identifying three customer segments.

3. Drag the **Discount** measure to **Rows**.

Tableau creates a vertical axis and displays a bar chart—the default chart type when there is a dimension on the **Columns** shelf and a measure on the **Rows** shelf.

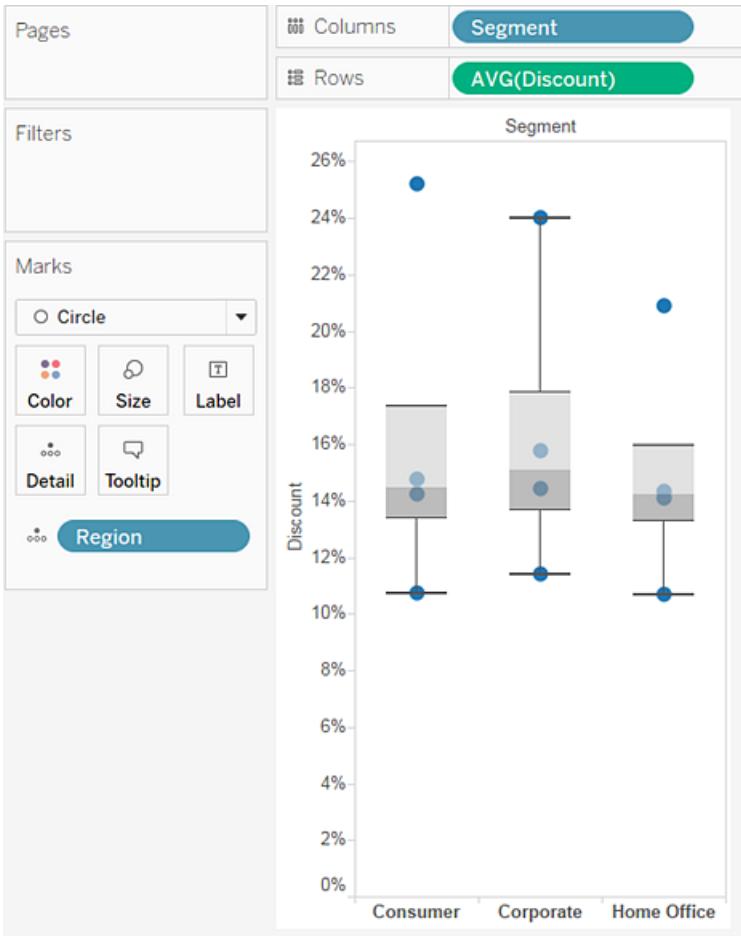
4. Drag the **Region** dimension to **Columns**, and drop it to the right of **Segment**.

Now you have a two-level hierarchy of dimensions from left to right in the view, with regions (listed along the bottom) nested within segments (listed across the top).

5. Click **Show Me** in the toolbar, then select the box-and-whisker plot chart type.

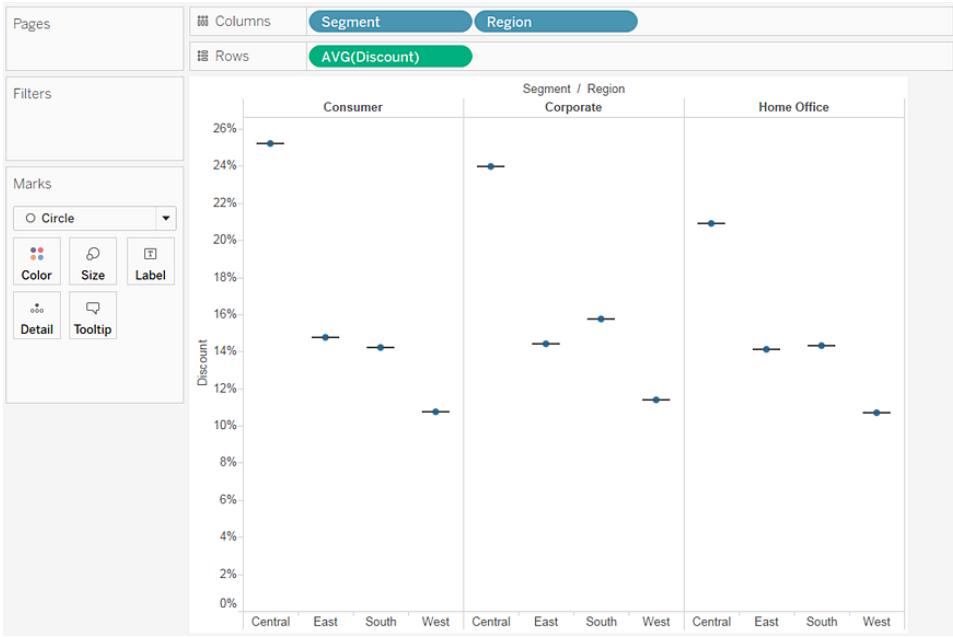


Tableau displays the following box plot:



Notice that there are only a few marks in each box plot. Also, Tableau reassigned **Region** from the **Columns** shelf to the **Marks** card. When you changed the chart type to a box plot, Tableau determined what the individual marks in the plot should represent. It determined that the marks should represent regions. We'll change that.

6. Drag **Region** from the **Marks** card back to **Columns**, to the right of **Segment**.



The horizontal lines are flattened box plots, which is what happens when box plots are based on a single mark.

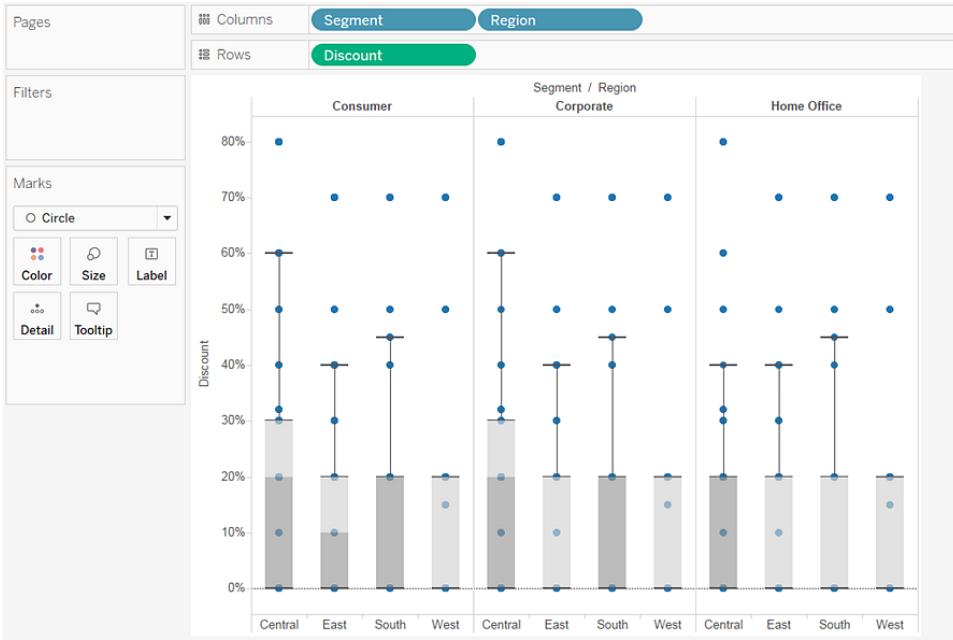
Box plots are intended to show a distribution of data, and that can be difficult when data is aggregated, as in the current view.

7. To disaggregate data, select **Analysis > Aggregate Measures**.

This command turns aggregation on or off, and because data is aggregated by default in Tableau, the first time you select this command, it disaggregates the data.

For more information, see [Disaggregating Data on page 338](#).

Now, instead of a single mark for each column in the view, you see a range of marks, one for each row in your data source.



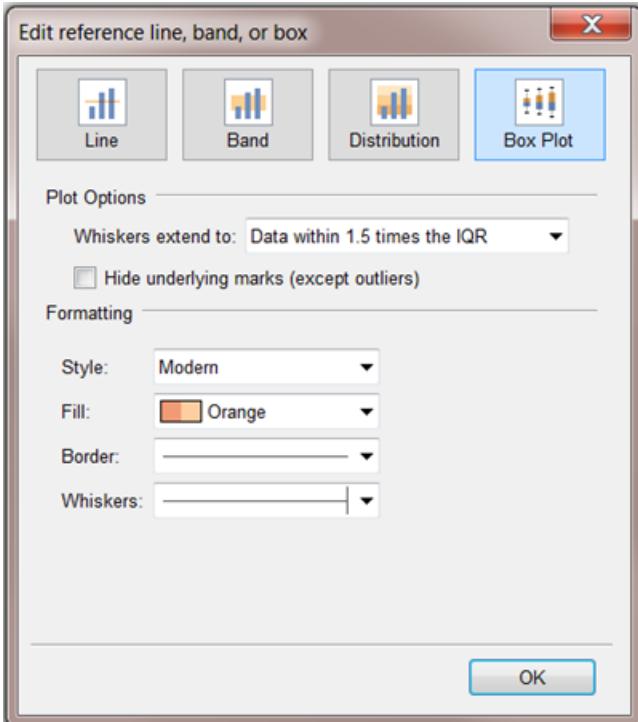
The view now shows the information we want to see. The remaining steps make the view more readable and appealing.

8. Click the **Swap** button to swap the axes:

The box plots now flow from left-to-right:



9. Right-click (control-click on Mac) the bottom axis and select **Edit Reference Line**.
10. In Edit Reference Line, Band, or Box dialog box, in the **Fill** drop-down list, select an interesting color scheme.



For more on these options, see [Adding Box Plots](#) on page 970.

Now your view is complete:



You can see that the discount was the same for all segments in the West. You can also see that the interquartile range (from the 25th percentile to the 75th percentile) for discount was greatest in the Central region for the consumer and corporate segments.

For more information about box plots, see [Reference Lines, Bands, Distributions, and Boxes](#) on page 948.

Build a Packed Bubble Chart

Use packed bubble charts to display data in a cluster of circles. Dimensions define the individual bubbles, and measures define the size and color of the individual circles.

To create a basic packed bubble chart that shows sales and profit information for different product categories, follow these steps:

1. Connect to the **Sample - Superstore** data source.

2. Drag the **Category** dimension to **Columns**.

A horizontal axis displays product categories.

3. Drag the **Sales** measure to **Rows**.

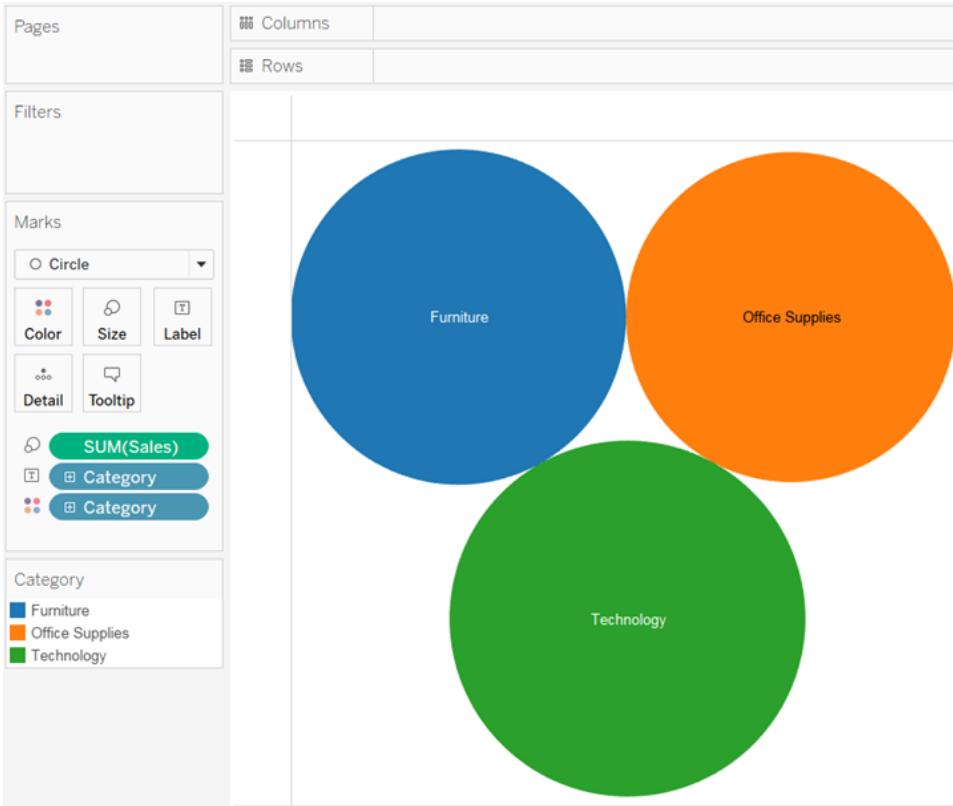
The measure is aggregated as a sum and a vertical axis appears.

Tableau displays a bar chart—the default chart type when there is a dimension on the **Columns** shelf and a measure on the **Rows** shelf.

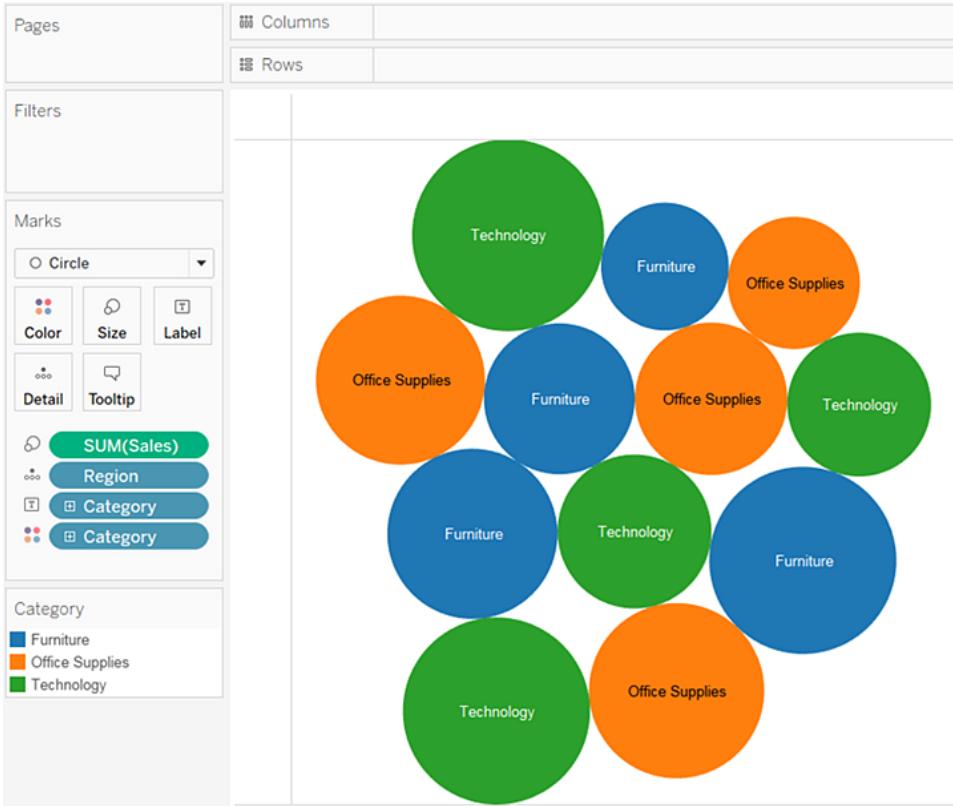
4. Click **Show Me** on the toolbar, then select the packed bubbles chart type.



Tableau displays the following packed bubble chart:

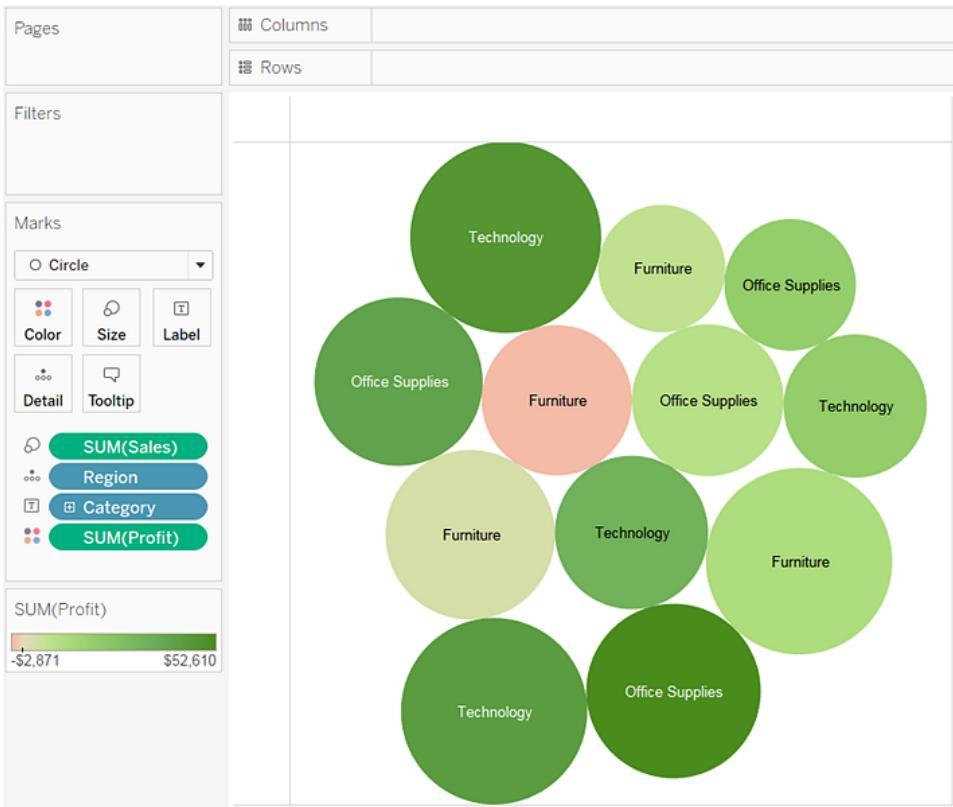


5. Drag **Region** to **Detail** on the **Marks** card to include more bubbles in the view.

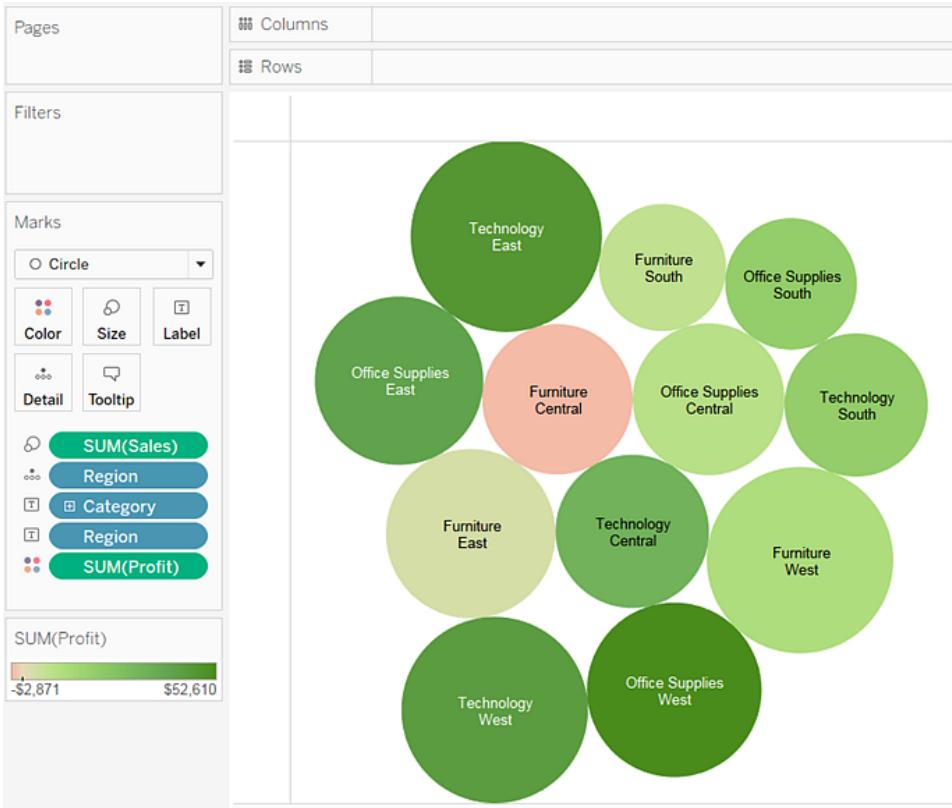


Next we'll add another layer of information to the view.

6. Drag **Profit** to **Color** on the **Marks** card:



- Drag **Region** to **Label** on the **Marks** card to clarify what each bubble represents.



The size of the bubbles shows the sales for different combinations of region and category. The color of the bubbles shows the profit (the darker the green, the greater the profit).

For information about formatting mark labels, see [Show and Hide Mark Labels](#) on page 549.

To further develop this view, you might edit the colors for **Profit** to show negative profit in a different color, or create a calculated field to shows profit divided by sales (that is, profit margin) and then drop that on **Color** instead of absolute profit. For more information, see [Color Properties](#) on page 192.

Build a Map View

Use map views to display and analyze geographic data. To create map views, add geographic fields to the view, and then add measures or continuous dimensions to the **Marks** card.

The first exercise in this topic shows you how to create a map view and customize your map appearance. The second exercise in this topic shows you how to create a map view with pie marks.

Build a map view

This exercise demonstrates how to create a custom map view that shows the amount of sales

by U.S. city.

1. Connect to the **Sample – Superstore** data source.
2. In the **Data** pane, double-click the **City** dimension. Tableau adds **Country**, **State**, and **City** to **Detail** on the **Marks** card and creates a map view.

Because Tableau assigns the **Country**, **State**, **City**, and **Postcode** dimensions a geographic role, a map view is created when you double-click one of these dimensions:

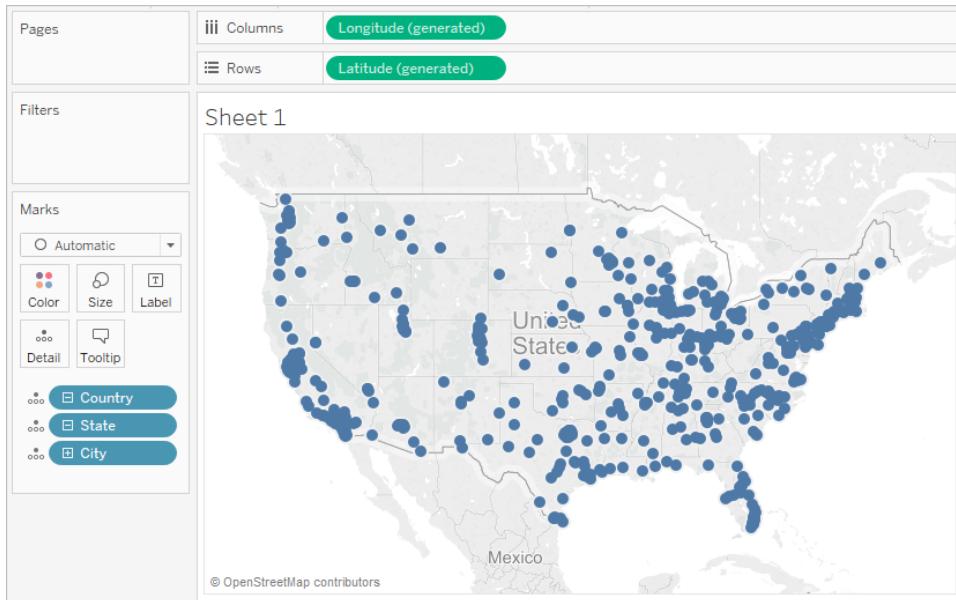
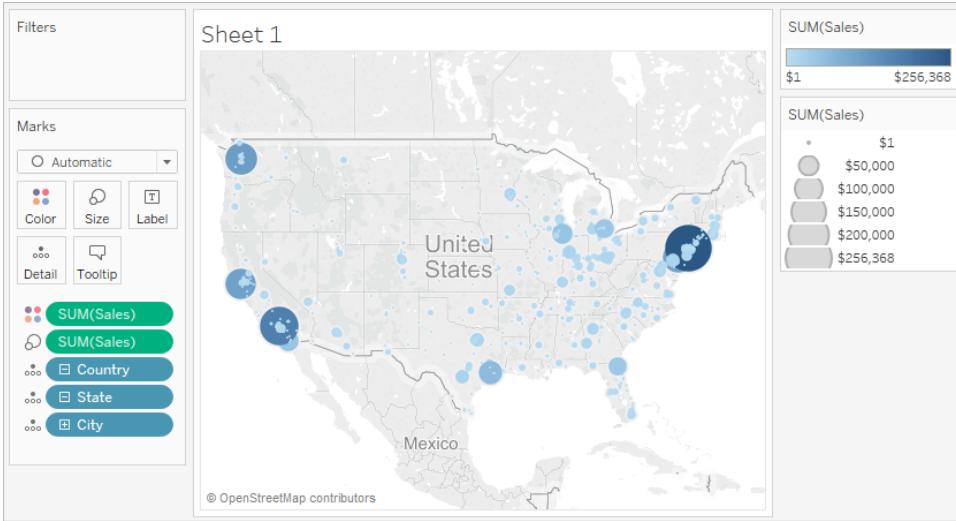
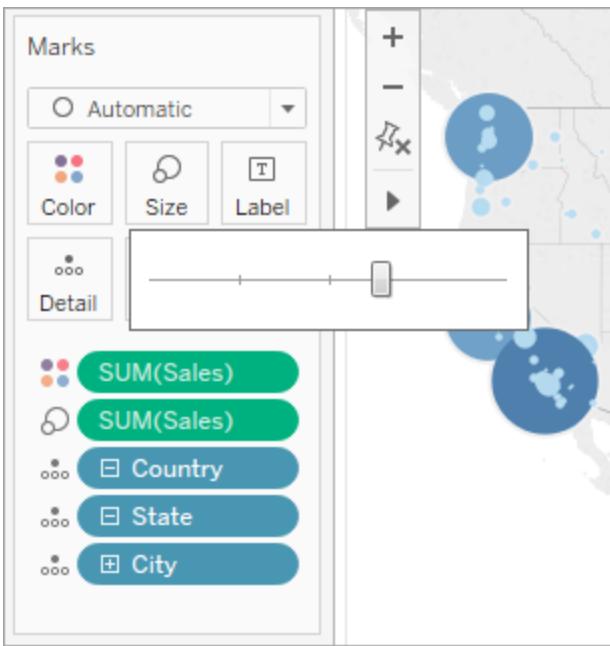


Tableau looks for geographic fields in any data source. For more information on how Tableau interprets geographic data in data sources, see [Prepare Your Geographic Fields](#) on page 735.

3. From **Measures**, drag **Sales** to **Color** on the **Marks** card. The marks in the view color according to the amount of sales in each city.
4. From **Measures**, drag **Sales** to **Size** on the **Marks** card. The marks in the view resize according to sales in each city.



- To adjust the size of the marks in the view, click **Size** on the **Marks** card, and then drag the slider to the right.

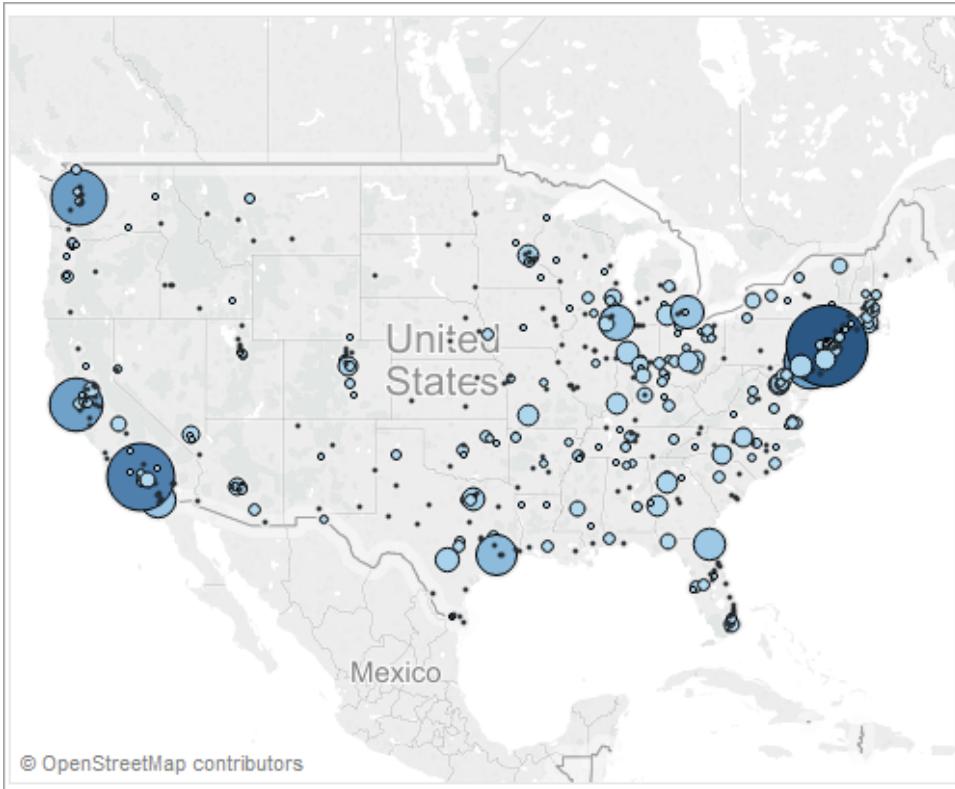


The marks in the view are now larger and easier to inspect, though they overlap. Overlapping marks make it difficult to see a difference in the size of some marks.

To make the map view easier to analyze, you can add a border to the marks in the view.

- To add a border to the marks in the view, on the **Marks** card, click **Color**, and then click the **Border** drop-down list to select a color.

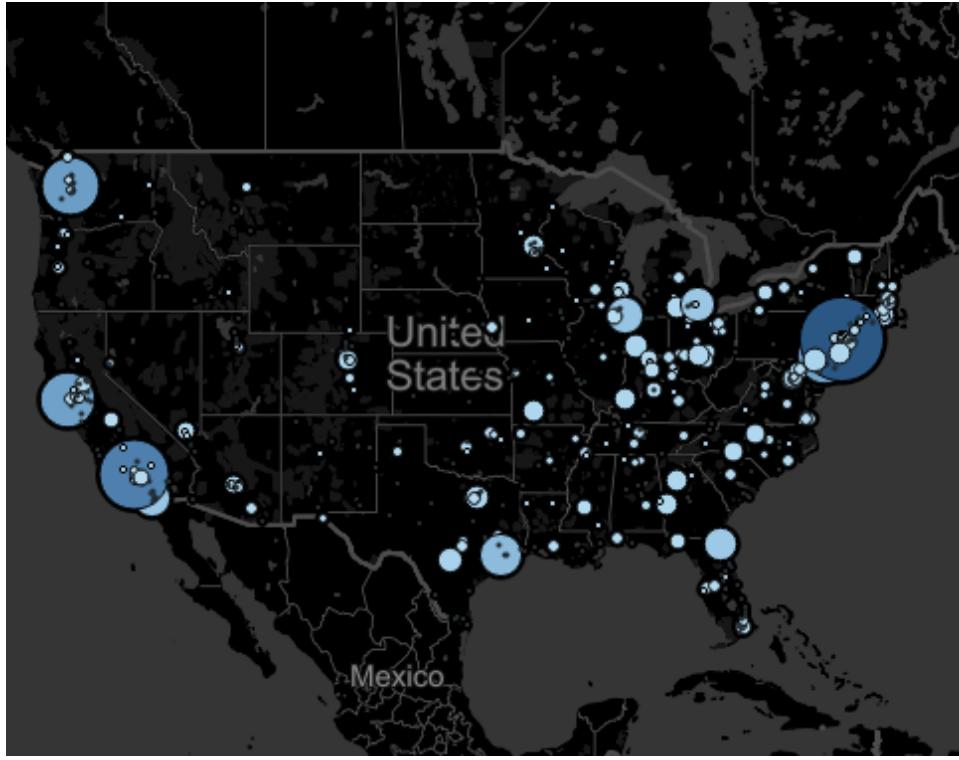
The marks in the view update to display the selected border:



After you plot your data on a map view, you can configure the background map style, map layers, and data layers using the **Map Layers** pane.

7. To open the **Map Layers** pane, select **Map > Map Layers**. In the **Map Layers** pane, click the **Style** drop-down list, and then select **Dark**.

The background map changes from light to dark.

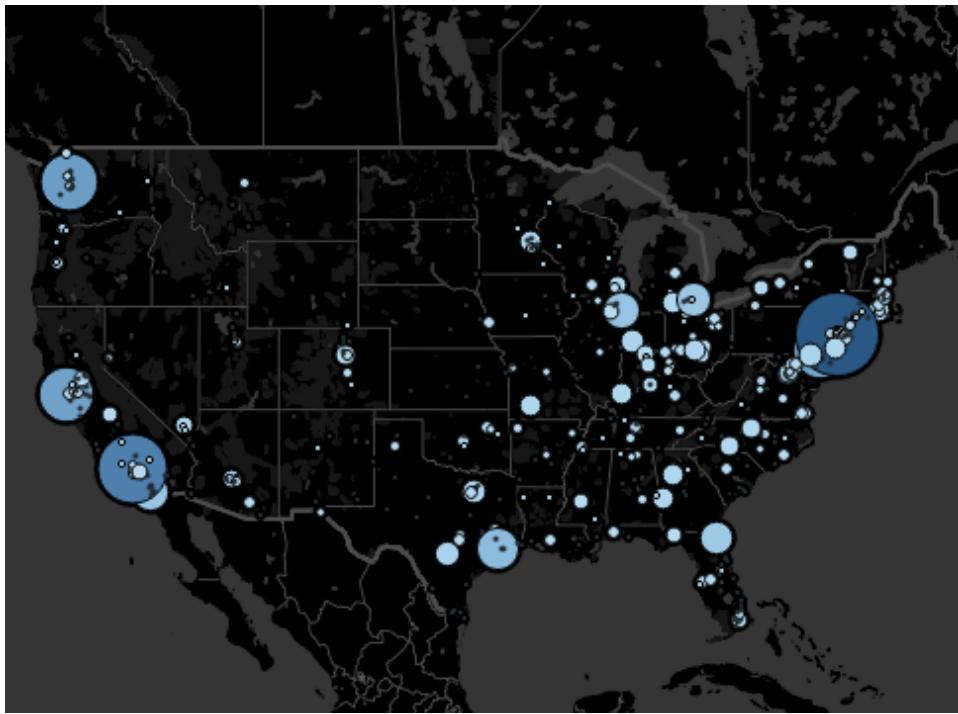


You can now add or subtract map layers to provide context to your data, or to simplify the view.

8. In the **Map Layers** pane, clear the **Country/Region Names** and **State/Province Names** map layers to simplify the map view.

The country/region and state/province names are removed from the view.

Your view is now complete. Your map view shows total sales for each city in your data source across the United States. The size and the color of each mark corresponds to the total sales for that city.



For more information about creating map views, see [Build and Use Maps](#) on page 702.

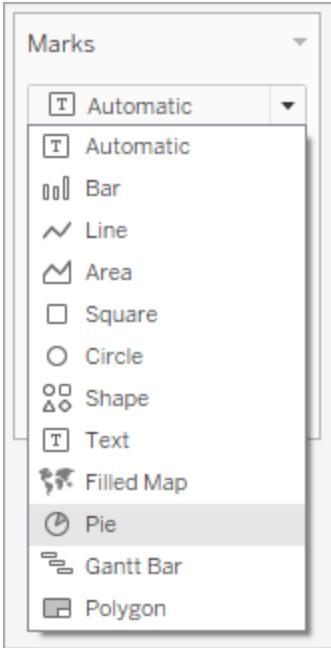
Build a map with pie marks

In this exercise, you will create a map with a pie mark on each state. The overall size of the mark represents sales, and the size of each section represents the sales for each ship mode (First Class, Same Day, Second Class, Standard Class).

1. Connect to the **Sample - Superstore** data source.
2. In the **Data** pane, double-click the **State** dimension.

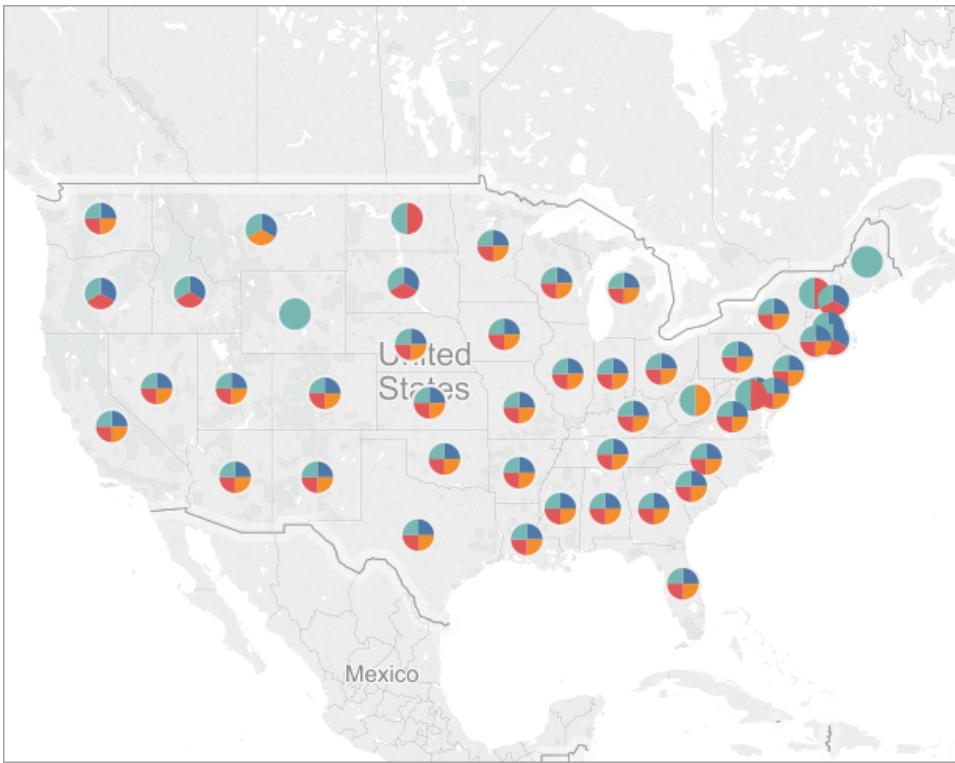
Tableau adds **Country** and **State** to **Detail** on the **Marks** card.
3. From **Measures**, drag **Sales** to **Size** on the **Marks** card.

The marks in the view resize according to sales in each city.
4. On the **Marks** card, click the **Mark Type** drop-down list, and then select **Pie**.



5. From **Dimensions**, drag **Ship Mode** to **Color** on the **Marks** card.
6. To make the marks in the view larger, on the **Marks** card, click **Size**, and then drag the slider to the right.

Your view is now complete. Your map view shows pie chart marks for each state. The size of each pie corresponds to the total sales for that state, and the slices of the pie show what proportions of the total were provided by each of the four available ship modes.



The Tableau Environment

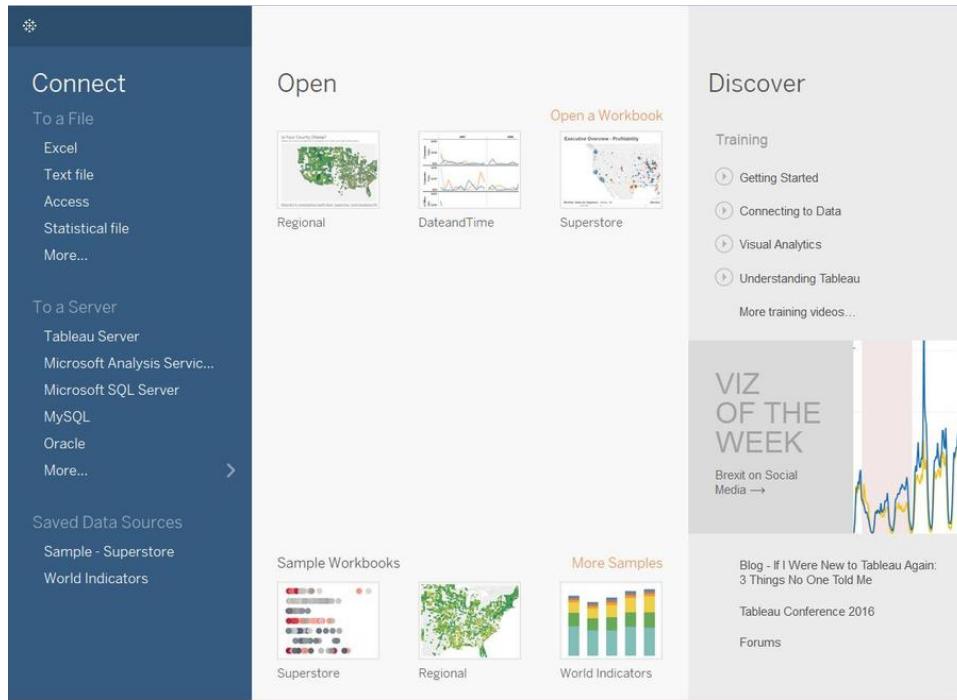
This section provides information on the various panes, shelves, pages, icons, and other elements that comprise the Tableau Desktop environment.

Start Page

The start page in Tableau Desktop is a central location from which you can do the following:

- Connect to your data
- Open your most recently used workbooks, and
- Discover and explore content produced by the Tableau community.

The start page consists of three panes: **Connect**, **Open**, and **Discover**.



Connect

Connect to data and open saved data sources.

This screenshot focuses on the 'Connect' pane of the Tableau desktop application. It lists various data sources under 'To a File' (Excel, Text file, Access, Statistical file, More...) and 'To a Server' (Tableau Server, Microsoft Analysis Service..., Microsoft SQL Server, MySQL, Oracle, More...). Below these are 'Saved Data Sources' (Sample - Superstore, World Indicators). The main area displays a large list of available connectors, including Tableau Server, Kognitio, SAP Sybase IQ, Actian Matrix, MapR Hadoop Hive, Snowflake, Actian Vector, MarkLogic, Spark SQL, Amazon Aurora, MemSQL, Splunk, Amazon EMR, Microsoft Analysis Services, Teradata, Amazon Redshift, Microsoft PowerPivot, Teradata OLAP Connect, Aster Database, Microsoft SQL Server, Web Data Connector, Cisco Information Server, MonetDB, Other Databases (ODBC), Cloudera Hadoop, MySQL, DataStax Enterprise, OData, EXASolution, Oracle, Firebird, Oracle Essbase, Google Analytics, Pivotal Greenplum Database, Google BigQuery, PostgreSQL, Google Cloud SQL, Presto, Google Sheets, Progress OpenEdge, Hortonworks Hadoop Hive, QuickBooks Online, HP Vertica, Salesforce, IBM BigInsights, SAP HANA, IBM DB2, SAP NetWeaver Business Warehouse, IBM PDA (Netezza), SAP Sybase ASE, and SAP Sybase IQ.

On the **Connect** pane, you can do the following:

- **Connect to data:** Under **To a File**, connect to data stored in Microsoft Excel files, text files, Access files, Tableau data extract files, and statistical files, such as SAS, SPSS, and R. Under **To a Server**, connect to data stored in databases like Microsoft SQL Server or Oracle. The server names listed in this section change based on which servers you connect to and how often.
- **Open saved data sources:** Quickly open data sources that you have previously saved to your My Tableau Repository directory. Also, Tableau provides sample saved data sources that you can use to explore Tableau Desktop functionality. To follow along with examples in the Tableau Desktop documentation, you'll usually use the **Sample – Superstore** data source.

Open

Open recent workbooks, pin workbooks to the start page, and explore sample workbooks.

The screenshot shows the 'Open' pane in Tableau. At the top left, it says 'Open'. Below that, there's a heading 'Open a Workbook' followed by three cards: 'Regional', 'DateandTime', and 'Superstore'. Each card has a thumbnail image of its respective dashboard. At the bottom left, there's a heading 'Sample Workbooks' followed by two cards: 'Superstore' and 'Regional'. Each card has a thumbnail image of its respective dashboard. On the right side, there's a heading 'More Samples' followed by one card: 'World Indicators'. This card also has a thumbnail image of its respective dashboard.

On the **Open** pane, you can do the following:

- **Open recently opened workbooks:** When you open Tableau Desktop for the first time, this pane is empty. As you create and save new workbooks, the most recently opened workbooks appear here. Click the workbook thumbnail to open a workbook, or if you don't see a workbook thumbnail, click the **Open a Workbook** link to find other workbooks that are saved to your computer.
- **Pin workbooks:** You can pin workbooks to the start page by clicking the pin icon that appears in the top-left corner of the workbook thumbnail. Pinned workbooks always appear on the start page, even if they weren't opened recently. To remove a recently opened or pinned workbook, hover over the workbook thumbnail, and then click the "x" that appears. The workbook thumbnail is removed immediately but will show again with your most recently used workbooks the next time you open Tableau Desktop.
- **Explore sample workbooks:** Open and explore sample workbooks.

Discover

See popular views in Tableau Public, read blog posts and news about Tableau, and find training videos and tutorials to help you get started.

Discover

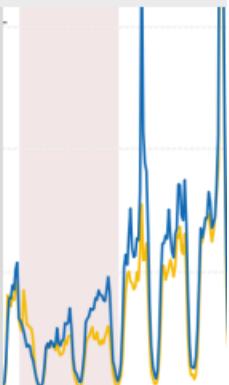
Training

- ▶ Getting Started
- ▶ Connecting to Data
- ▶ Visual Analytics
- ▶ Understanding Tableau

[More training videos...](#)

VIZ OF THE WEEK

Brexit on Social
Media →



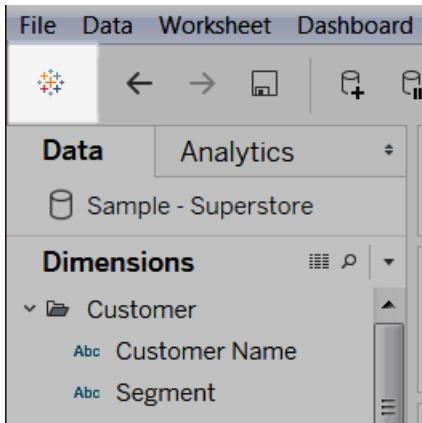
[Blog - If I Were New to Tableau Again:
3 Things No One Told Me](#)

[Tableau Conference 2016](#)

[Forums](#)

Navigating back to the start page

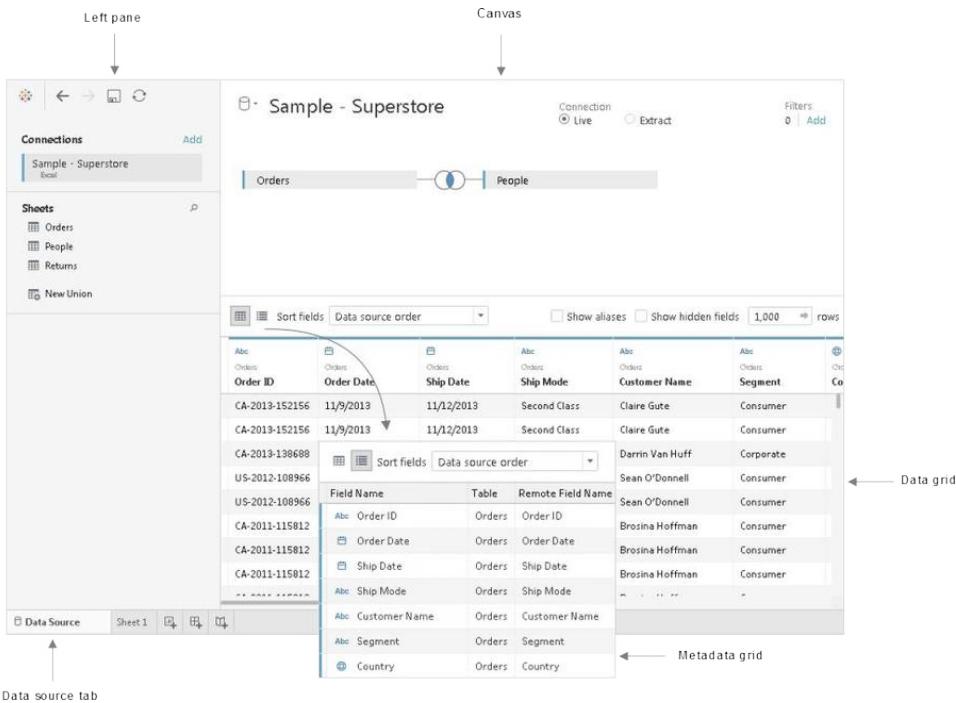
When Tableau Desktop is already open, you can navigate back to the start page by clicking the Tableau icon in the upper-left corner of the Tableau Desktop workspace.



Data Source Page

Before or during your analysis, you may want to make changes to the Tableau data source. You can do that on the data source page. Tableau takes you to the data source page after you establish the initial connection to your data. You can also access the data source page by clicking the **Data Source** tab from any location in the workbook.

Although the look of the page and the options available vary depending on the type of data that you are connected to, the data source page generally consists of four main areas: left pane, canvas, data grid, and metadata grid.



Left pane

The left pane of the data source page displays details about your data, that is the data that Tableau Desktop is connected to. For file-based data, the left pane might display the file name and the worksheets in the file. For relational-based data, the left pane might display the server, the database or schema, and the tables in the database.

You can also use the left pane to add more connections to the data source to create cross-database joins.

The left pane does not display for cube (multidimensional) data.

Canvas

When connected to most relational and file-based data, you can drag one or more tables to the canvas area to set up your Tableau data source.

When connected to cube data, the top of the data source page displays the available catalog or queries and cubes to select from to set up your Tableau data source.

Data grid

Use the data grid to review the fields and the first 1,000 rows of the data contained in the Tableau data source. You can also use the data grid to make general modifications to the Tableau data source like sorting or hiding fields; renaming fields or resetting field names; creating calculations; changing the column or row sort; or adding aliases. For the Web Data Connector, file- and relational-based data sources in Extract mode, you can see extract data in the grid, including extract only calculations.

To select multiple fields in the grid, click a column, and then drag the mouse to select more columns. To select all fields, click the area in the upper-left corner of the grid like in the example, below.

The diagram illustrates two types of data grids in Tableau:

- Top Grid (Cube/Multidimensional Data):** Shows a single row (Row ID 1) highlighted with a red border. The columns are labeled Row ID, Order ID, and Order Date. The data rows are: Row ID 1, Order ID CA-2013-152156, Order Date 11/9/2013; Row ID 2, Order ID CA-2013-152156, Order Date 11/9/2013.
- Bottom Grid (Extract-Only Data):** Shows three rows of data. The first row (Row ID 1) is highlighted with a dotted border. The columns are labeled Row ID, Order ID, and Order Date. The data rows are: Row ID 1, Order ID CA-2013-152156, Order Date 11/9/2013; Row ID 2, Order ID CA-2013-152156, Order Date 11/9/2013; Row ID 3, Order ID CA-2013-138688, Order Date 6/13/2013.

The grid does not display for cube (multidimensional) data.

Metadata grid

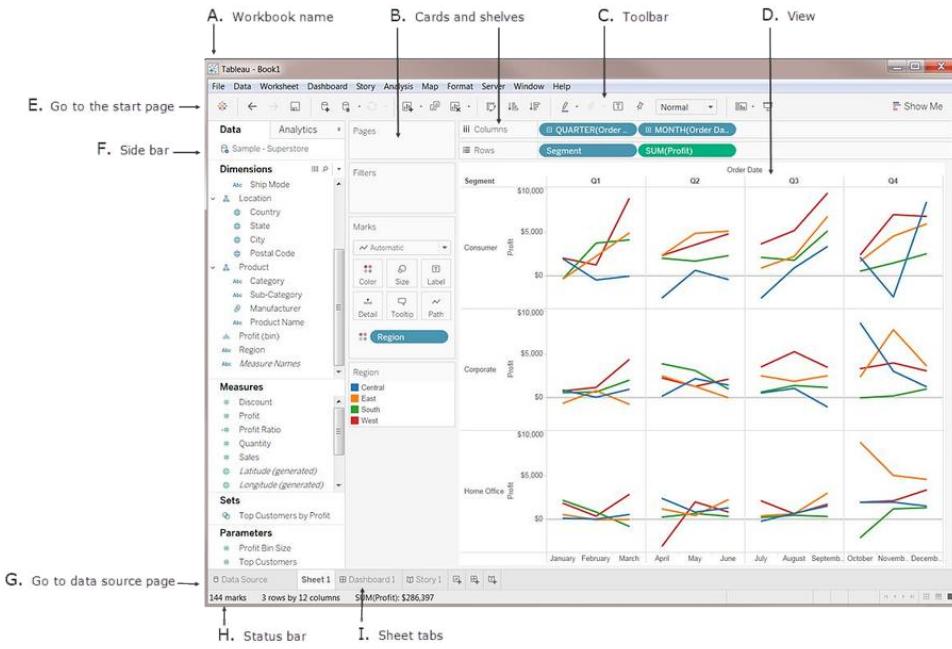
Depending on the type of data that you are connected to, click the metadata grid button to navigate to the metadata grid. The metadata grid displays the fields in your data source as rows so that you can quickly examine the structure of your Tableau data source and perform routine management tasks, such as renaming fields or hiding multiple fields at once.

When connected to cube or some extract only data, the metadata grid displays by default.

The Tableau Workspace

The Tableau workspace consists of menus, a toolbar, the Data pane, cards and shelves, and one or more sheets. Sheets can be worksheets, dashboards, or stories.

Worksheets contain shelves and cards that you can drag and drop data fields on to build views.



A. Workbook name.

B. Cards and shelves - Drag fields to the cards and shelves in the workspace to add data to your view.

C. Toolbar - Use the toolbar to access commands and analysis and navigation tools.

D. View - This is the workspace where you create your data visualizations.

E. Go to the start page. For more information, see [Start Page on page 125](#).

F. Side Bar. The side bar provides two panes: the **Data** pane and the **Analytics** pane. For more information, see [The Side Bar below](#).

G. Go to the data source page. For more information, see [Data Source Page on page 130](#).

H. Status bar - Displays information about the current view.

I. Sheet tabs - Tabs represent each sheet in your workbook. This can include worksheets, dashboards and stories. For more information, see [Sheets on page 242](#).

The Side Bar

The Side Bar provides two panes: the Data pane and the Analytics pane.

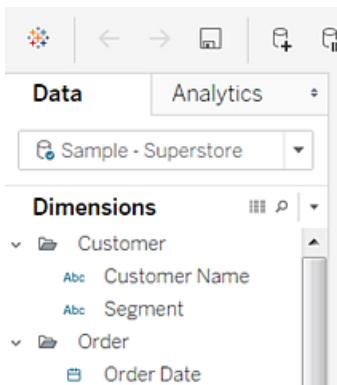
You can hide and show the Side Bar by clicking the icon in the upper right corner.

To open the Side Bar again, look for that same icon in the lower left corner of the Tableau window:



Data Pane

Tableau displays your data fields in the **Data** pane on the left side of the workspace. You can toggle between the **Data** and **Analytics** panes:



Current data sources appear at the top of the **Data** pane.

Below the data sources in the **Data** pane are the fields that are available in the currently selected data source.

Perform common tasks in the Data pane

To view all the data sources used in the workbook, click the drop-down arrow next to the data source that is displayed.

To view a context menu for the data source, click **Data** in the top menu and then click on the data source in the menu list. For more information, see [Edit Data Sources on page 458](#).

To search for fields in the Data pane, click the magnifying class icon and then type in the text box.

To see the underlying data, click the **View Data** icon at the top of the **Data** pane. For more information , see [View Data on page 573](#)

To convert a measure to a dimension, drag the measure and drop it into the **Dimensions** area in the **Data** pane.

View Data: Sample - Superstore

9,994 rows Show aliases

Category	City	Country	Customer Name
Office Supplies	Houston	United States	Darren Powers
Office Supplies	Naperville	United States	Phillina Ober
Office Supplies	Naperville	United States	Phillina Ober
Office Supplies	Naperville	United States	Phillina Ober
Office Supplies	Philadelphia	United States	Mick Brown
Office Supplies	Athens	United States	Jack O'Briant
Office Supplies	Los Angeles	United States	Lycoris Saunders
Furniture	Henderson	United States	Maria Etezadi
Office Supplies	Henderson	United States	Maria Etezadi

How the Data pane is organized

The **Data** pane is organized into several areas:

- **Dimensions** - fields that contain data like text or dates. This is called *category data*.
- **Measures** - fields that contain numbers. *Numerical data* can be aggregated. For more information, see [Dimensions and Measures on page 268](#).
- **Sets** - subsets of data that you define, for example if you want to create a report that shows only data you specify. For more information, see [Sets on page 512](#).
- **Parameters** - dynamic placeholders that can replace constant values in calculated fields and filters so that view designers can give data consumers choices about what they see. For more information, see [Parameters on page 989](#).

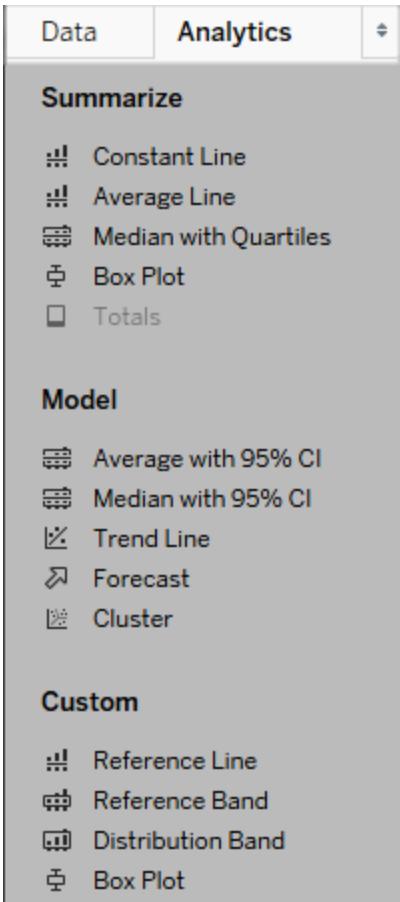
By default, Tableau treats all relational fields containing numbers as measures. However, you might decide that some of these fields should be treated as dimensions. For example, a field containing ages might be categorized as a measure by default in Tableau because it contains numeric data. However, if you want to look at each individual age rather than an axis, you can convert the **Age** field to a dimension.

Drag the **Age** measure and drop it into the Dimensions area in the Data pane. When you drag the **Age** field to the **Rows** or **Columns** shelf, it creates column headers (1, 2, 3, etc.) instead of a continuous axis.

For information on converting measures to dimensions—or dimensions to measures—see [Dimensions and Measures on page 268](#).

Analytics Pane

The Analytics pane provides quick and easy access to common analytic features in Tableau. You can drag reference lines, box plots, trend lines forecasts, and other items into your view from the **Analytics** pane, which appears on the left side of the workspace. Toggle between the **Data** pane and the **Analytics** pane by clicking one of the tabs at the top of the Side Bar:

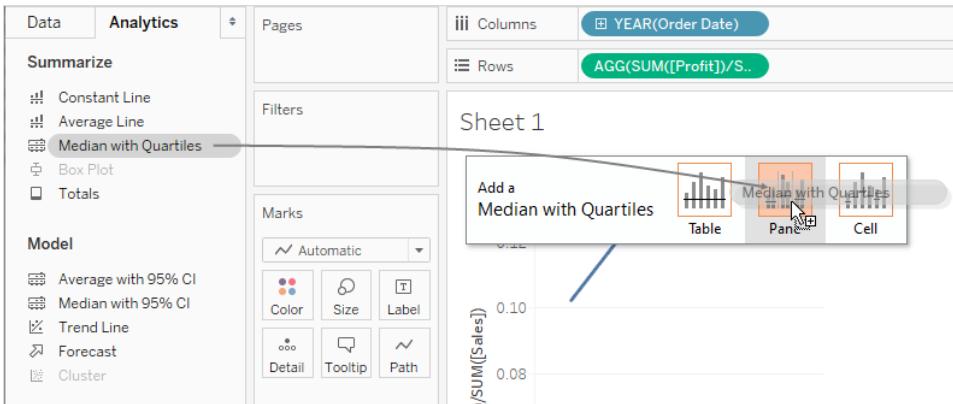


There isn't anything you can add to a view from the **Analytics** pane that you couldn't add by some other means—for example, reference lines and bands are available when you edit an axis, and trend lines and forecasts are available from the Analysis menu. The **Analytics** pane just makes the process easier by offering drag-and-drop access for the various options.

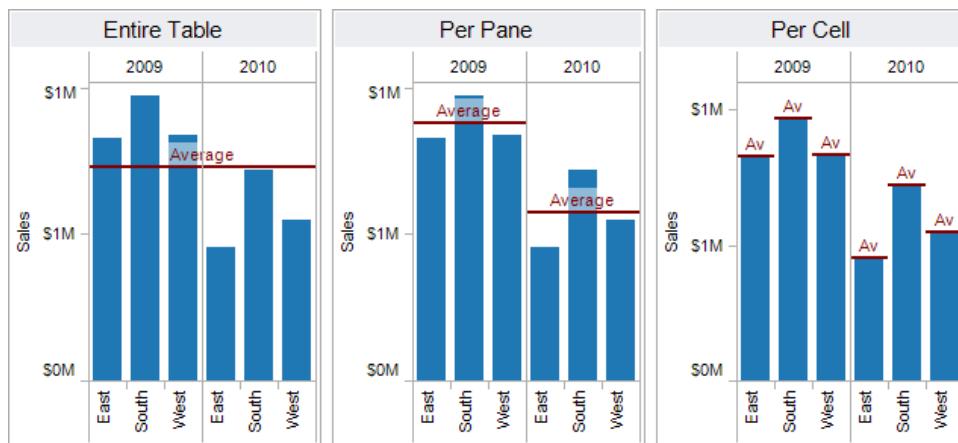
Add Analytics Pane Item

To add an item from the **Analytics** pane, drag it into the view. When you drag an item from the **Analytics** pane, Tableau shows the possible destinations for that item. The range of choices varies depending on the type of item and the current view.

In a simple case, the drop target area would offer these three options:



The terms **Table**, **Pane** and **Cell** define the scope for the item:

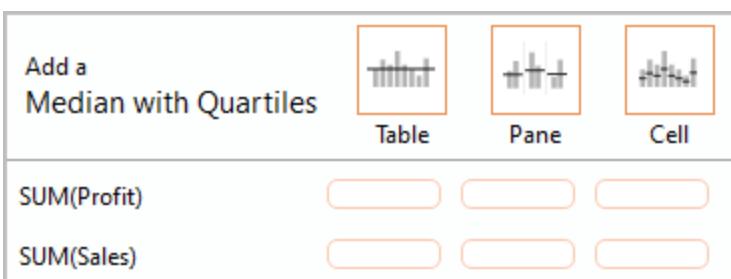


Adds a reference line to the entire table across all panes.

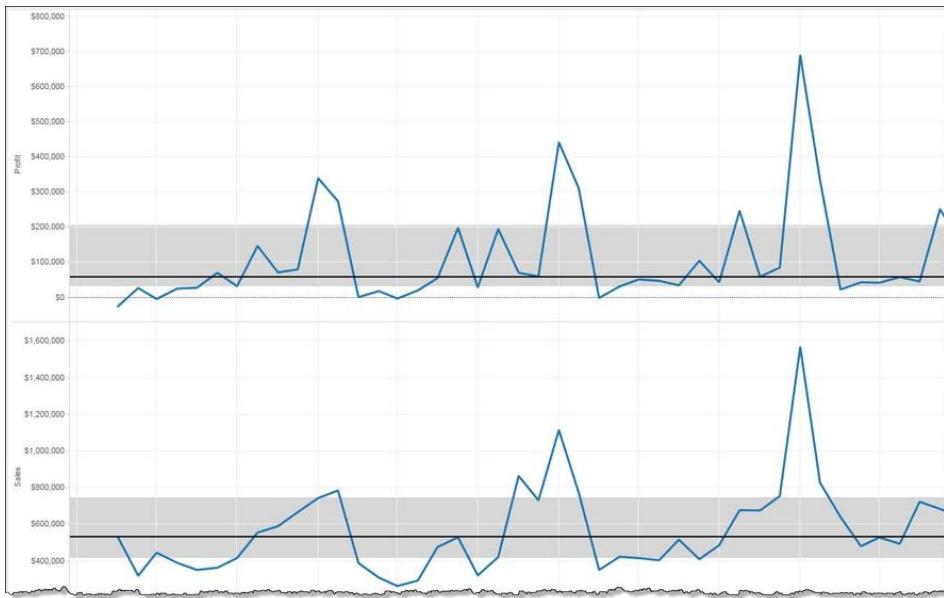
Adds a reference line on a per pane basis. Computed reference lines are recalculated for each pane in the view.

Adds a reference line within each cell. Computed reference lines are recalculated for each cell in the view.

For a more complicated view—for example, if the view contained a line chart with multiple or dual axes—Tableau would show you a drop target area that looked like this:



If you dropped the item in one of the three larger boxes in the header at the top of the drop target area--for example, the Table box--a separate median with quartiles would be added for each axis:



But if you drop the item in any of the six lower boxes aligned with a specific measure, the median with quartiles would only be added on the corresponding axis, with the specified scope.

Delete Analytics Pane Item

You can delete an item you just added from the Analytics pane by clicking Undo. You can also drag items off the view to delete them.

You can also click on an item and choose **Remove** from the tooltip.

Note: Some **Analytics** pane items (**Median with Quartiles** and **Average with 95% CI**) actually add both a reference line and a reference distribution. Unless you are using Undo, you would need to delete these items separately.

Edit Analytics Pane Item

To edit an item you have added from the Analytics pane, click on the item and choose **Edit** from the tooltip. For additional editing options, see the section for a particular item type under Analytics Pane Item Definitions, below.

[Analytics Pane Items](#)

The following items can be dragged from the **Analytics** pane and dropped in the view.

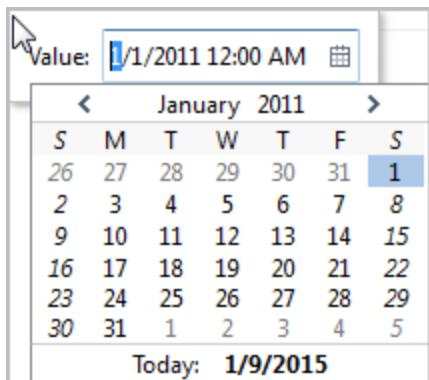
Constant Line

Adds one or more constant lines to the view. You can add a constant line for a specific

measure, for all measures, or for date dimensions. When you add a constant line, Tableau displays a Value prompt where you specify the value for the constant:



For a date value, the Value prompt is a calendar control:



You can click on a resulting constant line and choose **Edit**, **Format**, or **Remove**. Choosing **Edit** opens the Edit Reference Line dialog box. For details, see [Editing Reference Lines, Bands, Distributions, or Boxes on page 975](#). Another way to edit a line is to right-click (control-click on Mac) the relevant axis and choose **Edit Reference Line**.

Average Line

Adds one or more average lines to the view. You can add an average line for a specific measure or for all measures.

You can click on a resulting average line and choose a different aggregation, such as Total or Sum. You can also choose **Edit**, **Format**, or **Remove**. Choosing **Edit** opens the Edit Reference Line dialog box. For details, see [Editing Reference Lines, Bands, Distributions, or Boxes on page 975](#). Another way to edit a line is to right-click (control-click on Mac) the relevant axis and choose **Edit Reference Line**.

Median with Quartiles

Adds one or more sets of median lines and distribution bands to the view. You can add a median with quartiles for a specific measure or for all measures.

The distribution bands are computed as quartiles; the middle two quartiles are shaded.

You can click on a resulting median line or distribution and choose **Edit**, **Format**, or **Remove**. Median lines and distributions must be edited, formatted, or removed separately. Choosing **Edit** opens the Edit Reference Line dialog box. You must click on the outer edge of a distribution band to see the options--clicking in the middle of the band has no effect. For details, see [Editing Reference Lines, Bands, Distributions, or Boxes on page 975](#). Another way

to edit a line or distribution is to right-click (control-click on Mac) the relevant axis and choose **Edit Reference Line**. A submenu will offer you two choices: **Quartiles** and **Median**.

For information on distribution types, including quartiles, see [Basic Reference Distributions on page 963](#).

Box Plot

Adds one or more box plots to the view. You can add box plots for a specific measure or for all measures. The scope for a box plot is always **Cell** (and never **Table** or **Pane**).

Click or hover over any of the horizontal lines in the box plot to see statistical information about the whiskers, quartiles, and median.

You can also choose **Edit**, **Format**, or **Remove** when you click on a line. Choosing **Edit** opens the Edit Reference Line dialog box. Another way to edit a box plot is to right-click (control-click on Mac) the relevant axis and choose **Edit Reference Line**.

Note: Box Plot appears twice in the Analytics pane. For the Box Plot option in the Summarize section, Tableau will automatically add a box plot for the specified target. For the Box Plot option in the Custom section, Tableau will open the Edit Reference Line, Band, or Box dialog box after you specify a target.

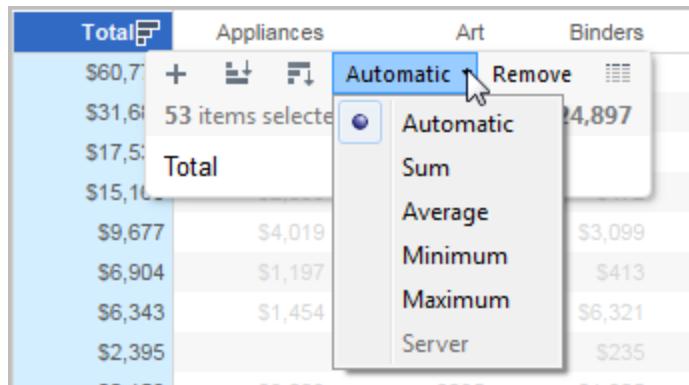
Totals

Adds totals to the view. When you add totals, the drop options are **Subtotals**, **Column Grand Totals**, and **Row Grand Totals**.

For details, see [Totals on page 976](#).

To remove totals, click the relevant column or row header and choose **Remove**.

You can also click on a totals column or row header after adding totals and set the aggregation for that row or column from the tooltip:



Average with 95% CI

Adds one or more sets of average lines with distribution bands; the distribution bands are

configured at a 95% confidence interval. You can add these items for a specific measure or for all measures.

The confidence interval distribution bands shade the region in which the population average will fall 95% of the time.

You can click on a resulting average line or distribution and choose **Edit**, **Format**, or **Remove**. Choosing **Edit** opens the Edit Reference Line dialog box. The average lines and distributions must be edited, formatted, or removed separately. You must click on the outer edge of a distribution band to see the options--clicking in the middle of the band has no effect. For details, see [Editing Reference Lines, Bands, Distributions, or Boxes on page 975](#). Another way to edit a line or distribution is to right-click (control-click on Mac) the relevant axis and choose **Edit Reference Line**. A submenu will offer you two choices: **Average** and **95% Confidence Interval**.

You can also remove lines and bands by dragging them off the view.

Median with 95% CI

Adds one or more sets of median lines with distribution bands; the distribution bands are configured at a 95% confidence interval. You can add these items for a specific measure or for all measures.

The confidence interval distribution bands shade the region in which the population median will fall 95% of the time.

You can click on a resulting median line or distribution and choose **Edit**, **Format**, or **Remove**. Choosing **Edit** opens the Edit Reference Line dialog box. The median lines and distributions must be edited, formatted, or removed separately. You must click on the outer edge of a distribution band to see the options--clicking in the middle of the band has no effect. For details, see [Editing Reference Lines, Bands, Distributions, or Boxes on page 975](#). Another way to edit a line or distribution is to right-click (control-click on Mac) the relevant axis and choose **Edit Reference Line**. A submenu will offer you two choices: **Median** and **95% Confidence Interval**.

You can also remove lines and bands by dragging them off the view.

Trend Line

Adds one or more trend lines to the view. When you add trend lines, the drop options identify the trend line model types available in Tableau: **Linear**, **Logarithmic**, **Exponential**, and **Polynomial**. For some views, only a subset of these options is available.

For details, see [Trend Line Model Types on page 660](#)

click on a trend line to remove or edit it, or to see a statistical definition. You can also remove a trend line by dragging it off the view.

Forecast

Adds a forecast to the view. Forecasting is only possible when there is at least one date and one measure in the view.

Forecasting is not supported for views based on multidimensional data sources. In addition, the view cannot contain any of the following:

- Table calculations
- Disaggregated measures
- Percent calculations
- Grand Totals or Subtotals
- Date values with aggregation set to Exact Date

A time series containing null values also imposes constraints.

For details, see [Forecasting on page 635](#).

To remove, edit, or read a description of the current forecast, go to the Analysis menu and choose **Forecast**.

Reference Line

You can add reference lines for a specific measure or for all measures in the view.

After you drag a reference line from the **Analytics** pane and drop it on a target, Tableau automatically opens the Edit Reference Line, Band or Box dialog box. See [Adding Reference Lines on page 953](#) for information on the available options. To return to this dialog box later, right-click (control-click on Mac) the relevant axis and choose **Edit Reference Line**. You can also click on the line and choose **Edit**, **Format**, or **Remove**.

Reference Band

You can add reference bands for a specific measure or for all measures in the view.

After you drag a reference band from the **Analytics** pane and drop it on a target, Tableau automatically opens the Edit Reference Line, Band, or Box dialog box. See [Adding Reference Bands on page 959](#) for information on the available options. To return to this dialog box later, right-click (control-click on Mac) the relevant axis and choose **Edit Reference Line**. You can also click on the band and choose **Edit**, **Format**, or **Remove**. You must click on the outer edge of a reference band to see the options--clicking in the middle of the band has no effect.

Distribution Band

You can add reference distributions for a specific measure or for all measures in the view.

After you drag a reference distribution from the **Analytics** pane and drop it on a target, Tableau automatically opens Edit Reference Line, Band, or Box dialog box. See [Adding Reference Distributions \(Bullet Graphs\) on page 963](#) for information on the available options. To return to this dialog box later, right-click (control-click on Mac) the relevant axis and choose **Edit Reference Line**. You can also click on the distribution and choose **Edit**, **Format**, or **Remove**. You must click on the outer edge of a distribution band to see the options--clicking in the middle of the band has no effect.

Box Plot

You can add box plots for a specific measure or for all measures in the view.

After you drag a box plot from the Custom section of the **Analytics** pane and drop it on a target, Tableau automatically opens Edit Reference Line, Band, or Box dialog box. See [Adding Box Plots on page 970](#) for information on the available options. The scope for a box plot is always **Cell** (and never **Table** or **Pan**e).

Click any of the horizontal lines in the box plot to see statistical information about the whiskers, quartiles, and median.

You can click on a resulting box plots and choose **Edit**, **Format**, or **Remove**. Choosing **Edit** opens the Edit Reference Line dialog box. Another way to edit a box plot is to right-click (control-click on Mac) the relevant axis and choose **Edit Reference Line**.

Shelves and Cards

Every worksheet in Tableau contains shelves and cards. By placing fields on shelves or cards, you can create the rows and columns of a data view, exclude data from the view, create pages, and control mark properties. You should experiment placing fields on different shelves and cards to find the optimal way to look at your data.

Tableau can also help you determine the best way to display your data using Show Me. See [Create a Line Chart with Show Me on page 40](#) and [Create a Scatter Plot with Show Me on page 37](#).

Columns and Rows Shelves

The **Columns** shelf creates the columns of a table, while the **Rows** shelf creates the rows of a table. You can place any number of fields on these shelves.

When you place a dimension on the **Rows** or **Columns** shelves, headers for the members of that dimension are created. When you place a measure on the **Rows** or **Columns** shelf, quantitative axes for that measure are created. As you add more fields to the view, additional headers and axes are included in the table and you get an increasingly detailed picture of your data.

In the view shown below, the members of the **Segment** dimension are displayed as column headers, while the **Profit** measure is displayed as a vertical axis.

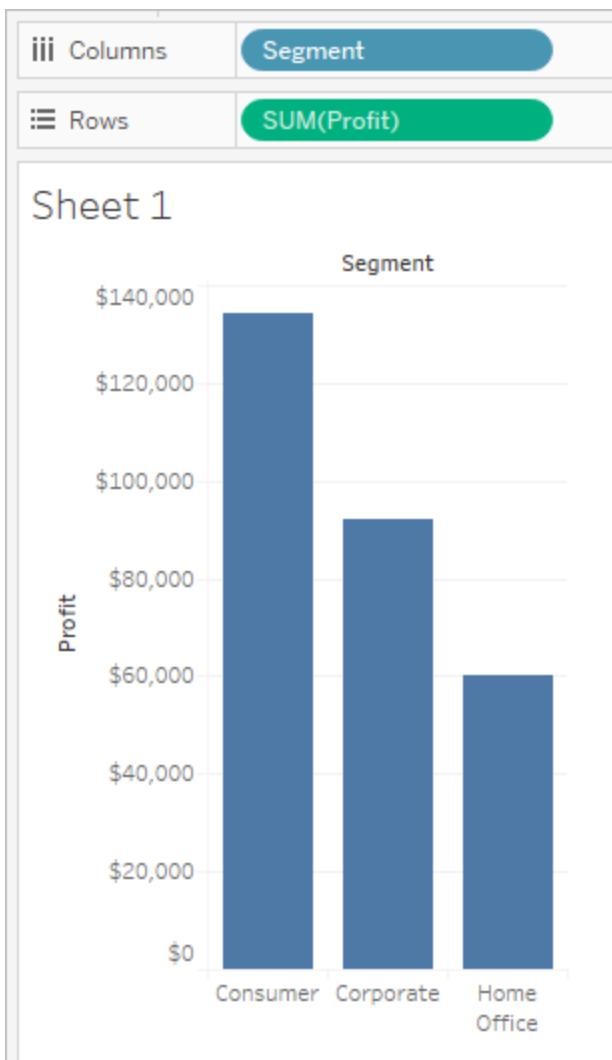
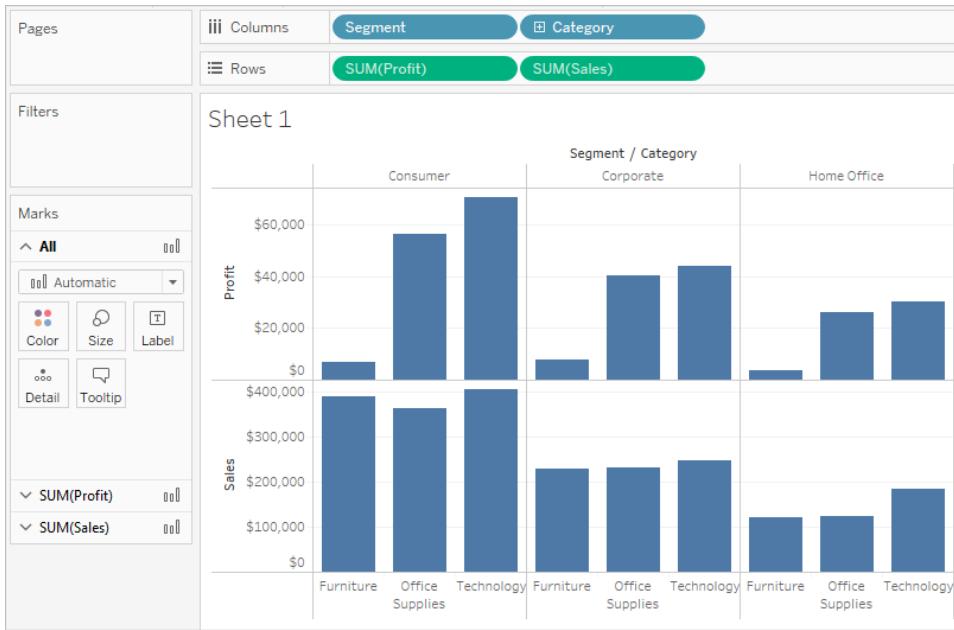


Tableau displays data using marks, where every mark corresponds to a row (or a group of rows) in your data source. The inner fields on the **Rows** and **Columns** shelves determine the default mark type. For example, if the inner fields are a measure and a dimension, the default mark type is a bar. You can manually select a different mark type using the Marks card drop-down menu. For more information, see [Mark Types](#) on page 168.

Adding more fields to the **Rows** and **Columns** shelves adds more rows, columns, and panes to the table.



Hide Rows and Columns

Generally you will add dimensions and measures to create the rows and columns of the table and you'll either include all data or add filters to only show a subset. However, when you filter data it is also excluded from calculations and other computations performed on the summarized data in the table. Instead of filtering the data, you can hide the row or column so it doesn't display in the view but it is still included in calculations.

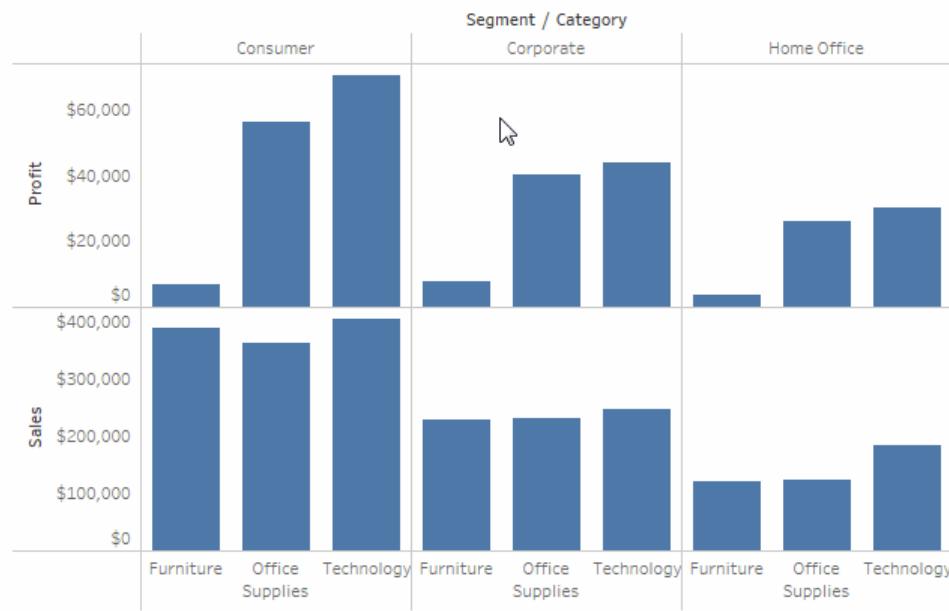
Hiding columns is especially useful when using table calculations that compare to previous or next. In that case, there is always a row or column that doesn't show data because there is no data to compare to. You can simply hide the empty column without modifying the table calculation. For example, when calculating year-over-year growth, the first year doesn't have a previous year to compare to, so the column is left blank. Filtering the first year will remove it from the view but it will also remove it from the calculation so now the second year doesn't have a previous year to compare to and is left blank. Instead of filtering, you can hide the column that you don't want to show without changing the calculation.

To hide a row or column:

Right-click (control-click on Mac) the row or column you want to hide, and then select **Hide**.

iii Columns	Segment	<input checked="" type="checkbox"/> Category
Rows	SUM(Profit)	<input checked="" type="checkbox"/> SUM(Sales)

Sheet 1



To show hidden data:

Open the field menu for a field that has hidden columns or rows and select **Show Hidden Data**.

iii Columns	Segment	Category
Rows	SUM(Profit)	SUM(Sales)

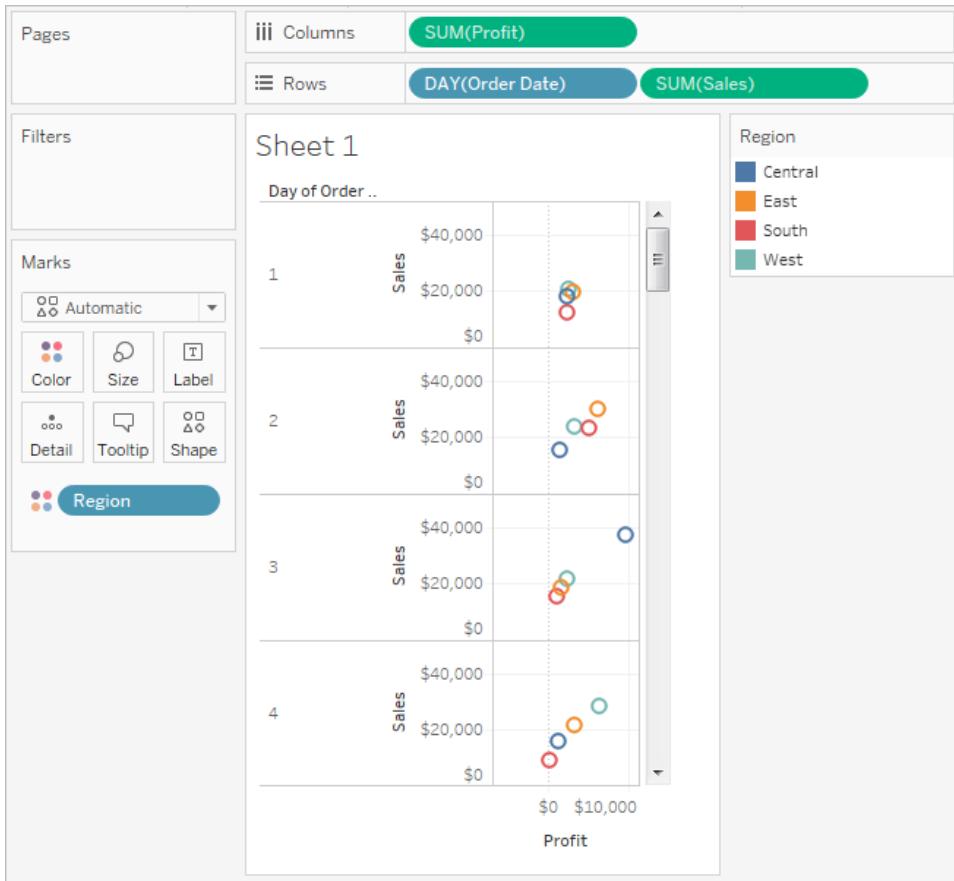
Sheet 1



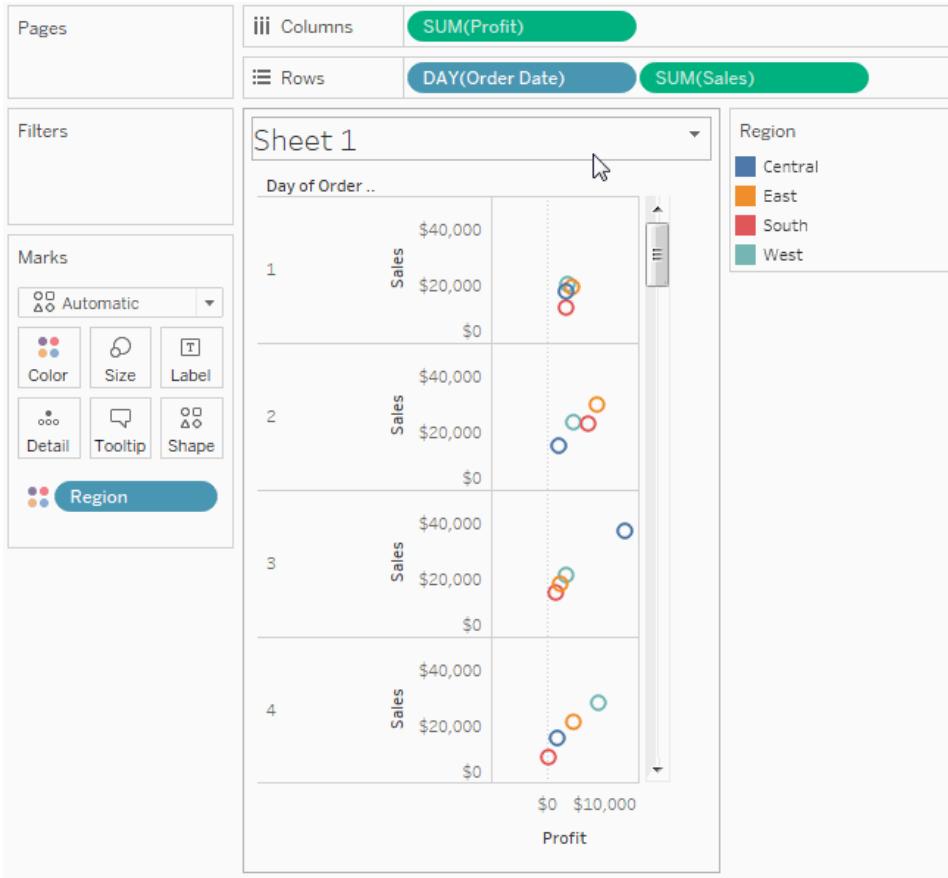
Pages Shelf

The **Pages** shelf lets you break a view into a series of pages so you can better analyze how a specific field affects the rest of the data in a view. When you place a dimension on the **Pages** shelf you are adding a new row for each member in the dimension. When you place a measure on the **Pages** shelf, Tableau automatically converts the measure into a discrete measure.

The **Pages** shelf creates a set of pages, with a different view on each page. Each view is based on a member of the field you placed on the **Pages** shelf. You can easily flip through the views and compare them on a common axis, using the controls that get added to the view when you move a field to the **Pages** shelf. For example, the view below shows the **Profit vs. Sales** by **Region** for each day throughout the month. The image below shows days 1, 2, 3, and 4. You would have to scroll down to see other days in the month.



To make this view more user-friendly, move **DAY(Order Date)** to the **Pages** shelf and use the associated control to flip through the pages (one for each day). You can quickly discover hidden insights. In this example, it is interesting that the 19th is an especially big day in terms of sales and profit in the Western region.



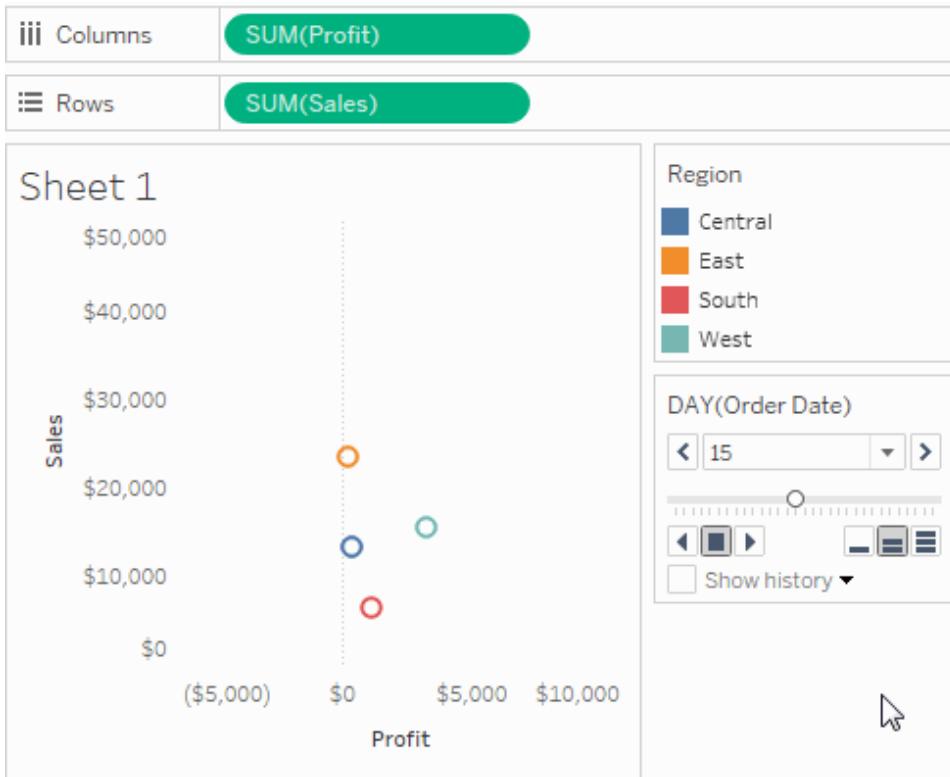
When you add a field to the **Pages** shelf, a page control is automatically added to the right of your view.

Use this control to navigate through the pages. There are three ways to navigate through the pages in a view:

- Jump to a specific page
- Manually advance through the pages
- Automatically advance through pages

[Jump to a specific page](#)

Select the member or value you want to view from the drop-down list to display a specific page.



Manually Advance through the pages

You can manually advance through the sequence of pages by doing any of the following:

- Use the forward and back buttons on either side of the drop-down list to navigate through the pages one at a time.
- Use the Page Slider to quickly scroll forward and backward in the sequence of pages.
- Use any of the following keyboard shortcuts to scroll forward and backward in the sequence of pages.

F4	Starts and stops forward playback
SHIFT + F4	Starts and stops backward playback
CTRL + .	Skip forward one page
CTRL + ,	Skip backward one page

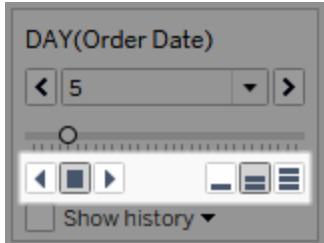
On a Mac, these are the equivalent keyboard shortcuts.

F4	Starts and stops forward playback
----	-----------------------------------

Shift-F4	Starts and stops backward playback
Command-period	Skip forward one page
Command-comma	Skip backward one page

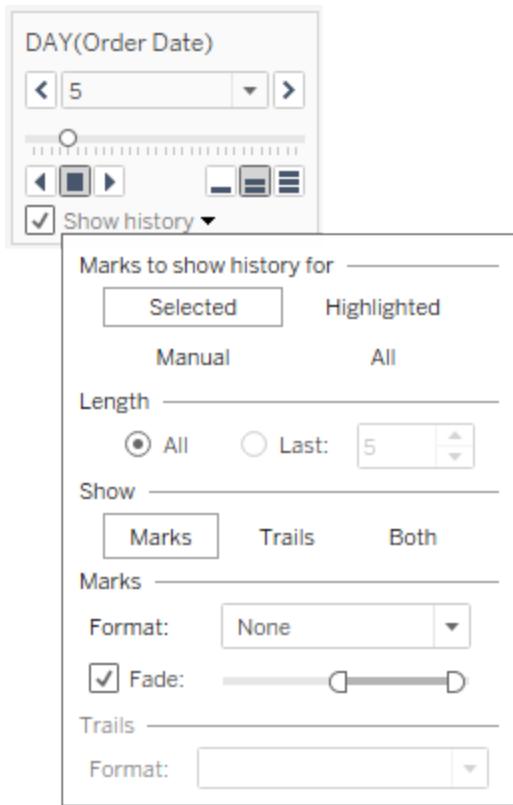
Automatically Advance through the pages

Use the playback controls to watch a slide show of the pages in the view. You can play forward or backward, and stop the playback at any time. You can control the speed of playback with the speed controls in the bottom right corner of the control. The smallest bar indicates the slowest playback speed.



Page History

Show page history using the Show History check box. With page history, marks from previous pages are shown on the current page. Open the drop-down control for history to specify what marks to show and when to show them.



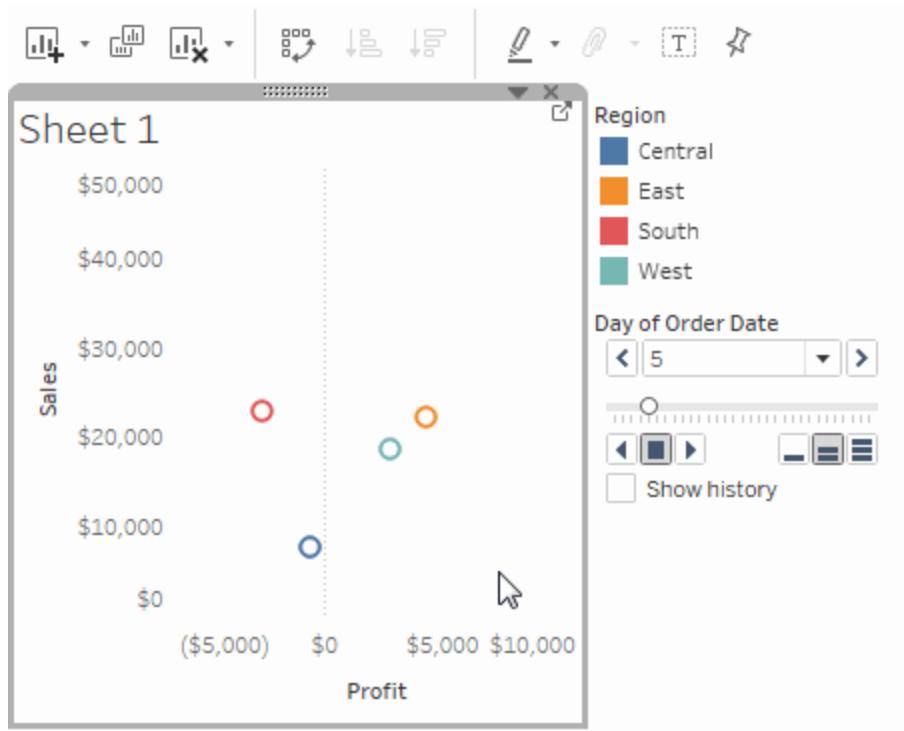
The history drop-down control has the following options:

- **Marks to show history for** – select whether you want to show history for just selected marks, highlighted marks, marks that you've manually selected to show history for, or all marks. To manually show history for marks, right-click (control-click on Mac) the mark in the view and select an option on the Page History menu.
- **Length** – select the number of pages to show in the history.
- **Show** – specify whether to show the historical marks, a line tracing through the previous values (trails), or both.
- **Marks** – format the historical marks including the color and how much to fade them if the color is set to automatic, the marks will either use the default mark color or the color encoding on the Color shelf.
- **Trails** – format the lines that are drawn through the historical marks. This option is only available if Trails is selected in the Show options.

Page trails may not display if there are multiple marks per color on a page. Make sure that the level of detail for the view is less than or equal to the level of detail on the **Pages** shelf and on the Color target. Also, trails are only supported for discrete mark types such as squares, circles, or shapes. They are not supported when the mark type is Automatic.

Pages on Dashboards

When a dashboard contains multiple views that use the same field on the **Pages** shelf, you can control all of the views with a single page control by selecting the **Synchronized** option. This option is only available on the page control shown on a dashboard.



Filters Shelf

The Filters shelf allows you to specify which data to include and exclude. For example, you might want to analyze the profit for each customer segment, but only for certain shipping containers and delivery times. By placing fields on the Filters shelf, you can create such a view.

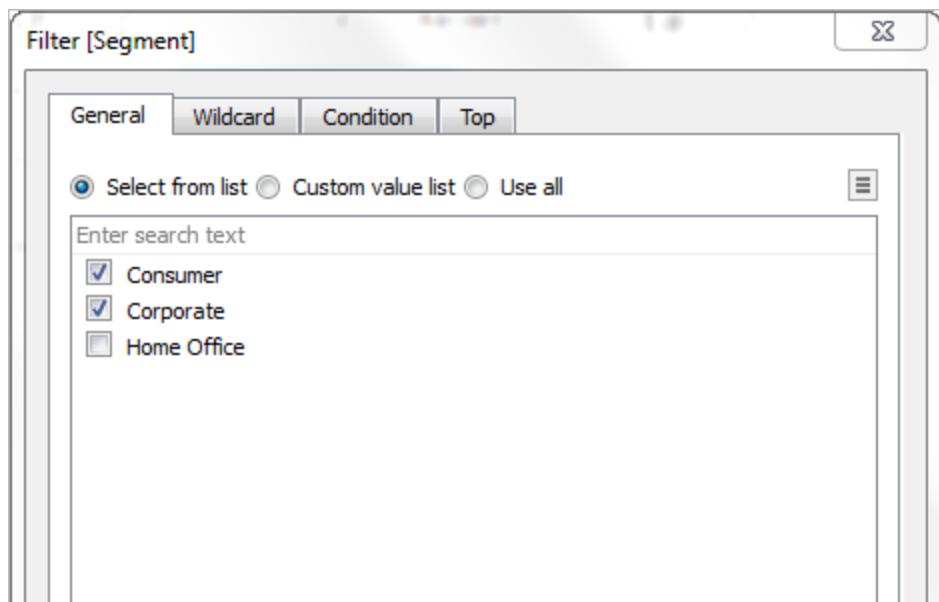
Note: This section presents a brief overview of filtering. For more information on filtering, see [Filter Data from Your Views on page 672](#).

You can filter data using measures, dimensions, or both at the same time. Additionally, you can filter data based on the fields that make up the columns and rows of the table. This is called an internal filter. You can also filter data using fields that don't contribute headers or axes to the table. This is called an external filter. All filtered fields display on the Filters shelf.

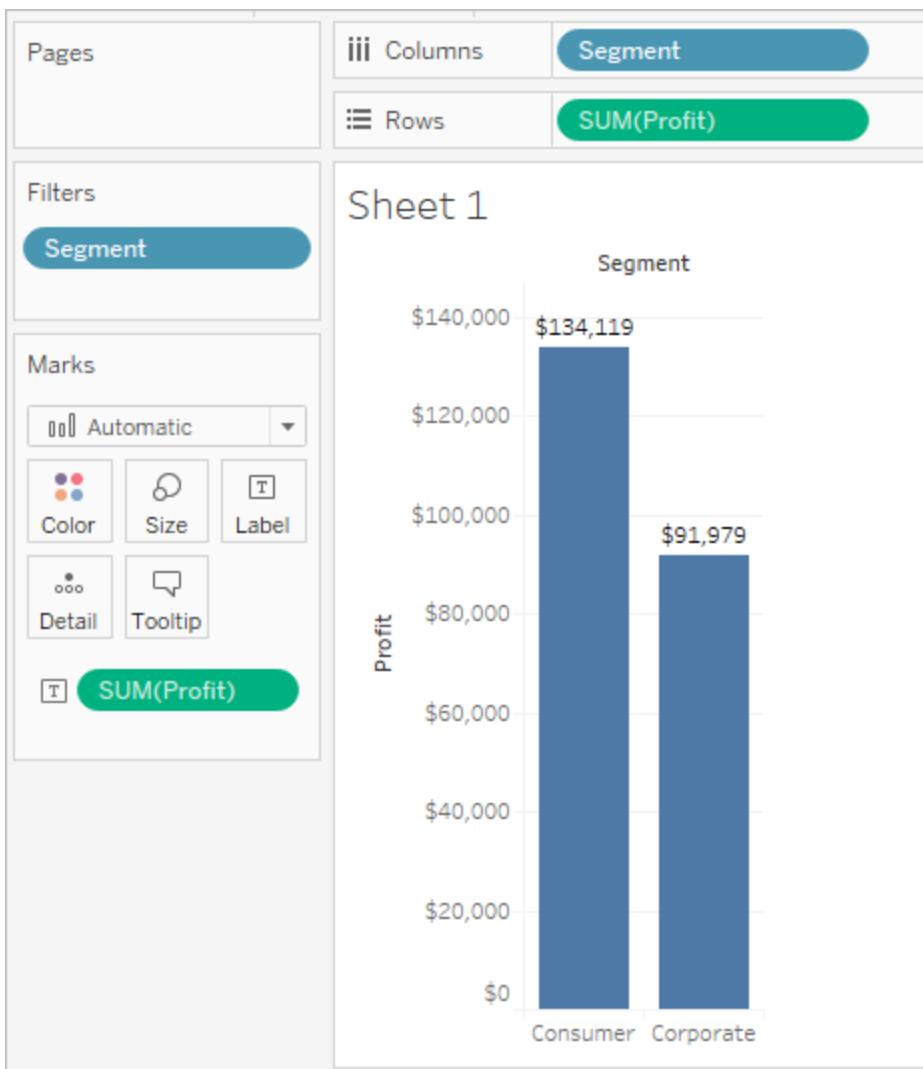
To illustrate the basic concepts of filtering, consider the following view.



Suppose you are not interested in the Home Office data. You can remove this column from the view by filtering the **Segment** dimension. To do so, select **Filter** on the field menu or drag the **Segment** dimension to the **Filters** shelf. The Filter dialog box opens. By default all members are selected. Clear the check box for **Home Office** to exclude it from the view. All selected members will be included.



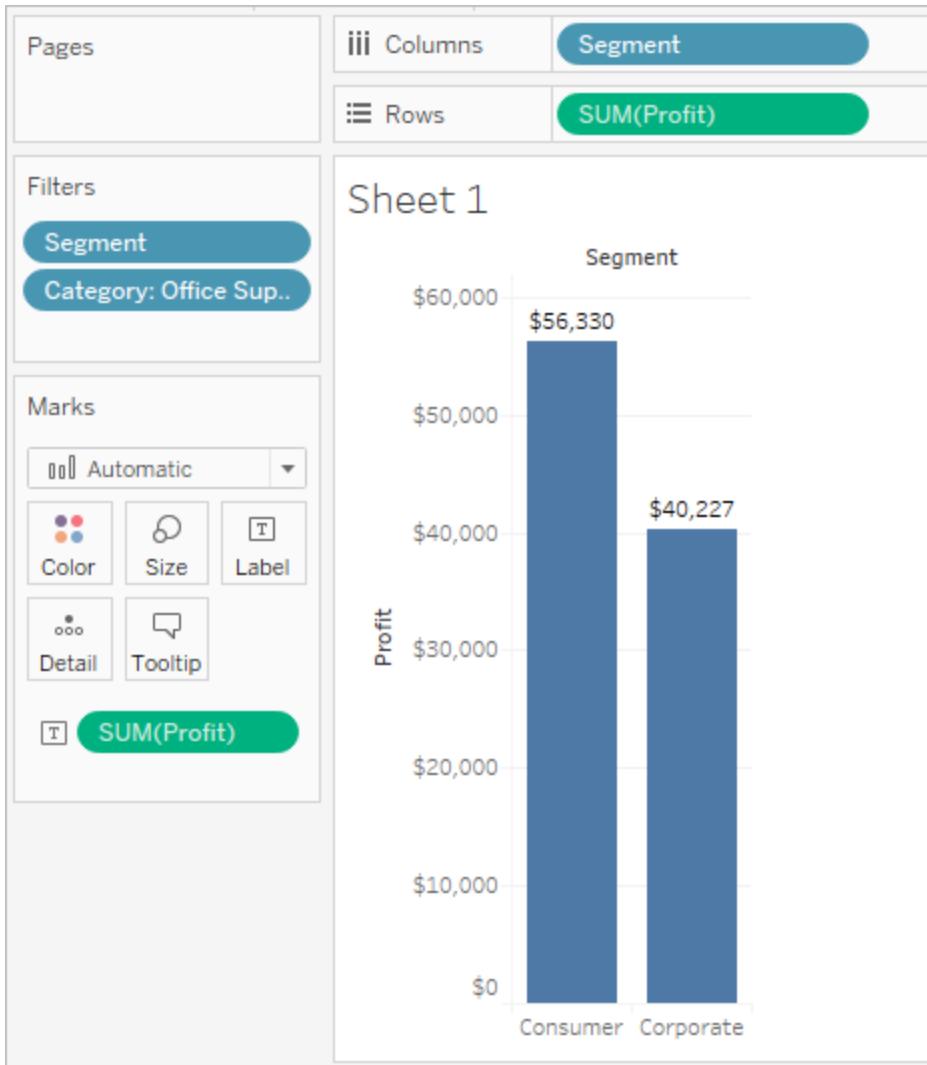
As shown below, the view updates and the Home Office column is removed. The filter is indicated by the **Segment** field on the **Filters** shelf.



Suppose you want to only view profit for a category of the products. Even though the **Category** field is not used on the **Rows** and **Columns** shelves or on the Marks card, you can still add a filter. Drag the **Category** dimension to the **Filters** shelf. This is an example of an external filter because Category is not part of the view.

The Filter dialog box automatically opens. By default, none of the members are selected. Select the members you want to keep as part of the view. All cleared members are excluded. In this example, Office Supplies is selected.

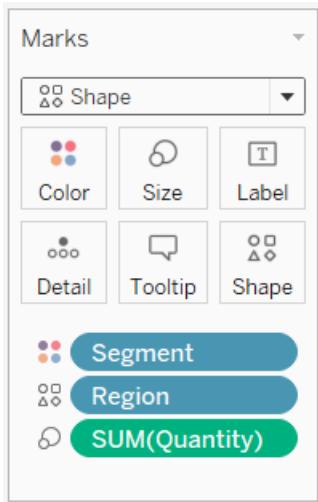
The modified data view is shown below. The mark label shows that the sum of the profit for the Consumer segment has decreased to \$56,330. This number is derived by summing all the rows in the data source that are associated with the Corporate market and are part of the Office Supplies category.



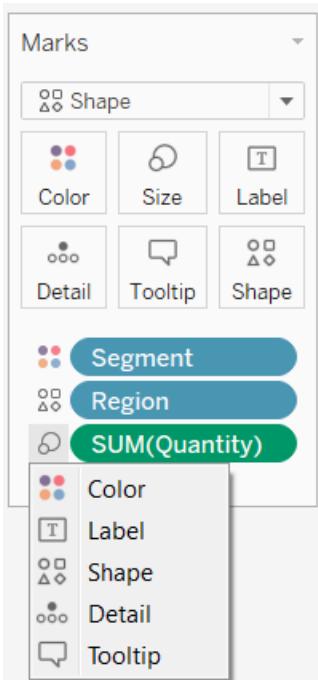
The order of fields placed on the Filters shelf does not affect the data view because the filters are independent. In other words, the result of filtering by customer segment, and then by container is the same as filtering by container and then by customer segment. For more information about the filtering order of operations, see [Filter Data from Your Views](#) on page 672.

Marks Card

The Marks card is where you drag fields to control mark properties such as type, color, size, shape, and so on. The fields on the Marks card are listed at the bottom of the card. Each field has an icon next to it to identify the mark property it is setting. For example, the Marks card shown below has three fields: Segment is on Color, Region is on Shape, and Quantity is on Size. For more information, see [Marks](#) on page 167, [Mark Types](#) on page 168, and [Mark Properties](#) on page 190.



After a field has been added to the Marks card, you can click the icon next to the field to change the property it is controlling.



Many properties can have multiple fields. For example, you can add multiple fields to Label, Detail, Tooltip, and Color. Size and Shape can only have one field at a time.

Note: By default, dragging a new field to **Color** replaces the existing fields. To add a new field to color without replacing the existing field, hold the SHIFT key on your keyboard while dragging a new field to **Color** on the Marks card.

Additional Shelves, Legends, Cards, and Controls

Some shelves, legends, cards, or controls are only displayed as a result of things that you do as you work with views. For example, the Color legend is only displayed when there is a field on Color.

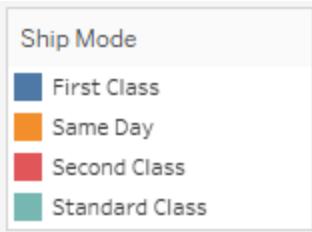


Tableau provides controls for moving or otherwise customizing these elements of the view.

The following list describes each such shelf, legend, card, or control.

- **Measure Values Shelf**— Measure Values is a special field that always appears at the bottom of the Measures area of the Data pane and contains all the measures of your data collected into one field. Tableau automatically adds Measure Values to the view when multiple measures are sharing the same axis. When Measure Values is in the view, Tableau displays a Measure Values shelf that shows which measures are being included. You can add measures to or remove measures from this card. See [Measure Values and Measure Names](#) on page 302.
- **Color Legend**— Shows how colors are allocated when there is a field on Color. For more information, see [Color Properties](#) on page 192.
- **Shape Legend**— Shows how shapes are allocated when there is a field on Shape. For more information, see [Shape Properties](#) on page 221.
- **Size Legend**— Shows how sizes are allocated when there is a field on Size. For more information, see [Size Properties](#) on page 207.
- **Map Legend**— Shows the legend for the symbols and patterns on a map. The map legend is not available for all map providers.
- **Parameter Controls**— A separate parameter control is available for every parameter in the workbook. For more information, see [Parameter Controls](#) on page 1000.
- **Title**— A title is displayed by default for every view. The default title is the sheet name. Double-click a title (Control-click on a Mac) to edit it.
- **Caption**— Choose **Show caption** from the Worksheet menu to display a caption for the view. For more information, see [Captions](#) on page 236.
- **Summary Card**— Choose **Show summary** from the Worksheet menu to display a summary card for the view. For more information, see [Summary Card](#) on page 572.

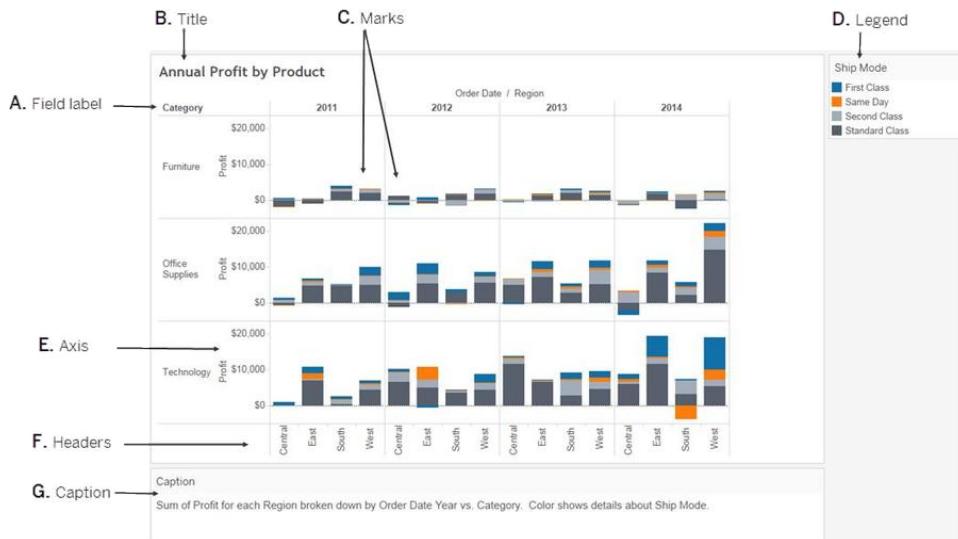
- **Page Control** – Provides options for navigating through pages when there is a field on the Pages shelf. For more information, see [Pages Shelf on page 147](#).

Parts of the View

This section describes the basic components of the views you can create in Tableau. The parts of a view can be categorized as either table components, which are part of every view, or optional components, which can be turned on or off.

Table Components

Data views are displayed in a table on every worksheet. A table is a collection of rows and columns, and consists of the following components: Headers, Axes, Panes, Cells, and Marks. In addition to these, you can choose to show Titles, Captions, Field Labels, and Legends.



A. Field label - The label of a discrete field added to the row or column shelf that describes the members of that field. For example, Category is a discrete field that contains three members; Furniture, Office Supplies and Technology.

B. Title - The name that you give your worksheet, dashboard, or story. Titles display automatically for worksheets and stories and you can turn them on to display them in your dashboards.

C. Marks - The data that represents the intersection of the fields (dimensions and measures) included in your view. Marks can be represented using lines, bars, shapes, maps and so on.

D. Legend - A key that describes how the data is encoded in your view. For example if you use shapes or colors in your view, the legend describes what each shape or color represents.

E. Axis - Created when you add a measure (fields that contain quantitative, numerical information) to the view. By default, Tableau generates a continuous axis for this data.

F. Headers -The members of a field.

G. Caption - Text that describes the data in the view. Captions can be automatically generated and can be toggled on and off.

Headers

Headers are created when you place a dimension or discrete field on the **Rows** shelf or the **Columns** shelves. The headers show the member names of each field on the shelves. For example, in the view below the column headers show the members of the **Order Date** field and the row headers show the members of the **Sub-Category** field.

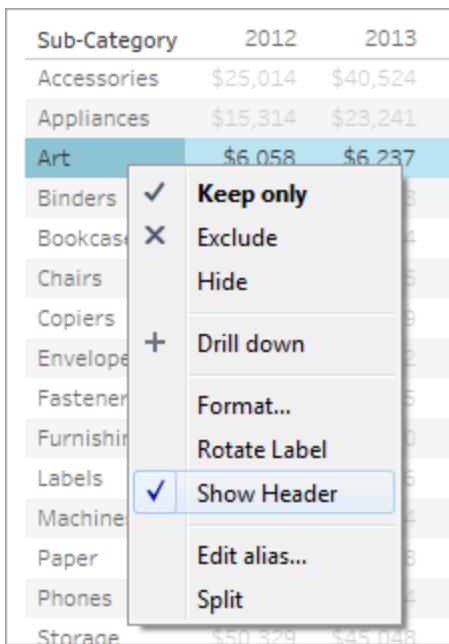
Sheet 1

Order Date				
Sub-Category	2012	2013	2014	2015
Accessories	\$25,014	\$40,524	\$41,896	\$59,946
Appliances	\$15,314	\$23,241	\$26,050	\$42,927
Art	\$6,058	\$6,237	\$5,910	\$8,914
Binders	\$43,488	\$37,453	\$49,485	\$72,986
Bookcases	\$20,037	\$38,544	\$26,275	\$30,024
Chairs	\$77,242	\$71,735	\$83,919	\$95,554
Copiers	\$10,850	\$26,179	\$49,599	\$62,899
Envelopes	\$3,856	\$4,512	\$4,730	\$3,379
Fasteners	\$661	\$545	\$960	\$858
Furnishings	\$13,826	\$21,090	\$27,874	\$28,915
Labels	\$2,841	\$2,956	\$2,827	\$3,861
Machines	\$62,023	\$27,764	\$55,907	\$43,545
Paper	\$14,835	\$15,288	\$20,638	\$27,718
Phones	\$77,391	\$68,314	\$78,660	\$105,643
Storage	\$50,329	\$45,048	\$58,632	\$69,834
Supplies	\$14,394	\$1,952	\$14,278	\$16,049
Tables	\$46,088	\$39,150	\$60,833	\$60,894

You can show and hide row and column headers at anytime.

To hide headers:

- Right-click (control-click on Mac) the headers in the view and select **Show Header**.



A screenshot of a spreadsheet application showing a context menu for the header row 'Sub-Category'. The menu options are:

- ✓ Keep only
- ✗ Exclude
- Hide
- + Drill down
- Format...
- Rotate Label
- ✓ Show Header
- Edit alias...
- Split

The 'Show Header' option is highlighted with a blue border.

Sub-Category	2012	2013
Accessories	\$25,014	\$40,524
Appliances	\$15,314	\$23,241
Art	\$6,058	\$6,237
Binders		
Bookcases		
Chairs		
Copiers		
Envelopes		
Fastener		
Furniture		
Labels		
Machines		
Paper		
Phones		
Storage	\$50,324	\$45,048

To show headers:

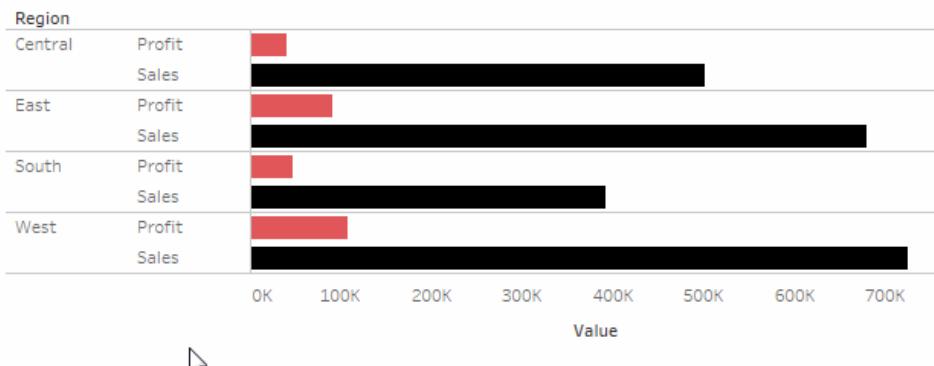
- Select the field in the view whose headers you want to show and select **Show Header** on the field menu.

The screenshot shows a Tableau data view titled "Sheet 1". The data is organized by year (2012) and sub-category. The "Sub-Category" column has a context menu open, with the "Show Header" option highlighted. The menu also includes options like Filter..., Show Filter, Show Highlighter, Sort..., Format..., Include in Tooltip, Edit Aliases..., Dimension, Attribute, Measure, Edit in Shelf, and Remove.

Sub-Category	2012	2012	2012	2012
Electronics	\$25,014	\$40,520	\$15,314	\$23,200
Apparel	\$6,058	\$6,200	\$43,488	\$37,400
Books	\$20,037	\$38,500	\$77,242	\$71,700
Home & Garden	\$10,850	\$26,100	\$3,856	\$4,500
Automotive	\$661	\$500	\$13,826	\$21,000
Food & Beverage	\$2,841	\$2,900	\$62,023	\$27,700
Leisure	\$14,835	\$15,200	\$77,391	\$68,314
				\$78,660 \$105,643

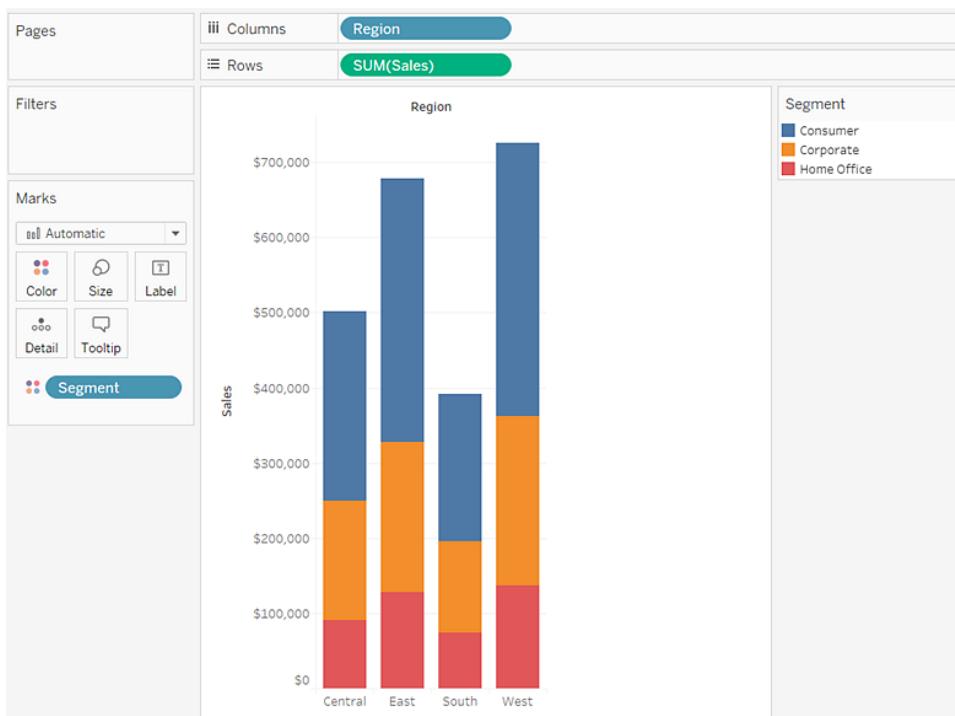
Hiding headers can be useful when you are working with multiple measures. For example, the view below shows both the sales and profit for each region along a single axis. You can see the view looks cluttered with the Measure Names headers showing. Because Measure Names is also indicated by the mark color, you can hide the excess headers to clean up the view.

Sheet 1



Axes

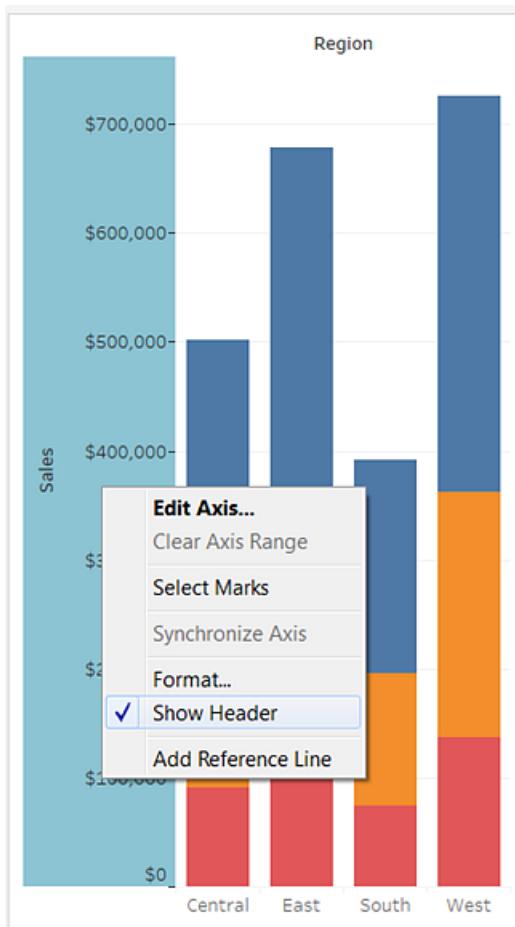
Axes are created when you place a measure or continuous field on the **Rows** or **Columns** shelves. By default, the values of the measure field are displayed along a continuous axis.



You can show and hide axes at any time.

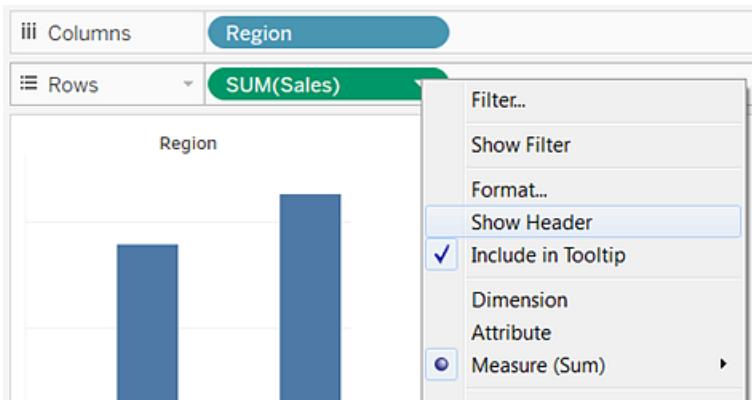
To hide axes:

- Right-click (control-click on Mac) the axis in the view and select **Show Header** to clear the check mark next to this option.



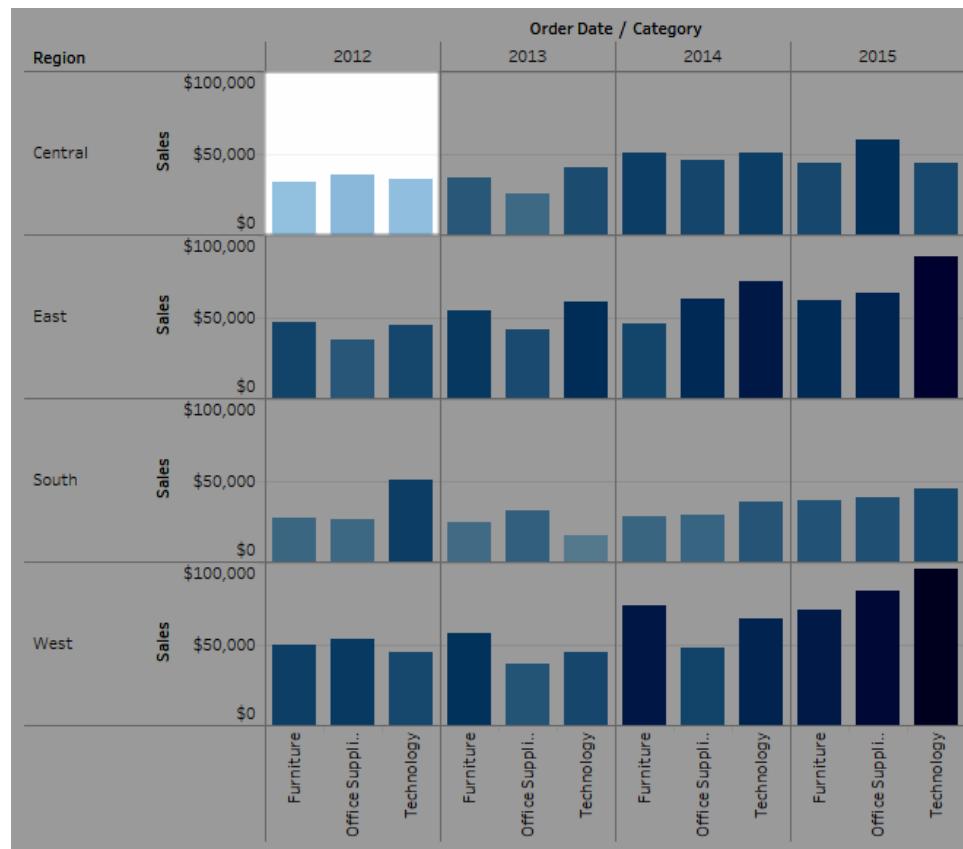
To show axes:

- Right-click (control-click on Mac) the measure in the view whose axis you want to show and select **Show Header** on the field menu.



Panes

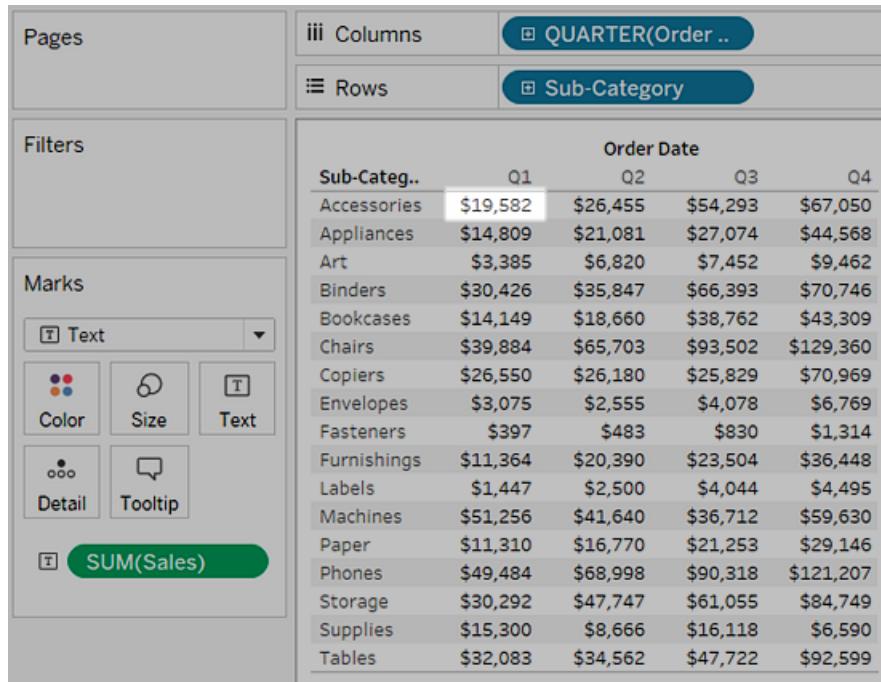
Panes are created by the intersection of rows and columns in a table. An example of a pane is highlighted in the view below.



Note: Tableau can display up to 1 million panes. If your view contains more than 1 million panes, you will need to filter your data so there are fewer panes in the view. For more information on filtering, see [Filter Data from Your Views](#) on page 672.

Cells

Cells are the basic components of any table you can create in Tableau. For a text table, the cell is the intersection of a row and a column, and is where the text is displayed. For example, one of the 68 cells is highlighted in the view below.



The screenshot shows a Tableau interface with a text table. The top navigation bar has 'iii Columns' and 'QUARTER(Order ..)' selected under 'Pages'. Below it, 'Rows' and 'Sub-Category' are selected under 'Rows'. The left sidebar shows 'Filters' and 'Marks'. Under 'Marks', 'Text' is selected, and the 'Text' section includes 'Color', 'Size', 'Text', 'Detail', 'Tooltip', and 'SUM(Sales)'. The main area displays a table titled 'Order Date' with columns for 'Sub-Categ..', 'Q1', 'Q2', 'Q3', and 'Q4'. The 'Q1' column for 'Accessories' is highlighted in yellow, indicating it is the active cell. The data rows include: Accessories (\$19,582), Appliances (\$14,809), Art (\$3,385), Binders (\$30,426), Bookcases (\$14,149), Chairs (\$39,884), Copiers (\$26,550), Envelopes (\$3,075), Fasteners (\$397), Furnishings (\$11,364), Labels (\$1,447), Machines (\$51,256), Paper (\$11,310), Phones (\$49,484), Storage (\$30,292), Supplies (\$15,300), and Tables (\$32,083).

Sub-Categ..	Q1	Q2	Q3	Q4
Accessories	\$19,582	\$26,455	\$54,293	\$67,050
Appliances	\$14,809	\$21,081	\$27,074	\$44,568
Art	\$3,385	\$6,820	\$7,452	\$9,462
Binders	\$30,426	\$35,847	\$66,393	\$70,746
Bookcases	\$14,149	\$18,660	\$38,762	\$43,309
Chairs	\$39,884	\$65,703	\$93,502	\$129,360
Copiers	\$26,550	\$26,180	\$25,829	\$70,969
Envelopes	\$3,075	\$2,555	\$4,078	\$6,769
Fasteners	\$397	\$483	\$830	\$1,314
Furnishings	\$11,364	\$20,390	\$23,504	\$36,448
Labels	\$1,447	\$2,500	\$4,044	\$4,495
Machines	\$51,256	\$41,640	\$36,712	\$59,630
Paper	\$11,310	\$16,770	\$21,253	\$29,146
Phones	\$49,484	\$68,998	\$90,318	\$121,207
Storage	\$30,292	\$47,747	\$61,055	\$84,749
Supplies	\$15,300	\$8,666	\$16,118	\$6,590
Tables	\$32,083	\$34,562	\$47,722	\$92,599

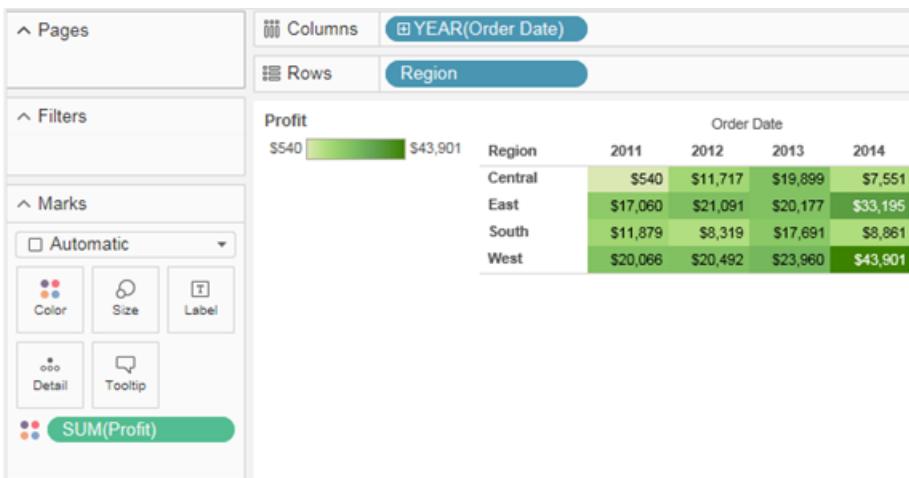
For other view types such as bar charts and scatter plots, identifying the cell is not always possible or useful.

Marks

When you drag fields to the view, the data are displayed using marks. Each mark represents the intersection of all of the dimensions in the view. For example, in a view with **Region** and **Year** dimensions, there is a mark for every combination of those two dimensions (East 2011, East 2012, West 2011, West 2012, etc.).

	Order Date			
Region	2011	2012	2013	2014
Central	Abc	Abc	Abc	Abc
East	Abc	Abc	Abc	Abc
South	Abc	Abc	Abc	Abc
West	Abc	Abc	Abc	Abc

Marks can be displayed in many different ways including lines, shapes, bars, maps, and so on. You can show additional information about the data using mark properties such as color, size, shape, labels, etc. The type of mark you use and the mark properties are controlled by the Marks card. Drag fields to the Marks card to show more data. For example, the same view above is shown again below but this time with **Profit** on Color. With this additional information, it is clear that the West region had the highest profit in 2014.

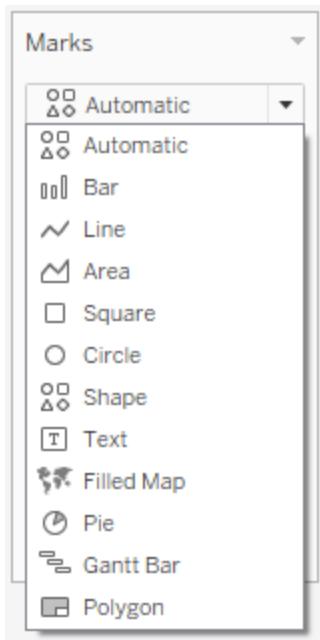


Control the marks in the view using the Marks card. Use the drop-down menu to specify the type of mark to show. Drag fields to the Marks card and use the drop-down controls to add more information to the view and control the color, shape, size, labels, and number of marks in the view.

For information about how to move marks in a view, see [Move Marks](#) on page 228.

Mark Types

Mark types are available from the Marks card drop-down menu at the top of the Marks card. After selecting a mark type, you can further modify the marks by adding fields to Color, Size, Shape, and so on. The available mark properties are dependent on the type of mark you select.



Automatic Mark

When the Marks card drop-down menu is set to **Automatic**, Tableau automatically selects the best mark type for your data view. The mark type that is automatically selected is determined by the inner fields on the **Rows** and **Columns** shelves. The icon in the Marks card drop-down menu indicates which type of mark was automatically selected. The mark types below are automatically selected for the following scenarios.

Text

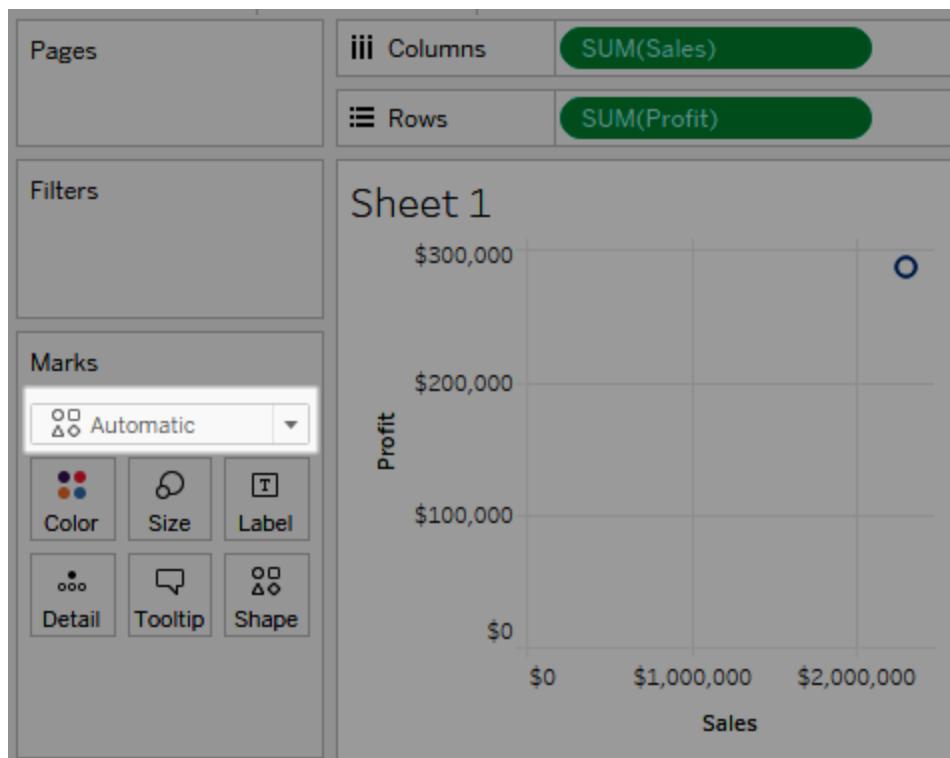
The Text mark type is automatically selected when there are dimensions as inner fields on both the **Rows** and **Columns** shelves.

The screenshot shows the Tableau Data Editor interface. In the top right, under 'Columns', 'YEAR(Order Date)' is selected. Under 'Rows', 'Region' is selected. The 'Marks' card in the bottom left is set to 'Automatic'. The main area displays a data table titled 'Sheet 1' with columns for Region (Central, East, South, West) and years (2012, 2013, 2014, 2015). Each cell contains the value 'Abc'. The data is presented as follows:

Order Date				
Region	2012	2013	2014	2015
Central	Abc	Abc	Abc	Abc
East	Abc	Abc	Abc	Abc
South	Abc	Abc	Abc	Abc
West	Abc	Abc	Abc	Abc

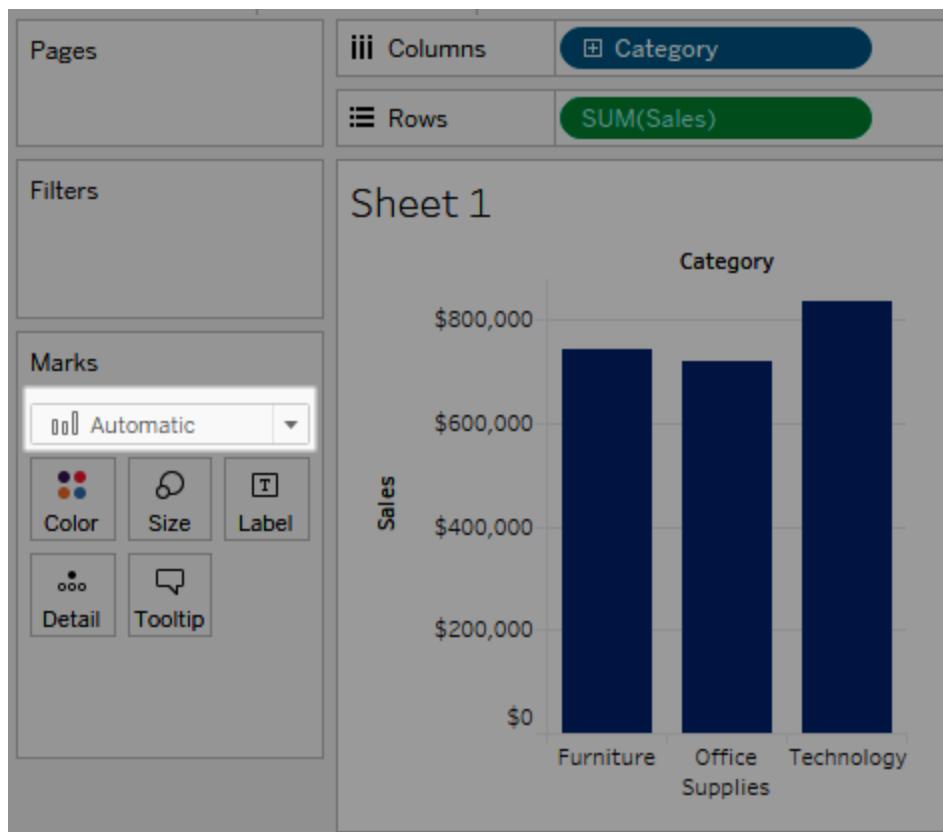
Shape

The Shape mark type is selected when there are measures as inner fields on both the **Rows** and **Columns** shelves.



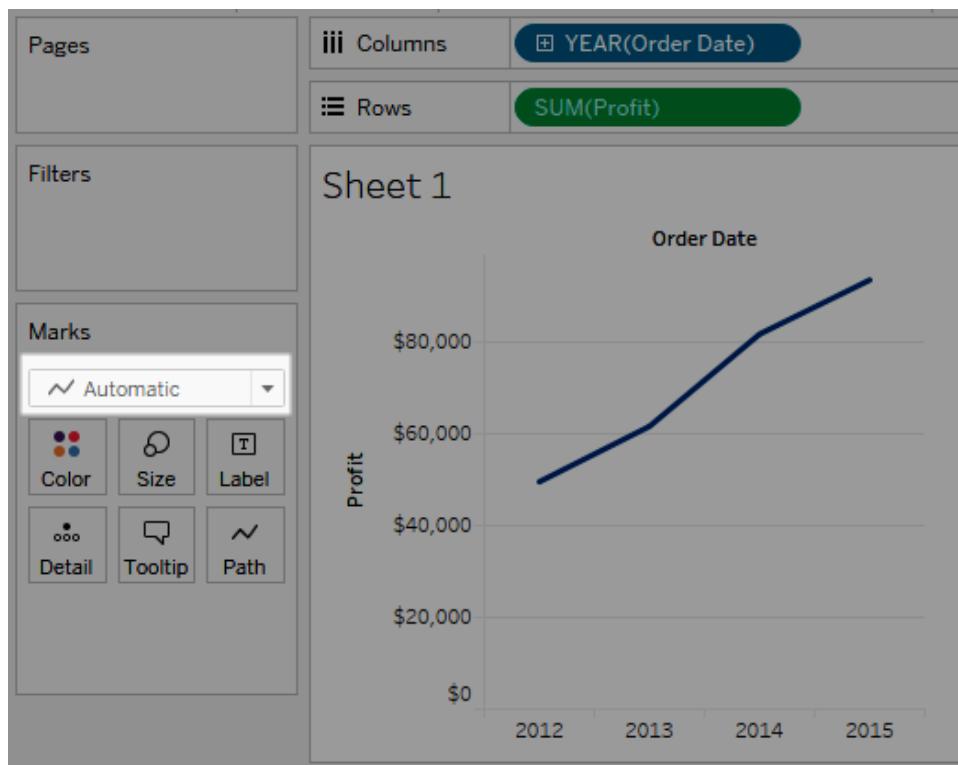
Bar

The Bar mark type is selected when there is a dimension and a measure as inner fields on **Rows** and **Columns** shelves.



Line

The Line mark type is selected when there is a date field and a measure as the inner fields on the **Rows** and **Columns** shelves.



Note: You can override the default selection and use any mark type that provides insight into your data. However, you should exercise some caution when manually selecting a mark type because the resulting view might hide important information about your data.

Bar Mark

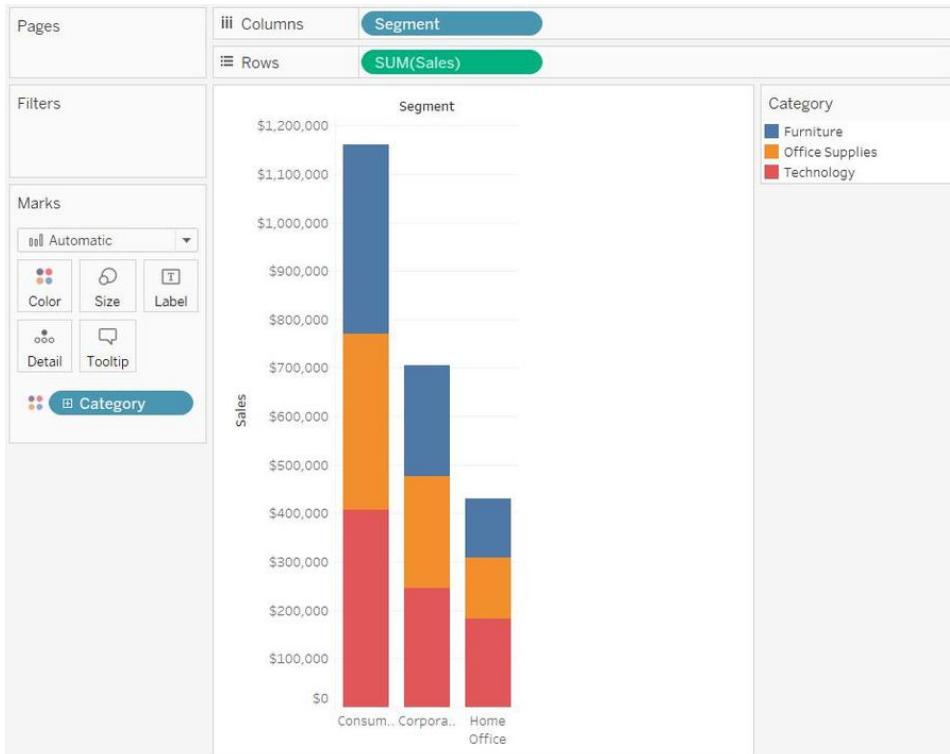
The Bar mark type is useful when you want to compare measures across categories, or when you want to break data down into stacked bars. Tableau displays your data using bars when:

- The Marks card drop-down menu is set to **Automatic**, and you place a dimension and a measure as the inner fields on the **Rows** and **Columns** shelves. If the dimension is a date dimension, the Line mark is used instead.
- You select **Bar** from the Marks card drop-down menu.

Marks are automatically stacked.

Note: For views where the mark type is Bar and there are continuous (green) fields on both **Rows** and **Columns**, Tableau supports additional options and defaults for sizing the bar marks on the axis where the bars are anchored. See the Continuous axis mark sizing section in [Size Properties on page 207](#) for details.

The data view shown below displays a dimension as the columns of the table, and a measure as the rows of the table. It is also color-encoded by a dimension (**Category**). Because the Marks card drop-down menu is set to **Automatic**, data is displayed using bars.



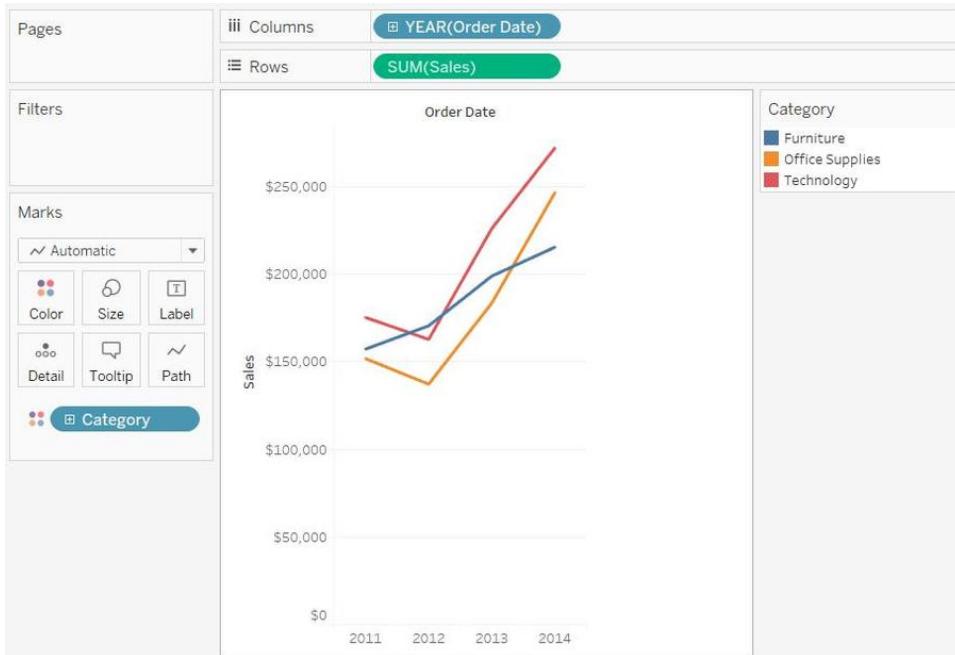
To try some hands-on exercises for building bar charts, see [Build a Bar Chart](#) on page 57.

Line Mark

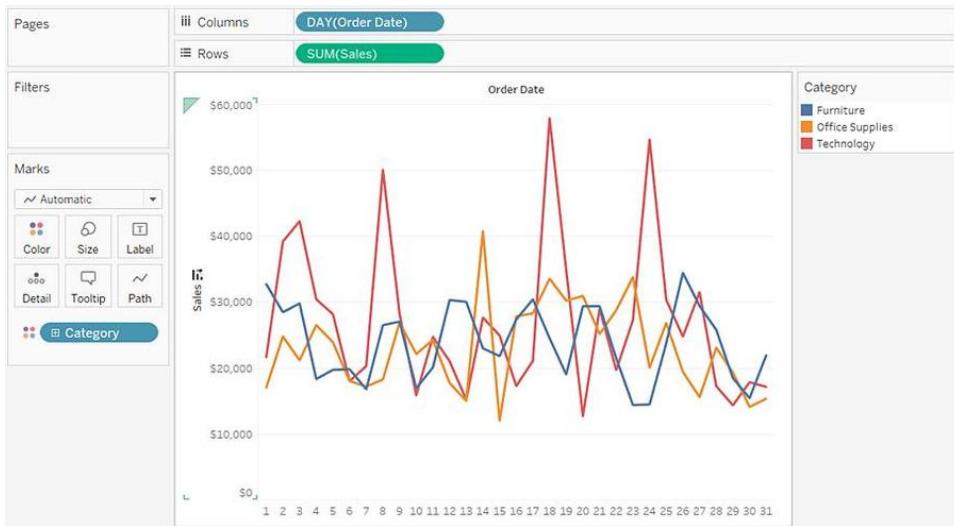
The line mark type is useful when you want to see trends in data over time, your data are ordered, or interpolation makes sense. Tableau displays data using lines when:

- The Marks card drop-down menu is set to **Automatic**, and you place one or more measures on either the **Columns** shelf or the **Rows** shelf, and then plot the measures against a date dimension or a continuous dimension.
- You select **Line** from the Marks card drop-down menu.

The data view shown below displays a date dimension as the columns of the table and a measure as the rows of the table.



As the density of data increases, trends are often easier to see when using lines. This view shows 93 data points.



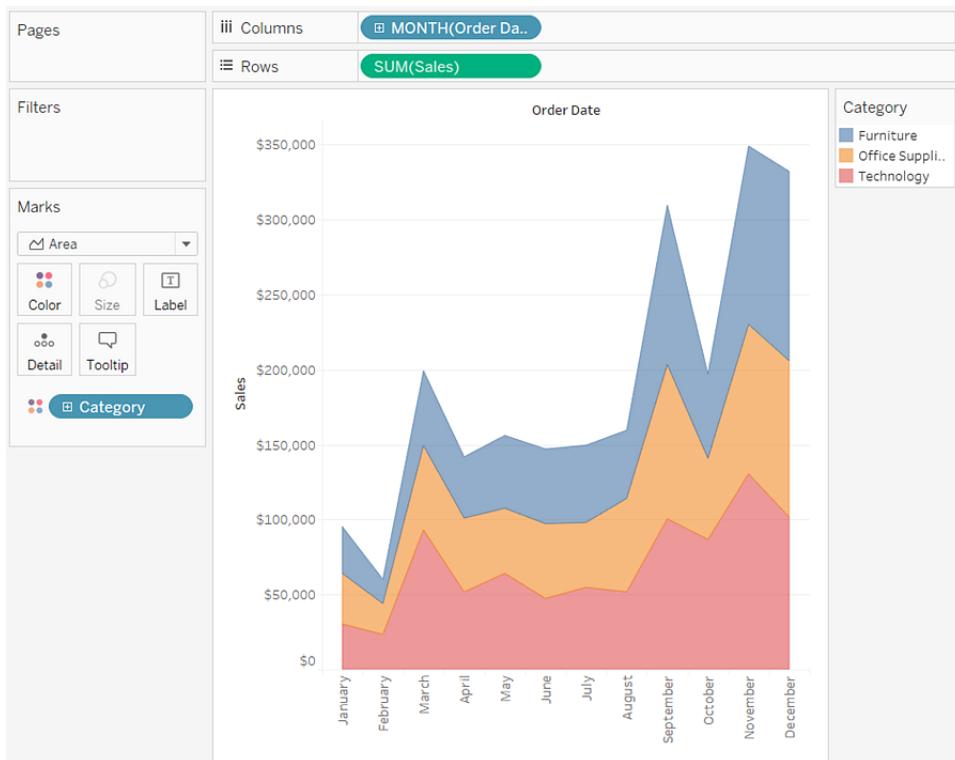
To try some hands-on exercises for building line charts, see [Building Line Charts](#) on page 72.

Area Mark

The Area mark type is useful when you have a view where the marks are stacked and do not overlap. For more information about stacked marks, see [Stack Marks on page 230](#). In an area chart, the space between each line and the next line is filled with a color. This type of chart is not the best way to show specific values along the line but it can clearly show the total values so you can get an idea of how a dimension is contributing to an overall trend. Tableau displays your data using areas when:

- The Marks card drop-down menu is set to **Automatic** and one or more measures is plotted against a date dimension or continuous dimension. The **Analysis > Stack Marks** option must also be selected.

The view below shows a date dimension on the **Columns** shelf and a measure on the **Rows** shelf. Each line represents the sales for a product category. The lines have been stacked so that they do not overlap.



Note: You can add labels, edit the color legend, and highlight areas. In addition, you can turn on color effects, such as mark borders. For more information, see [Mark Properties on page 190](#).

Square Mark

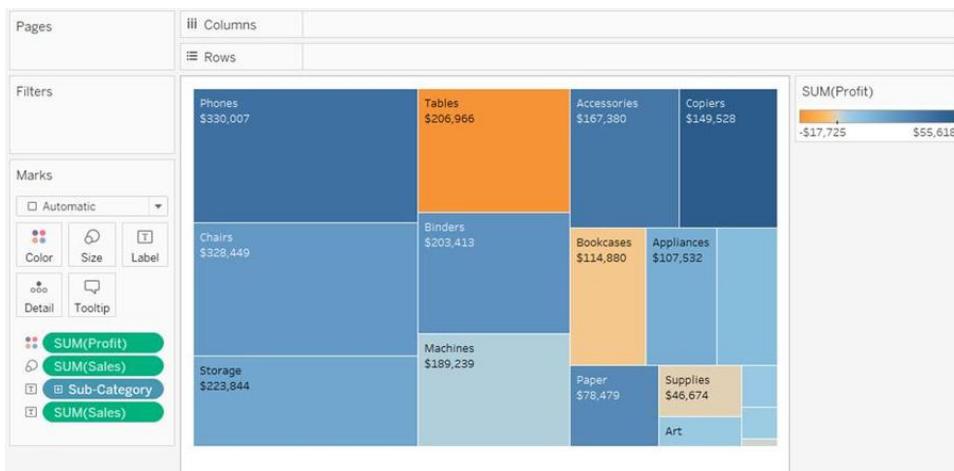
The square mark type is useful when you want to clearly see individual data points. When you select **Square** from the **Marks** card drop-down menu, Tableau displays your data using squares.

The view shown below displays several dimensions on both the rows and columns shelves. By default, the data is displayed using text. However, when you place a measure on **Color**, Tableau converts the marks to squares and creates a heat map. To try some hands-on exercises for building heat maps, see [Build a Heat Map on page 82](#).



When you add additional levels of detail by placing a dimension on **Detail**, **Color**, **Shape**, **Size**, or **Label** on the **Marks** card, the squares are displayed side-by-side and wrap to fill the cell. If the window is too small to show all of the squares, an ellipses indicates that there are more values than you can see.

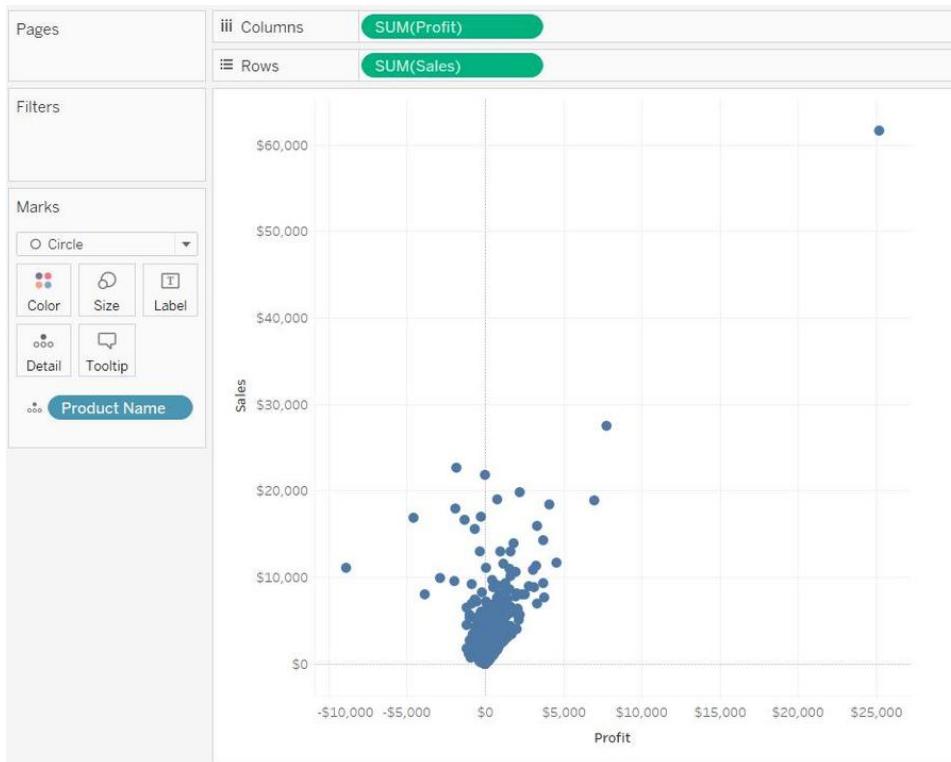
You can also use the square mark to create treemaps. A treemap displays hierarchical data as a set of nested rectangles. To try a hands-on exercise for building a treemap, see [Build a Treemap on page 104](#).



Circle Mark

When you select **Circle** from the Marks card drop-down menu, Tableau displays your data using filled circles.

If the mark type was set to **Automatic**, Tableau would display the data using a shape (that is, an open circle).

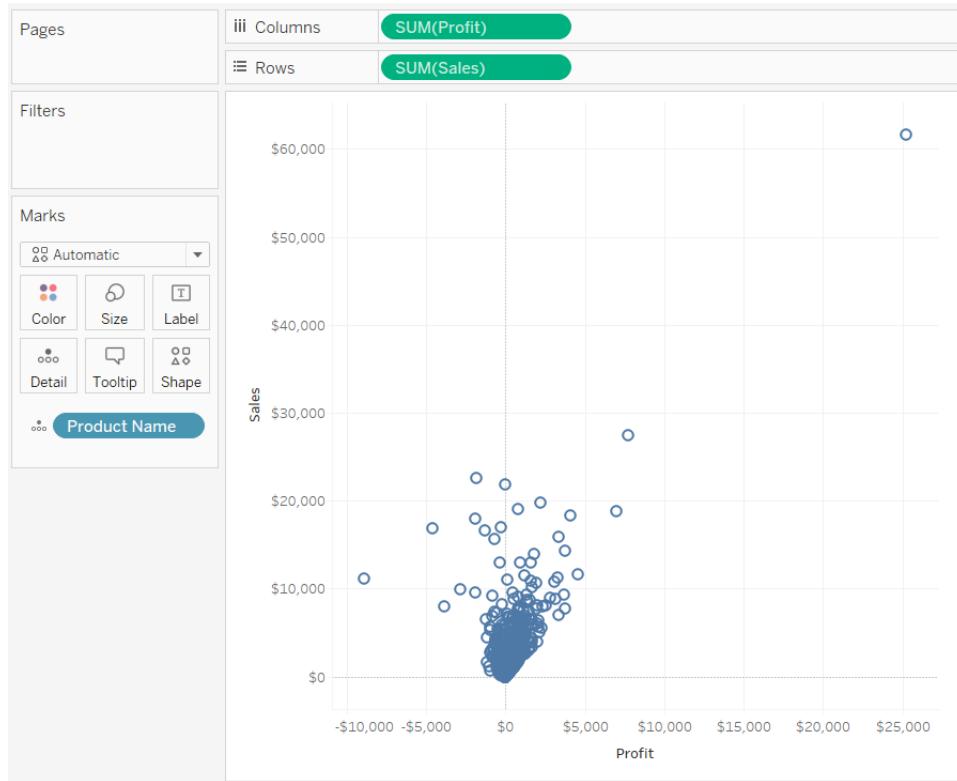


Shape Mark

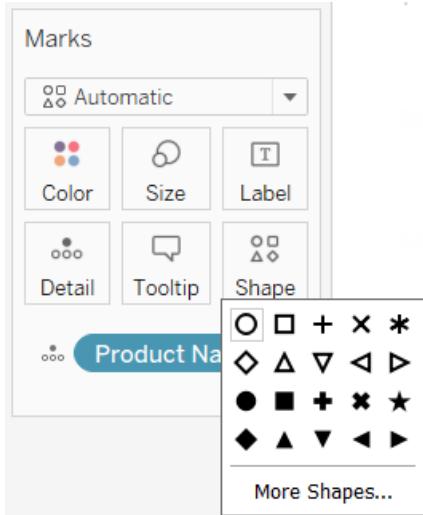
The shape mark type is useful when you want to clearly see individual data points while also viewing categories associated with those points. Tableau displays your data using a shape when:

- The Marks card drop-down menu is set to **Automatic**, and you place one or more measures on both the **Rows** and the **Columns** shelves.
- You select **Shape** from the Marks card drop-down menu.

The view shown below displays the data from two measures. Because the Marks card drop-down menu is set to **Automatic**, the data are displayed using a shape.

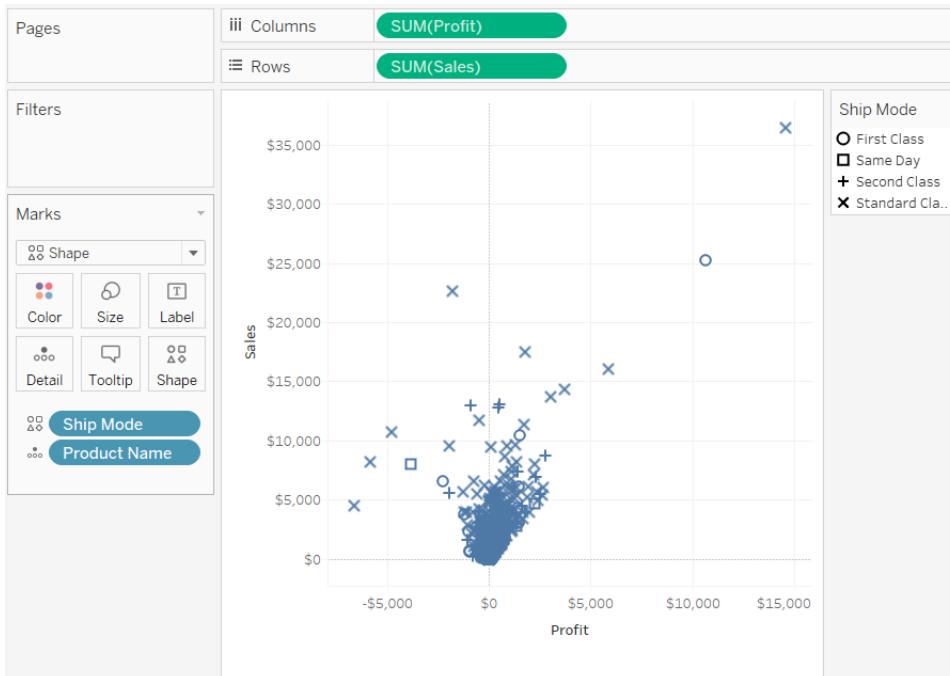


By default, the shape used is an open circle. To select a different shape, click **Shape** on the Marks card. Twenty unique shapes are available:



To add more data to the view, you can place a field on **Shape** on the Marks card. Tableau separates the marks according to the values in the field. If the field is a dimension, each member is assigned a unique shape. If the field is a measure, the measure is automatically binned into distinct buckets and each bucket is assigned a unique shape. The shape legend displays how shapes are distributed.

As shown below, the **Ship Mode** dimension is placed on **Shape** on the Marks card to encode each mark with information about how the order was shipped.



Text Mark

The Text mark type is useful when you want to display the numbers associated with one or more dimension members. This type of view is often called a text table, a cross-tab, or a pivot table. Tableau displays your data using text when:

- The Marks card drop-down menu is set to **Automatic**, and you place one or more dimensions as the inner fields on both the **Rows** and the **Columns** shelves.
- You select **Text** from the Marks card drop-down menu.

Initially, the data is displayed as **Abc**.

The screenshot shows the Tableau interface. On the left, the Data pane lists 'Pages' and 'Filters'. The Marks pane shows 'Automatic' selected, with options for Color, Size, Text, Detail, and Tooltip. In the center, 'Sheet 1' displays a text table titled 'Order Date'. The columns are 'Sub-Catego..', '2012', '2013', '2014', and '2015'. The rows list various categories: Accessories, Appliances, Art, Binders, Bookcases, Chairs, Copiers, Envelopes, Fasteners, Furnishings, Labels, Machines, and Paper. Each category has 'Abc' displayed under all years.

Sub-Catego..	2012	2013	2014	2015
Accessories	Abc	Abc	Abc	Abc
Appliances	Abc	Abc	Abc	Abc
Art	Abc	Abc	Abc	Abc
Binders	Abc	Abc	Abc	Abc
Bookcases	Abc	Abc	Abc	Abc
Chairs	Abc	Abc	Abc	Abc
Copiers	Abc	Abc	Abc	Abc
Envelopes	Abc	Abc	Abc	Abc
Fasteners	Abc	Abc	Abc	Abc
Furnishings	Abc	Abc	Abc	Abc
Labels	Abc	Abc	Abc	Abc
Machines	Abc	Abc	Abc	Abc
Paper	Abc	Abc	Abc	Abc

To complete the view, you must place a field (typically a measure) on **Text** on the Marks card. In the view below, the **Sales** measure, which is aggregated as a sum, is used to complete the table.

Pages

iii Columns YEAR(Order Date)

Rows Sub-Category

Filters

Marks

Automatic

Color Size Text

Detail Tooltip

SUM(Sales)

Sub-Catego..	Order Date			
	2011	2012	2013	2014
Accessories	\$25,014	\$40,524	\$41,896	\$59,946
Appliances	\$15,314	\$23,241	\$26,050	\$42,927
Art	\$6,058	\$6,237	\$5,910	\$8,914
Binders	\$43,488	\$37,453	\$49,485	\$72,986
Bookcases	\$20,037	\$38,544	\$26,275	\$30,024
Chairs	\$77,242	\$71,735	\$83,919	\$95,554
Copiers	\$10,850	\$26,179	\$49,599	\$62,899
Envelopes	\$3,856	\$4,512	\$4,730	\$3,379
Fasteners	\$661	\$545	\$960	\$858
Furnishings	\$13,826	\$21,090	\$27,874	\$28,915
Labels	\$2,841	\$2,956	\$2,827	\$3,861
Machines	\$62,023	\$27,764	\$55,907	\$43,545
Paper	\$14,835	\$15,288	\$20,638	\$27,718
Phones	\$77,391	\$68,314	\$78,660	\$105,643
Storage	\$50,329	\$45,048	\$58,632	\$69,834
Supplies	\$14,394	\$1,952	\$14,278	\$16,049
Tables	\$46,088	\$39,150	\$60,833	\$60,894

Note: To create a cross-tab of any data view, select **Worksheet > Duplicate as Crosstab**.

When you add additional levels of detail by placing a dimension on **Detail**, **Color**, **Shape**, **Size**, or **Text** on the Marks card, the values are shown side-by-side and wrap to fill the cell. Allowing the text to stack up makes it possible to create word cloud visualizations like the one shown below.

Pages

iii Columns

Rows

Filters

Marks

Text

Color Size Text

Detail Tooltip

SUM(Profit) SUM(Sales) State

If the application window is too small to fill all of the text, the cell displays an ellipses to indicate that there are more values than can fit.

The screenshot shows the Tableau Data Explorer interface. On the left, there's a sidebar with 'Pages' (selected), 'Filters', and 'Marks' (set to 'Automatic'). Under 'Marks', there are buttons for Color, Size, Text, Detail, Tooltip, and two selected items: 'Manufacturer' and 'SUM(Sales)'. At the top right, there are buttons for 'Columns' (labeled 'YEAR(Order Date)') and 'Rows' (labeled 'Sub-Category'). The main area displays a text table titled 'Order Date' with columns for Sub-Category and years 2012, 2013, 2014, and 2015. The data includes:

Sub-Catego..	2012	2013	2014	2015
Accessories	\$552 ...	\$438 ...	\$32 ...	\$522 ...
Appliances	\$715 ...	\$294 ...	\$114 ...	\$156 ...
Art	\$178 ...	\$98 ...	\$42 ...	\$153 ...
Binders	\$6 \$13 ...	\$17 \$6 ...	\$11 ...	\$9 \$64 ...
Bookcases	\$2,095 ...	\$7,406 ...	\$5,161 ...	\$3,954 ...
Chairs	\$461	\$1,084 ...	\$230 ...
Copiers	\$1,200 ...	\$2,700 ...	\$480 ...	\$480 ...
Envelopes	\$69 ...	\$51 ...	\$45 ...	\$9 \$64 ...
Fasteners	\$26 ...	\$22 ...	\$78 ...	\$68 ...

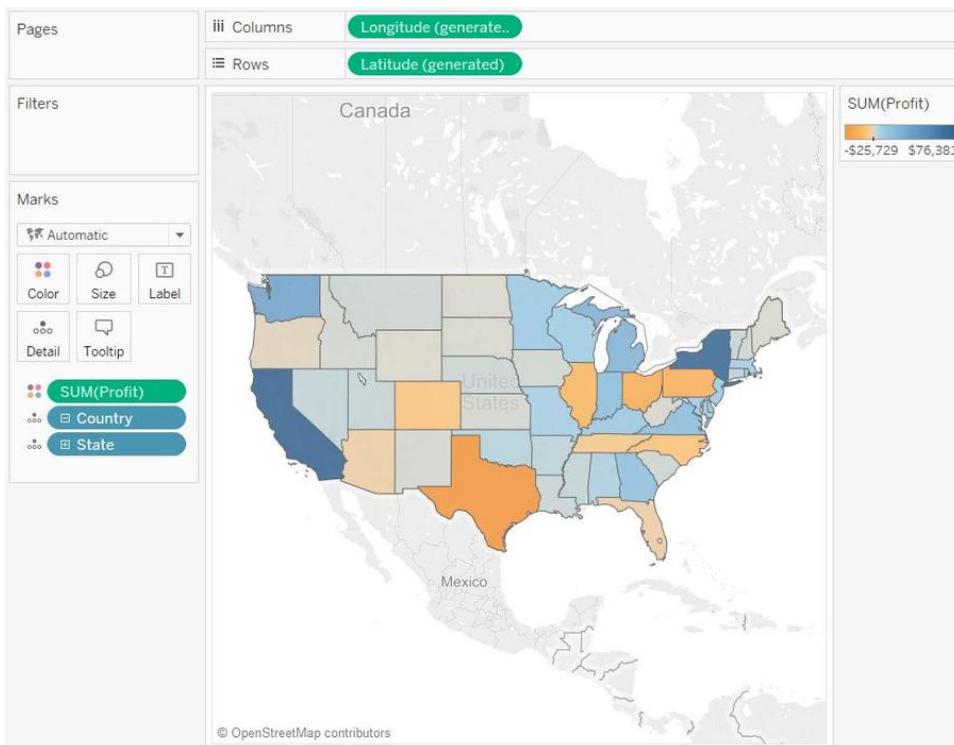
To try some hands-on exercises for building text tables, see [Build a Text Table](#) on page 65.

Filled Map Mark

The Filled Map mark type uses geocoding to fill a polygon with a color based on data. The primary use of the filled map mark type is for creating choropleth maps (also known as thematic maps or data maps). The area to be filled is defined by the geographic fields used in the view. Tableau displays data using a filled map when:

- A geographic dimension is on **Detail** on the Marks card, along with **Latitude** and **Longitude** on the **Rows** and **Columns** shelves. A measure or continuous dimension is added to **Color** on the Marks card.
- You select **Filled Map** from the Marks card drop-down menu.

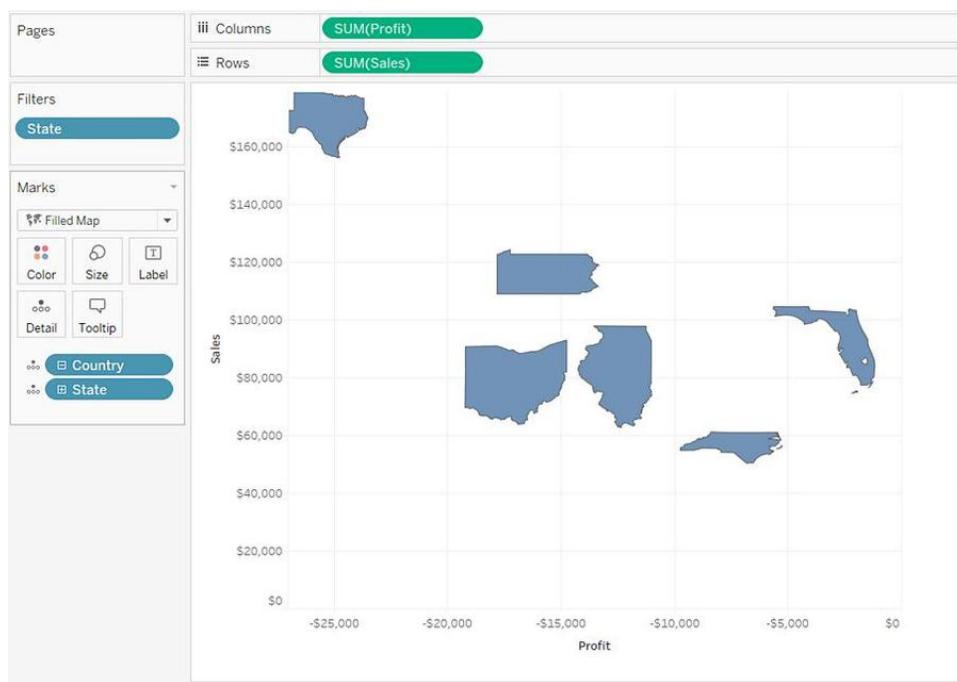
The data view below shows a map with the **State** geographic dimension on **Detail** on the Marks card. The **Profit** measure is on **Color** on the Marks card.



The polygon for each state is filled with a color based on the profit for that state. The profit in Texas is low (in fact, negative) while the profit in California is high.

You can also use the Filled Map mark type when you are plotting two measures against each other. By default, placing measures on both the **Rows** and **Columns** shelves creates a scatter plot. However, adding a geographic dimension allows you to change the mark type to Filled Map. Each mark becomes the area defined by the geographic field. For example, the view below shows **Sales** versus **Profit** for a selection of states. Rather than showing a round mark

with a label for each state, the view uses the Filled Map mark type to draw the outline of each state. While this works well for recognizable areas, it is not the best choice for areas that are similar in shape or difficult to recognize.



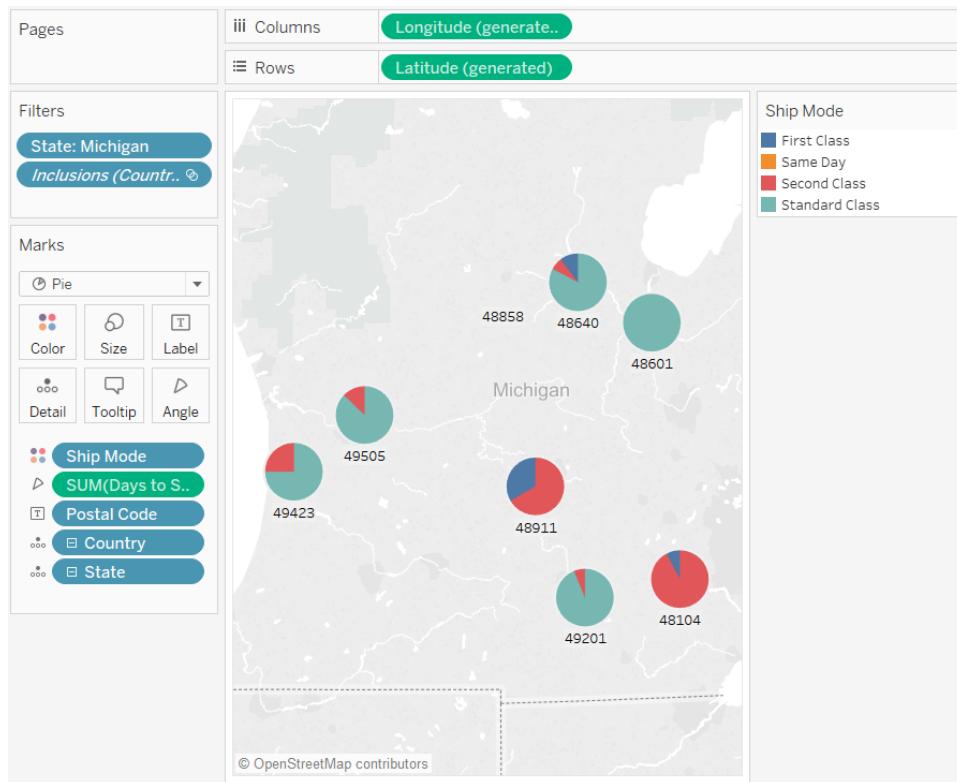
Pie Mark

You can use the Pie mark to show proportions. Although generally this type of information can be better shown using stacked bar charts, there are cases where pie marks can be very effective—for example, when trying to convey the percentage allocation of marketing expenses by state where the spending of geographically close states are very relevant.

Tableau will never use the Pie mark as an automatic mark type, but you can select **Pie** from the Marks card drop-down menu.

When you select the Pie mark type, an additional target named Angle is displayed on the Marks card. The Angle target determines the angular measure of the pie wedges. For example, if you place a measure such as **Sales** on **Angle** on the Marks card, the 360 degrees of the pie corresponds to the total sum of sales, and each wedge is divided by the values of the field on **Color** on the Marks card.

The view below shows the time it took to ship products by various ship modes. The data overlays a map and shows data by state. We can see that standard class shipping takes the longest in most areas.



To try some hands-on exercises for pie charts, see [Build a Pie Chart](#) on page 102 and [Build a Map View](#) on page 118.

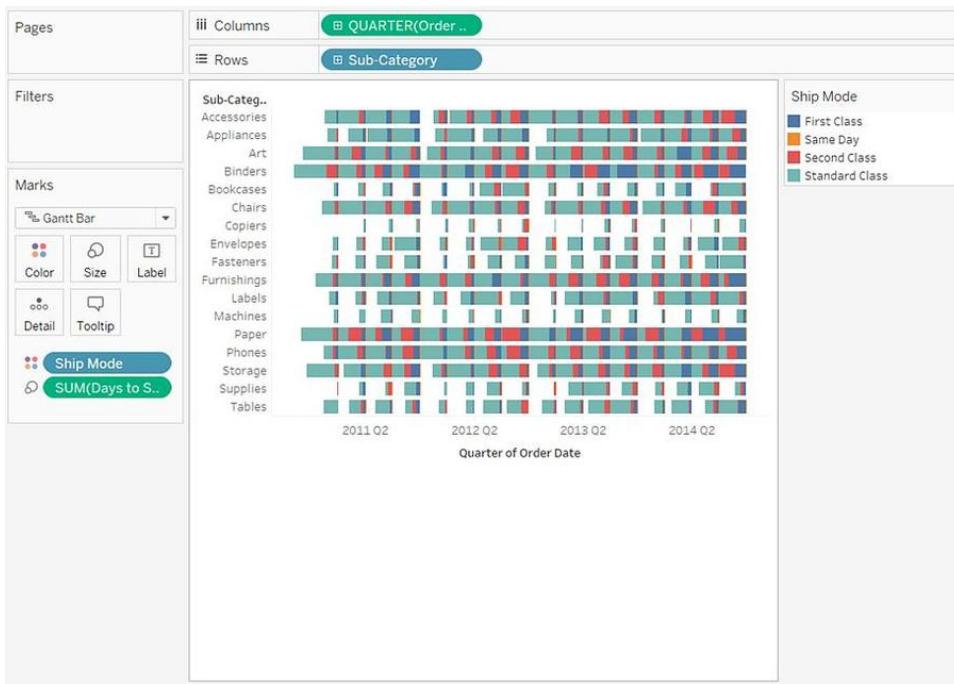
Gantt Bar Mark

The Gantt Bar mark type is useful for viewing dates, project plans, or the relationships between different quantitative variables. Tableau displays your data using Gantt bars when:

- The Marks card drop-down menu is set to **Automatic** and you place one or more dimensions on either the **Columns** shelf or the **Rows** shelf, and then plot the dimensions against a continuous quantity.
- You select **Gantt Bar** from the Marks card drop-down menu.

The distinguishing characteristic of Gantt Bars is that the length of every mark is proportional to the measure placed on **Size** on the Marks card.

The view below displays a dimension as a function of a continuous date. If the Marks card drop-down menu is set to **Automatic**, the data would be displayed using bars. Selecting **Gantt Bar** and adding additional fields will display a view like the following.



In particular, placing the **Days to Ship** measure on **Size** on the Marks card causes every bar in the view to be drawn with a length that indicates the delivery time of an order. Additionally, placing the **Ship Mode** dimension on **Color** on the Marks card causes each bar to be colored according to the ship mode.

To try a hands-on exercise for building a Gantt Chart, see [Build a Gantt Chart](#) on page 96.

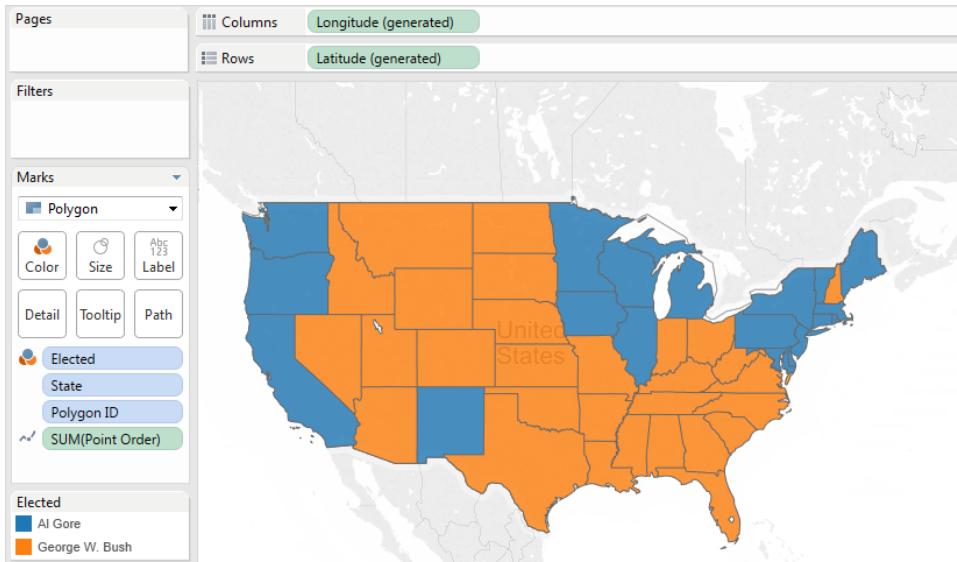
Polygon Mark

Polygons are points connected by lines enclosing an area. The polygon mark type is useful when you want to connect points to create data areas. Tableau displays data using polygons when you select Polygon from the **Mark** menu.

The polygon mark is not commonly used and often requires a specially constructed data source.

The view shown below comes from a specially constructed data source that holds geographic and election data. It displays the 48 contiguous US states as a function of latitude and longitude and color-encodes each state by the 2000 presidential election results.

If **Mark** is set to Automatic, the data will be displayed using a shape. Manually selecting Polygon and adding additional fields to the view causes a different view to be created.



Every state is considered to be a polygon in the data source. The **PolygonID** field on the **Detail** target is distinct for each US state. You can remove states from the view by filtering this field.

Additionally, you can specify the drawing order of the lines that constitute each polygon by placing a field on the **Path** target. In this example, the **PointOrder** measure is used to draw each state.

Mark Properties

You can control the mark properties using the Marks card. For example, you can control the colors, size, shape, etc. of the marks in the view. Drag fields to each property to encode the

marks using your data. Click each property on the marks card to open a drop-down control where you can further tune the mark properties.

Color Properties

All marks have a default color, even when there are no fields on **Color** on the **Marks** card. For most marks, blue is the default color; for text, black is the default color.

When you drop a field on **Color** on the **Marks** card, Tableau applies different colors to marks, based on the field's values. The effect of color-encoding your data view depends on whether the field you drop on **Color** is discrete, as indicated by a blue background:

 **Category**

, or continuous, as indicated by a green background:

 **SUM(Sales)**

. Typically, dimensions are discrete and measures are continuous. For discrete fields, Tableau assigns a categorical palette, and for continuous fields, a quantitative palette.

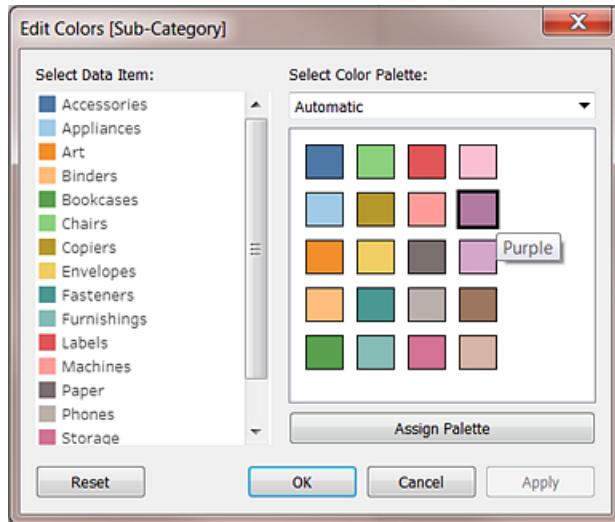
You can also use the **Color** drop-down control on the Marks card to specify other color properties such as opacity, borders, and halos. For more information, see [Configure color effects on page 197](#).

Note: Color selections for a field are shared across multiple worksheets that use the same data source. For example, if you assign the color orange to the West value for the **Region** field in one worksheet, West will automatically be orange in all other worksheets in the workbook. To set the default color for a field, right-click (control-click on Mac) the field in the **Data** pane, and then select **Default Properties > Color**.

Categorical palettes

When you drop a field with discrete values (typically a dimension) on **Color** on the **Marks** card, Tableau uses a categorical palette and assigns a color to each value of the field. Categorical palettes contain distinct colors that are appropriate for fields with values that have no inherent order, such as departments or shipping methods.

To change colors for values of a field, click in the upper-right corner of the color legend to open a context menu, and then select **Edit Colors** to open the Edit Colors dialog box.



Change the color for a value

1. Click a value on the left, under **Select Data Item**.
2. Click a new color in the palette on the right. You can hover over a swatch to identify the color.
3. Repeat for as many values that you want to change.
4. Click **OK** to exit the Edit Colors dialog box.

Select a different palette

The **Select Color Palette** drop-down list in the Edit Colors dialog box provides color palettes that you can use for discrete fields. The list contains both categorical and ordinal palettes.

At the top of the list are categorical palettes, such as *Tableau 10*. As noted above, categorical palettes are appropriate for discrete fields with no inherent order.

At the bottom of the list are ordinal palettes such as *Orange*. Ordinal palettes contain a range of related colors and are appropriate for fields that have an associated order, such as dates or numbers.

After you select a palette, click **Assign Palette** to automatically assign the new palette colors to the members in the field.

To return to the Automatic palette and the default color assignments, click **Reset** in the Edit Colors dialog box.

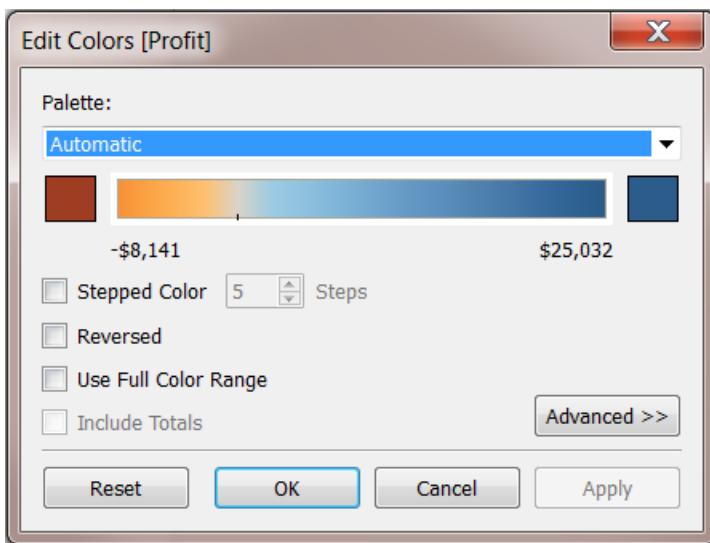
Quantitative palettes

When you drop a field with continuous values on the **Marks** card (typically a measure), Tableau displays a quantitative legend with a continuous range of colors.



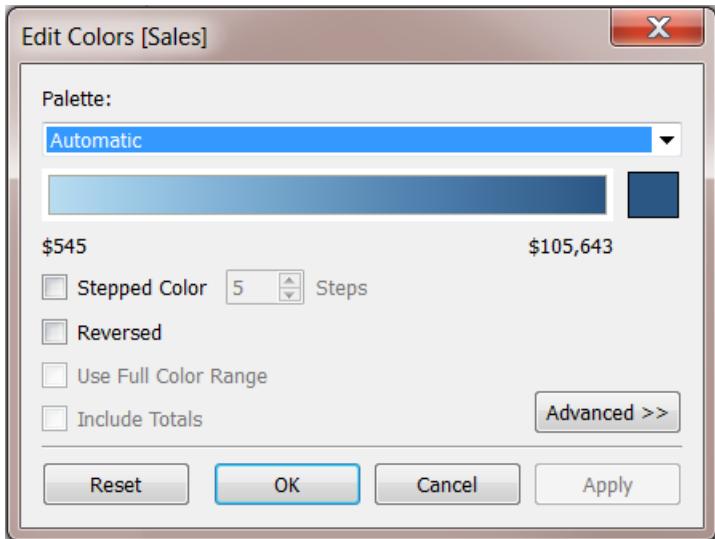
You can modify the colors used in the range, the distribution of color, and other properties. To edit colors, click in the upper right of the color legend to open a context menu, and select **Edit Colors** to open the Edit Colors dialog box.

When there are both negative and positive values for the field, then the default range of values will use two color ranges and the Edit Colors dialog box for the field has a square color box on either end of the range. This is known as a diverging palette.



Edit Colors dialog box for a diverging palette

When all values are positive, then the default range of values will use a single color range and the Edit Colors dialog box for the field has a square color box only at the right end of the range. This is known as a sequential palette.



Edit Colors dialog box for a sequential palette

You can specify whether Tableau uses a diverging or a sequential palette for a continuous field on **Color**, and also configure the range of colors for the field's values.

The **Palette** drop-down list provides a range of color palettes from which you can choose. There are two types of quantitative palettes available for continuous fields:

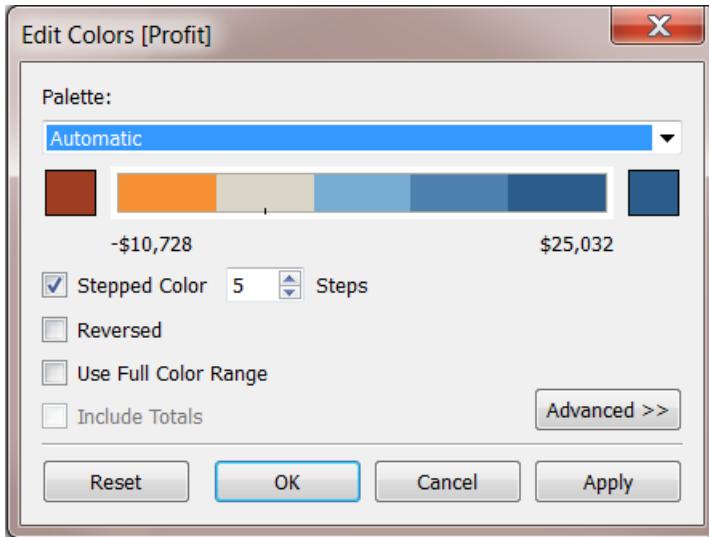
- All palettes with *Diverging* in the name are diverging quantitative palettes—for example, *Orange-Blue Diverging*. You can choose a diverging palette for any continuous field—it isn't necessary for the range of values to contain both positive and negative numbers. To change the colors for a diverging palette, click one of the square color boxes at either end of the palette spectrum to open a color configuration dialog box, which is part of your computer's operating system.
- All other palettes are sequential quantitative palettes. To change the colors for a sequential palette, click the square color box at the right end of the palette spectrum to open the color configuration dialog box, which is part of your computer's operating system.

Options for quantitative palettes

The following options are available in the Edit Colors dialog box for a continuous field.

Stepped Color

Click **Stepped Color** to group values into uniform bins, where each bin is associated with a color. Use the spin control to specify how many steps (bins) to create. For example, for a range of values from 0 to 100 you could specify five steps, to sort values into five bins (0-20, 20-40, etc.).



If a diverging color palette is selected, the point where the palette transitions between colors is shown on the color ramp with a small black tick mark. When the number of steps is odd, the mark is placed in the middle of the transitional step. When the number of steps is even, the mark is placed at the boundary between the steps where the color changes.

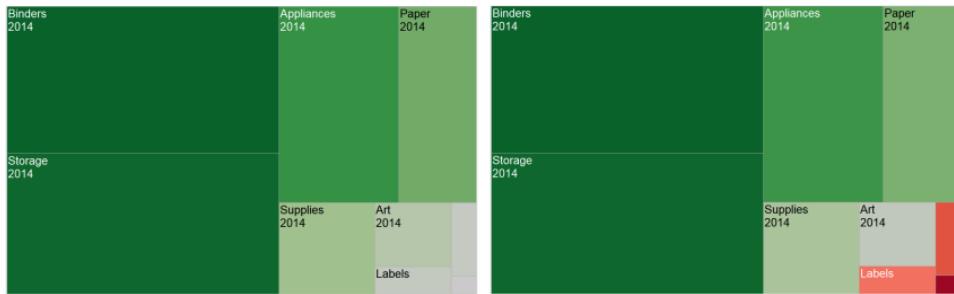
Reversed

Click **Reversed** to invert the order of colors in the range. For example, if you want lower values to have a darker intensity in a sequential palette, reverse the palette. For a diverging palette, reversing the color palette means swapping the two colors in the palette, in addition to inverting the shades within each color range.

Use Full Color Range

With a diverging (two-color) palette, you can select to **Use Full Color Range**. Tableau assigns both the starting number and the ending number a full intensity for both color ranges. So if the range is from -10 to 100, the color representing negative numbers will be adjusted to change in shade much more quickly than the color representing positive numbers. If you do not select **Use Full Color Range**, Tableau assigns the color intensity as if the range of values was from -100 to 100, so that the change in shade is the same on both sides of zero. This means there will be much less change on the negative side, where actual values only range from -10 to 0, than on the positive side, where values range from 0 to 100.

The image on the left below shows a red-green diverging color palette for values from -858 to 72,986. Without using the full color range, -858 (associated with the small box at the lower right of the chart) shows as gray, because -858 is only about 1% as far to the negative side as 72,986 is to the positive side. When the full color range is used, as in the image on the right, -858 shows as a dark red, equal in intensity to the maximum positive value.



Limit the color range

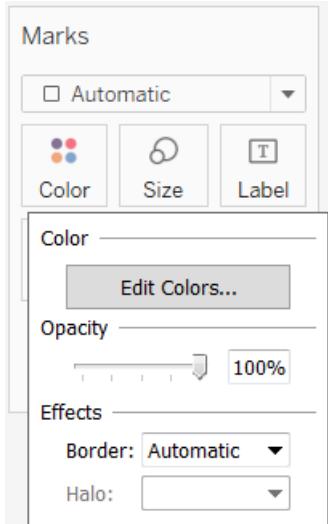
When you click **Advanced** in the Edit Colors dialog box, you can choose to specify the start, end, and center RGB values for the range by selecting the check box and typing a new value into the field. The **Start** value is the lower limit in the range, the **End** value is the upper limit, and the **Center** value is where the neutral color is located on a diverging color palette.

Reset the color range

To return to the Automatic palette and the default color assignments, click **Reset** in the Edit Colors dialog box.

Configure color effects

Click the **Color** drop down on the **Marks** card to configure additional **Color** settings not related to the actual colors shown.



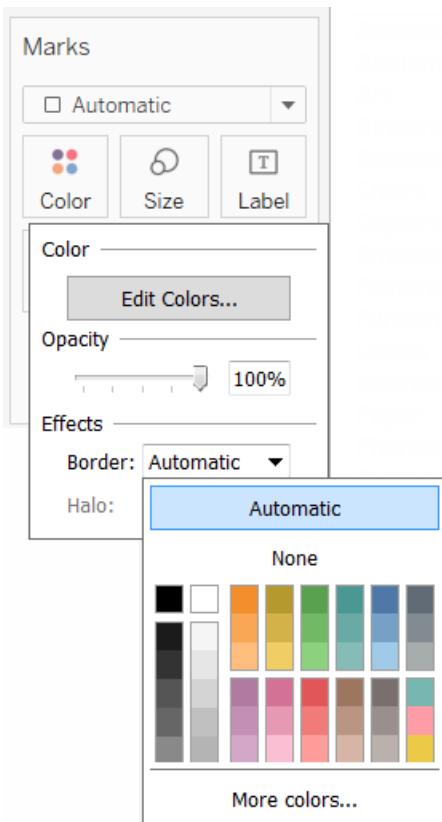
Opacity

Modify the opacity of marks by moving the slider.

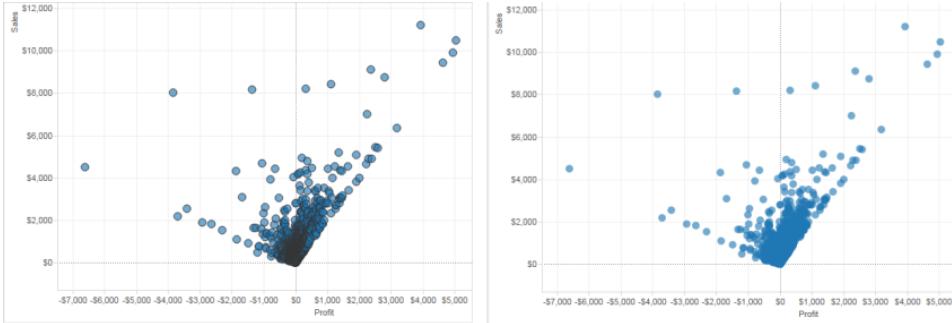
Adjusting opacity is especially useful in dense scatter plots or when you are looking at data overlaying a map or background image. As you slide the slider toward the left, marks become more transparent.

Mark borders

By default, Tableau displays all marks without a border. You can turn on mark borders for all mark types except text, line, and shape. On the **Color** drop-down control, select a mark border color.



Borders can be useful for visually separating closely spaced marks. For example, the views below show a scatterplot with mark borders turned on (left) and turned off (right). When borders are turned on, marks are easier to distinguish in areas where they are tightly clustered.



Note: You can also use the opacity setting to show the density of marks.

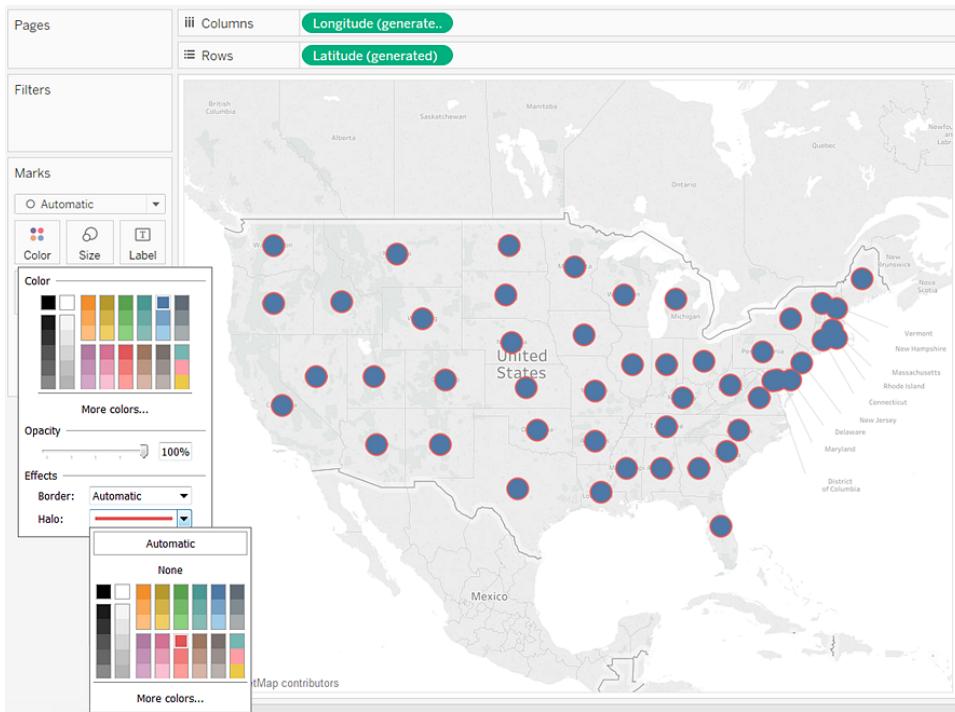
When you are viewing a large number of color-encoded small marks, it is usually better to leave mark borders off. Otherwise borders can dominate the view, making it difficult to see the color encoding.

For example, the views below show bars that are segmented by a large number of color-encoded dimension members. With mark borders turned on (right), some of the narrower marks are difficult to identify by color. With borders turned off (left), the marks are easy to distinguish.



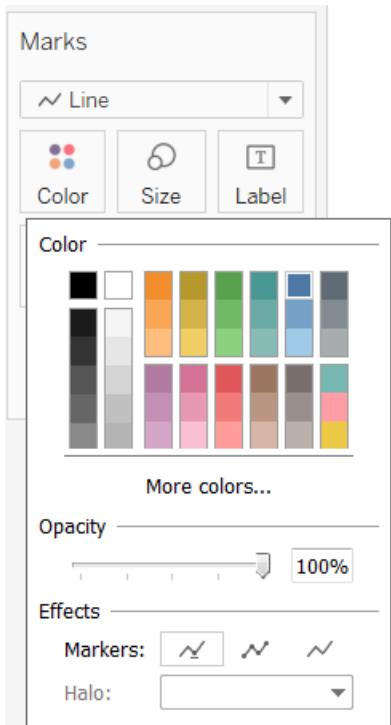
Mark halos

To make marks more visible against a background image or map, surround each mark with a solid contrasting color called a halo. Mark halos are available when you have a background image or a background map. On the **Color** drop-down control, select a mark halo color.



Markers

When you are using the Line mark type, you can add a marker effect to show or hide points along the line. You can show selected points, all points, or no points. On the **Color** drop-down control, select a marker in the **Effects** section.



Example – Multiple Fields on Color

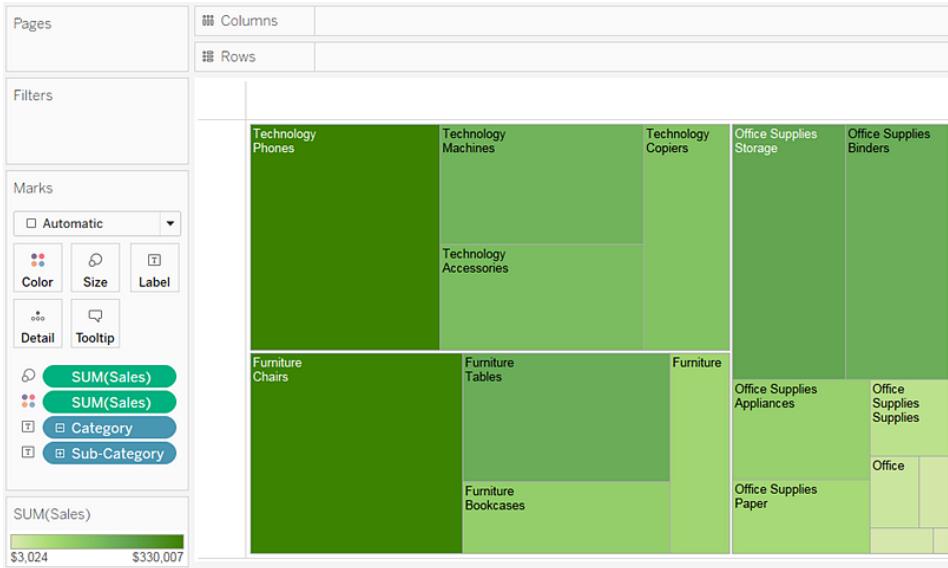
If you drop one field on Color and then drop a different field on Color, the second field replaces the first field. However, depending on the chart type, for example treemaps and bullet graphs, you can put multiple fields on color. You can use one field to set the hue, and the other to show gradations within that hue.

Follow these steps, using the Sample - Superstore data source, to build a treemap with two fields on Color.

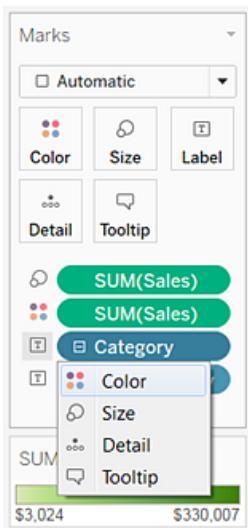
1. Drag **Category** and **Sub-Category** to **Columns**.
2. Drag **Sales** to **Size** on the **Marks** card.
3. Click **Show Me** in the toolbar, then select the Treemap chart type.



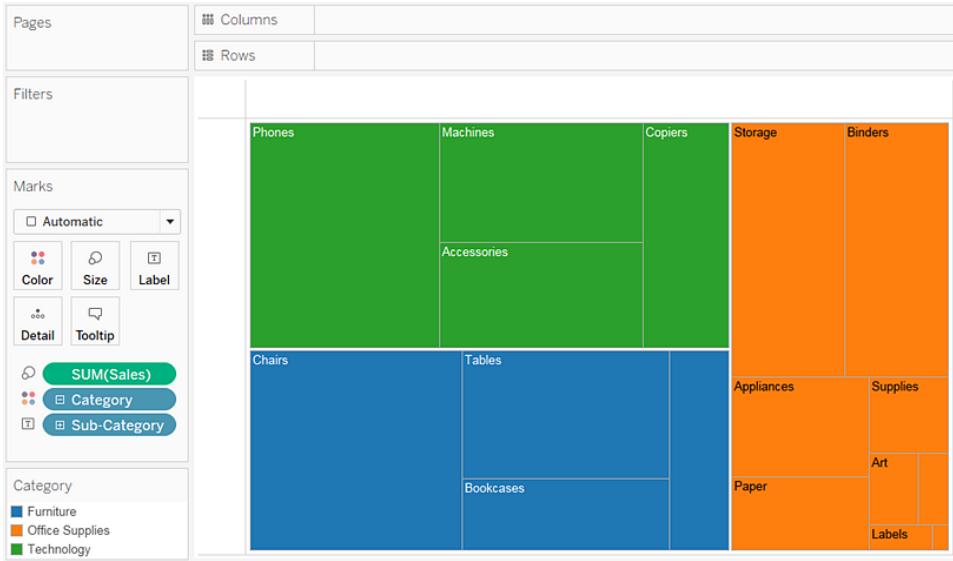
Tableau moves all fields to the **Marks** card, putting **SUM(Sales)** on both **Size** and **Color**, and **Category** and **Sub-Category** on **Label**:



4. Click the label icon to the left of **Category** on the **Marks** card and select **Color**:

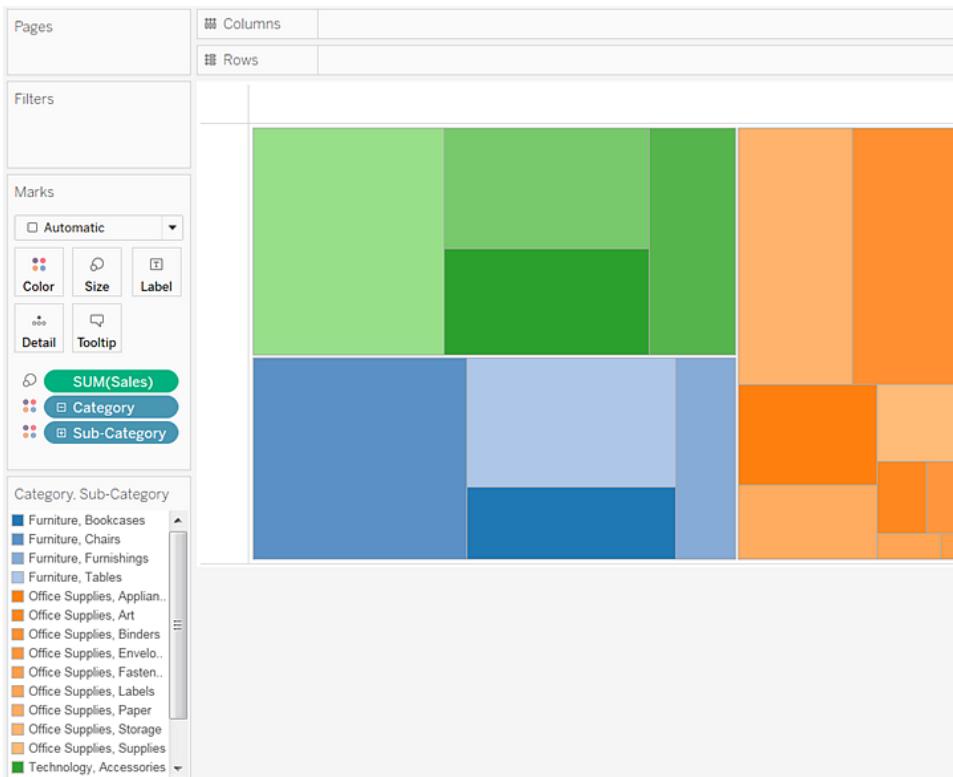


Category replaces **SUM(Sales)** on **Color**. The marks are still sized by the sum of **Sales**, but now they are colored by **Category**:



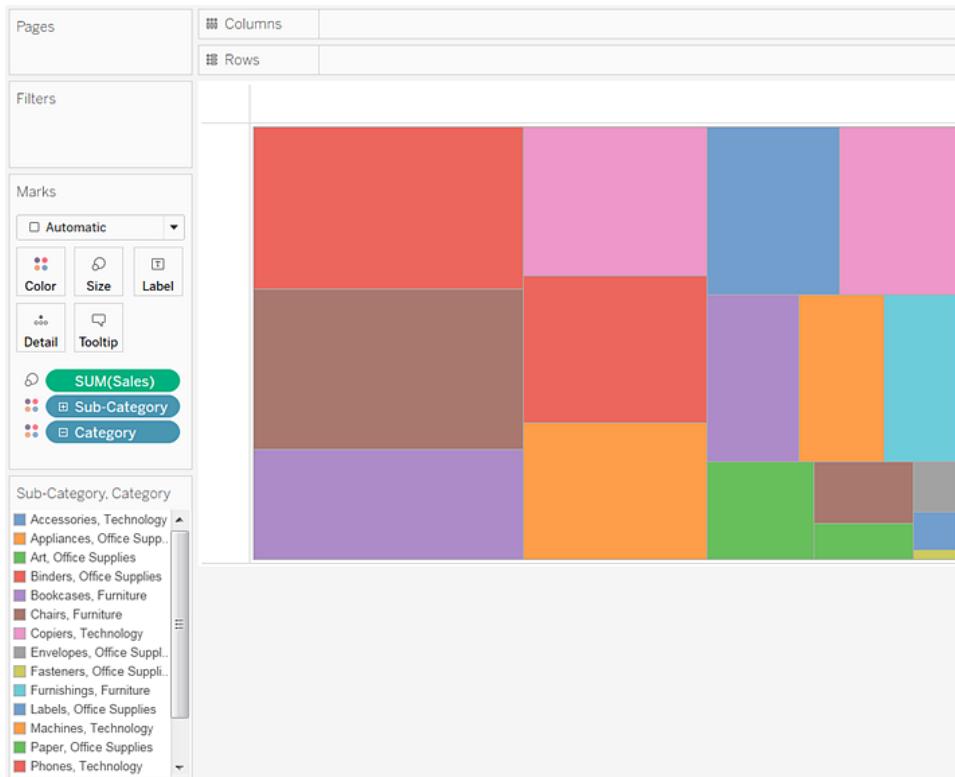
- Click the label icon to the left of **Sub-Category** on the Marks card and select **Color**.

Tableau uses distinct, categorical colors for the first field, **Category**, and a range of sequential shades to distinguish values for the second field, **Sub-Category**:



The size of the individual rectangles is still determined by **Sales**, per **Category** and **Sub-Category**.

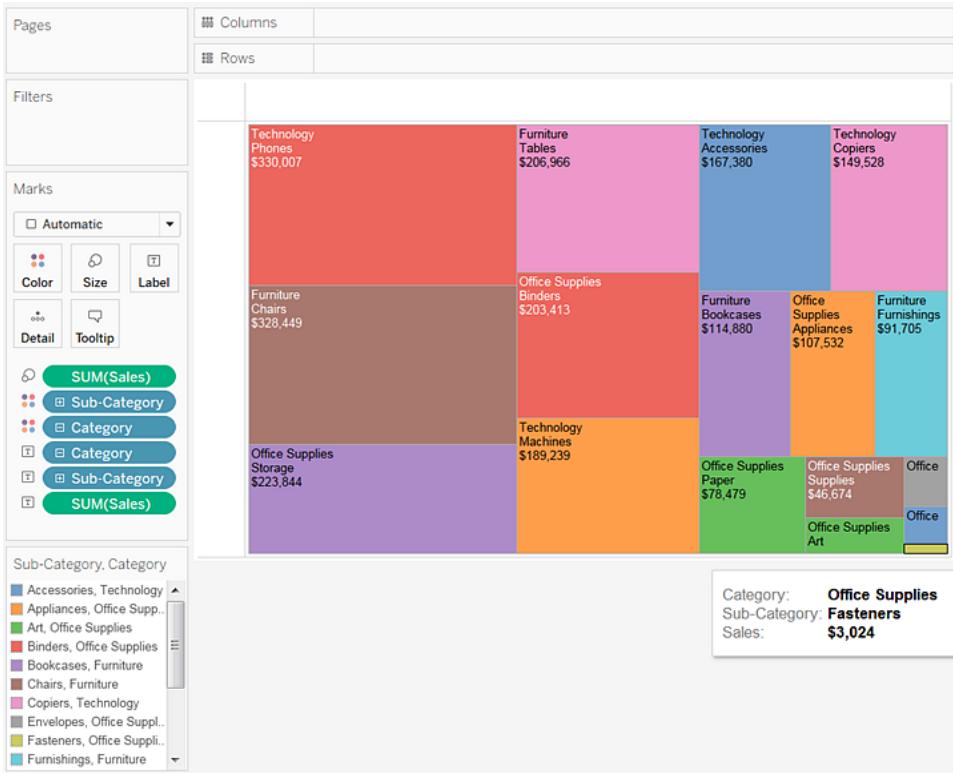
The two fields on **Color (Category and Sub-Category)** are related within a hierarchy, so if you swap their positions on the **Marks** card, moving **Sub-Category** to be above **Category**, the effect is the same as if you had removed **Category** from the view altogether. The treemap changes to show a uniquely colored rectangle for each **Sub-Category**:



When the two fields on Color are not related within a hierarchy, you can switch the order of the fields on the Marks card so that the field that was used for categorical colors was used for sequential shades, and vice versa.

If you aren't satisfied with the colors that Tableau used, double-click the color legend to change them.

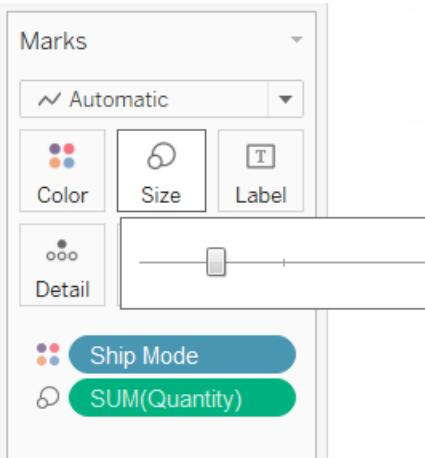
6. Make the view more readable by adding **Category**, **Sub-Category**, and **Sales** to **Label**. Users can hover to see tooltips for any rectangle that is too small to show text by default.



Size Properties

The **Size** property allows you to encode data by assigning different sizes to the marks in a data view. Depending on whether you use a discrete or continuous field, you will add either categorical or quantitative size encodings.

You can change the size of the marks using the slider on the **Size** drop-down control on the **Marks** card.



The Size slider affects different marks in different ways, as described in the following table.

Mark Type	Description
Circle, Square, Shape, Text	Makes the mark bigger or smaller.
Bar, Gantt Bar	Makes bars wider or narrower.
Line	Makes lines thicker or thinner.
Polygon	You cannot change the size of a polygon.
Pie	Makes the overall size of the pie bigger and smaller.

The size of your data view is not modified when you change marks using the **Size** slider. However, if you change the view size, the mark size might change to accommodate the new formatting. For example, if you make the table bigger, the marks might become bigger as well.

Categorical sizes

When you place a discrete field on **Size** on the **Marks** card, Tableau separates the marks according to the members in the dimension, and assigns a unique size to each member. Because size has an inherent order (small to big), categorical sizes work best for ordered data like years or quarters.

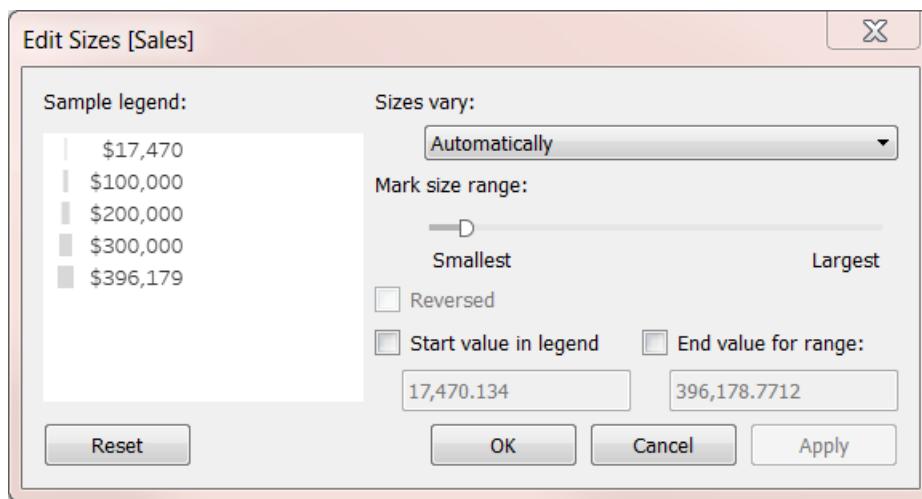
Size-encoding data with a discrete field separates the marks in the same way as the **Detail** property does, and then provides additional information (a size) for each mark. For more information about the **Detail** property, see [Detail Properties on page 216](#). When you add categorical size encoding to a view, Tableau displays a legend showing the sizes assigned to each member in the field on the **Size** target. You can modify how these sizes are distributed using the Edit Sizes dialog box.

Edit categorical sizes:

1. Click the drop-down arrow in the right-hand corner of the Size legend and select **Edit Sizes** from the legend menu. For more information about legends, see [Legends on page 239](#).
2. In the Edit Sizes dialog box, drag the slider to adjust the sizes assigned to each member.

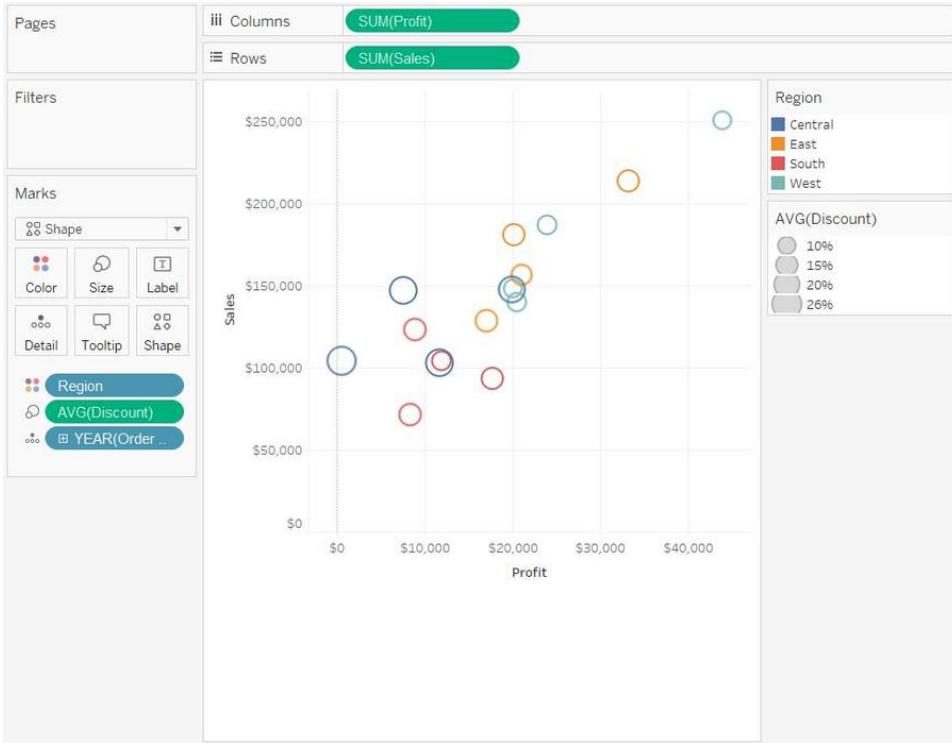
In the Edit Sizes dialog box, sizes are displayed on the left and a size range slider is shown on the right. The sizes assigned to each member are distributed across the specified range.

You can also select **Reversed** to assign the largest mark to the smallest value and the smallest mark to the largest value.



3. Click **OK** to save your changes.

The view below shows profit and order quantity broken out by region and order date. The discount customers received with each order is indicated by the size of the mark.



Quantitative sizes

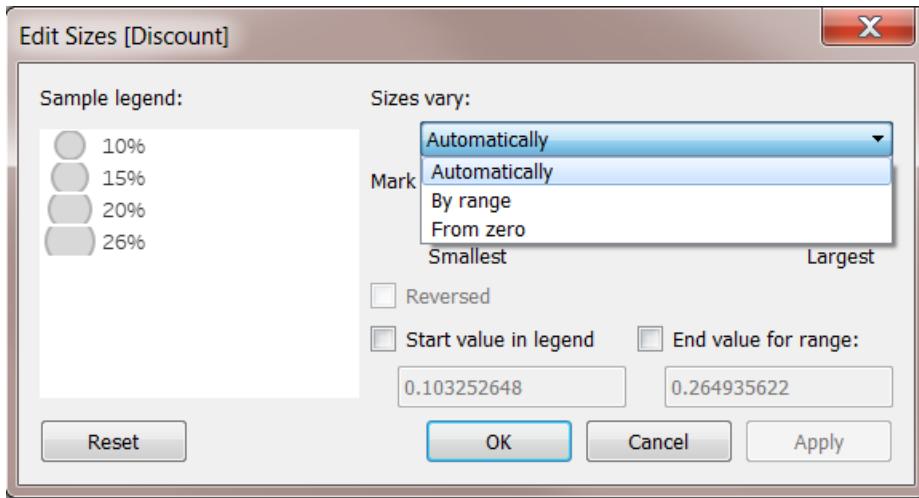
When you place a continuous field on **Size** on the **Marks** card, Tableau draws each mark with a different size using a continuous range. The smallest value is assigned the smallest sized mark and the largest value is represented by the largest mark.

When you add quantitative size encoding to the view, Tableau displays a legend showing the range of values over which sizes are assigned. You can modify how these sizes are distributed using the Edit Sizes dialog box.

Edit quantitative sizes:

1. Click the drop-down arrow in the right-hand corner of the Size legend and select **Edit Size** from the legend menu t.
2. In the **Edit Sizes** dialog box, in the **Sizes vary** drop-down box, select one of the following options to vary the sizes:
 - **Automatically** - Selects the mapping that best fits your data. If the data is numeric and does not cross zero (all positive or all negative), the **From zero** mapping is used. Otherwise, the **By range** mapping is used.
 - **By range** - Uses the minimum and maximum values in the data to determine the distribution of sizes. For example, if a field has values from 14 to 25, the sizes are distributed across this range.

- **From zero** - Sizes are interpolated from zero, assigning the maximum mark size to the absolute value of the data value that is farthest from zero.



3. Use the range slider to adjust the distribution of sizes. When the **From zero** mapping is selected from the **Sizes vary** drop-down menu, the lower slider is disabled because it is always set to zero.
4. Select **Reversed** to assign the largest mark to the smallest value and the smallest mark to the largest value. This option is not available if you are mapping sizes from zero because the smallest mark is always assigned to zero.
5. To modify the distribution of sizes, select the **Start value in legend** and **End value for range** check boxes and type beginning and end values for the range.
6. Click **OK** to save your changes.

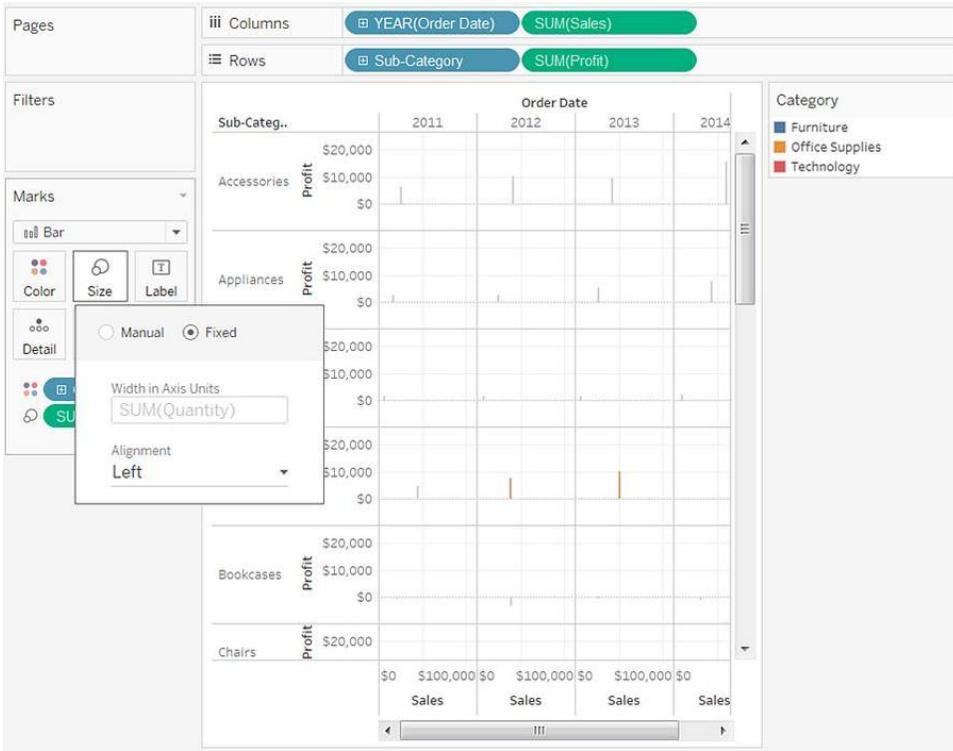
The view below analyzes the time it takes to ship products based on their ship mode, order date, and the size of order. The size of each mark represents the order quantity while the color represents the ship mode. You can see that most products ship within one or four days. However, larger orders tend to take approximately four to six days and are shipped Standard Class. Curiously, there are a couple of small Same Day orders that were not delivered same day.



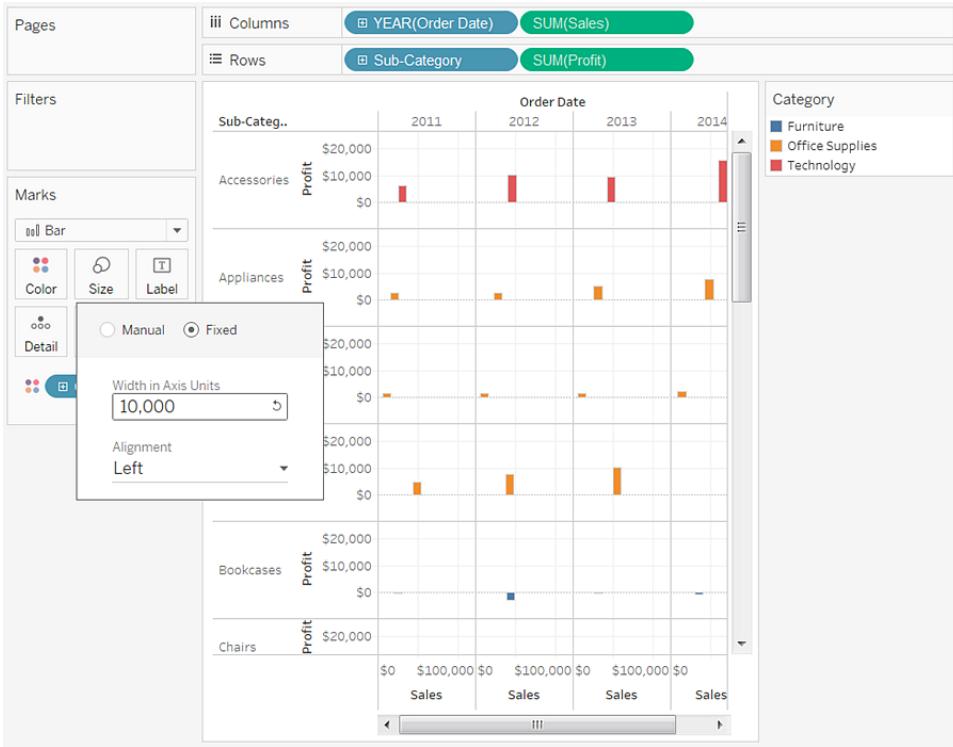
Continuous axis mark sizing

For views where the mark type is **Bar** and there are continuous (green) fields on both **Rows** and **Columns**, Tableau supports additional options and defaults for sizing the bar marks on the axis where the bars are anchored.

- The bar marks in histograms are continuous by default (with no spaces between the marks), and are sized to match the size of the bins. See [Build a Histogram on page 89](#) for an example.
- When there is a field on **Size**, you can determine the width of the bar marks on the axis where the bars are anchored by using the field on **Size**. To do this, click the **Size** card and select **Fixed**.



- When there is no field on **Size**, you can specify the width of the bar marks on the axis where the bars are anchored in axis units. To do this, click the **Size** card, choose **Fixed**, and then type a number in the **Width in axis units** field.



- When there is a continuous date field on the axis where the bars are anchored, the width of the marks is set to match the level of the date field. For example, if the level of the continuous date field is MONTH, the bars are exactly one month wide—that is, slightly wider for 31-day months than for 30-day months. You can configure the width of the bars by clicking the **Size** card, choosing **Fixed**, and then typing a number in the **Width in days** field, but the resulting bar widths don't take into account the varying lengths of time units such as months and years.

Label/Text Properties

The Label property allows you to encode data by assigning text labels to the marks. When working with a text table, this property is called Text, which allows you to view the numbers associated with a data view. The effect of text-encoding your data view depends on whether you use a dimension or a measure.

- Dimension – When you place a dimension on **Label** or **Text** on the Marks card, Tableau separates the marks according to the members in the dimension. The text labels are driven by the dimension member names.
- Measure – When you place a measure on **Label** or **Text** on the Marks card, the text labels are driven by the measure values. The measure can be either aggregated or disaggregated. However, dis-aggregating the measure is generally not useful because it often results in overlapping text.

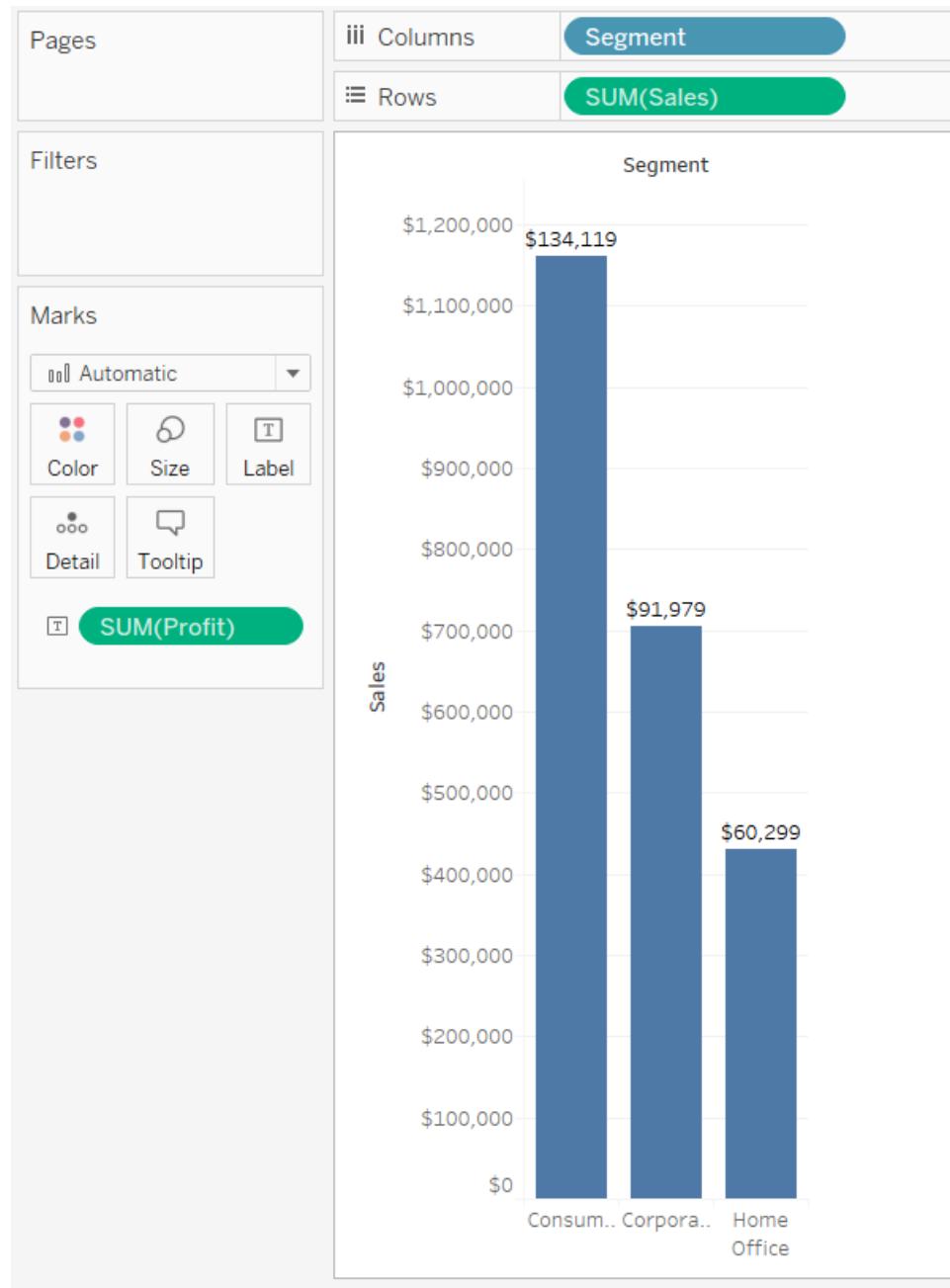
Text is the default mark type when dimensions are the inner fields for both the **Rows** shelf and the **Columns** shelf. This type of view is called a text table, which is also referred to as cross-tab or a PivotTable.

The screenshot shows the Tableau interface with a data view. On the left, there's a sidebar with 'Pages' (empty), 'Filters' (empty), and 'Marks' settings. Under 'Marks', 'Automatic' is selected, and 'Text' is chosen from the dropdown. Below that are buttons for 'Color', 'Size', 'Text', 'Detail', and 'Tooltip'. A green button labeled 'SUM(Sales)' is highlighted. The main area displays a text table titled 'Segment' with three columns: Sub-Categ.., Consumer, and Corporate. The table has two additional columns at the top: Home and Office. The data includes rows for Accessories, Appliances, Art, Binders, Bookcases, Chairs, Copiers, Envelopes, Fasteners, Furnishings, Labels, Machines, Paper, Phones, Storage, Supplies, and Tables, with their respective sales figures.

Sub-Categ..	Consumer	Corporate	Home	Office
Accessories	\$87,105	\$48,191	\$32,085	
Appliances	\$52,820	\$36,589	\$18,124	
Art	\$14,252	\$8,590	\$4,276	
Binders	\$118,161	\$51,560	\$33,691	
Bookcases	\$68,633	\$34,006	\$12,241	
Chairs	\$172,863	\$99,141	\$56,445	
Copiers	\$69,819	\$46,829	\$32,880	
Envelopes	\$7,771	\$5,943	\$2,763	
Fasteners	\$1,681	\$783	\$560	
Furnishings	\$49,620	\$25,001	\$17,084	
Labels	\$6,709	\$4,102	\$1,675	
Machines	\$79,543	\$60,277	\$49,419	
Paper	\$36,324	\$23,883	\$18,272	
Phones	\$169,933	\$91,153	\$68,921	
Storage	\$100,492	\$79,791	\$43,560	
Supplies	\$25,741	\$19,435	\$1,497	
Tables	\$99,934	\$70,872	\$36,160	

Note: You can display text labels with other mark types by dragging a field to the Label target on the Marks card. For more information on showing and hiding marks, see [Mark Labels on page 548](#).

In the view below, the heights of the bars are driven by the **Sales** measure and the labels show the aggregation (SUM) of the **Profit** measure.

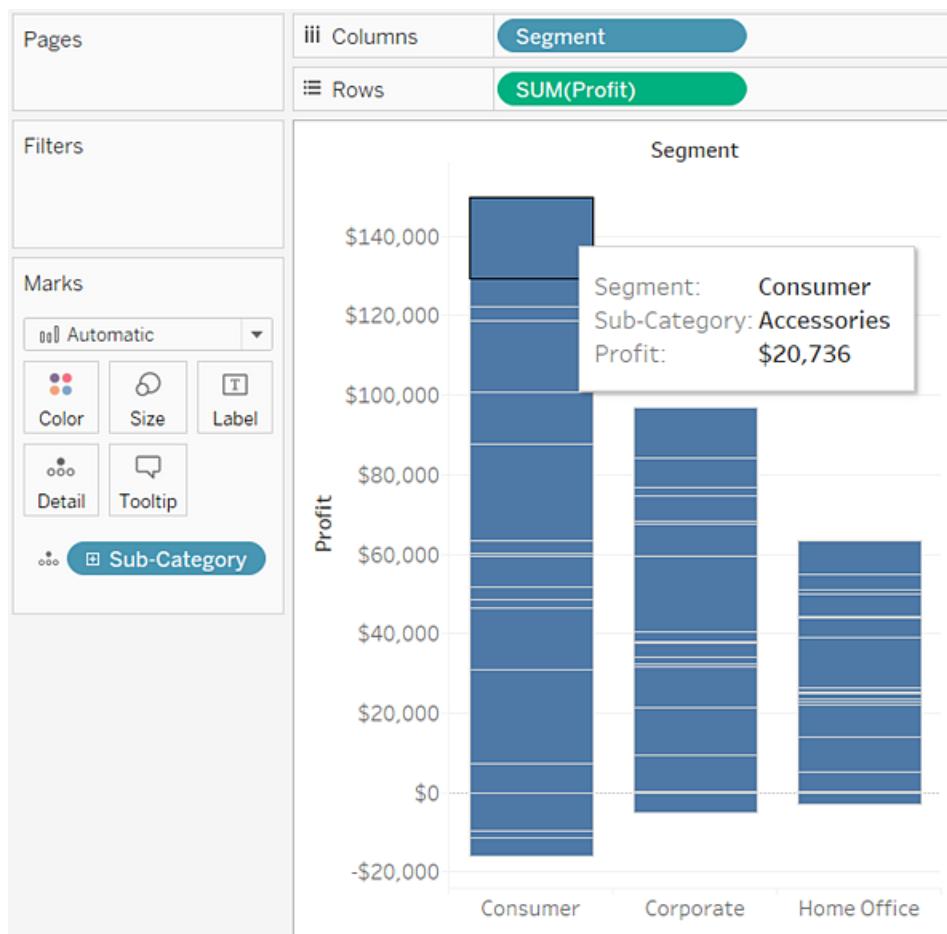


Detail Properties

Whenever you place a dimension on the **Rows** or **Columns** shelf, Tableau uses the categorical members of the dimension to create table headers. The headers show how Tableau is sorting the underlying row data into specific categories. For example, the **Segment** dimension separates the data source rows into four levels of detail: Consumer, Corporate, and Home Office.

Drop a dimension on **Detail** on the Marks card to separate the marks in a data view according to the members of a dimension. Unlike dropping a dimension on the **Rows** or **Columns** shelf, dropping it on **Detail** on the Marks card is a way to show more data without changing the table structure.

In the view below, the bars are separated into segments according to the members of the **Sub-Category** dimension. The size of each segment reflects the contribution to the profit for a particular member. For example, the Accessories sub-category in the Consumer market has a profit of \$20,736.



You can place any number of dimensions on **Detail** on the Marks card. In fact, placing all dimensions on this shelf is one way to display all the rows of your data source.

Note: The Detail property works only if the measures that contribute axes to the table are aggregated. If the measures are disaggregated, then it isn't possible to separate the marks into additional levels of detail because all levels of detail are already shown.

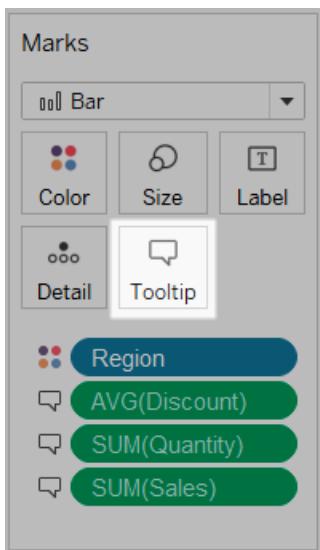
Placing a measure on **Detail** has no effect because measures do not contain members. However, you can place measures on this shelf if you want to export their values to Microsoft Access, copy their values to the Clipboard, or use the values in actions.

Tooltip Properties

Tooltips are details that appear when you hover over one or more marks in the view. Tooltips are also convenient for quickly filtering or removing a selection, or viewing underlying data. You can edit a tooltip to include both static and dynamic text. You can also modify which fields are included in a tooltip.

Add a tooltip

1. Drag a field to **Tooltip** on the **Marks** card.
2. Click **Tooltip** on the **Marks** card to open the Edit Tooltip dialog box, where you can add text, rearrange the tooltip contents, and insert more fields.

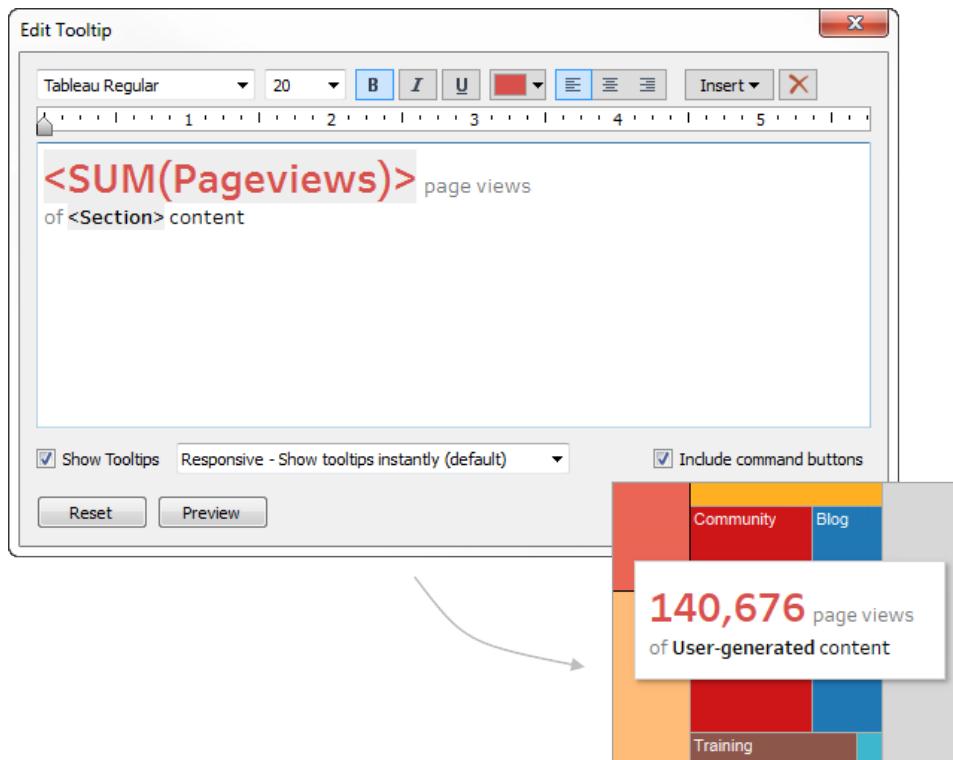


Note: Dimensions are added to the tooltip using the ATTR aggregation, which means the tooltip may sometimes display as an asterisk. The asterisk indicates that there are multiple dimension members that apply to the mark you are pointing at. For example, a mark may represent the aggregated sales for all regions. Adding the Region field to the tooltip results in an asterisk because the mark represents more than one region. To avoid showing an asterisk, add the dimension to Detail on the Marks card or use it elsewhere in the view to ensure the marks are at the same level of detail.

Tooltip options

After you open the Edit Tooltip dialog box, there are several options that you can choose from to format the tooltips in your view and configure their behavior. You can choose from the

options below.



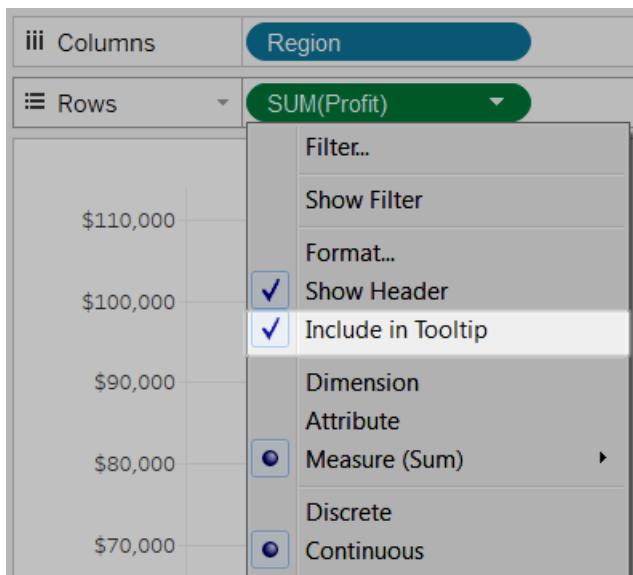
- **Format tooltips:** Tooltips are specified on a per-sheet basis and can be formatted using the tools on the top of the Edit Tooltip dialog box.
- **Adding dynamic text:** Use the **Insert** menu at the top of the dialog box to add dynamic text such as field values, sheet properties, and more. The **All Fields** command on the **Insert** menu adds all field names and values that are used in the view to the tooltip for any mark in the view.
- **Show tooltips:** Tooltips are shown by default. If you prefer to hide automatic tooltips, clear the **Show Tooltips** check box.
- **Configure tooltip behavior:** Configure how tooltips behave by selecting one of the following options in the drop-down list next to the **Show Tooltips** check box:
 - **Responsive:** Select this option to show tooltips instantly as you move the cursor over the marks in the view. This is the default for all views.
With this option, tooltips appear without command buttons. You must click a mark in the view first to see the command buttons.
 - **On Hover:** Select this option to show tooltips only after you rest the cursor on a mark. However, with this option, command buttons appear on the tooltip without

any further action from you (unless you specify otherwise in the Edit Tooltips dialog box).

- **Show commands:** Select the **Include command buttons** check box to add **Keep Only**, **Exclude**, **Group Members**, **Create Set** and **View Data** buttons at the top of the tooltip. These command buttons are available both in Tableau Desktop and when the view is published to the web or viewed on a mobile device.

For more information about grouping members, see [Groups on page 497](#). For more information about creating sets, see [Sets on page 512](#).

- **Add or remove fields:** To add and remove fields when using the automatic tooltip, right-click (control-click on Mac) the field on one of the shelves in the view and select **Include in Tooltip**.



Note: The **Include in Tooltip** option is only available if you have not customized the tooltip. If you have customized the tooltip, you can return to the automatic tooltip by clicking **Tooltip** on the Marks card and then clicking **Reset** in the Edit Tooltip dialog box.

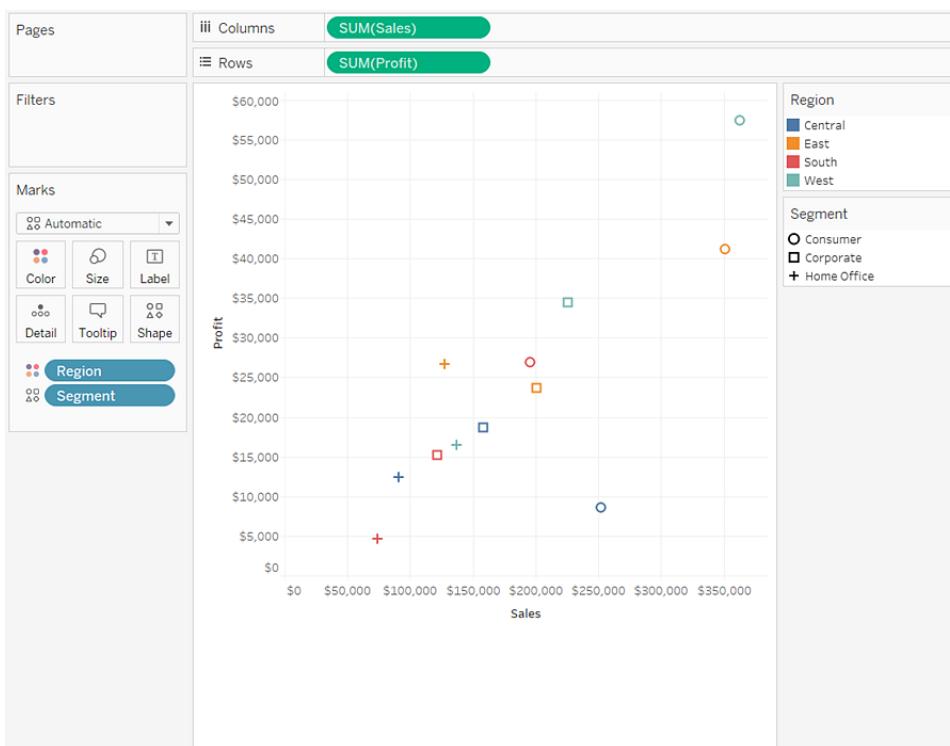
Shape Properties

The Shape property allows you to encode data by assigning different shapes to the marks in a data view.

When you place a dimension on **Shape** on the **Marks** card, Tableau separates the marks according to the members in the dimension, and assigns a unique shape to each member. Tableau also displays a shape legend, which shows each member name and its associated shape. When you place a measure on **Shape** on the **Marks** card, the measure is converted to a discrete measure.

Shape-encoding data separates the marks in the same way as the Detail property does, and then provides additional information (a shape) for each mark. Shape is the default mark type when measures are the inner most fields for both the **Rows** shelf and the **Columns** shelf.

In the view below, the marks are separated into different shapes according to the members of the **Customer Segment** dimension. Each shape reflects the customer segment's contribution to profit and sales.

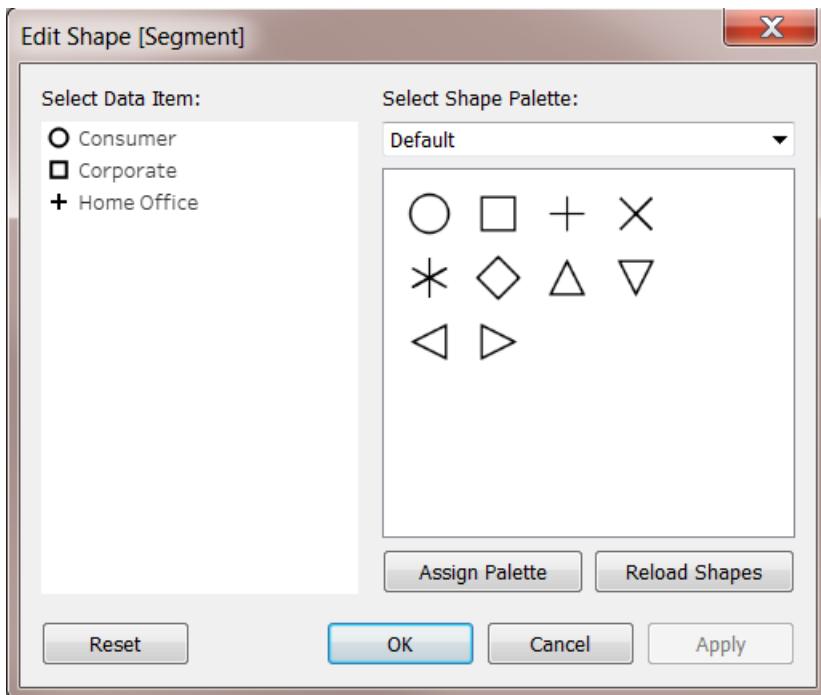


Editing shapes

By default, ten unique shapes are used to encode dimensions. If you have more than 10 members, the shapes repeat. In addition to the default palette, you can choose from a variety of

shape palettes, including filled shapes, arrows, and even weather symbols.

1. Click **Shape** on the **Marks** card, or select **Edit Shape** on the legend's card menu.
2. In the Edit Shape dialog box, select a member on the left and then select the new shape in the palette on the right. You can also click **Assign Palette** to quickly assign the shapes to the members of the field.



Select a different shape palette using the drop-down menu in the upper right.

Note: Shape encodings are shared across multiple worksheets that use the same data source. For example, if you define Furniture products to be represented by a square, they will automatically be squares in all other views in the workbook. To set the default shape encodings for a field, right-click (control-click on Mac) the field in the **Data** pane and select **Default Properties > Shape**.

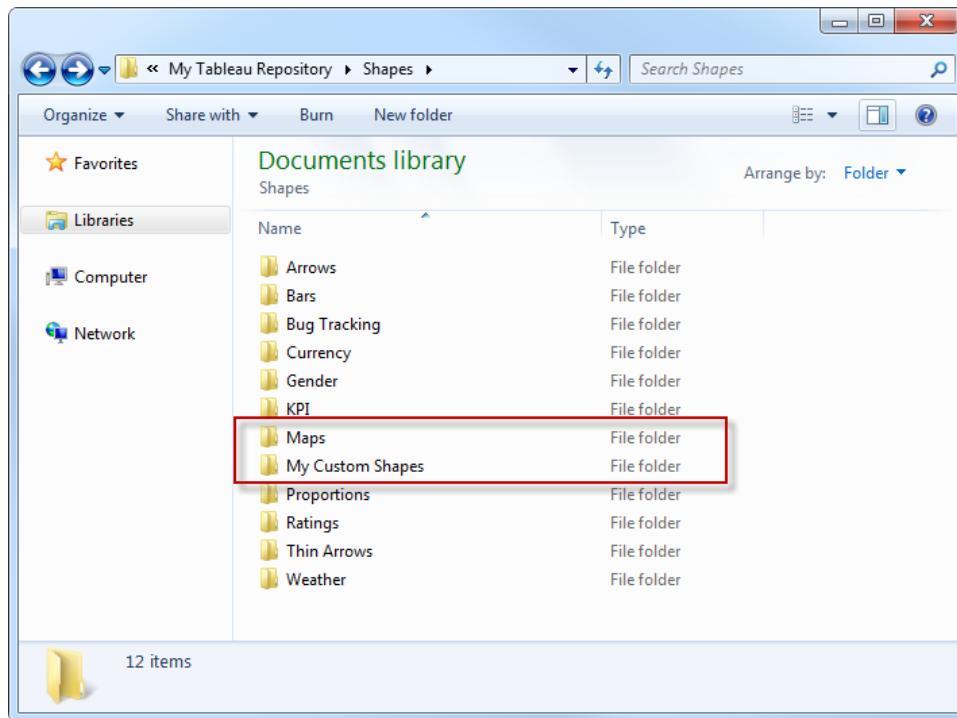
Custom shapes

You can add custom shapes to a workbook by copying shape image files to the Shapes folder in your Tableau Repository, which is located in your Documents folder. When you use custom shapes, they are saved with the workbook. That way the workbook can be shared with others.

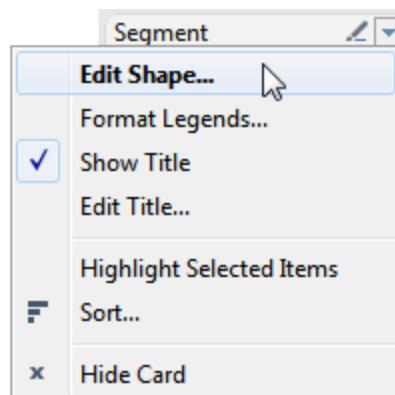
1. Create your shape image files. Each shape should be saved as its own file and can be in any of several image formats including bitmap (.bmp), portable network graphic (.png),

.jpg, and graphics interchange format (.gif).

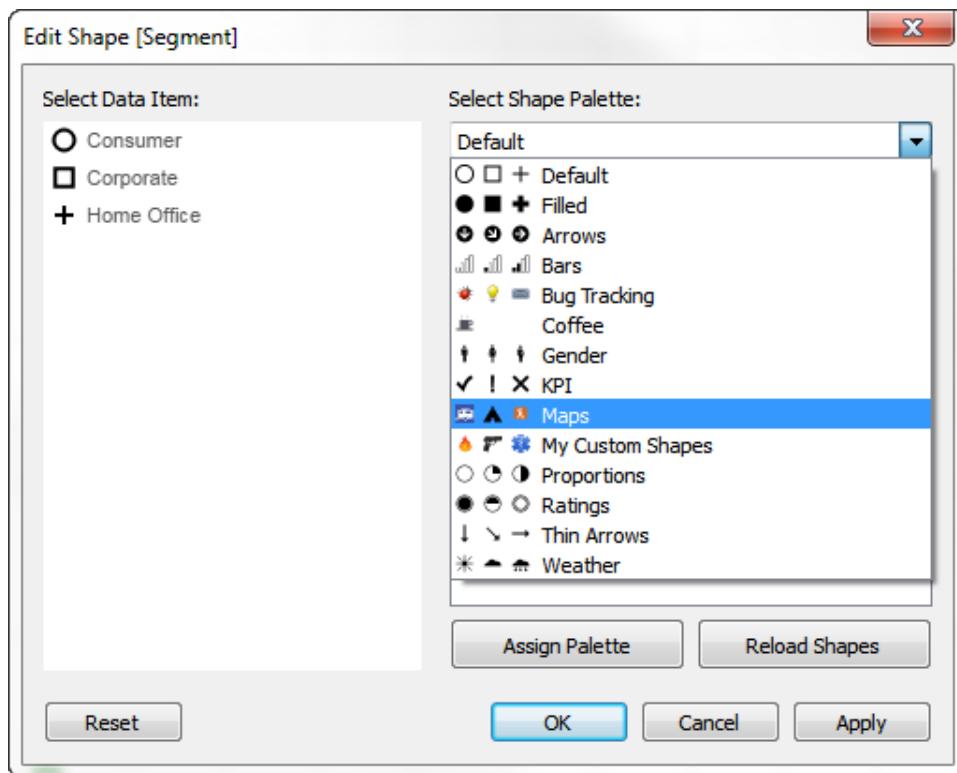
2. Copy the shape files to a new folder in the My Tableau Repository\Shapes folder in your Documents folder. The name of the folder will be used as the name of the palette in Tableau. In the example below, two new palettes are created: Maps and My Custom Shapes.



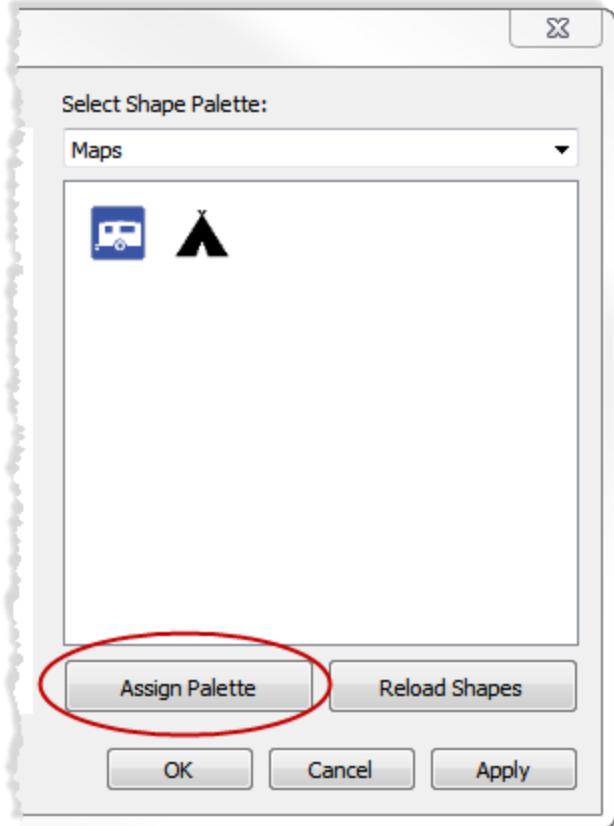
3. In Tableau, click the drop-down arrow on the shape legend, and select **Edit Shape**.



4. Select the new custom palette in the drop-down list. If you modified the shapes while Tableau was running, you may need to click **Reload Shapes**.

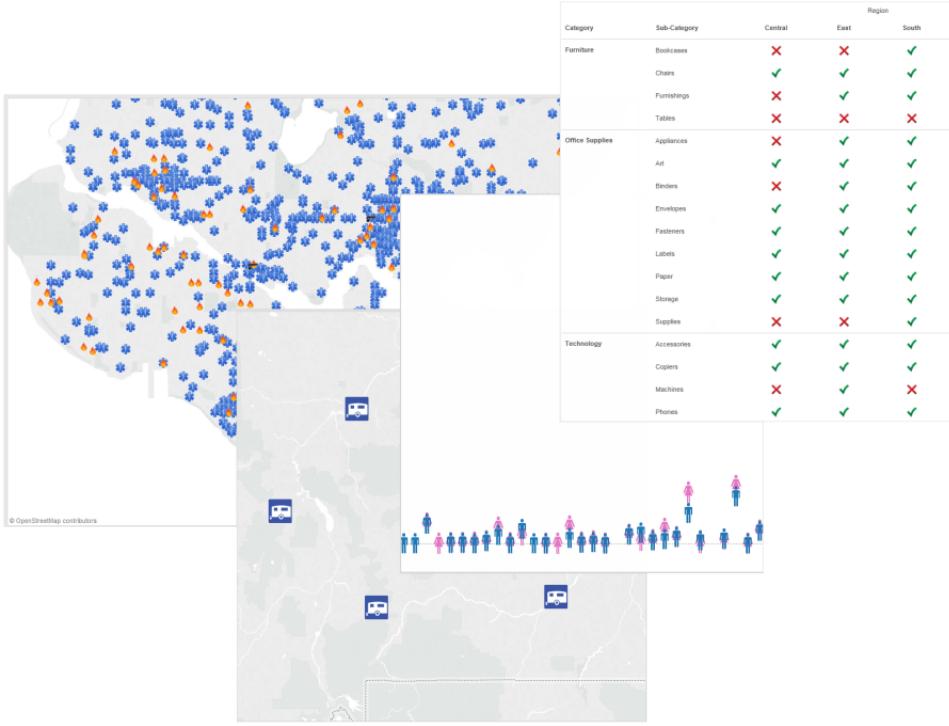


5. You can either assign members shapes one at a time, or click **Assign Palette** to automatically assign the shapes to the members.



Note: You can return to the default palette by clicking the **Reset** button. If you open a workbook that uses custom shapes that you don't have, the workbook will show the custom shapes because the shapes are saved as part of the workbook. However, you can click **Reload Shapes** in the Edit Shapes dialog box to use the ones in your repository instead.

Below are some examples of views that use both the default and custom shape palettes.



Tips for creating custom shapes

When you create custom shapes there are a few things that you can do to improve how your shapes look and function in the view. If you are creating your own shapes, we recommend following general guidelines for making icons or clip art.

- **Suggested size** - Unless you plan on using Size to make the shapes really large, you should try to make your original shape size close to 32 pixels by 32 pixels. However, the original size depends on the range of sizes you want available in Tableau. You can resize the shapes in Tableau by clicking Size on the **Marks** card, or by using the cell size options on the **Format** menu.
- **Adding color encoding** - If you plan to also use Color to encode shapes, you should use a transparent background. Otherwise, the entire square of the image will be colored rather than just the symbol. GIF and PNG file formats both support transparency. GIF files support transparency for a single color that is 100% transparent, while PNG files support alpha channels with a range of transparency levels available on every pixel in the image. When Tableau color encodes a symbol, the amount of transparency for each pixel won't be modified, so you can maintain smooth edges.
- **File formats** - Tableau doesn't support symbols that are in the Enhanced Meta File format (.emf). The shape image files can be in one of the following formats: .png, .gif, .jpg, .bmp, and .tiff.

Path Properties

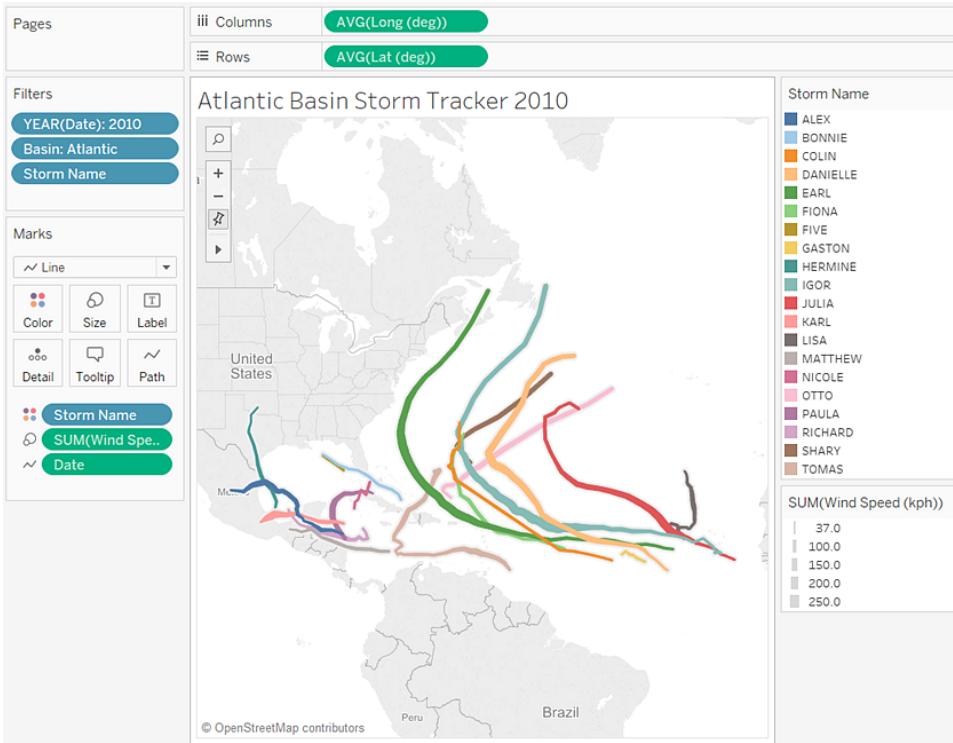
The Path property allows you to encode data by connecting marks using a particular drawing order. You can path-encode your data using either a dimension or a measure.

- Dimension – When you place a dimension on **Path** on the Marks card, Tableau connects the marks according to the members in the dimension. If the dimension is a date, the drawing order is driven by the date order. If the dimension includes words such as customer names or product types, the drawing order is driven by the order of the members in the data source. You can change the order that data points are connected by changing the sort order of the members. For more information, see [Sorting](#).
- Measure – When you place a measure on **Path** on the **Marks** card, Tableau connects the marks according to the values of the measure. The measure can be aggregated or disaggregated.

The Path property is available only when you select the **Line** or **Polygon** mark type from the **Marks** card drop-down menu. For more information, see [Mark Types](#) on page 168.

To create a useful path-encoded view, your data table should contain at least one measure. You can't create a path that connects only categorical data (dimensions).

The view below was created using storm data from the Atlantic basin in 2010. The view uses line marks with the path determined by the date of the storm. This lets you see the path of the storm. By placing the continuous date on **Path** on the **Marks** card, this tells Tableau to draw the lines in chronological order.



Move Marks

In a dual-axis view that contains many marks, some marks can become hidden behind others, making it difficult to see details in your view. Tableau provides a **Move Marks** option to move selected marks forward or backward, depending on the axis that you select.

To move marks forward or backward, right-click on one of your axes and select one of the following options:

- **Move Marks to Front**
- **Move marks to Back**

Example - Move marks forward

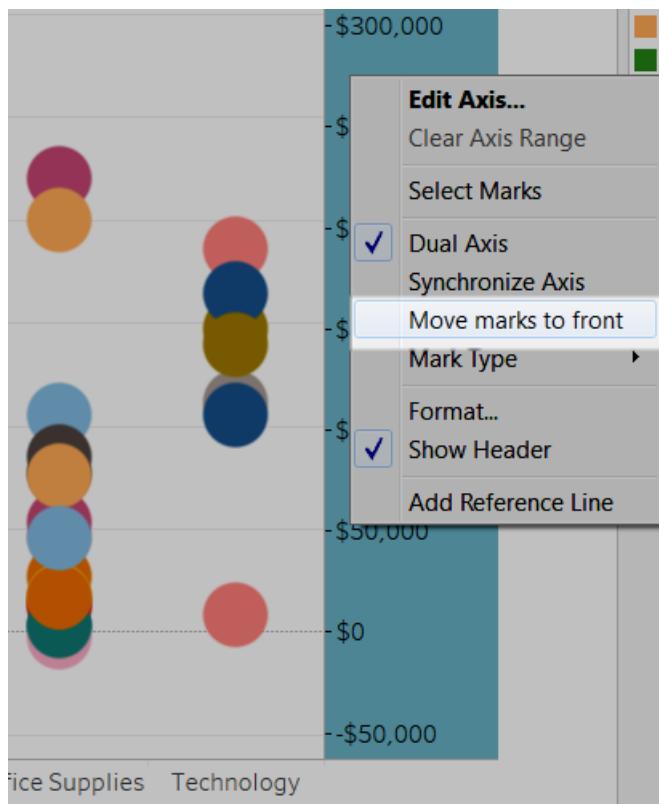
In this example, suppose you have a view that uses the **Circle** mark type on a dual axis and shows department **Sales** and **Profit** for each **Category**.

The shape of each mark represents **Sales** and **Profit**, while **Category** is encoded as **Color**. The right axis represents the **Sales** mark, while the left represents the **Profit** mark.

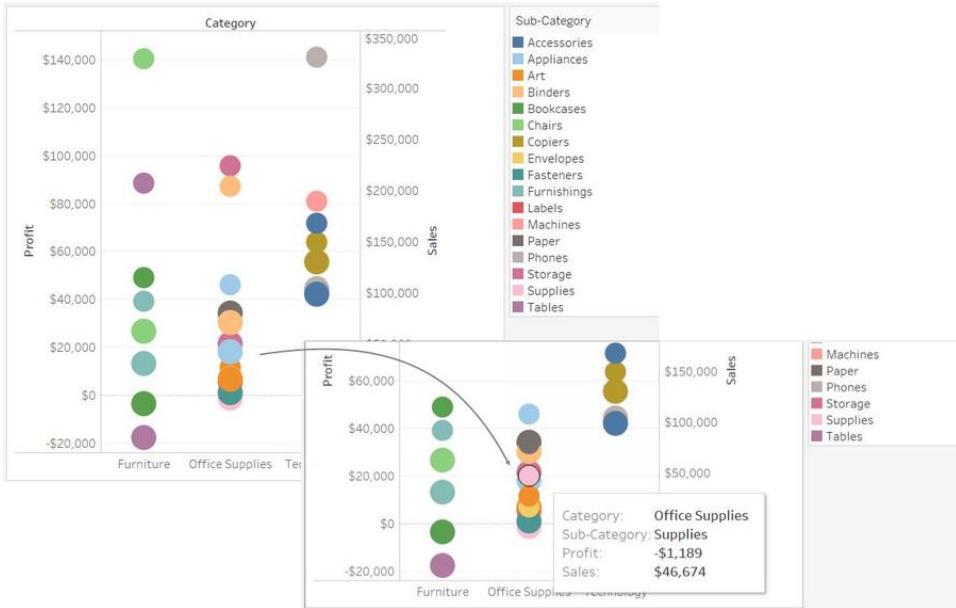
Because the profit marks are in front, it is difficult to see the **Sales** marks in the **Office Supplies** category.

To move the **Sales** marks in front of the **Profit** marks, do the following:

1. Right-click on the **Sales** axis.
2. Select **Move marks to front** from the context menu.



By moving the **Sales** mark to the front, you can now see that rubber bands are below \$100,000 in sales, whereas they were nearly invisible before.



Stack Marks

Stacking marks is relevant when your data view includes numeric axes. That is, at least one measure has been placed on the **Rows** or **Columns** shelves. When marks are stacked, they are drawn cumulatively along an axis. When marks are not stacked, they are drawn independently along an axis. That is, they are overlapping.

Stacking marks is particularly useful for bar charts which is why Tableau automatically stacks bars. You might find that stacking marks is useful for other marks such as lines as well.

You can control whether marks are stacked or overlapping in any given view by selecting the **Analysis > Stack Marks** menu item.

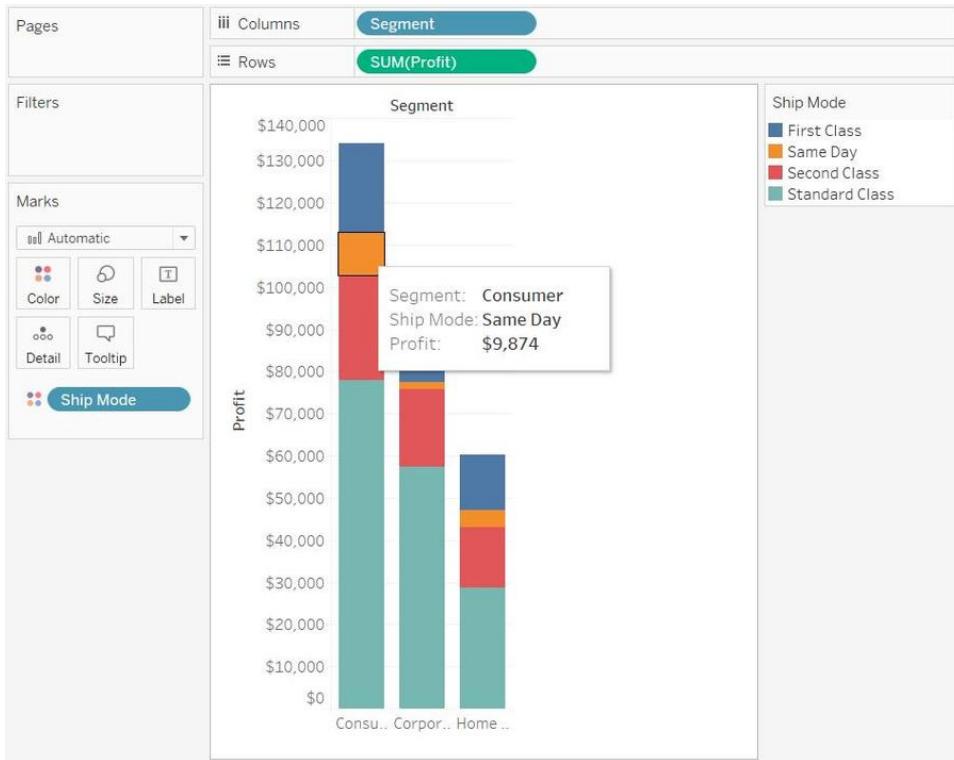
You can either allow Tableau to automatically select whether the marks are stacked or you can specify **On** or **Off**. The default setting is **Automatic**. When you are in automatic mode, the Stack Marks menu shows whether stacked marks is on or off.

Selecting **On** or **Off** on the **Stack Marks** menu, switches into manual mode. Your selection remains throughout any changes you make to the view.

The following examples illustrate stacking marks.

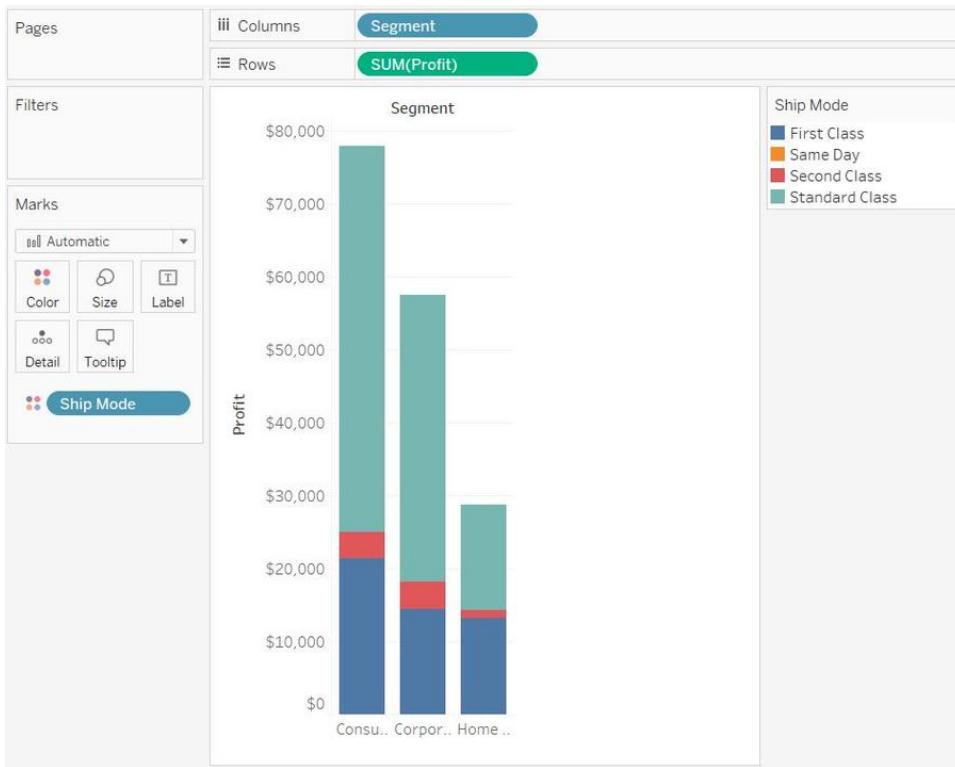
Example – Stacking Bars

Consider the stacked bars view shown below. It was created by placing a dimension on the **Columns** shelf, placing a measure on the **Rows** shelf, and color-encoding the data by a dimension (that is, dropping a dimension on **Color** on the Marks card).



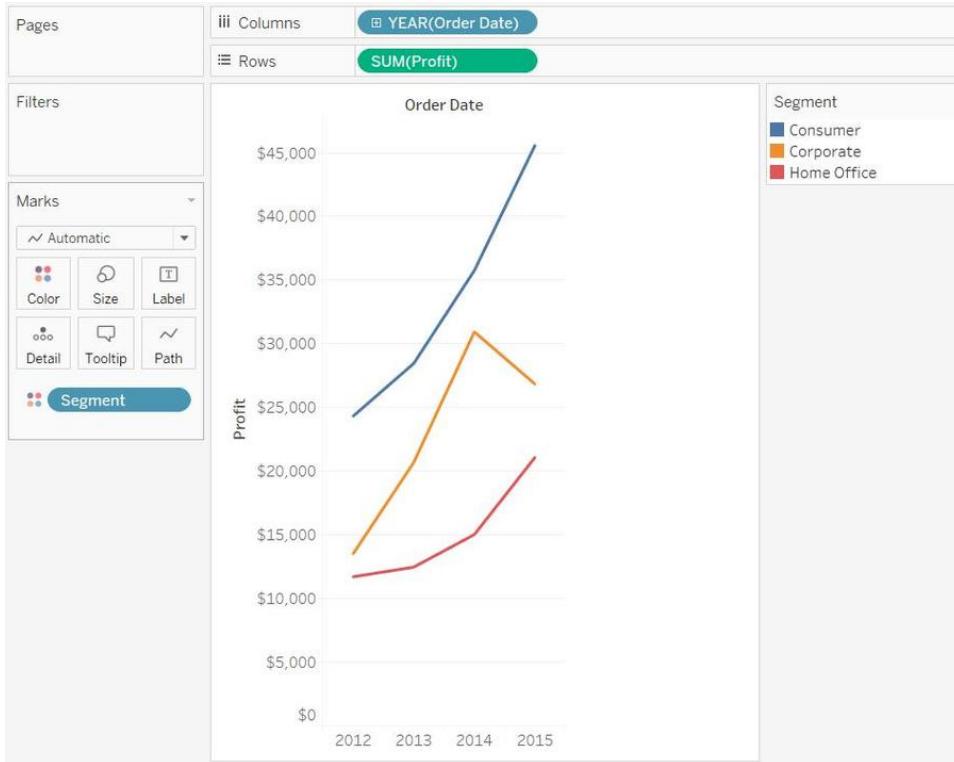
Because the mark type is a bar, Tableau automatically stacks the marks. This means that the marks are drawn cumulatively and the height of each stacked segment within each bar represents the value for that segment, with each segment listed alphabetically. For example, the sum of the profit for products shipped with Same Day shipping (orange bar segment) in the Consumer market is \$9,874.

If you turn off **Stack Marks** (under the **Analysis** menu), the marks start from the horizontal axis. As shown below, you can still view the individual bar segments. Be aware, however, because un-stacked marks overlap, it is possible to create a view where bar segments are not visible. For example, in the view below, Same Day shipping (orange bar segment) is no longer visible in the view for any market.



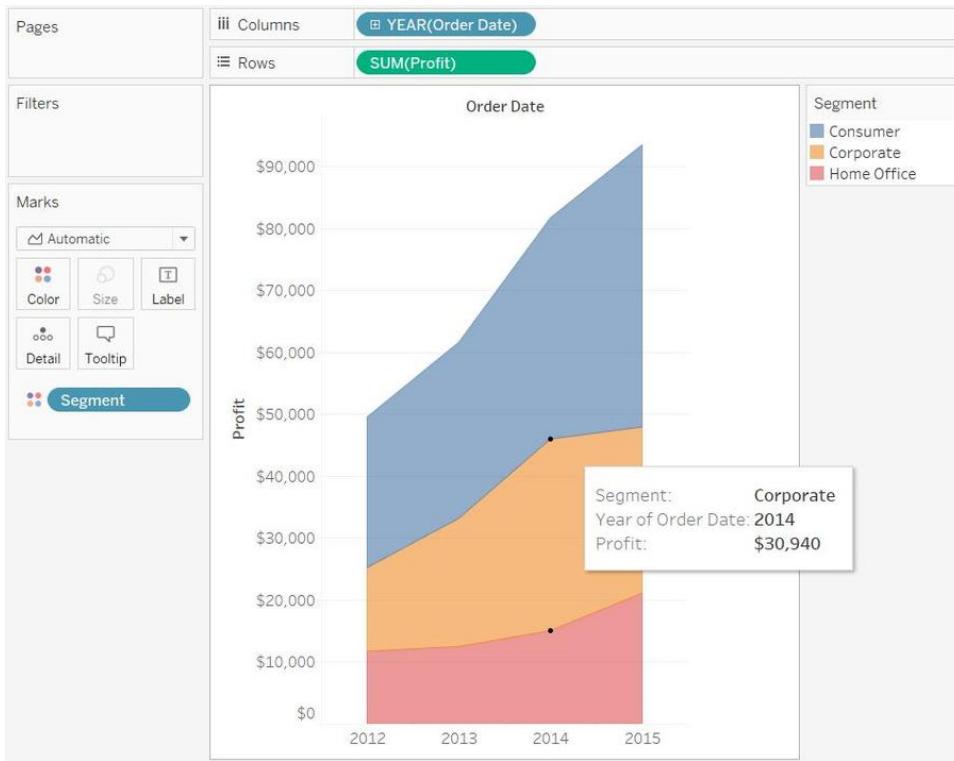
Example – Stacking Lines

Consider the data view shown below. It was created by placing a date dimension on the **Columns** shelf, placing a measure on the **Rows** shelf, and color-encoding the data by a dimension (that is, dropping a dimension on **Color** on the Marks card). Because the mark type is Line, the marks are not automatically stacked. Instead, they are drawn independently from the horizontal axis.



Interpret any data point by reading the associated values from the horizontal and vertical axes. For example, in the year 2014, the Corporate (orange) profit totaled \$30,940. That is, the space between that data point and the horizontal axis is equal to the sum of the profit for the Corporate market.

Now, stack the marks by selecting the **Analysis > Stack Marks > On** menu item. Tableau automatically switches to the Area mark type.



In this view, the lines are no longer independent of each other. Instead, they are drawn cumulatively. The stacking order is driven by the order of the dimension members in the data source. This order is reflected in the color legend, from bottom to top.

Therefore, the stacked Home Office (red) area is the same as its un-stacked version because it's at the bottom of the stacking list. The stacked Corporate (orange) area is derived by adding its un-stacked values to the un-stacked Home Office values. The stacked Consumer(blue) area is derived by adding its un-stacked values to the stacked Corporate data.

The vertical axis gives the new scale for the stacked marks. Interpret the filled area as the sum of the profit.

For example, notice that the label for the 2014 Corporate data still shows the profit as \$30,940. The interpretation is that the space between the Corporate data and the Home Office data yields the sum of the profit for the Corporate market.

Titles

You can show titles on any worksheet, dashboard, or story. For worksheets and stories, a title is displayed by default, but you can remove it. For dashboards, you can add a title. By default, the title is the name of the sheet, but you can edit the title to change the text and include dynamic values such as page number and sheet name. For more information about how to format titles, see [Format Titles, Captions, Tooltips and Legends](#) on page 1077.

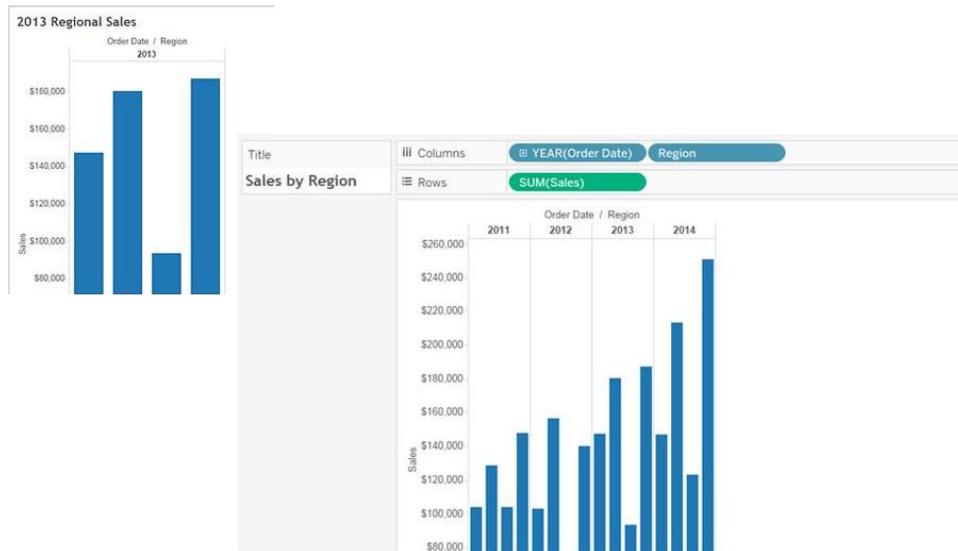
Show and hide titles

To show and hide titles for worksheets, dashboards, and stories, review the following sections.

Worksheets

Titles are shown by default for worksheets and are included as part of the worksheet, shown at the top of the view. You can move the title to the sides or the bottom of the view. However, when you move the title from the top of the view, it becomes a **Title** card and displays like any other card in the view.

Note: If you move a title from the top position and then hide it; when you show the title again, it appears back at the top of the worksheet in its default position.



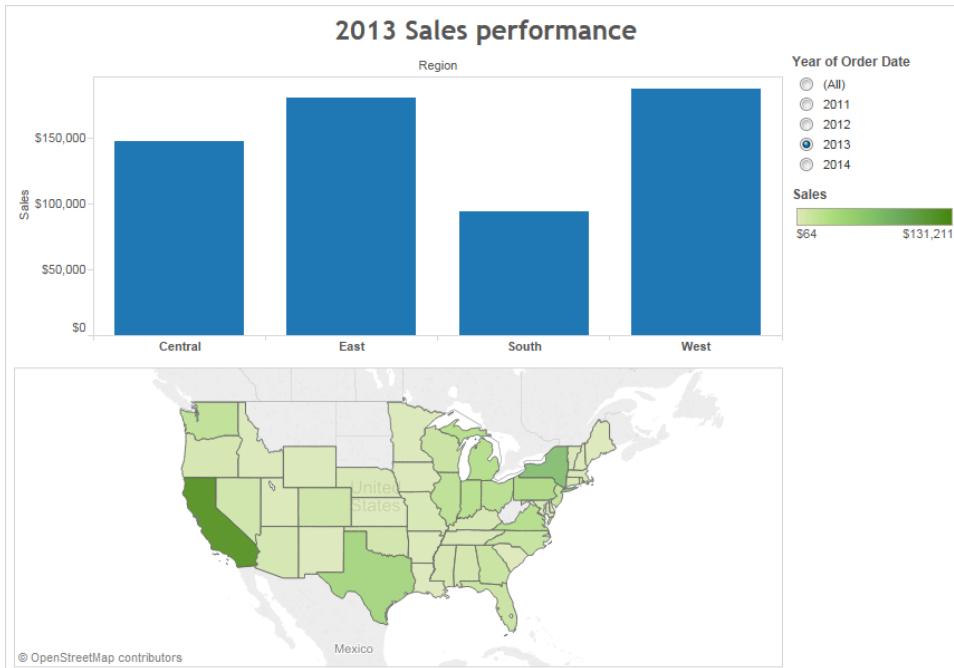
To show and hide titles in a worksheet, do one of the following:

- From the toolbar menu, click **Worksheet > Show Title**.
- On the toolbar, click the drop-down arrow on the **Show/Hide Cards** button and select **Title** from the context menu.
Toggle the check mark on or off to show or hide the title.

Dashboards

You can turn on titles for dashboards. The title appears as part of the dashboard.

To show or hide titles on a dashboard, from the toolbar menu, select **Dashboard > Show Title**.



When you add worksheets to the dashboard, the title of the worksheet automatically shows, even if you turned off the title on the worksheet itself. To turn off the title for the worksheet on the dashboard, do the following steps:

1. In the dashboard, select the worksheet to highlight it.
2. In the top right corner of the highlighted worksheet, click the drop-down arrow and select **Title** in the context menu to clear the check mark.

Stories

Story titles are displayed by default. To toggle story titles on or off, in the top menu, select **Story > Show Title** to add or remove the check mark.

Captions

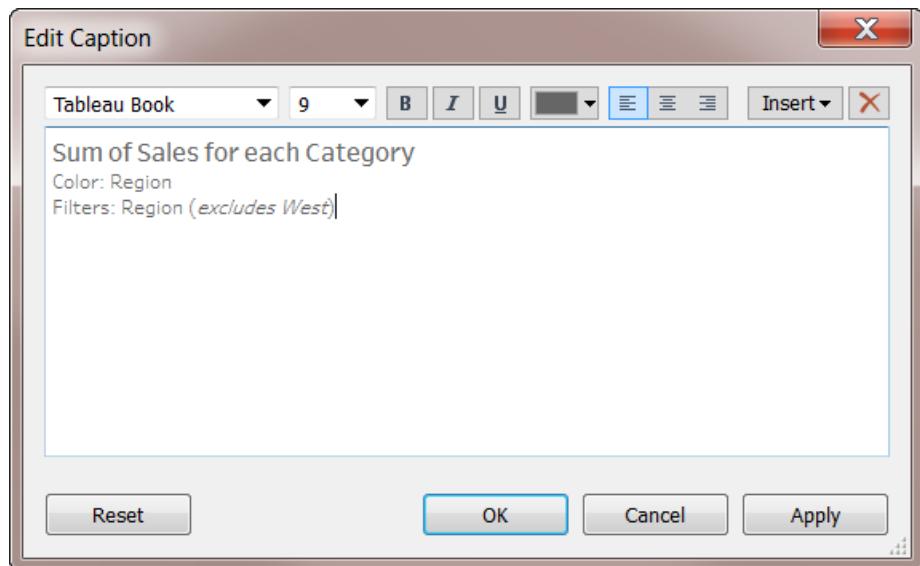
All views can have a caption that is either automatically generated or manually created. The caption is displayed on the Caption card. To show the caption, select it on the **Show/Hide**

Cards toolbar menu or select **Worksheet > Show Caption**.

Caption

Sum of Sales for each Category. Color shows details about Region. The view is filtered on Region, which keeps Central, East and South.

The caption is automatically generated by default. However, you can edit the caption by double clicking the Caption card. In the Edit Caption dialog box, you can use change the font, size, color, and alignment and style.



Use the Insert menu to add automatic text such as page number, sheet name, and field and parameter values.

The caption can optionally be included when printing, printing to PDF, and publishing to Tableau Server. When you export the view as an image to another application like Microsoft PowerPoint, you can optionally include the caption.

Field Labels

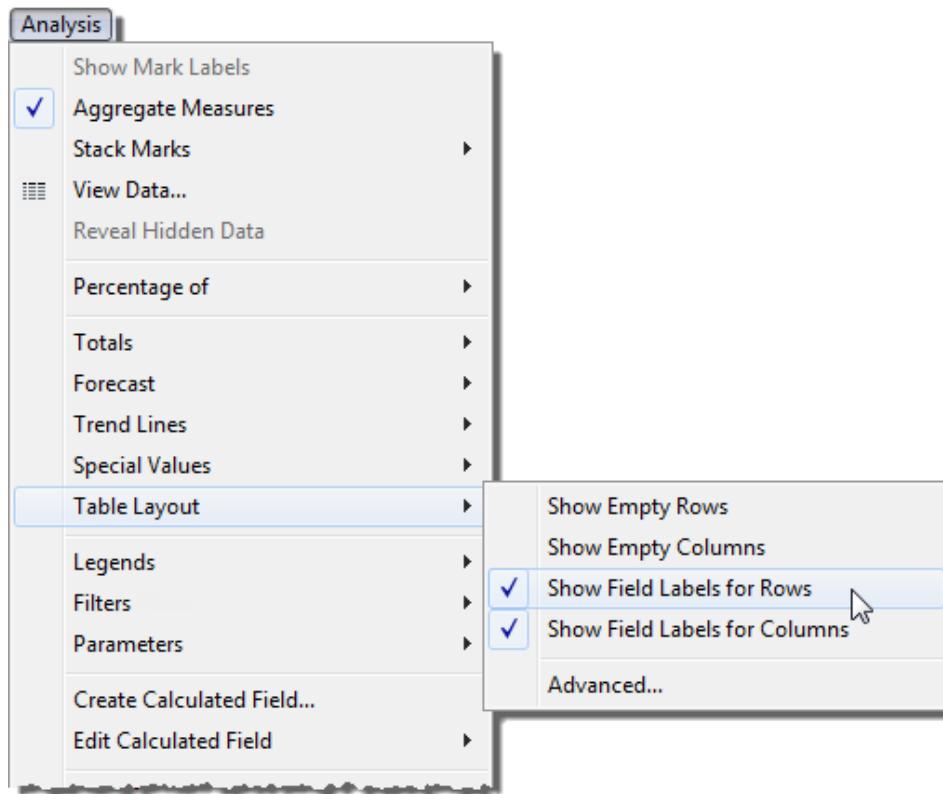
Placing discrete fields on the rows and column shelves creates headers in the view that display the members of the field. For example, if you place a field containing products on the rows shelf, each product name is shown as row headers. In addition to showing these headers, you can show field labels, which are labels for the headers. In this example, the rows are labeled **Category**, thus indicating that the discrete category names are members of the **Category** field.

The screenshot shows the Tableau Data Explorer interface. At the top, there are two tabs: 'Columns' and 'Rows'. The 'Columns' tab is active, showing 'Region' and 'YEAR(Order Date)' as columns. The 'Rows' tab is also present. Below these tabs is a table structure with a header row 'Region / Order Date' and a sub-header 'East'. The table has 'Category' as the first column and years from 2012 to 2015 as subsequent columns. The data for Furniture, Office Supplies, and Technology categories across these years is displayed.

Category	Region / Order Date			
	East			
	2012	2013	2014	2015
Furniture	47,233	53,817	46,387	60,854
Office Supplies	35,969	42,655	61,645	65,247
Technology	45,479	59,859	72,497	87,138

Field labels apply only to discrete fields. When you add continuous fields to the view, Tableau creates an axis. The axis is labeled with a header.

By default, field labels are shown. You can hide or show field labels at anytime by selecting **Analysis > Table Layout > Show Field Labels for Rows** or **Show Field Labels for Columns**.

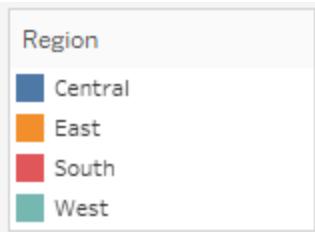


You can format the fonts, alignment, shading, and separators for field labels.

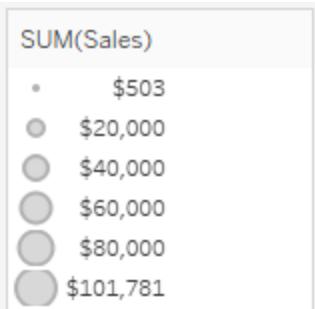
Legends

When you add fields to Color, Size, and Shape on the Marks card, a legend displays to indicate how the view is encoded with relation to your data.

Color Legend



Size Legend



Shape Legend



Not only do legends help you understand encodings, you can also use legends to sort, filter, and highlight specific sets of data. For more information, see [Legend Highlighting on page 1123](#).

Workbooks and Sheets

Tableau uses a workbook and sheet file structure, much like Microsoft Excel. A workbook contains sheets, which can be a worksheet, a dashboard, or a story.

A worksheet contains a single view along with shelves, legends, and the Data pane.

A dashboard is a collection of views from multiple worksheets.

A story contains a sequence of worksheets or dashboards that work together to convey information.

Within a workbook, you can create new sheets, clear an entire worksheet, duplicate sheets, hide or show a worksheet, and delete a sheet. Tableau has several ways to view and organize the sheets in your workbook.

[Sharing workbooks between Tableau Desktop on Windows and on the Mac](#)

When working in Tableau Desktop on the Mac, all sheets and workbooks will be fully functional for viewing and editing when opened in Tableau Desktop for Windows. The reverse is also true for workbooks created in Tableau Desktop on Windows.

However, Tableau Desktop on Windows is able to connect to additional data sources that may not be compatible when opened in Tableau Desktop on the Mac.

For more information about opening a Windows workbook that contains data sources that are not supported in Tableau Desktop on the Mac, see [Open an existing Windows workbook on a Mac](#) on the next page in the Workbooks topic.

[Creating and Opening Workbooks](#)

Tableau workbook files are much like Microsoft Excel workbooks.

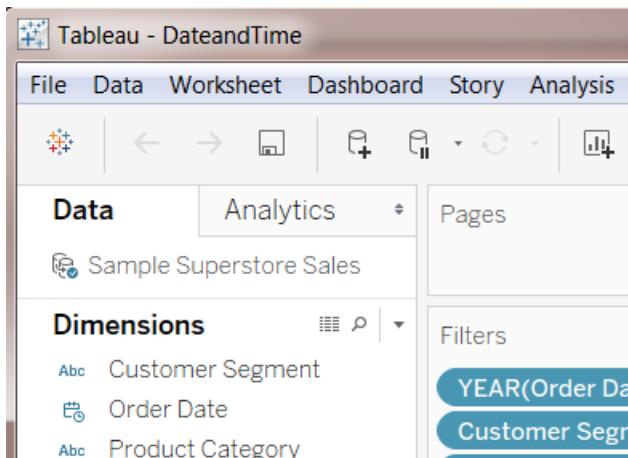
A workbook contains one or more sheets, which can be worksheets, dashboards, or stories. You can use workbooks to organize, save, and share your results.

When you open Tableau, a new, blank workbook opens automatically.

To create a new workbook, select **File > New**.

To open an existing workbook, click the thumbnail image of the workbook on the start page.

When you open a workbook, the workbook name appears in the title bar.



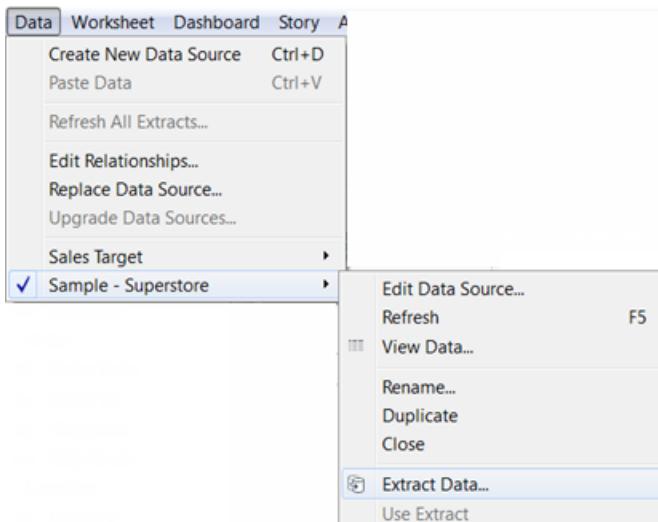
You can open multiple workbooks simultaneously. Each workbook appears in its own window.

When you open multiple workbooks in Tableau Desktop on the Mac, multiple instances of the application are created, each with its own icon in the Dock. This differs from typical Mac application behavior, where one instance of the application handles all open documents.

Open an existing Windows workbook on a Mac

When working in Tableau Desktop on the Mac, you might need to open a Windows workbook that contains data sources that are not supported in Tableau Desktop on the Mac. To do so, follow the steps below:

1. In Tableau Desktop on Windows, save the workbook as an extract. To create and save an extract, on the **Data** menu, select a data source, and then select **Extract Data**.



2. Open the Tableau Extract (.tde file) you saved in Tableau Desktop on your Mac.

Note: When opening a workbook with Microsoft Excel or text files from Tableau version 9.2 or earlier, you will be prompted to upgrade the workbook. If the workbook is unable to upgrade, create an extract of the workbook in Windows, then open the extract on your Mac.

Tip: You can also open new workbooks in Windows using one of the following methods:

- Select **File > Open** and navigate to the location of your workbook using the Open dialog box. Tableau workbooks have a .twb or .twbx file extension.
- Double-click on any workbook file in the Windows explorer. On a Mac, click on any workbook file in Finder.

- In Windows, you can also drag any workbook file onto the Tableau desktop icon or onto the running application.

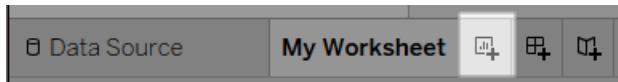
Sheets

Each workbook can contain different types of sheets: views (also known as worksheets), dashboards, and stories. A worksheet is where you build views of your data by dragging and dropping fields onto shelves. A dashboard is a combination of several views that you can arrange for presentation or to monitor. A story is a sequence of views or dashboards that work together to convey information. The sheets display along the bottom of the workbook as tabs. In this section you'll learn how to create, open, duplicate, hide, and delete sheets. You'll also learn how to organize sheets in a workbook.

Creating New Worksheets, Dashboards, and Stories

There are several ways to create new sheets in a workbook, dashboard, or a story. You can create as many sheets in a workbook as you want.

To create a new worksheet, dashboard, or story, click the **New Worksheet, **New Dashboard**, or **New Story** button at the bottom of the workbook.**



The **New Worksheet** button.

To rename a new worksheet, dashboard, or story, right-click (Ctrl-click on a Mac) the tab and then select the **Rename** command.

More ways to create new worksheets

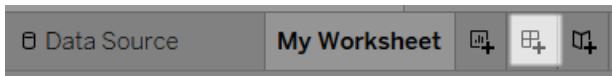
Create a new worksheet by doing one of the following:

- Select **Worksheet > New Worksheet**.
- Right-click any open tab in the workbook, and select **New Worksheet** from the menu.
- On the toolbar, click the drop-down arrow on the **New Worksheet** button and then select **New Worksheet**.
- Press **Ctrl + M** on your keyboard (**Command-M** on a Mac).

More ways to create new dashboards

Create a new dashboard by doing one of the following:

- Select **Dashboard > New Dashboard**.
- Click the **New Dashboard** button at the bottom of the workbook.

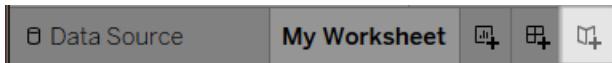


- Right-click on any open tab in the workbook, and select **New Dashboard** from the menu.
- On the toolbar, click the drop-down arrow on the **New Worksheet** button and then select **New Dashboard**.

More ways to create new stories

Create a new story by doing one of the following:

- Select **Story > New Story**.
- Click the **New Story** button at the bottom of the workbook.



- Right-click on any open tab in the workbook, and select **New Story** from the menu.
- On the toolbar, click the drop-down arrow on the **New Worksheet** button and then select **New Story**.

[Undo, Redo, and Clear Sheets](#)

Every Tableau workbook contains a history of steps you have performed on the worksheets, dashboards, and stories in that workbook.

To move backward through the history, click **Undo** on the toolbar or press Ctrl + Z on your keyboard (Command-Z on a Mac).

To move forward through the history, click **Redo** on the toolbar or press Ctrl + Y (Command-Y on a Mac) on your keyboard.

To remove all fields, formatting, sizing, axis ranges, filters, sorts, and context filters

in the sheet, click **Clear Sheet** on the toolbar.

To clear specific aspects of the view, use the the Clear Sheet drop-down menu.

Note: Using the clear commands on the toolbar does not clear the history. If you decide that you didn't want to clear the sheet, click the **Undo** button.

Duplicating Sheets

Duplicating a sheet allows you to easily make a copy of a view in our workbook. The **Duplicate** option creates a new version of a worksheet, dashboard or story that you can then modify without losing the original version. To duplicate the active sheet, right-click the sheet tab (control-click on Mac) and select **Duplicate**.

Note: When you duplicate a dashboard, a new version of the dashboard is created, but it still references the original worksheets that were used to create the dashboard.

A crosstab (sometimes referred to as a Pivot Table) is a table that summarizes data in rows and columns of text. It is a convenient way to display the numbers associated with the data view.

To quickly create a cross-tab from a view, right-click the sheet tab (control-click on Mac) and select **Duplicate as Crosstab**. You can also select **Worksheet > Duplicate as Crosstab**. This command inserts a new worksheet into your workbook and populates the sheet with a cross-tab view of the data from the original worksheet. (Dashboards and stories cannot be duplicated as crosstabs.)

There are other ways to see the numbers behind the data views. For example, you can mouse-over any mark to display the associated numbers in a tooltip. Click the **View Data** command at the top of the tooltip to view underlying data. You can copy and paste the data into Excel or another application.

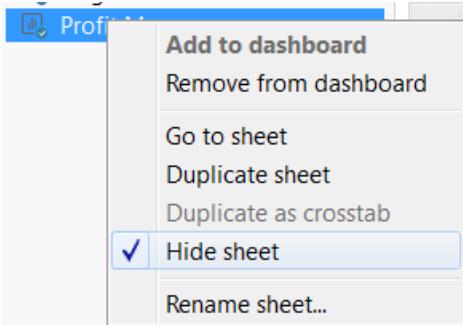
Hide and Show Worksheets in Dashboards or Stories

You can hide or show worksheets in a dashboard or story, for example, if you want to share a story with others but not clutter the view with all the supporting worksheets.

To hide a worksheet, right-click the worksheet tab (control-click on Mac) and select **Hide Sheet**.

Dashboards

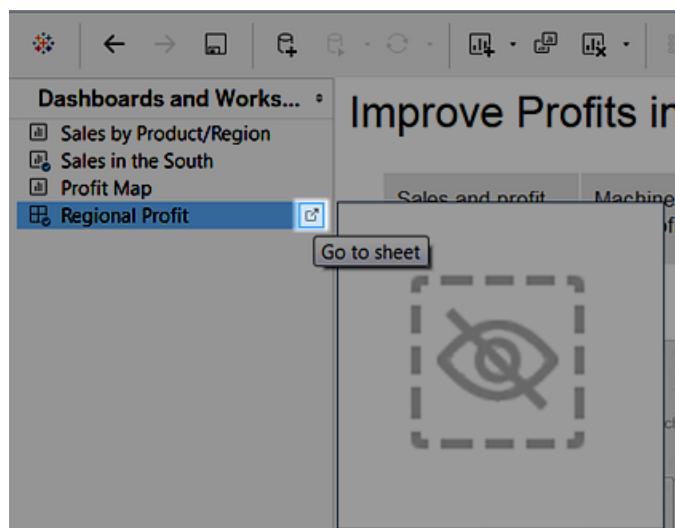
To show a hidden sheet from a dashboard that uses it, in the **Dashboard** pane, right-click the worksheet that was hidden and select **Hide Sheet** to clear the selection and show the worksheet.



To show a sheet until you switch to another sheet, select the dashboard section that references the hidden worksheet, click the drop-down arrow in the top right corner, and then select **Go to Sheet**.

Stories

To view a hidden dashboard or worksheet, hover over the icon to the right of the sheet name and click to go to the underlying dashboard or sheet.



Though you can see the sheet, it is still hidden, so it won't show when you publish the workbook or after you close and reopen it.

To unhide a sheet or dashboard, right-click it (Ctrl-click on Mac) and choose **Unhide Sheet**.

Note: If a story uses a hidden dashboard that in turn uses a hidden worksheet, to see that hidden worksheet you first go to the hidden dashboard from the story, and then to the hidden worksheet from the dashboard.

Deleting Sheets

Deleting a sheet removes it from the workbook. To delete the active sheet, right-click (control-click on Mac) the sheet in the sheet tab along the bottom of the workbook and select **Delete**. Worksheets used in a dashboard or story cannot be deleted, but they can be hidden. For more information, see [Hide and Show Worksheets in Dashboards or Stories on page 244](#).

There must always be at least one worksheet in a workbook.

Organizing Sheets

You can navigate between sheets in a workbook, and you can reorder, create, or duplicate or delete sheets.

There are three ways to navigate and view the sheets in a workbook:

- Tabs at the bottom of the workbook
- In the filmstrip view
- In the sheet sorter view

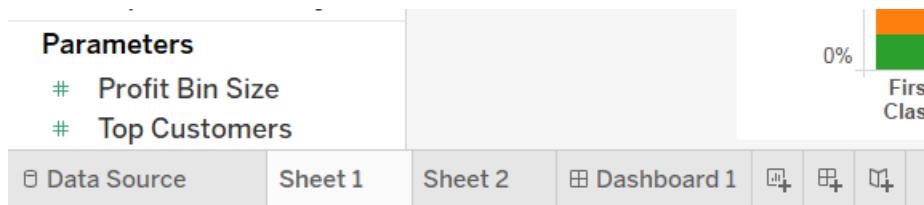
The tabs are useful when you want to quickly navigate between a small number of sheets. If your workbook has a large number of sheets, you can use the sheet sorter to easily navigate them all.

You can also drag and drop to do the following:

- Reorder the sheets
- Create new sheets
- Duplicate or delete existing sheets from any of the views

Navigate with sheet tabs

Each sheet is represented as a tab along the bottom of the workbook. Select any tab to open the corresponding worksheet.



In the bottom right corner of the application window, there are several controls that you can use to advance through each sheet or quickly jump to the first or last sheet in the workbook. These controls are only available when there are too many sheet tabs to show across the bottom of the application window.



You can also navigate between sheets using the window menu or move through the multiple worksheets by pressing the left or right arrow keys on your keyboard.

To navigate through multiple worksheets, select a worksheet tab at the bottom of the workbook first.

Note: Navigating within a story is a different matter—there, you use the Navigator to move between story points. See [Update a Story](#) on page 1216.

Show thumbnails with the filmstrip

Similar to the sheet tabs, the filmstrip displays along the bottom of the workbook. However, instead of just sheet names, the filmstrip also shows a thumbnail image of each sheet. The filmstrip is useful when you are using Tableau to present your analysis and works well when you are working in Presentation mode.

Open the filmstrip by clicking the **Filmstrip** button on the status bar (bottom right corner) of the workbook. Just like with the tabs, select the thumbnail image for the sheet you want to open. You can right-click the images to specify commands that apply to each sheet.

Note: When viewed from a Retina display, workbooks that are created on a standard resolution device will show only the first thumbnail in the filmstrip in high resolution. To display all thumbnails in the filmstrip in high resolution, resave the workbook on a computer with Retina display.



Manage many sheets with sheet sorter

The sheet sorter shows all sheets in a workbook as thumbnail images on a single page. The sheet sorter is useful when you have a large number of sheets in a workbook. Open the sheet

sorter by clicking the sheet sorter button on the status bar (bottom right corner of the workbook).



Right-click a sheet (Control-click on the Mac) to see these commands. You can also right-click (Control-click on the Mac) to refresh the thumbnail image of a particular sheet or **Refresh All Thumbnails** at once. If you have a large number of sheets, refreshing your view can take some time.

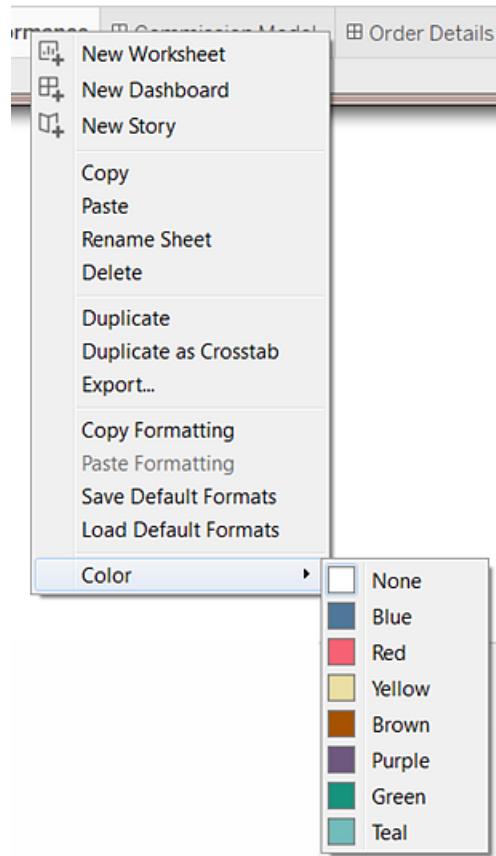
Sheet commands

Use sheet commands to manage and organize your worksheets. For example you can create new sheets, duplicate sheets, copy formatting, apply color, or delete the sheet entirely.

You can access sheet commands on the right-click menu (Control-click on a Mac) in the worksheet, sheet sorter, or the filmstrip view. To apply commands to multiple sheets at once, press the Ctrl key (Shift key on a Mac), and then select the sheets.

To make it easier to identify and group sheets, you can assign a color to sheets. You can select from seven different colors. Selecting **None** clears the color.

To assign a color to sheets, select one or more sheets, right-click the sheets (Control-click on the Mac), select **Color**, and then pick a color.



The color strip appears on the bottom of the tab or sheet.



Files and Folders

You can save your work using several different Tableau specific file types: workbooks, bookmarks, packaged data files, data extracts, and data connection files. Each of these file types are described below.

- **Workbooks (.twb)** – Tableau workbook files have the .twb file extension. Workbooks hold one or more worksheets, plus zero or more dashboards and stories.
- **Bookmarks (.tmb)** – Tableau bookmark files have the .tmb file extension. Bookmarks contain a single worksheet and are an easy way to quickly share your work. For more information, see [Bookmarks on page 1273](#).

- **Packaged Workbooks (.twbx)** – Tableau packaged workbooks have the .twbx file extension. A packaged workbook is a single zip file that contains a workbook along with any supporting local file data and background images. This format is the best way to package your work for sharing with others who don't have access to the original data. For more information, see [Packaged Workbooks on page 1267](#).
- **Data Extract (.tde)** – Tableau data extract files have the .tde file extension. Extract files are a local copy of a subset or entire data that you can use to share data, work offline, and improve database performance.
- **Data Source (.tds)** – Tableau data source files have the .tds file extension. Data source files are shortcuts for quickly connecting to the original data that you use often. Data source files do not contain the actual data but rather the information necessary to connect to the actual data as well as any modifications you've made on top of the actual data such as changing default properties, creating calculated fields, adding groups, and so on. For more information, see [Export Data Sources on page 461](#).
- **Packaged Data Source (.tdsx)** – Tableau packaged data source files have the .tdsx file extension. A packaged data source is a zip file that contains the data source file (.tds) described above as well as any local file data such as Extract files (.tde), text files, Excel files, Access files, and local cube files. Use this format to create a single file that you can then share with others who may not have access to the original data stored locally on your computer. For more information, see [Export Data Sources on page 461](#).

These files can be saved in the associated folders in the My Tableau Repository directory, which is automatically created in your My Documents folder when you install Tableau. Your work files can also be saved in other locations, such as your desktop or a network directory.

[Changing the Repository Location](#)

You can specify a new location for the Tableau repository if you are not using the default location in your Documents folder. For instance, if you are required to have your data on a network server instead of on your local machine, you can point Tableau at the remote repository.

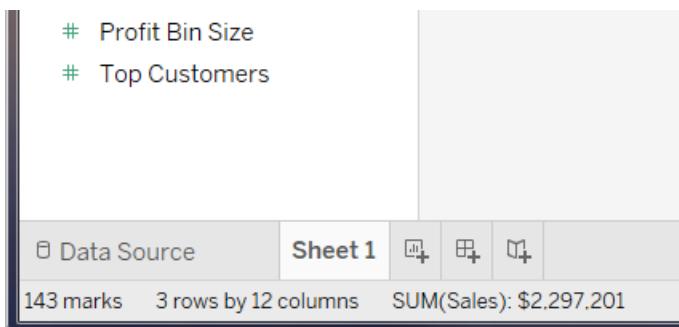
1. Select **File > Repository Location**.
2. Select a new folder that will act as the new repository location in the Select a Repository dialog box.
3. Restart Tableau so that it uses the new repository.

Changing the repository location does not move the files contained in the original repository. Instead, Tableau creates a new repository where you can store your files.

Status Bar

The status bar is located at the bottom of the Tableau workspace. It displays descriptions of menu items as well as information about the current view. For example, the status bar below

shows that the view has 143 marks shown in 3 rows and 12 columns. It also shows that the SUM(Sales) for all the marks in the view is \$2,297,201.



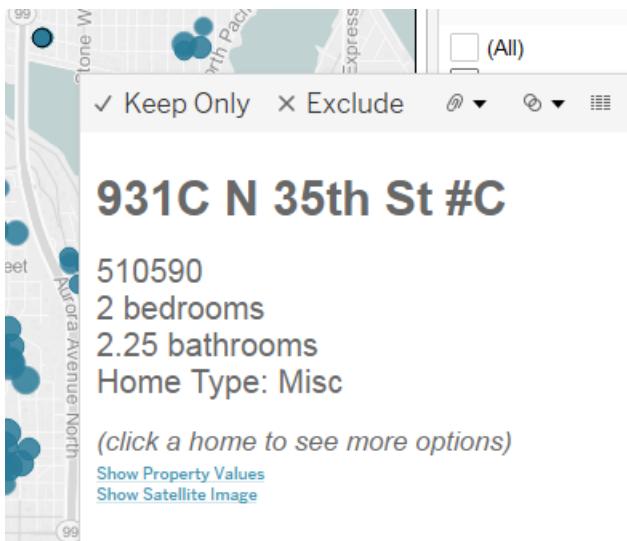
You can hide the status bar by selecting **Window > Show Status Bar**.

Occasionally, Tableau will display warning icons in the bottom right corner of the status bar to indicate errors or warnings. Below are the possible warning icons and what they mean.

Warning Icon	Description
	Cancel query indicator: When you cancel multiple queries, an indicator appears to show you how many queries are still running on the database and using resources. See Abandoned Queries on page 1835 for more information about managing these queries.
	Precision warning: Some fields are more precise in the database than Tableau can model. When you add these fields to the view a precision warning is displayed in the status bar. See Precision Warnings on page 1835 for more information about this warning.
	Special values indicator: If your data contains null values, unknown geographic locations, or negative or zero values on a logarithmic axis; the values are shown with an indicator in the lower right corner of the view. Click the indicator for options for handling these values. See Special Values on page 326 to learn more about this indicator and how to handle these values.

Toolips

Toolips are additional data details that display when you rest the pointer over one or more marks in the view. Toolips also offer convenient tools to quickly filter marks, view underlying data, group data, and create sets. Toolips consist of a body, action links, and commands.



Body

The body of a tooltip contains details about a specific mark or a selection of multiple marks. For example, in a bar chart showing sales by region, the tooltip body may include the actual sales amount and the region name. The default tooltip is based on the fields used in the view. You can customize what is shown in the tooltip by dragging fields to **Tooltip** on the marks card. Click **Tooltip** on the marks card to further customize the tooltip including how the text is formatted. Alternatively, you can select **Worksheet > Tooltip**. See [Format Titles, Captions, Tooltips and Legends on page 1077](#) to learn more about formatting the body of the tooltips.

Action links

If the sheet has any actions, the action links are listed below the body of the tooltip. An action adds context and interactivity to your data through filters, highlighting, and links to external resources. See [Actions on page 1116](#) to learn more about adding actions to your workbook.

Commands

The top of the tooltip lists commands to quickly filter data and view the underlying data. For example, you can use the tooltip to quickly remove an outlier in a scatter plot. Each of the commands are described below.

- **Keep Only** - creates a filter that removes all other data. See [Select to keep or exclude data points in your view on page 673](#) to learn more.
- **Exclude** - creates a filter that removes the selected data. See [Select to keep or exclude data points in your view on page 673](#) to learn more.
- **Group Members** - creates a group based on the selection. If the selection contains multiple dimensions, you can group on one dimension or all dimensions. See [Groups on page 497](#) to learn more.
- **Create Set** - creates a new set containing the selected members. You can create a new

set or add members to an existing set. See [Sets on page 512](#) to learn more.

- **View Data** - opens a window displaying the data. You can view the summarized data or the underlying data. See [View Data on page 573](#) to learn more.

These commands are visible by default. You can disable the commands in the Edit Tooltip dialog box.

[Disable tooltip commands](#)

1. Click Tooltip on the Marks card or select **Worksheet >Tooltip**.
2. In the Edit Tooltip dialog box, clear the **Include command buttons** check box in the bottom left corner.

Tooltip settings apply to the active worksheet and can be different for each sheet in the workbook.

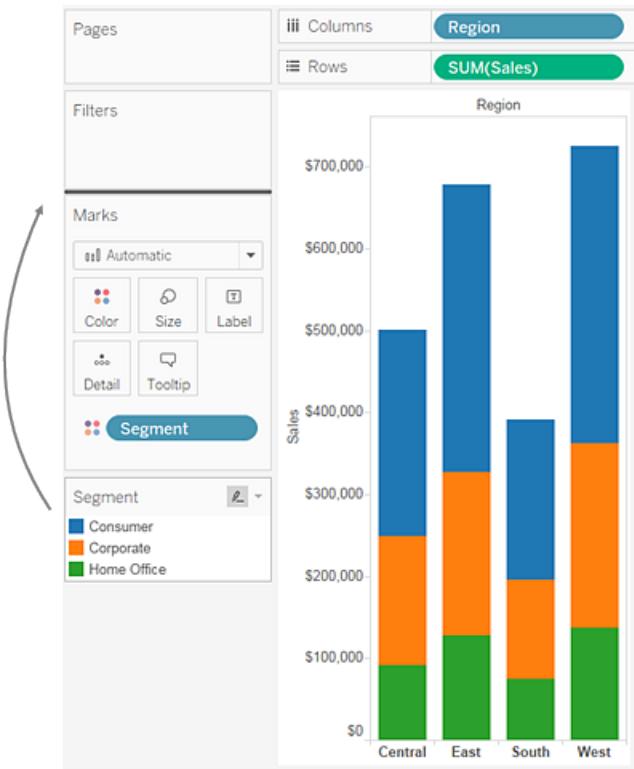
[Reorganizing the Workspace](#)

Every worksheet contains a variety of cards, shelves, legends, and so on. You can reorganize the workspace by rearranging cards, hiding and showing specific parts of the workspace, and hiding everything but the view using Presentation Mode.

[Rearranging Cards](#)

A worksheet contains several cards that contain shelves, legends, and other controls. Each card can be rearranged to create a custom workspace. To move a card, point the cursor at the

title area of the card you want to move. When the cursor becomes the move symbol , click and drag the card to a new position. As you drag the card around the worksheet, the possible positions for it are highlighted with a black bar.



You can restore the worksheet windows to their default state by selecting **Reset Cards** on the Show/Hide Cards toolbar control.

Showing and Hiding Parts of the Workspace

Just about everything in the workspace can be turned on and off so you can avoid cluttering the worksheet with unnecessary cards, shelves, etc.

- To show and hide the toolbar or status bar, select **Window** and then select what you want to hide.
- To show and hide the window on the left side, which is either the Data pane (for worksheets), the Dashboard pane (for dashboards), or the Dashboards and Sheets pane (for stories); click the Minimize button  in the upper right corner of the pane. The pane is minimized to the bottom left corner of the workbook. Click the same button again to restore the pane.
- To show or hide a card click **Show/Hide Cards**  on the toolbar and then select the card you want to show or hide.

You can restore the worksheet windows to their default state by selecting **Reset Cards** on the **Show/Hide Cards** toolbar control.

Presentation Mode

Sometimes you may want to use Tableau to present your findings. Rather than hiding each card or shelf one at a time, you can switch to presentation mode. Presentation mode hides everything on the sheet except for the view and its associated legends, filter cards, parameter controls, and worksheet tabs.

- To switch in and out of presentation mode, click the Presentation Mode  button on the toolbar or select **Window > Presentation Mode**.

Use the presentation mode controls in the bottom right corner to move between sheets and more. Each presentation mode control is described below.

	Show Filmstrip - shows the sheets as thumbnails at the bottom of the workspace.
	Show Tabs - shows the sheet tabs at the bottom of the workspace.
	Previous/Next Sheet - advances forward or backward through the sheets in a workbook.
	Enter/Exit Full Screen - switches between expanding the workbook to fill the entire screen and showing it in a window.
	Exit Presentation Mode - returns the workbook to showing the entire workspace including the menus, toolbar, and the Data pane.

Language and Locale

Tableau Desktop is localized into several languages.

When you first run Tableau, it recognizes your computer locale and uses the appropriate language if it is supported. If you are using an unsupported language, the application defaults to English.

You can configure Tableau to display the user interface (menus, messages, etc.) by choosing **Help > Choose Language**. After you change this setting, you'll need to restart the application for the changes to take effect. You do not need to change this setting for every workbook.

To configure date and number formatting, choose **File > Workbook Locale**. By default, the locale is set to **Automatic**, which means the locale will match the locale when the workbook is opened. This can be useful if you are authoring a workbook that will be viewed in many different

languages and you want the dates and numbers to update accordingly. When you select a specific locale, the workbook will not change regardless of who opens it.

Tableau checks the following, in order, to determine the workbook locale:

- Workbook Locale (explicit setting)
- Windows Locale or Mac language
- Tableau Language

If none of the above is set, then the workbook locale defaults to English.

Day of the Week Sorting

If you are working in a language for which Tableau does not provide a local version, set your workbook locale to assure that Tableau can sort the days of the week in the correct chronological order. Otherwise, Tableau will sort the names of the days alphabetically. If none of the supported locales is appropriate, you can sort the days of the week manually. See [Manual Sorting on page 495](#).

Visual Cues and Icons in Tableau Desktop

Tableau provides many visual cues to help you evaluate the type of data that's displayed in the Data pane and the state of a data view.

Data sources in the Data pane

The following table explains each of the icons used to describe the type of data sources in the Data pane. Each icon in the table can be modified by one of two indicators.

- A blue check mark indicates that the data source is the primary data source in the workbook. 
- An orange check mark indicates that the data source is the secondary data source in the workbook. 

Visual Cue	Description
	The workbook is directly connected to a relational data source or file.
	

Visual Cue	Description
	The workbook is connected to a cube (multidimensional) data source. In Tableau, cube data sources are supported only in Windows.
	The workbook is connected to an extract that still references the underlying data.
	The workbook is connected to a data source that has been published to Tableau Server.

Fields in the Data Pane

The following table explains each of the icons displayed in the Data pane. Each icon in the table can be modified by one of four indicators.

- Blue icons indicate that the field is discrete.  [Abc](#)
- Green icons indicate that the field is continuous.  <#>
- Icons preceded by the equal sign (=) indicate that the field is a user-defined calculation or a copy of another field.  [=Abc](#)

Visual Cue	Description
 Abc	The field contains text values.

Visual Cue	Description
	The field contains numeric values.
	The field is a calculation defined on the server.
	The field contains only date values.
	The field contains both date and time values.
	The field contains geographical data and has been assigned a geographic role. Use these fields when building map views. See Build and Use Maps on page 702 .
	The field contains geographical data from an active custom geocoding file. See Custom Geocode Your Data on page 718 .
	The field contains boolean (true or false) values.
	The field is a calculation that is defined in the database by an administrator. These fields are marked with a cylinder icon and are not available for all data sources.
	The field is a user-defined set. See Sets on page 512 .
	The field is a server named set.
	The field is a set that was automatically created as a result of an action.

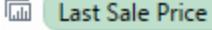
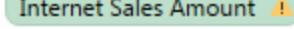
Visual Cue	Description
	The field is a user filter, used when publishing to the web. See Restrict Access at the Data Row Level on page 1252.
	The field is a numeric bin. See Create Bins from a Continuous Measure on page 597.
	The field is a group. See Groups on page 497.
	The field is a relational hierarchy. See Hierarchies (For Relational Databases) on page 299.
	A folder that contains one or more fields. Folders are used to organize fields in the Data pane. See Organize the Data Pane on page 305.
	The field is an attribute of a cube (multidimensional) data source. In Tableau, cube data sources are supported only in Windows.
	The field is a varying attribute of a cube (multidimensional) data source.
	The field is a level in a multidimensional hierarchy. Levels greater than five are shown without numbers.
	The field is blended with a field from another data source. See Primary and secondary data sources on page 371
	The field is not blended with a field from another data source. See Primary and secondary data sources on page 371

Fields on Shelves

Fields placed on shelves use a combination of icons, colors, and text styles as visual cues.

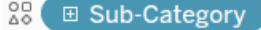
Visual Cue	Description
	A blue field on a shelf indicates a discrete field. In most cases, adding a dimension to a shelf results in a blue field. Blue fields are discrete—they contain a finite number of values. Adding a

Visual Cue	Description
	blue field to a shelf creates headers. See Headers on page 161.
	A green field on a shelf indicates a continuous field. In most cases, adding a measure to a shelf results in a green field. Green fields are continuous—they contain an infinite number of values. Adding a green field to a shelf creates an axis. See Axes on page 164.
	The Sort icon indicates a field that has either a computed or manual sort order applied. See Computed Sorting on page 480.
	The sigma icon indicates a slicing filter in a multidimensional (cube) data source. See Create Slicing Filters on page 699.
	The Venn diagram icon indicates a set. See Sets on page 512.
	A field name shown in italics indicates a filtered set.
	A gray field on the Filters shelf indicates a context filter. See Improve View Performance with Context Filters on page 694.
	The delta icon indicates that the field is a table calculation. See Transform Values with Table Calculations on page 843.
	The plus and minus controls appear when the field is part of a hierarchy that you can traverse.
	The arrow icon indicates that a forecast is being displayed for the field. This icon is also used on the Forecast Indicator field, which is an automatic field used to distinguish between actual and forecast values. See Forecasting on page 635.

Visual Cue	Description
 SUM(Sales ..)	The field is from a secondary data source. See Blend Your Data on page 369 .
 Last Sale Price	The field is assigned to a specific worksheet.
 Segment	The field is assigned to all worksheets with the same data source.
 Internet Sales Amount	The field is incompatible with one or more other fields in the view.

Fields on the Marks card

Fields placed on the Marks card use specific icons to describe how they appear in the view. See [Mark Properties on page 190](#).

Visual Cue	Description
 Segment	The field is applied to Color on the Marks card.
 SUM(Profit) ▾	The field is applied to Size on the Marks card.
 AVG(Discou..) ▾	The field is applied to Label on the Marks card.
 Sub-Category	The field is applied to Shape on the Marks card.
 SUM(True Average)	The field is applied to Detail on the Marks card.

Visual Cue	Description
 AVG(Math)	The field is applied to Tooltip on the Marks card.
 YEAR(Date)	The field is applied to Path on the Marks card. Path is only available when the Line or Polygon mark type is selected from the Marks drop-down menu. See Path Properties on page 227.

Sheets in the Dashboards and Worksheets pane

The following table explains each of the icons used to describe the type of sheet that can be placed in a story. A blue check mark indicates that a sheet is being used in one or more story points. 

Visual Cue	Description
 	The sheet is a worksheet.
 	The sheet is a dashboard.

Tableau Concepts

Why are some fields dimensions and others measures?

What effect will adding a filter have on my view?

Why is the background color blue for some fields, and green for others?

The topics in this section attempt to clarify these and other questions about what you can see and experience as you use Tableau Desktop.

If you're new to Tableau Desktop, also consider working through the [Build-It-Yourself Exercises](#) on page 57, and check out the [Free Training Videos](#) on the Tableau website.

Data Types

All fields in a data source have a data type. The data type reflects the kind of information stored in that field, for example integers (410), dates (1/23/2015) and strings ("Wisconsin"). The data

type of a field is identified in the Data pane by one of the icons shown below.

Data type icons in Tableau

Icon	Data type
	Text (string) values
	Date values
	Date & Time values
	Numerical values
	Boolean values (relational only)
	Geographic values (used with maps)

You can change the data type for a field either on the **Data Source** page or in the **Data** pane.

Change the data type for a field in the Data Source page

Sometimes Tableau incorrectly interprets the data type of a field. For example, Tableau might interpret a field that contains dates as an integer data type, rather than a date data type.

You can change the data type for a field that was part of the original data source (as opposed to a calculated field created in Tableau) on the **Data Source** page.

1. Click the field type icon for the field (as shown in the table above).
2. Choose a new data type from the drop-down list:

The screenshot shows the Tableau Data pane with a data source containing several fields: Ship Mode, Customer Name, Segment, Country, and City. The 'Customer Name' field is selected, indicated by a blue border around its column header. A context menu is open over this field, listing various data types: Number (Decimal), Number (Whole), Date & Time, Date, String, Boolean, Default, and Geographic Role. The 'Default' option is currently selected, as indicated by a checked checkbox icon next to it. Other options like 'Number (Decimal)' and 'String' also have small checkbox icons next to them.

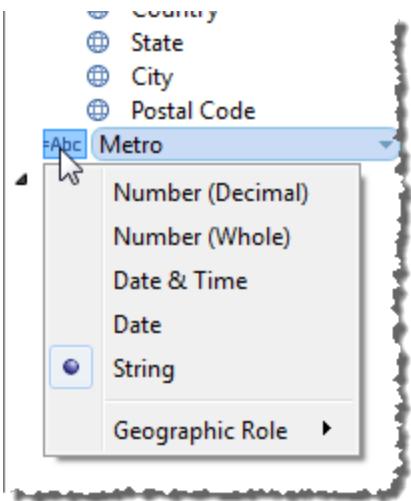
Ship Mode	Customer Name	Segment	Country	City
Orders	Orders	Orders	Orders	Orders
Second Class	Number (Decimal)	Consumer	United States	Hende
Second Class	Number (Whole)	Consumer	United States	Hende
Second Class	Date & Time	Corporate	United States	Los An
Standard Class	Date	Consumer	United States	Fort La
Standard Class	String	Consumer	United States	Fort La
Standard Class	Boolean	Consumer	United States	Los An
Standard Class	Default	Consumer	United States	Los An
Brosina Hoffman	Geographic Role	Consumer	United States	Los An
Brosina Hoffman		Consumer	United States	Los An
Brosina Hoffman		Consumer	United States	Los An

Tip: Be sure to change data types before you create an extract. Otherwise, your data may not be accurate. For example, if a floating-point field in the original data source is interpreted as an integer by Tableau, and you create your extract before you change the field's data type, the resulting floating-point field in Tableau will have some of its precision truncated.

For information on changing data types on the **Data Source** page, see [Data Source Page on page 130](#).

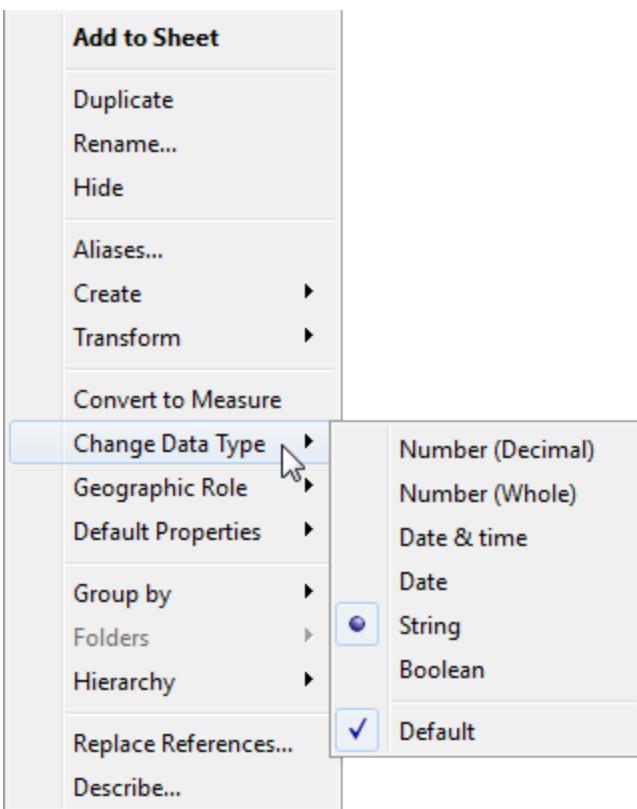
Change the data type for a field in the Data pane

To change the data type of a field in the **Data** pane, click the icon to the left of the field name, and then choose a new data type from the drop-down list.



Change the data type for a field in the view

To change a field's data type in a view, right-click (control-click on a Mac) the field in the **Data** pane, choose **Change Data Type**, and then select the appropriate data type from the drop-down list.



Note: Sometimes the data in your database is more precise than Tableau can model. When you add these values to the view, a precision warning appears in the right corner of the status bar. See [Precision Warnings](#) on page 1835.

Mixed data types in data from files

Most columns in a Microsoft Excel, Microsoft Access, or CSV (comma-separated value) file contain values that are all of the same data type (Booleans, dates, numbers, or text). When you connect to the file, Tableau creates a field in the appropriate area of the **Data** pane for each column. Dates and text values are dimensions, and numbers are measures.

However, files that you connect to might include columns that have a mixture of data types, such as numbers and text, or numbers and dates. When you connect to the file, the mixed-value column is mapped to a field with a single data type in Tableau. Therefore, a column that contains numbers and dates might be mapped as a number data type (making it a measure) or it might be mapped as a date data type (in which case Tableau treats it as a dimension.)

Tableau determines how to map mixed-value columns to data types by the data types of the first 10,000 rows in an Excel data source, and the first 1,024 rows in a CSV data source. For example, if most of the first 10,000 rows are text values, the entire column is mapped to use the text data type.

Note: Empty cells also create mixed-value columns because their formatting is different from text, dates, or numbers.

When Tableau determines a data type for each field, if the values in a field don't match that data type, Tableau handles the field in one of several different ways, depending on the data type. For example, sometimes Tableau populates those fields with Null values, as shown in the following table:

Mapped data type	Treatment of other data types in the field.
Text	Dates and numbers are treated as text. Nulls are not created.
Dates	Text is treated as Null. A number is treated as the day in numeric order from 1/1/1900.
Numbers	Text is treated as Null. A date is treated as the number of days since 1/1/1900.
Boolean	Text, dates, and numbers are treated as Null.

If using fields that are based on mixed-value columns introduces difficulties when analyzing your data, you can do one of the following:

- Format empty cells in your underlying data source so that they match the data type of the column.
- Create a new column that does not contain the mixed values.

Field Types

When you connect to a new data source, Tableau assigns each field in the data source to either the Dimensions area or the Measures area of the Data pane, depending on the type of data the field contains. If a field contains categorical data (such as names, dates, or geographical data), Tableau assigns it to the Dimensions area. If a field contains numbers, Tableau assigns it to the Measures section.

So is it correct to say that a dimension is a field that contains categorical data, such as names, dates, or geographical data, and that measure is a field that contains numbers? Those assertions are accurate enough as a starting point, but as you work in Tableau, remember that you control the definition of a field in the view. Most fields can be used as either a dimension or as a measure, and can be either continuous or discrete, according to the requirements of the user.

Tableau's initial assignment of fields to either the Dimensions area or the Measures area establishes a default. When you click and drag a field from the Data pane to a view, Tableau continues to provide a default definition for the field. If you are dragging a field from the Dimensions area, the resulting field in the view will be discrete (with a blue background). If you are dragging a field from the Measures area, the resulting field will be continuous (with a green background).

Note: If you want to be able to tell Tableau how to categorize a view you drag to the view, so as to override the default, right-click it (Control-click on a Mac) before you drag it to the view and Tableau will prompt you to specify how you want the field to be used in the view when you drop it.

By default, dimensions are discrete and measures are continuous, but in fact all four combinations are possible:

discrete dimensions	<code>Product Name</code>
continuous dimensions (possible only with Date dimensions)	<code>+ QUARTER(Order Date)</code>
discrete measures	<code>SUM(Profit)</code>

continuous measures	SUM(Profit)
---------------------	-------------

Note: With a cube (multidimensional) data source (supported only on Windows), the options for changing data roles are limited. You can change some measures from continuous to discrete, but in general, you cannot change data roles for fields in cube data sources.

Dimensions and Measures

When you connect to a data source, Tableau assigns each field in the data source as playing one of two possible data roles: dimension or measure. What effect do these assignments have when you start working with data in Tableau?

Dimensions

When you first connect to a data source, Tableau assigns any fields that contain discrete categorical information (for example, fields where the values are strings or Boolean values) to the **Dimensions** area in the **Data** pane.

When you click and drag a field from the **Dimensions** area to **Rows** or **Columns**, Tableau creates column or row headers.

The screenshot shows the Tableau Data pane with the following structure:

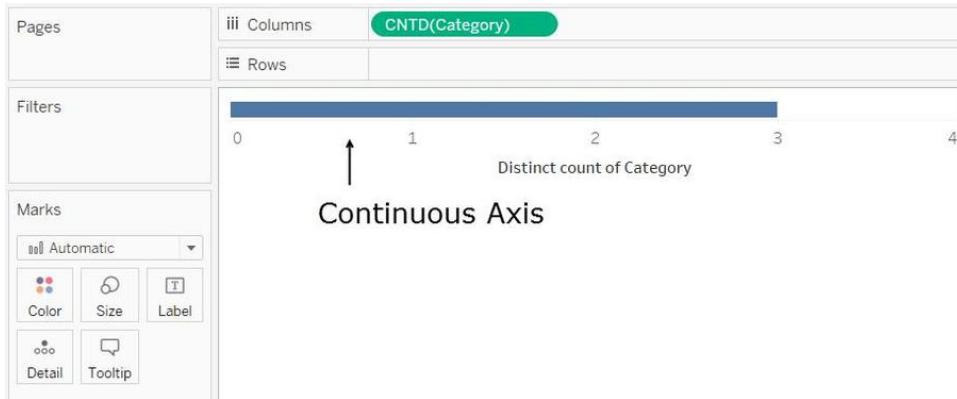
- Dimensions:**
 - Location (Country, State, City, Postal Code)
 - Product (Category, Sub-Category, Manufacturer, Product Name)
 - Profit (bin)
- Measures:**
 - Discount
 - Profit
 - Profit Ratio
 - Quantity
 - Sales

The **Category** field is selected in the Dimensions section, indicated by a blue background. In the main workspace, there is a table with three columns: Furniture, Category, and Technology. The Furniture column has a single row with value 'Abc'. The Category column has two rows: 'Furniture' with value 'Abc' and 'Office Supplies' with value 'Abc'. The Technology column has a single row with value 'Abc'. An arrow points from the text "Column Headers" to the right edge of the Category column header.

Any field you drag from the **Dimension** area will initially be discrete when you add it to a view, with a blue background. For this reason, it might be easy to assume (incorrectly, as it turns out) that a field's background color indicates whether it is a dimension or a measure. But Date dimensions and numeric dimensions can be either discrete or continuous, and all measures

can be discrete or continuous. A field's background color indicates whether it is discrete (blue) or continuous (green).

After you drag a dimension to **Rows** or **Columns**, you can change the field to a measure just by clicking the field and choosing **Measure**. Now the view will contain a continuous axis instead of column or row headers, and the field's background will become green:



If you want to make a dimension continuous (without first converting it into a measure), your options are limited. This is only possible with Date dimensions, which can be discrete or continuous but are always dimensions, and with numeric dimensions. You cannot convert dimensions containing strings or Boolean values.

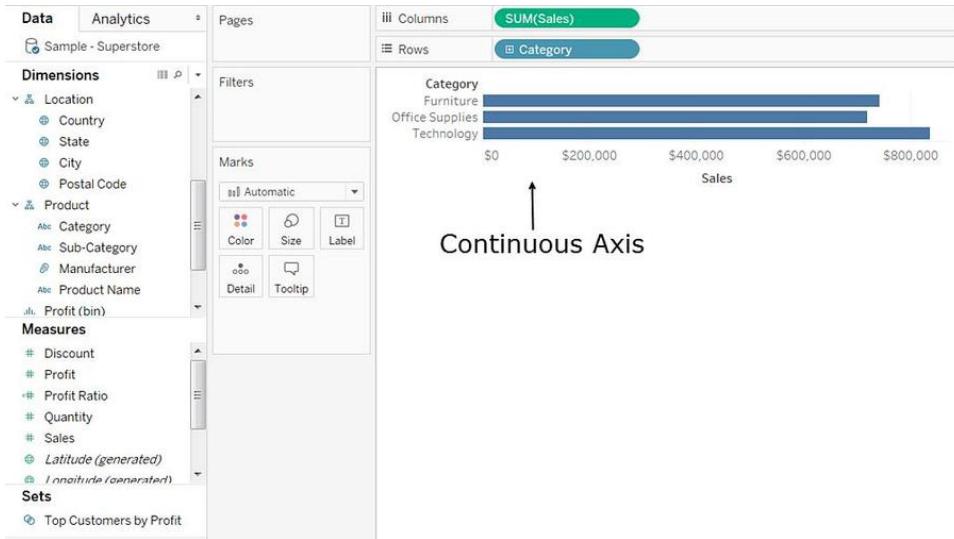
Tableau does not aggregate dimensions. If you want a field's values to be aggregated, that field must be a measure. When you convert a dimension into a measure, Tableau will always prompt you to assign it an aggregation (Count, Average, etc.). Aggregation means collecting multiple values (individual numbers) into a single number by, for example, counting the number of individual values, averaging them, or displaying the smallest individual value for any row in the data source. For a discussion of the different types of aggregation Tableau can perform, see [Aggregations on page 329](#).

In Tableau queries, dimensions in the view are expressed in SQL as "Group By" clauses.

Measures

When you first connect to a data source, Tableau assigns any fields that contain quantitative, numerical information (that is, fields where the values are numbers) to the **Measures** area in the **Data** pane.

When you drag a field from the Measures area to **Rows** or **Columns**, Tableau creates a continuous axis.



Any field that you drag from the Measures area will initially be continuous when you add it to the view, and so its background will be green. But if you then click the field and choose **Discrete**, the values become column headers.

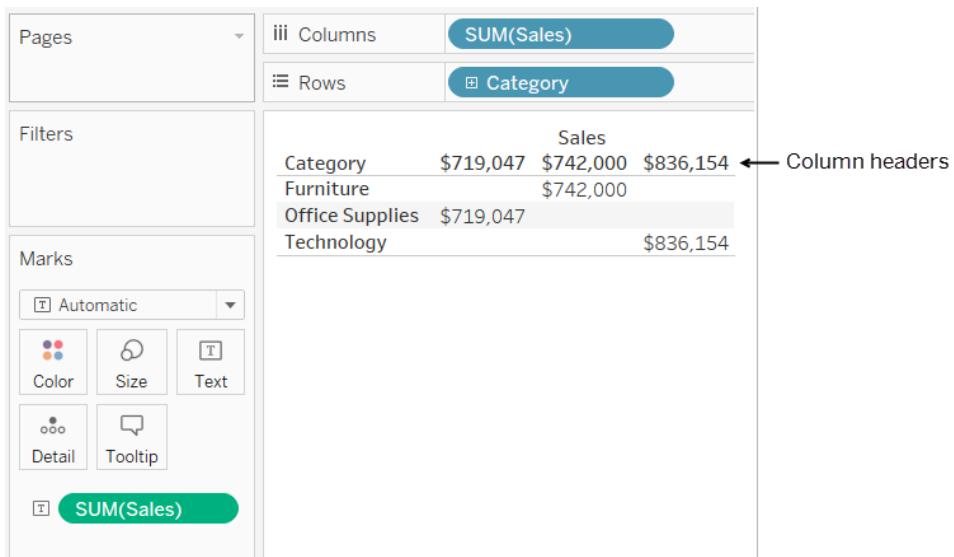


Tableau continues to aggregate values for the field, because even though the field is now discrete, it is still a measure, and Tableau will always aggregate measures. You can take the process one step further and convert the measure to a dimension if you like. Only then will Tableau stop aggregating its values. See [Convert a Measure to a Dimension](#) on the next page for an example of this process.

Visual identification of dimensions and measures in a view

If you are looking at a view in Tableau Desktop and you're not sure whether a field is a measure

or a dimension, a quick visual cue is that measures are aggregated: **SUM(Quantity)**

and dimensions are not: **Region**.

But there are exceptions:

- If the entire view is disaggregated, then by definition no field in the view is aggregated. See [Disaggregating Data on page 338](#).
- If you are using a multidimensional data source, fields are aggregated in the data source and measures fields in the view do not show that aggregation.

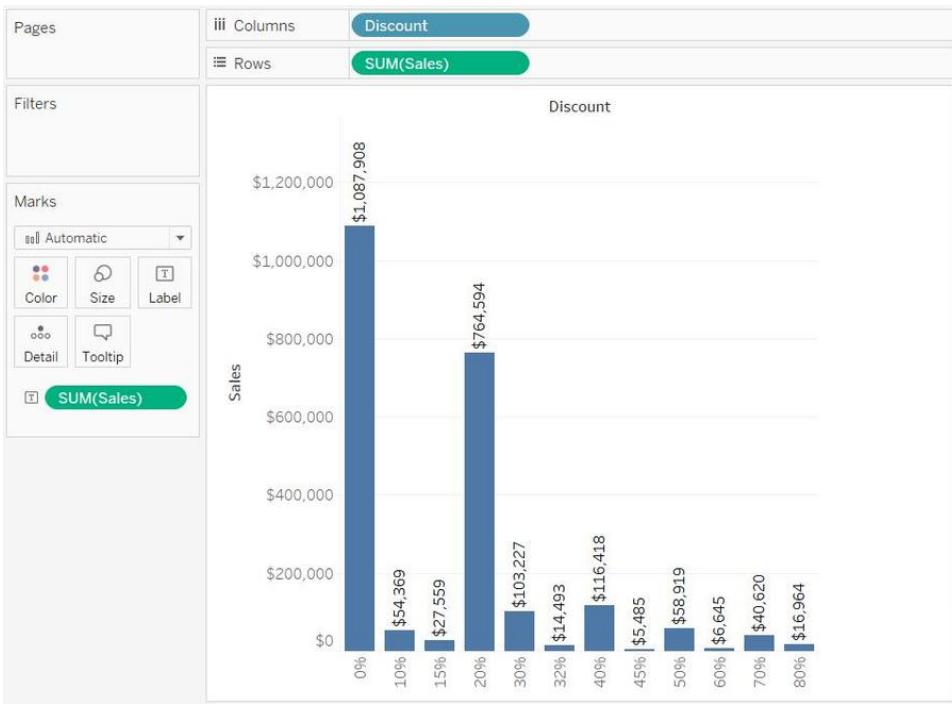
Convert a Measure to a Dimension

You can convert a field in a view from a measure to a dimension.

Or, if you want the change to affect all future uses of the field in the workbook, you can convert a field in the **Data** pane from a measure to a dimension.

Convert a measure in the view into a discrete dimension

You can drag a field from the **Measures** area in the **Data** pane but then use it as a dimension in the view. For example, suppose you want to know the aggregated sales totals for each possible discount rate. The view you are aiming for looks like this:



The **Discount** field contains numeric data, so when you connect to the data source, Tableau assigns it to the **Measures** area in the **Data** pane. In the **Sample - Superstore** data source, which is included with Tableau Desktop, the values for **Discount** range from 0% to 80%.

Here are the steps to create the view shown above:

1. Drag **Sales** to **Rows** and **Discount** to **Columns**. Tableau shows you a scatter plot—this is the default chart type when you put one measure on **Rows** and another on **Columns**.

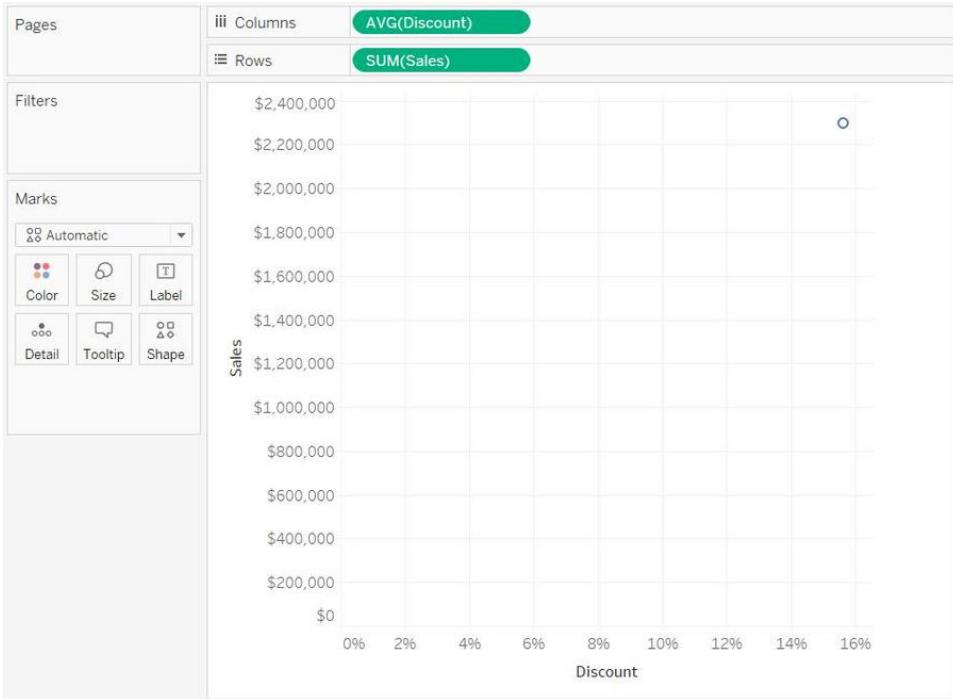
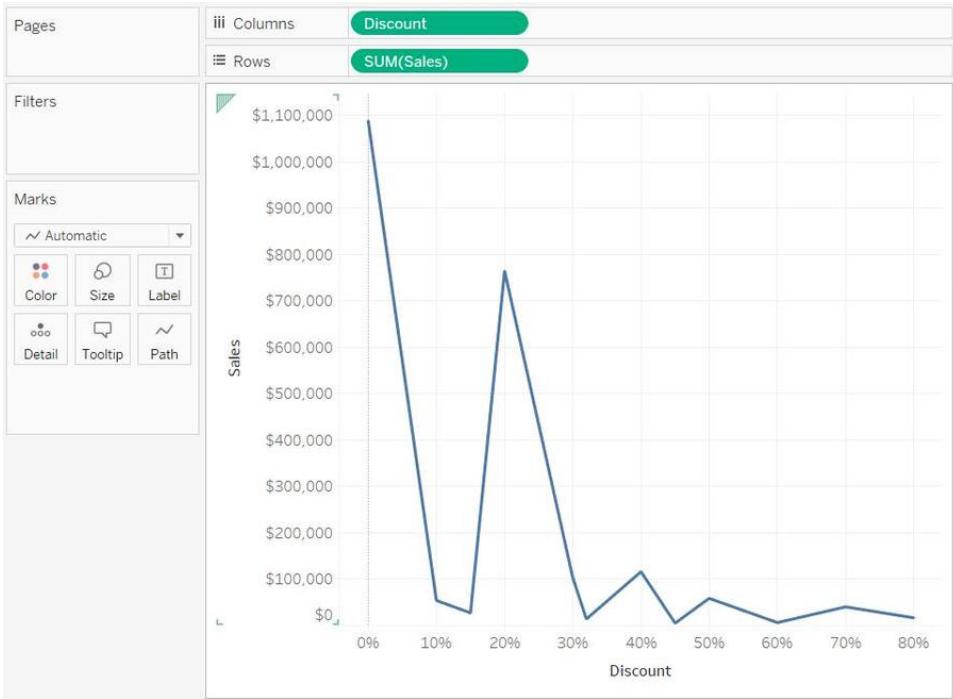


Tableau aggregates **Discount** as AVG, and **Sales** as SUM. The fields are both continuous, so along the bottom and left side of the view Tableau displays axes (and not column or row headers).

2. To treat **Discount** as a dimension, click the drop-down arrow on the field (on the **Columns** shelf) and select **Dimension** from the context menu. Tableau no longer aggregates the values for **Discount**, so what you see now is a line. But the values for **Discount** are still continuous, so Tableau still shows continuous axes for both fields:



- To complete the process, click the drop-down arrow on the **Discount** again and select **Discrete** from the context menu. The transformation of **Discount** is now complete. You now see the bar chart as in the initial image at the beginning of this topic. Across the bottom, you now see column headers (0%, 10%, 20%, etc.) instead of an axis.

Let's review how we got to this point:

Action	Result
Convert Discount from a measure to a dimension...	Sales values are no longer aggregated according to discount rate, resulting in a line chart instead of a scatter plot.
Convert Discount from continuous to discrete...	Tableau shows headers at the bottom of the view, instead of a continuous axis.

The only thing left to do is to drag **Sales** to Label and then format the labels for readability.

The resulting chart is somewhat useful because there are only 12 unique values for **Discount** in the data source. Had there been a unique value for each row, which would not have been unusual for a numeric field, the number of individual bars in the resulting view would have been equal to the number of rows in the data source, which would probably not result in a useful visualization.

Converting a measure in the Data pane into a dimension

When you first connect to a data source, Tableau assigns most fields that contain quantitative,

numerical information (that is, fields where the values are numbers) to the Measures area in the Data pane. The exception is for fields where the name suggests the data type, such as Year or Month (which Tableau would identify as Date dimensions) or fields containing words like “ID” and “Key,” which Tableau would categorize as dimensions, even when they are numeric.

However, you might decide that some of these fields that Tableau has categorized as measures should actually be dimensions. Postal codes are a classic example—they often consist entirely of numbers, but the information is categorical and not continuous—you would never want to aggregate postal codes by adding or averaging them. Similarly, a field containing individuals' ages may be categorized as a measure by default in Tableau because it contains numeric data. In some cases you may want to add or average ages, but you might also want to look at each individual age as a bin or category, in which case you want Tableau to create headers for this field rather than an axis. If this is how you want to use age in your view, you can convert the field to a dimension.

To convert a measure to a dimension in the **Data** pane, do either of the following.

- Click and drag the field from the measures area of the **Data** pane and drop it into the dimensions area.

The screenshot shows the Data pane of Tableau. The top navigation bar has 'Data' selected. Below it, the 'Analytics' tab is visible. The main area displays the 'Dimensions' and 'Measures' sections.

Dimensions

- Country
- State
- City
- Postal Code
- Product
 - Category
 - Sub-Category
 - Manufacturer
 - Product Name
 - Profit (bin)
 - Region
- Measure Names
SUM(Quantity)

Measures

- Quantity
- Sales
- Latitude (generated)

Sets

- Top Customers by Profit

Parameters

- Profit Bin Size
- Top Customers

- Right-click (control-click on a Mac) the measure in the **Data** pane and select **Convert to Dimension**.

If you place a field that you converted from a measure to a dimension on a shelf, it now produces headers instead of an axis.

The screenshot shows a Tableau interface with a data view. At the top, there are two buttons: 'Columns' (grayed out) and 'Category' (highlighted in blue). Below them is another button 'Rows' (grayed out) and 'Quantity' (highlighted in blue). The main area displays a data table with four columns: 'Quantity' (row header), 'Furniture' (\$18,491), 'Office Supplies' (\$17,224), and 'Technology' (\$17,536). The 'Quantity' column has 14 rows from 1 to 14. The other three columns show corresponding values for each quantity.

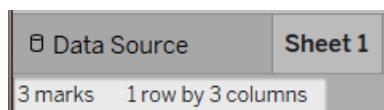
Quantity	Furniture	Office Supplies	Technology
1	\$18,491	\$17,224	\$17,536
2	\$87,642	\$94,617	\$106,832
3	\$137,952	\$142,844	\$141,265
4	\$85,081	\$88,262	\$150,328
5	\$127,640	\$136,841	\$151,180
6	\$73,934	\$65,456	\$67,732
7	\$69,862	\$74,779	\$95,267
8	\$43,690	\$36,475	\$37,595
9	\$55,255	\$32,040	\$41,211
10	\$13,438	\$3,864	\$6,755
11	\$7,035	\$3,695	\$5,042
12	\$8,783	\$2,342	\$3,743
13	\$8,300	\$15,895	\$8,143
14	\$4,896	\$4,712	\$3,524

Dimensions and the Level of Detail

As you add dimensions to **Rows** or **Columns**, the number of marks in the view increases. To understand why adding dimensions increases the number of marks in the view, do the following:

1. Drag **Segment** to **Columns**.

The status bar at the bottom of the Tableau window shows you that there are now three marks in the view:



Those marks just contain placeholder text, `Abc`, because you are only building the view's structure at this point.

2. Drag **Region** to **Columns**.

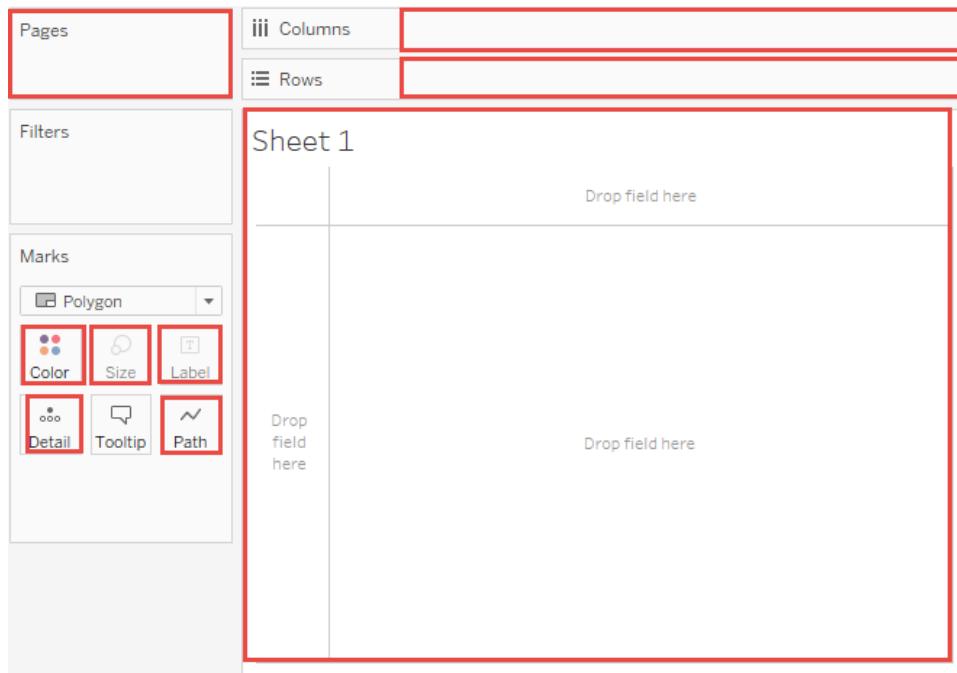
Now there are 12 marks. Three values in **Segment** multiplied by four values in **Region** is 12.

3. Drag **[Ship Date]** to **Rows**.

The total is now 57 marks (three segments by four regions by five years is 60, but there are three combinations of the dimensions in the view for which there is no data in the data source).

We could continue adding dimensions to **Rows** and **Columns** and observe as the number of total marks continues to increase. Dragging a dimension to a location on the Marks card such as Color or Size will also increase the number of marks, though it will not increase the number of headings in the view. The process of adding dimensions to the view to increase the number of marks is known as setting the *level of detail*.

Adding a dimension to any of the following locations in Tableau affects the level of detail:



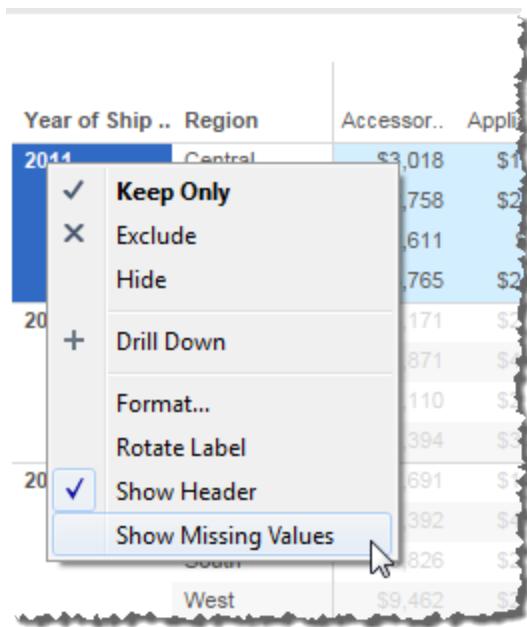
4. The view now contains 57 separate instances of Abc—the view is all structure and no content. Rectify this by dragging **Sales** to Text. The view can now be considered complete:



Notes

- In some cases, adding a measure to the view can increase the number of marks in the view. For example, if you dropped **Sales** on **Rows** in the view above, the number of marks would be 57. But if you then also dropped **Profit** on **Rows**, the number of marks would increase to 114. But this is not the same as changing the view's level of detail.
- The number of marks in the view is not guaranteed to correspond to the number you would get by multiplying the number of dimension values in each of the dimensions that make up the level of detail. There are multiple reasons why the number of marks could be lower. To increase the number of marks in this view from 57 to 60 in the view above, right-click (Control-click on a Mac) on one of the Date headers in the view and the date or bin headers and choose **Show Missing Values**. For more information about how to

show missing values, see [Missing Values](#) on page 602



Continuous and Discrete

Continuous and *discrete* are mathematical terms. Continuous means "forming an unbroken whole, without interruption"; discrete means "individually separate and distinct."

In Tableau, fields can be either continuous or discrete. When you drag a field from the **Measures** area to **Columns** or **Rows**, the values are continuous by default and Tableau creates an axis. When you drag a field from the **Dimensions** area of the **Data** pane to **Columns** or **Rows**, the values are discrete by default and Tableau creates column or row headers. Let's consider why this is so.

Continuous fields produce axes

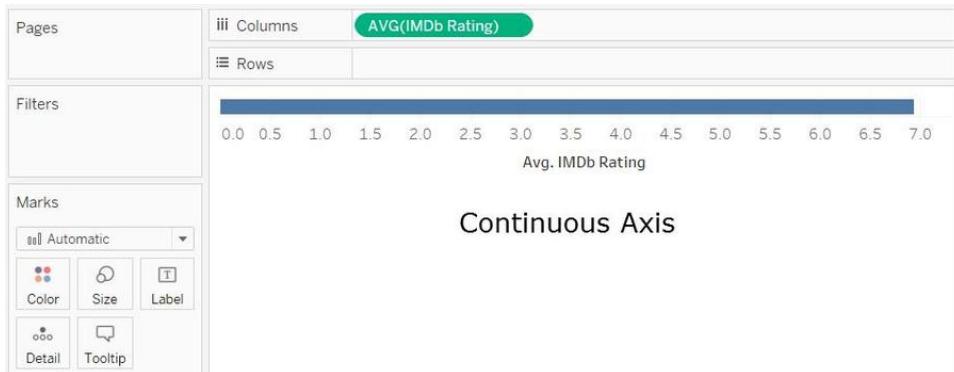
If a field has values that are numbers that can be added, averaged, or otherwise aggregated, Tableau assigns that field to the **Measures** area of the **Data** pane when you first connect to a data source. Tableau is assuming that the values are continuous.

Tableau displays an axis when you drop a continuous field on **Rows** or **Columns**. An axis is a measuring line that shows values between a minimum and a maximum. Rulers and analog thermometers are examples of physical objects that display axes.

Tableau must be able to show a range of actual and potential values, because in addition to the initial values in the data source, it is always possible that new values will emerge as you work with a continuous field in the view.

While there are value labels on a continuous axis (0, 0.5, ... 3.0 in the following image), actual marks don't have to align with these labels as they would with column headers. For example, in

the following image, the blue bar actually extends to a value of 6.940 on the horizontal axis, not 7.0 exactly.



The number of potential values for continuous fields is impossible to anticipate. For example, if you have a field named **Ratings** and the initial values are 1, 3, 3.5, 3.6, and 4, that's five distinct values. But if you drop **Ratings** on **Rows**, Tableau automatically aggregates that value as SUM (which you would then immediately change to AVG, because it's more logical to average grades than to add them), and that would then create a sixth value (3.02) that didn't exist until you added the field to the view. And if you then applied a filter that eliminated two of the initial values, the average would change as well, so that would be yet another value. And then if you changed the aggregation, ... You get the idea. The number of potential values is, if not infinite, then certainly immense.

The fact that a field contains numbers does not automatically indicate that those values are continuous. Postal codes are the classic example: though they are often composed entirely of numbers, they are actually string values which should never be added or averaged. If Tableau assigns such a field to the **Measures** area, you should drag it up to the **Dimensions** area.

Discrete fields create headers

If a field contains values that are names, dates, or geographical locations—anything other than numbers—Tableau assigns that field to the **Dimensions** area of the **Data** pane when you first connect to a data source. Tableau treats the values as discrete.

Tableau creates headers when you drop a discrete field on **Columns** or **Rows**. The individual values for a discrete field become the row or column headings. (Because such values are never aggregated, no new field values are created as you work with your view, so there is no need for an axis).

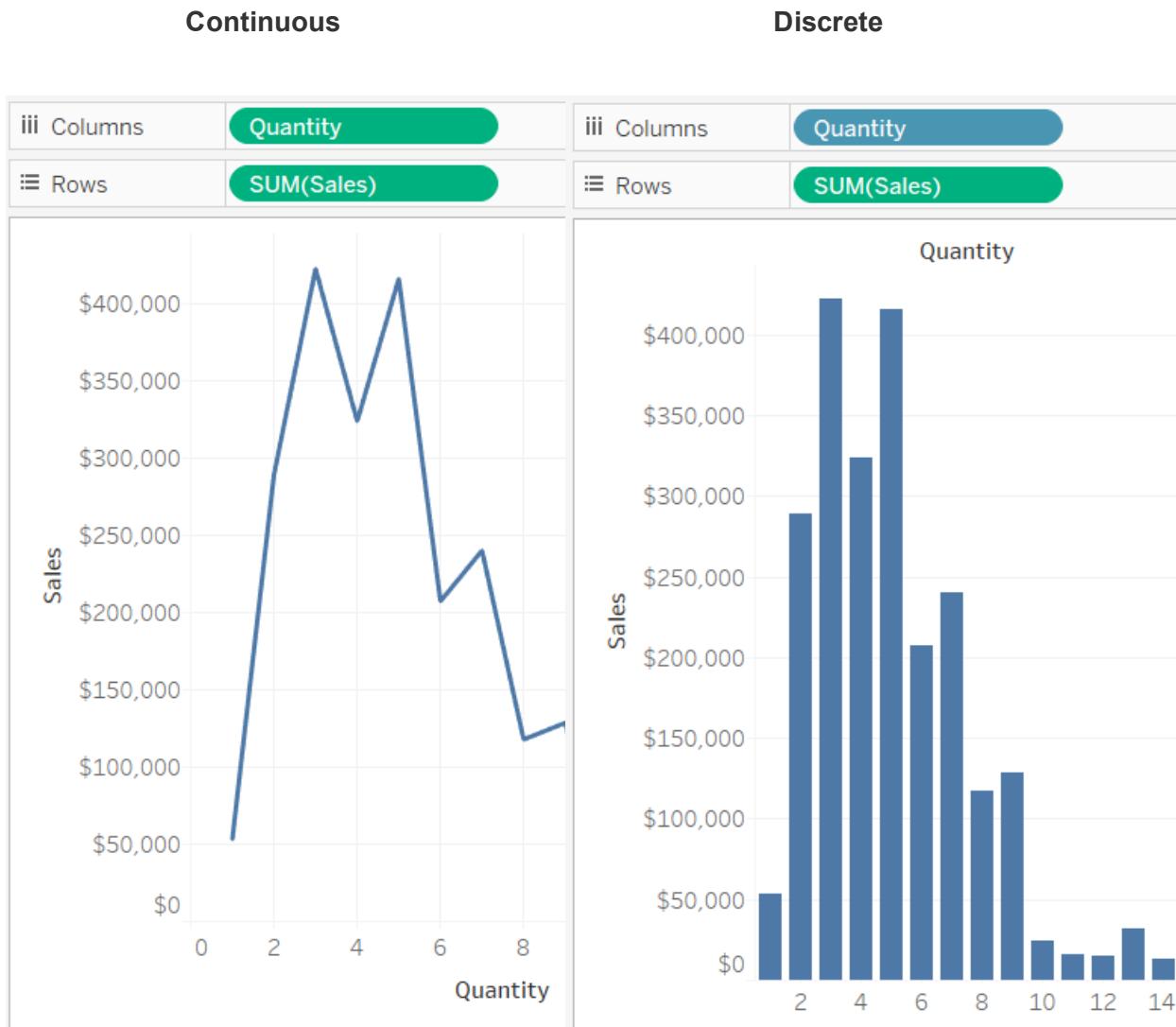
Recognizing the difference

If a field is continuous, the background color is green. If it is discrete, the background color is blue. Background color does not indicate dimension vs. measure—it indicates continuous vs. discrete.

What tells you whether a field in the view is a measure or a dimension is whether it is aggregated.

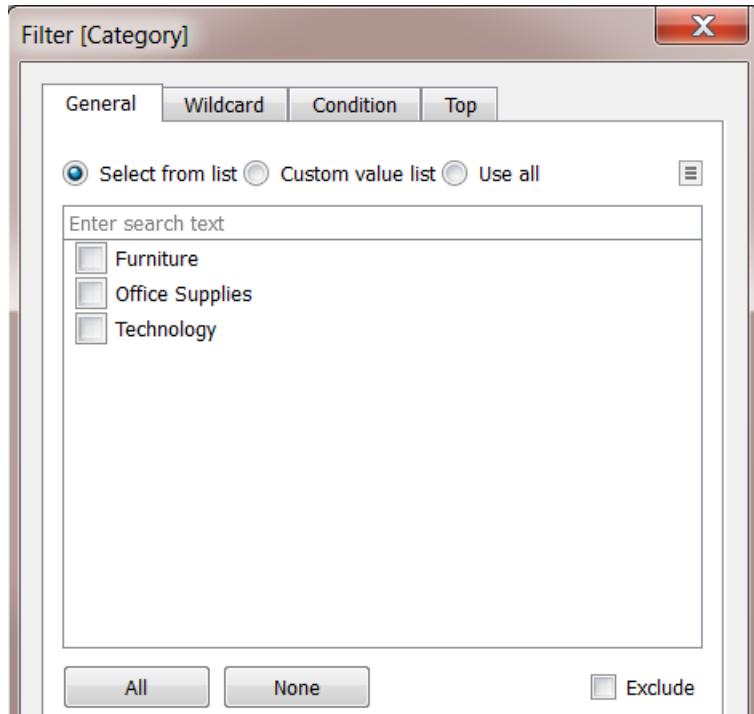
On the left below you see a view where **Quantity**, which was dragged from the **Measures** area of the **Data** pane, has been converted from a measure to a dimension but is still continuous, with an axis along the bottom of the view. We know that the field is continuous because of the axis and because the background is green. We know that it is a dimension because it is not aggregated.

On the right, you see a view where that view has been further modified by clicking **Quantity** on the **Columns** shelf and choosing **Discrete**. Now there are headers along the bottom of the view, instead of an axis.



Implications for filters

When you drop a discrete field on the **Filters** shelf, Tableau prompts you to choose which "members" of the discrete field to include in the view.



When you drop a Date field on Filters, the result can be a discrete filter or a continuous filter.

For more information, see [Filter Data from Your Views on page 672](#).

When you drop a continuous measure on Filters, Tableau first prompts you to choose an aggregation for the filter, and then prompts you to specify how to filter the continuous range of values.

When you drop a continuous dimension on Filters (other than a Date), Tableau prompts you to specify how to filter the continuous range of values.

For more on filtering various types of fields, see [Drag dimensions, measures, and date fields to the Filters shelf on page 675](#).

Implications for color

When you drop a discrete field on Color, Tableau displays a categorical palette and assigns a color to each value of the field.

When you drop a continuous field on Color, Tableau displays a quantitative legend with a continuous range of colors.

For more information about color palettes, see [Color Properties on page 192](#).

Convert Fields between Discrete and Continuous

You can convert measures from discrete to continuous or from continuous to discrete. And you can convert date dimensions and other numeric dimensions to be either discrete or continuous.

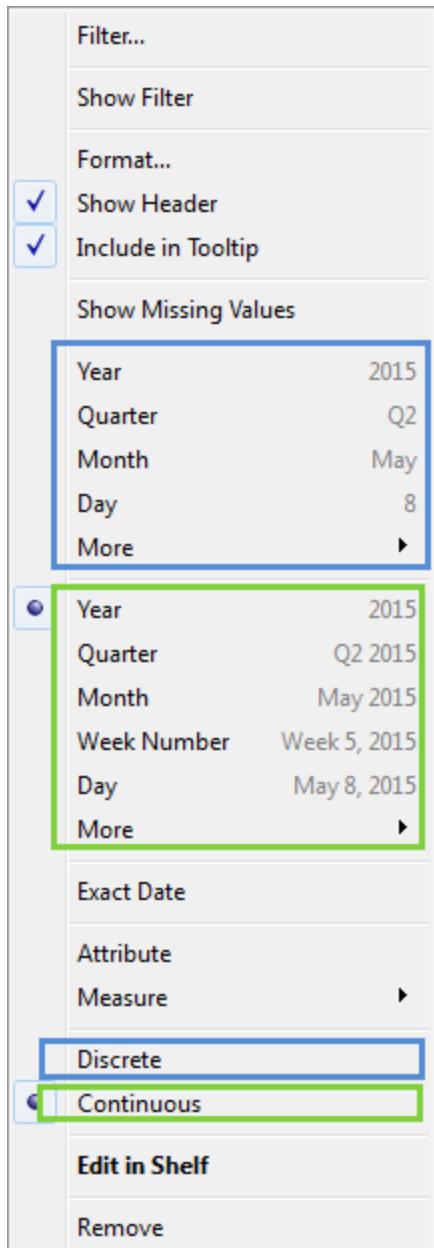
Convert measures

You can convert measures from discrete to continuous or from continuous to discrete. Click the field and choose **Discrete** or **Continuous**. The field is green when it is continuous, and blue when it is discrete.

For measures in the **Data** pane, right-click the field and choose **Convert to Discrete** or **Convert to Continuous**. The color of the field changes accordingly.

Convert date fields

You can convert Date fields between discrete and continuous. Click any Date field in the view and choose one of the options on the context menu to change it from discrete to continuous or from continuous to discrete:



Click any of the options in the blue areas to configure the field as a discrete date.
Choosing one of these options creates what is known as a "date part."

Click any of the option in the green areas to configure the field as a continuous date.
Choosing one of these options creates what is known as a "truncated date."

To convert a Date field in the **Data** pane (and thus to determine the default result when you drag it into a view), right-click the field and choose **Convert to Discrete** or **Convert to Continuous**.

Tableau's Order of Operations

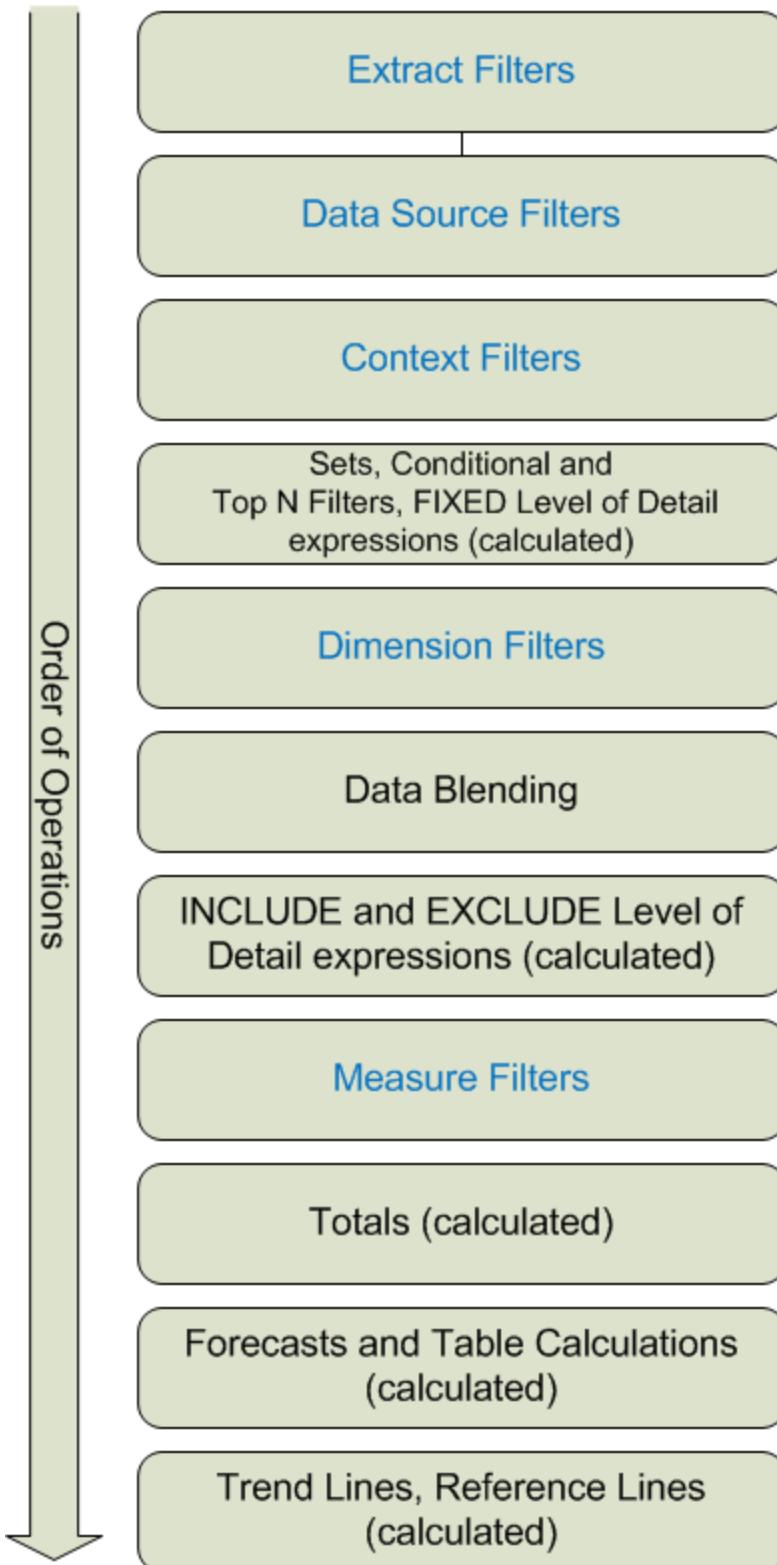
The order of operations in Tableau, sometimes called the query pipeline, is the order in which Tableau performs various actions. (Actions are also known as operations).

Many operations apply filters, which means that as you build a view and add filters, those filters always execute in the order established by the order of operations.

Sometimes, you might expect Tableau to execute filters in one order, but the order of operations dictates that they be executed in a different order, which gives you unexpected results. When this happens, you can sometimes change the order in which operations are executed in the pipeline.

This topic provides two scenarios for updating a view to correct problems resulting from the order of operations: converting a dimension filter to a context filter, and converting a table calculation to a FIXED level of detail expression.

The Tableau order of operations includes all the elements in the following illustration. Filters are shown in blue; other operations, which are mostly calculations, are shown in black.



Example 1: Convert a Dimension Filter to a Context Filter

This and the following example use the **Sample – Superstore** data source provided with Tableau Desktop.

In this example, the view addresses the following question: Who are the top 10 customers, by total sales, in New York City?

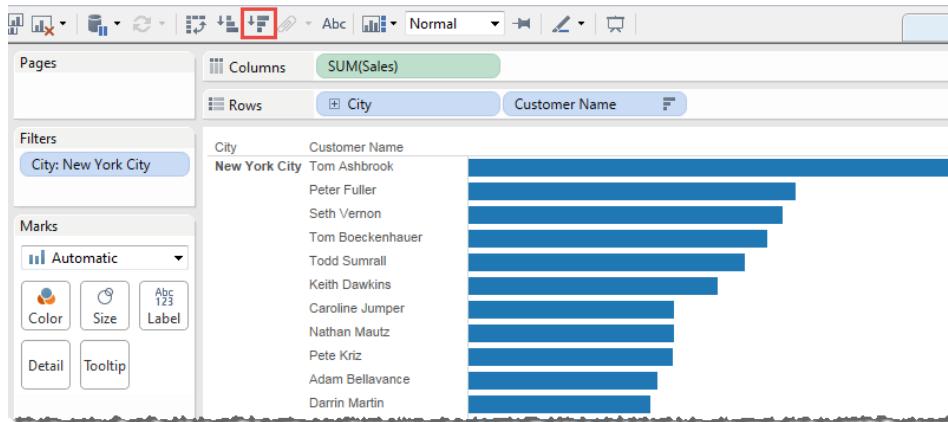
The view contains two dimension filters, one that you create on the **General** tab in the Filters dialog box, and the other on the **Top N** tab. The problem is that these filters are executing simultaneously, whereas you would like to general filter to be applied before the top n filter, so that the top n filter can act on the results as previously filtered by the general filter. The solution is to redefine one of the filters as a context filter so that a clear order of precedence is established.

Here are the steps for building this view.

1. Drag **Sales** to **Columns**.
2. Drag **City** and **[Customer Name]** to **Rows**.
3. Drag **City** from the **Data** pane again, this time to **Filters**. On the **General** tab in the Filter dialog box, set the filter to show just a single value: New York City. Do this by clicking **None** and then choosing **New York City**.

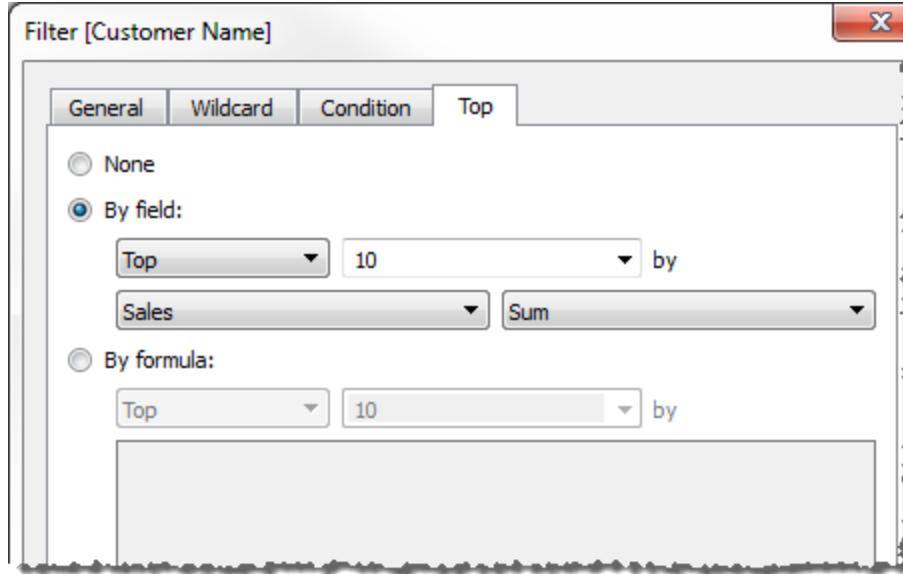
This creates a general dimension filter.

4. Click the **Sort Descending** button () on the toolbar. Your view now looks like this:

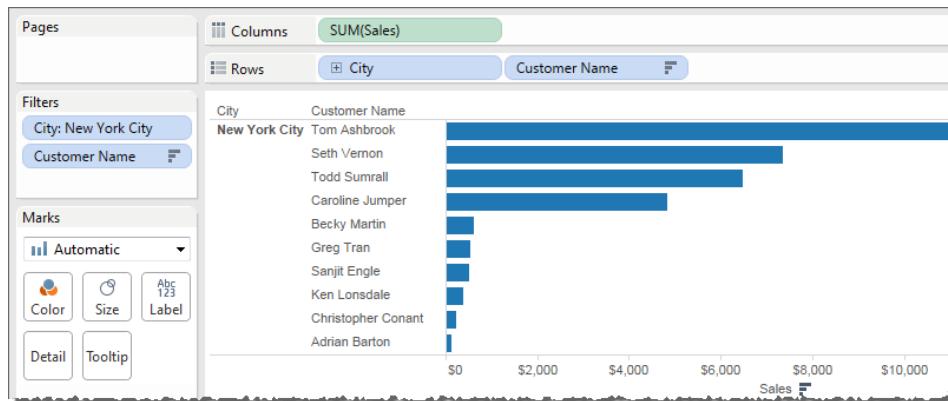


Note the first few names in the list: Ashbrook, Fuller, Vernon, etc.

5. Now drag **[Customer Name]** from the **Data** pane to **Filters**, and create a Top 10 Filter, to show only the top 10 customers by total sales:

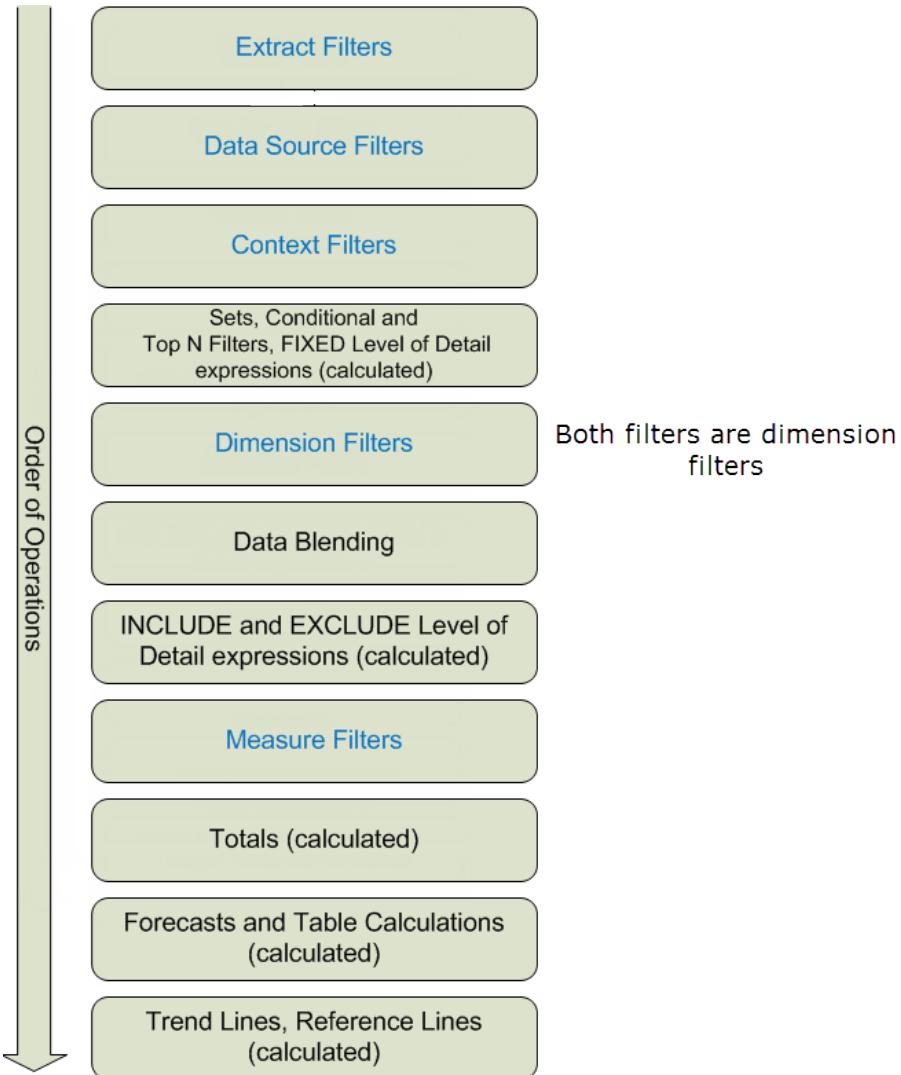


After you apply this second filter, the view looks right, but notice that the names shown are no longer the same as before:



What happened to Peter Fuller, formerly in second place? The goal was to show the top 10 customers in New York City, but now the view is actually showing the top 10 customers overall.

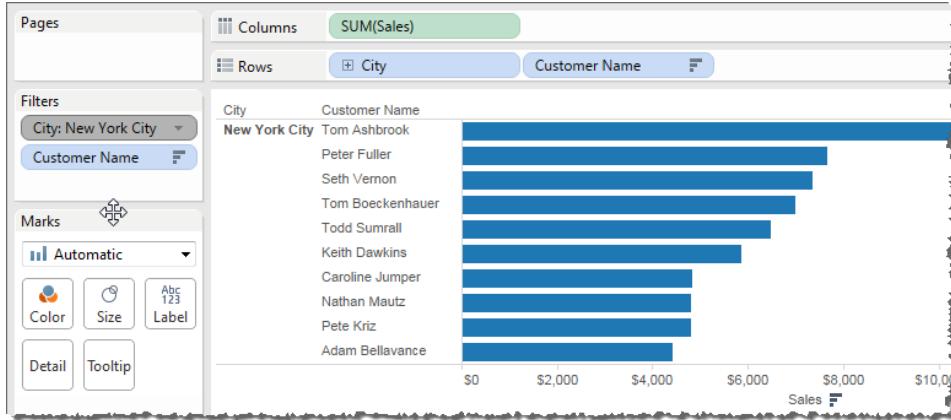
The problem is that top and general dimension filters are applied simultaneously—they are both dimension filters, and they appear in the same place in the Tableau order of operations:



The solution is to add the general dimension filter (on **City**) to context—that is, by turning it into a context filter, which is executed before any other filter that you create in a worksheet.

For details, see [Improve View Performance with Context Filters](#) on page 694.

- Right-click **City** on the **Filters** shelf (Control-click on a Mac) and choose **Add to Context**. As a context filter, this filter now takes precedence over the dimension filter, and so the view now shows what it's supposed to:



Example 2: Convert a Table Calculation to a FIXED Level of Detail Expression

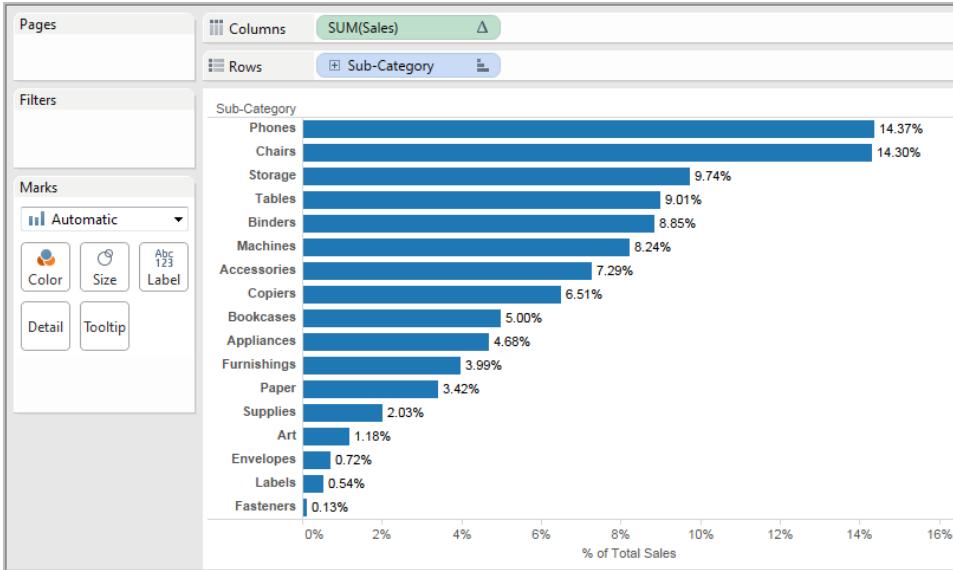
In this example, the view addresses the following question: What is the percent of total sales by product sub-category?

The view contains a dimension filter and a table calculation. Tableau applies the dimension filter before executing the table calculation. To invert the order of these operations, use a **FIXED** level of detail expression instead of a table calculation.

Here are the steps for building this view.

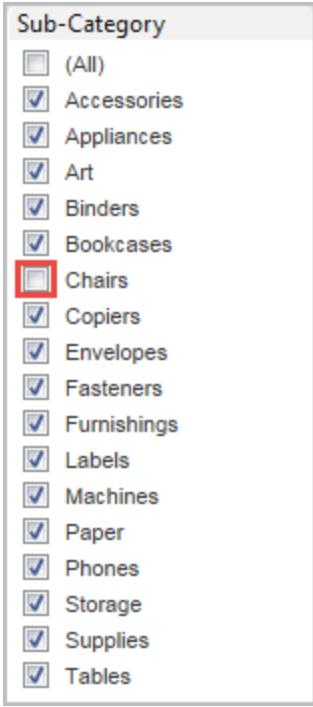
1. In a new worksheet, drag **Sales** to **Columns**.
2. Drag **Sub-Category** to **Rows**.
3. Right-click **SUM(Sales)** on **Columns** and select a quick table calculation – **Percent of Total**.
4. Click the **Sort Descending** button () on the toolbar to sort the categories from most to least.
5. Click the **Show Mark Labels** button () on the toolbar to display measure values in the view.

Your view now looks like this:



Note the percentages for the first few items: 14.37%, 14.30%, etc.

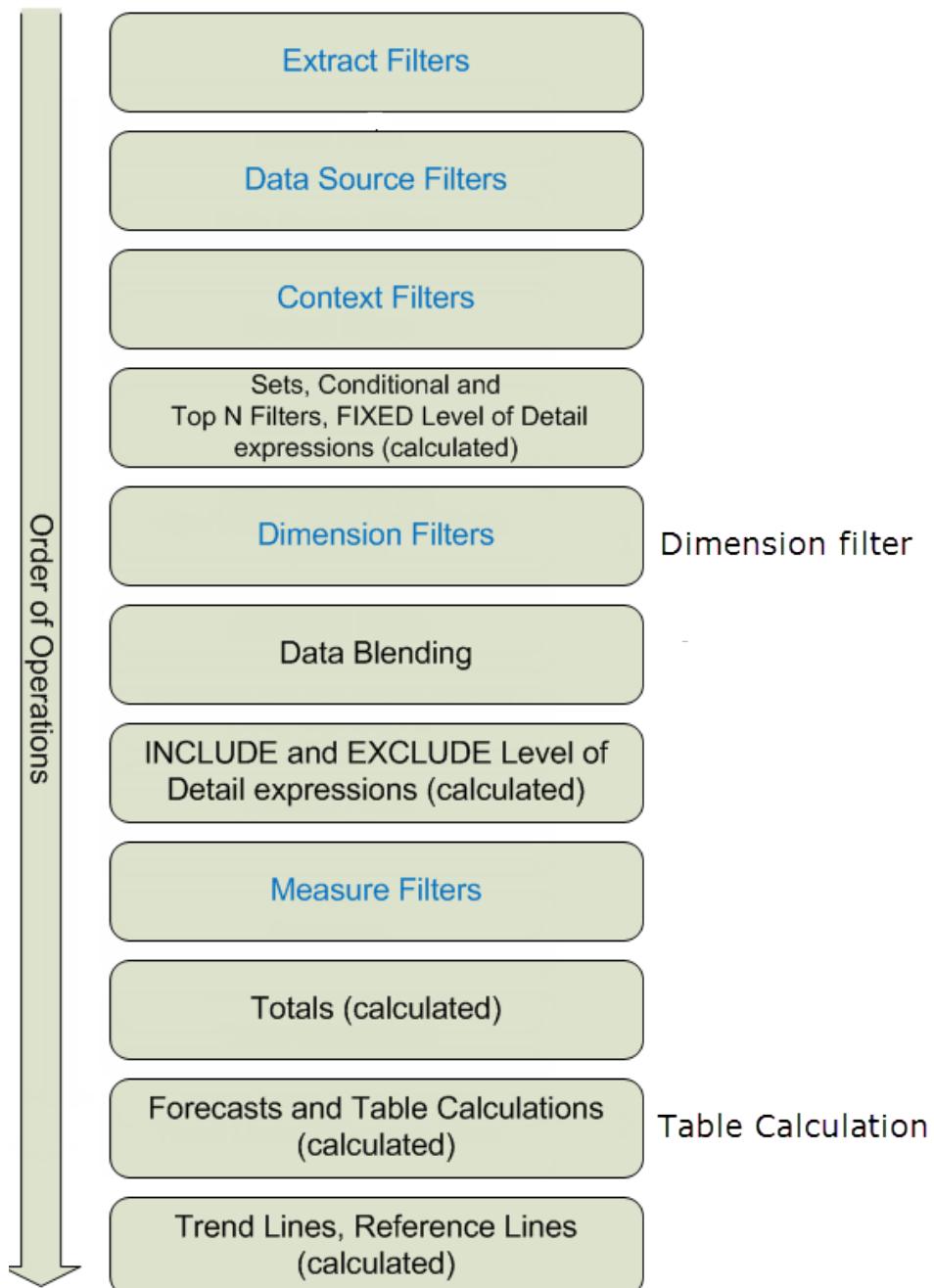
6. Right-click **Sub-Category** on **Rows** and choose **Show Filter**.
7. Clear the check mark for **Chairs** in the filter.



In the view, the percentages are now different—the highest percentage is now over 16%. In some cases, this may be just the result that you want (that is, for percentages to be

recalculated as you work with the quick filter). But in other cases you may want the percentages to hold steady even as you filter items in or out. That's what we want in this case.

In the order of operations, a dimension filter is applied before a table calculation:



To have Tableau calculate the percentages before it acts on the quick filter, you create a **FIXED** level of detail expression, and then use that instead of the table calculation.

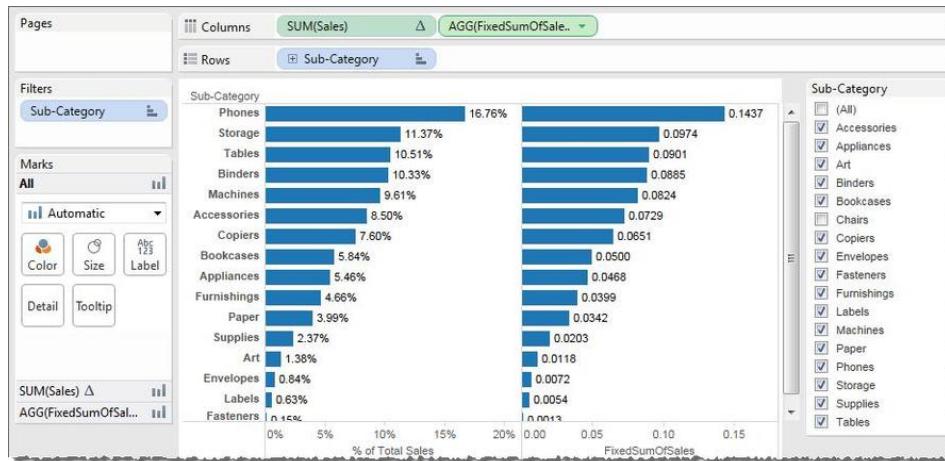
FIXED level of detail expressions compute a value using the specified dimensions, without reference to the dimensions in the view. In this case you'll use it to establish percentages for the various sub-categories—percentages that won't be affected by your general dimension filter. Why? Because **FIXED** level of detail expressions are computed before dimension filters are applied.

For details, see [Level of Detail Expressions on page 899](#).

- The **FIXED** level of detail expression must divide the sum of **Sales** (for a particular measure value) by the total sum of **Sales** for the view. Because the numerator is aggregated, the denominator must be as well, so the expression you write is:

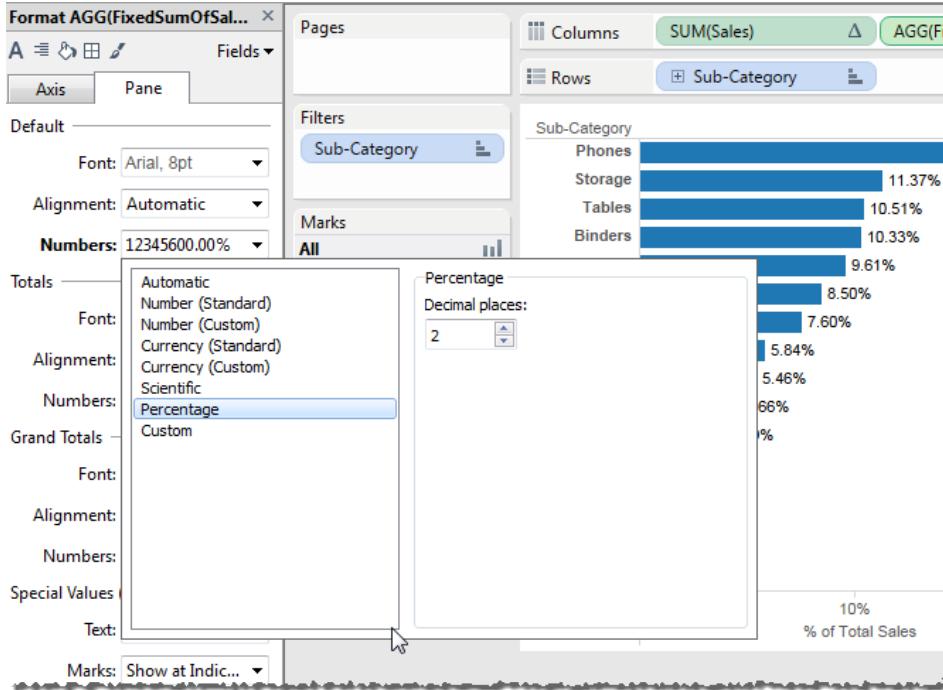
```
SUM([Sales]) / SUM({FIXED : SUM([Sales])})
```

- Save that expression as **FixedSumOfSales** and then drag it from the **Data pane** to **Columns**, dropping it to the right of the existing **SUM(Sales)** field that uses the table calculation. (Keep them both in the view for comparison.) Here is what your view now looks like:

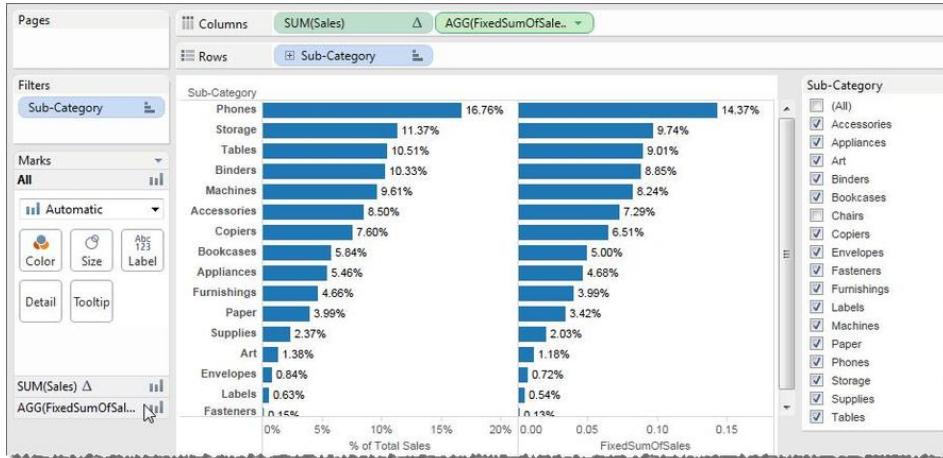


The percentage numbers in the chart on the right are now consistent, regardless of which fields you select or don't select with the quick filter. All that remains is to format the values for **FixedSumOfSales** so that they show as percentages.

- Right-click **FixedSumOfSales** on **Columns** and choose **Format**. In the **Format** pane, choose **Numbers** and then **Percentage**:



This gives you the final view:



As you select or clear items in the **Sub-Category** quick filter, the percentages in the bar chart on the left change, but the percentages in the bar chart on the right do not.

Understanding Data Fields

The data in all data sources are categorized into fields such as Customer, Sales, Profit, Temperature, etc. These fields are made from the columns in your data source. When you connect to a data source with Tableau, the fields are displayed along the left side of the workbook in the Data pane. The fields are what you will use to build views of your data. Each

field is automatically assigned a data type (such as integer, string, and date) and a pair of data roles.

Understanding the Data Pane

All data sources contain fields. In Tableau, these fields appear in the Data pane. For cube (multidimensional) data sources, the fields are determined by the dimensions and measures of a cube. In Tableau, cube data sources are supported only in Windows. For relational data sources, the fields are determined by the columns of a table or view. Each field contains a unique attribute of the data such as customer name, sales total, product type, and so on. For example, some of the fields of an Excel worksheet are shown below.

	A	B	C	D	E	F	G	H
1	Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Customer Name	Segment
2		1 CA-2013-152156	11/9/2014	11/12/2014	Second Class	CG-12520	Claire Gute	Consumer
3		2 CA-2013-152156	11/9/2014	11/12/2014	Second Class	CG-12520	Claire Gute	Consumer
4		3 CA-2013-138688	6/13/2014	6/17/2014	Second Class	DV-13045	Darrin Van H	Corporate
5		4 US-2012-108966	10/11/2013	10/18/2013	Standard Class	SO-20335	Sean O'Donn	Consumer
6		5 US-2012-108966	10/11/2013	10/18/2013	Standard Class	SO-20335	Sean O'Donn	Consumer
7		6 CA-2011-115812	6/9/2012	6/14/2012	Standard Class	BH-11710	Brosina Hoff	Consumer
8		7 CA-2011-115812	6/9/2012	6/14/2012	Standard Class	BH-11710	Brosina Hoff	Consumer
9		8 CA-2011-115812	6/9/2012	6/14/2012	Standard Class	BH-11710	Brosina Hoff	Consumer
10		9 CA-2011-115812	6/9/2012	6/14/2012	Standard Class	BH-11710	Brosina Hoff	Consumer
11		10 CA-2011-115812	6/9/2012	6/14/2012	Standard Class	BH-11710	Brosina Hoff	Consumer
12		11 CA-2011-115812	6/9/2012	6/14/2012	Standard Class	BH-11710	Brosina Hoff	Consumer
13		12 CA-2011-115812	6/9/2012	6/14/2012	Standard Class	BH-11710	Brosina Hoff	Consumer
14		13 CA-2014-114412	4/16/2015	4/21/2015	Standard Class	AA-10480	Andrew Allen	Consumer
15		14 CA-2013-161389	12/6/2014	12/11/2014	Standard Class	IM-15070	Irene Maddo	Consumer
16		15 US-2012-118983	11/22/2013	11/26/2013	Standard Class	HP-14815	Harold Pawl	Home Office
17		16 US-2012-118983	11/22/2013	11/26/2013	Standard Class	HP-14815	Harold Pawl	Home Office
18		17 CA-2011-105893	11/11/2012	11/18/2012	Standard Class	PK-19075	Pete Kriz	Consumer
19		18 CA-2011-167164	5/13/2012	5/15/2012	Second Class	AG-10270	Alejandro Gr	Consumer
20		19 CA-2011-143336	8/27/2012	9/1/2012	Second Class	ZD-21925	Zuschuss Doi	Consumer
21		20 CA-2011-143336	8/27/2012	9/1/2012	Second Class	ZD-21925	Zuschuss Doi	Consumer
22		21 CA-2011-143336	8/27/2012	9/1/2012	Second Class	ZD-21925	Zuschuss Doi	Consumer

After you connect to your data and set up the data source with Tableau, the data source fields appear on the left side of the workbook in the Data pane.

The screenshot shows the Tableau Data pane with the following structure:

- Data** tab selected.
- Analytics** tab available.
- Dimensions** section:
 - Category
 - City
 - Country
 - Customer ID
 - Customer Name
 - Order Date
 - Order ID
 - Postal Code
 - Product ID
- Measures** section:
 - Discount
 - Profit
 - Quantity
 - Sales

The Data pane organizes fields into different areas:

- **Dimensions** – Fields that typically hold discrete qualitative data. Examples of dimensions include dates, customer names, and customer segments.
- **Measures** – Fields that typically hold numerical data that can be aggregated. Examples of measures include sales, profit, number of employees, temperature, frequency, and pressure.
- **Sets** – An additional area that stores custom fields based on existing dimensions and criteria that you specify. Named sets from an MS Analysis Services server or from a Teradata OLAP connector also appear in Tableau in this area of the Data pane. You can interact with these named sets in the same way you interact with other custom sets in Tableau.
- **Parameters** – An additional area that stores parameters that you have created. Parameters are dynamic variables that can be used as placeholders in formulas.

Note: For cube data sources, fields are explicitly defined as dimensions or measures when the database is created. For relational data sources, Tableau automatically organizes the fields. By default, fields containing text, date or boolean values are dimensions, while fields containing numerical values are measures.

The Data pane for an Excel worksheet (a relational database) is shown below. The **Discount** and **Sales** fields contain numbers and appear as measures, in the lower part of the Data pane. The **Segment** field contains text and the **Order Date** field contains dates. These fields appear as dimensions in the upper part of the Data pane.

The screenshot shows the Data pane in Excel with the following structure:

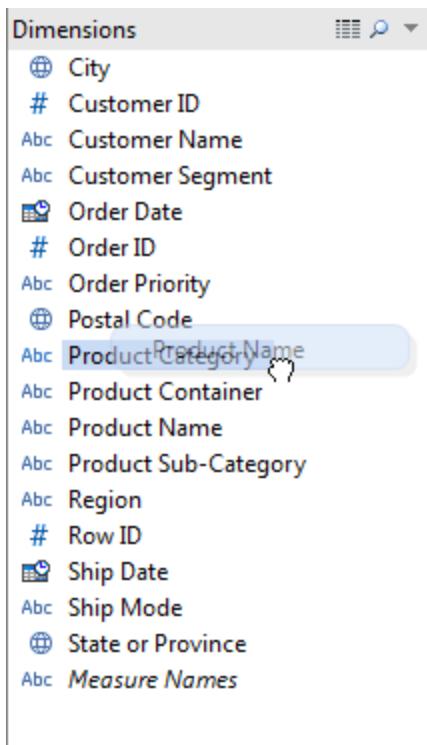
- Dimensions** section (top half):
 - Category
 - City
 - Country
 - Customer ID
 - Customer Name
 - Order Date** (highlighted with a green oval)
 - Order ID
 - Postal Code
 - Product ID
 - Product Name
 - Region
 - Row ID
 - Segment** (highlighted with a green oval)
 - Ship Date
 - Ship Mode
 - State
 - Sub-Category
 - Measure Names
- Measures** section (bottom half):
 - Discount** (highlighted with a green oval)
 - Profit
 - Quantity
 - Sales** (highlighted with a green oval)
 - Latitude (generated)
 - Longitude (generated)
 - Number of Records
 - Measure Values

Note: By default the field names defined in the data source are displayed in the Data pane. You can rename fields as well as member names.

Hierarchies (For Relational Databases)

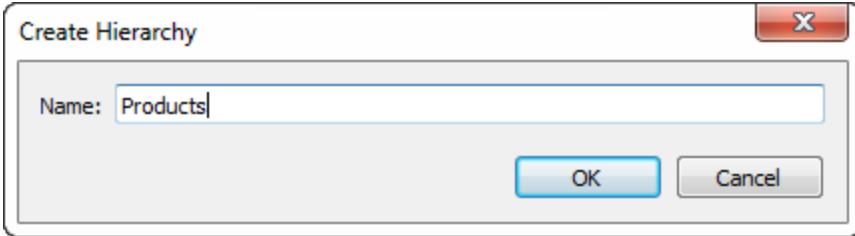
Unlike cube (multidimensional) data sources, relational data sources don't have built-in hierarchies. However, often relational data sources have related dimensions that have an inherent hierarchy. For example, a data source may have fields for Country, State, and City. These fields could be grouped into a hierarchy called Location. You can assemble relational hierarchies by dragging and dropping in the Data pane.

1. Drag a field in the Data pane and drop it directly on top of another field.



Note: When you want to create a hierarchy from a field inside a folder, right-click (control-click on a Mac) the field and then select **Create Hierarchy**.

2. When prompted, enter a name for the relational hierarchy and click **OK**.



3. Drag any additional fields into the hierarchy. You can also re-order fields in the hierarchy by dragging them to a new position.

A screenshot of a 'Dimensions' pane. On the left is a tree view of dimensions. A plus sign indicates expandable categories. One such category, 'Products', is expanded, showing its sub-fields: 'Product Name', 'Product Sub-Category', 'Product Category', and 'Product Name'. Other dimensions listed include City, Customer ID, Customer Name, Customer Segment, Order Date, Order ID, Order Priority, Postal Code, Product Container, Region, Row ID, Ship Date, Ship Mode, State or Province, and Measure Names.

Hierarchies support single-click navigation up and down the levels. When you use the fields in the view, a plus button displays on the field so you can drill down and up in the hierarchy.

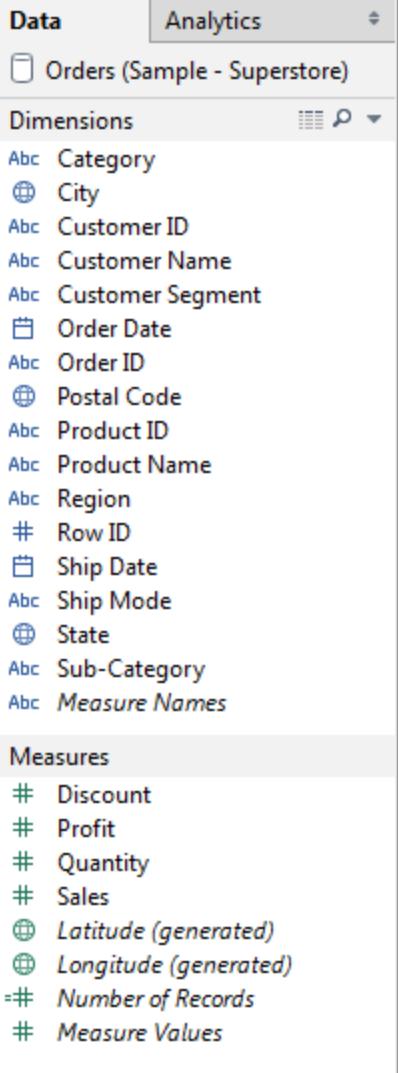
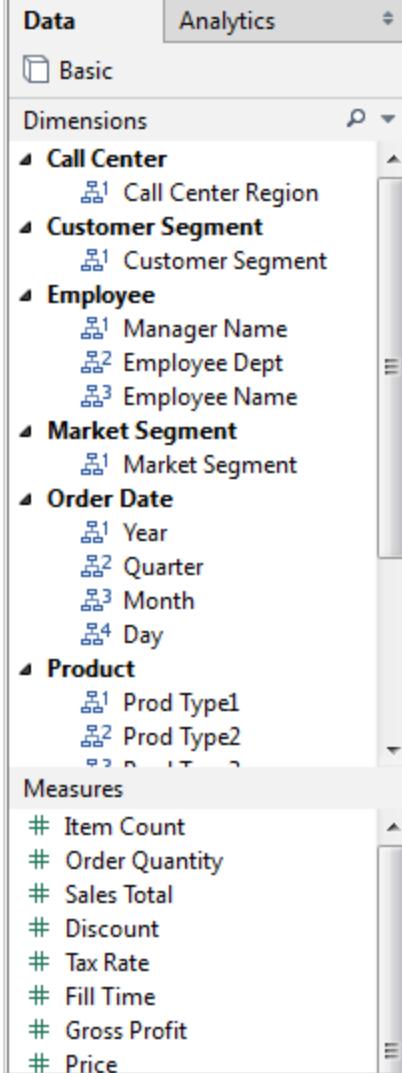
A screenshot of a 'Data' pane. It shows a 'Columns' section with 'Product Category' and 'Product Sub-Category' selected. Below it is a 'Rows' section with 'SUM(Sales)' selected. The 'Product Sub-Category' field has a plus sign next to it, indicating it is a hierarchical field.

Relational and Cube Data

The Data pane for a relational and cube data source are shown below. Note that the panes look essentially the same for both data sources in that the fields are organized into dimensions and measures. However, the cube data source contains hierarchies for dimensions. For example,

notice that the Employee dimension in the cube Data pane contains hierarchical members such as Manager Name and Employee Dept.

Note: In Tableau, cube (multidimensional) data sources are supported only in Windows.

Relational	Cube
	

You can expand or collapse the various hierarchies in both relational and cube Data panes by clicking the arrow. You can hide the Data pane all together by clicking the minimize button  in the upper-right corner of the Data pane.

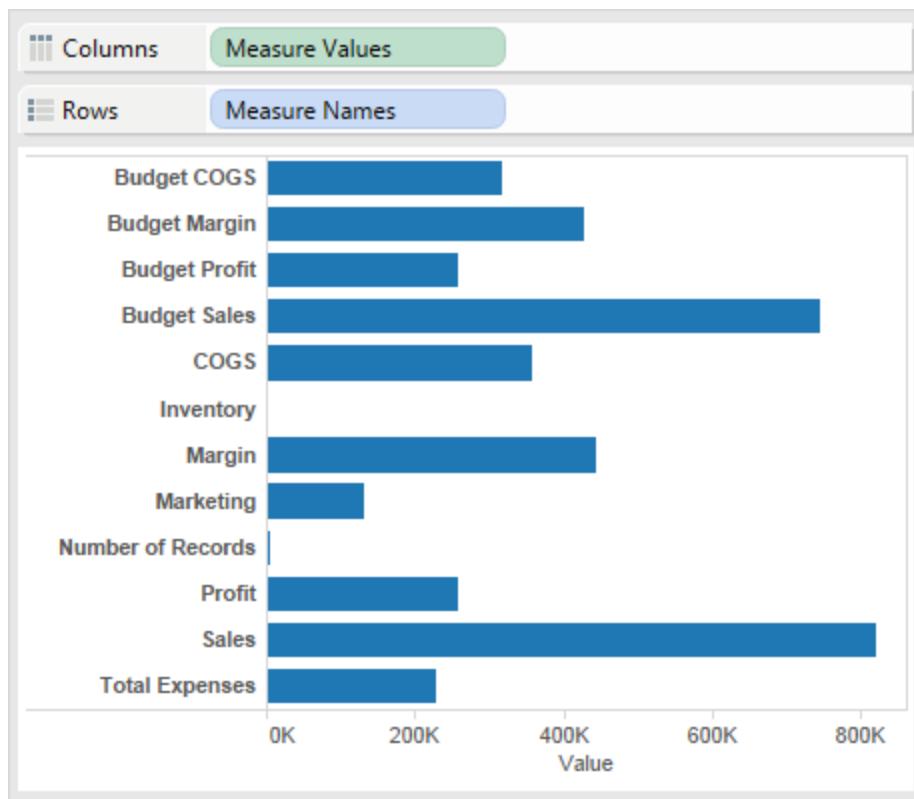
Measure Values and Measure Names

The Data pane contains a few fields that are not part of your data, two of which are **Measure Values** and **Measure Names**.

The **Measure Values** field always appears at the bottom of the **Measures** area of the Data pane and contains all the measures of your data collected into one field.

The **Measure Names** field always appears at the bottom of the **Dimensions** area of the Data pane and contains all the names of the measures collected into a single dimension.

Tableau automatically creates these fields so that you can build certain types of views that involve multiple measures. In particular, use these fields if you want to display multiple measures in the same pane simultaneously. As shown below, creating a view with **Measure Values** and **Measure Names** is one way to display all the data in your data source.



Where do these fields come from?

When you connect to a data source, Tableau automatically creates fields to contain all of the measure names and measure values.

For example, below is a simple data source that contains three measures. From this data source, Tableau creates the fields shown on the right.

The diagram illustrates the relationship between a data source and two cards. On the left is a data table with columns: Date, Region, Sales, Profit, and Discount. A curved arrow points from this table to a blue-bordered card labeled "Measure Names". This card contains three items: "Sum of Sales", "Sum of Profit", and "Sum of Discount". Another curved arrow points from the "Measure Names" card to a green-bordered card labeled "Measure Values". This card displays a table with three columns: Sum of Sales, Sum of Profit, and Sum of Discount. The data rows correspond to the measures listed in the "Measure Names" card.

	A	B	C	D	E	F
1	Date	Region	Sales	Profit	Discount	
2	1/1/2009	East	\$100	\$50	0%	
3	1/2/2009	West	\$300	\$100	10%	
4	1/3/2009	Central	\$500	\$200	30%	
5	1/4/2009	East	\$400	\$160	40%	
6	1/5/2009	South	\$600	\$500	0%	
7	1/6/2009	West	\$800	\$750	0%	
8	1/7/2009	West	\$400	\$250	0%	
9	1/8/2009	Central	\$100	\$65	20%	
10	1/9/2009	East	\$300	\$254	50%	
11	1/10/2009	South	\$200	\$89	75%	
12	1/11/2009	South	\$100	\$40	30%	
13						

Measure Names

Sum of Sales
 Sum of Profit
 Sum of Discount

Measure Values

Sum of Sales	Sum of Profit	Sum of Discount
\$100	\$50	0%
\$300	\$100	10%
\$500	\$200	30%
\$400	\$160	40%
\$600	\$500	0%
\$800	\$750	0%
\$400	\$250	0%
\$100	\$65	20%
\$300	\$254	50%
\$200	\$89	75%
\$100	\$40	30%

Common uses

Create a details table

Use a details table to show all of the measure values for specified categories.

For example, the text table below shows financial measures for each product type. When you use the **Measure Names** and **Measure Values** fields, the Measure Values card automatically opens below the Marks card. You can use this card to quickly add and remove measures from the view.

Drag measures to or from the Measure Values card to add them to or remove them from the view.

The screenshot shows the Tableau Data pane with the following structure:

- Product Type**: Columns
- Measure Names**: Rows
- Marks**: A dropdown menu showing options for Color, Size, and Text.
- Measure Values**: A list of calculated fields:
 - SUM(Budget COGS)
 - SUM(Budget Margin)
 - SUM(Budget Profit)
 - SUM(Budget Sales)
 - SUM(COGS)
 - AVG(Inventory)
 - SUM(Margin)
 - SUM(Marketing)
 - SUM(Number of Record..)
 - SUM(Profit)
 - SUM(Sales)
 - SUM(Total Expenses)

	Product Type			
	Coffee	Espresso	Herbal Tea	Tea
Budget COGS	\$96,880	\$90,800	\$73,060	\$57,140
Budget Margin	\$131,740	\$122,680	\$97,400	\$76,460
Budget Profit	\$83,880	\$71,900	\$58,720	\$44,260
Budget Sales	\$228,620	\$213,480	\$170,460	\$133,600
COGS	\$90,696	\$97,000	\$92,810	\$78,166
Inventory	761	672	785	792
Margin	\$121,572	\$121,172	\$110,000	\$90,294
Marketing	\$33,366	\$38,216	\$34,154	\$26,738
Number of Records	1,056	1,176	1,056	960
Profit	\$74,683	\$68,620	\$63,254	\$52,986
Sales	\$216,828	\$222,996	\$207,214	\$172,773
Total Expenses	\$60,025	\$64,603	\$57,814	\$47,220

Blend measures on a single axis

To learn how to blend measures on a single axis, see [Blended Axes](#) on page 606.

Number of Records

In addition to the Measure Names and Measure Values fields, the Data pane contains a **Number of Records** field that is also not part of the underlying data. This field represents the

number of rows in the data source. It is useful when you are working with a data source that is primarily categorical resulting in very few measures.

[Latitude and Longitude \(generated\)](#)

If you have defined any fields to be geographic fields, that is, they can be used with maps, Tableau automatically geocodes your data and includes **Latitude (generated)** and **Longitude (generated)** fields. You can use these fields to overlay your data on live maps.

Data Pane Features and Functions

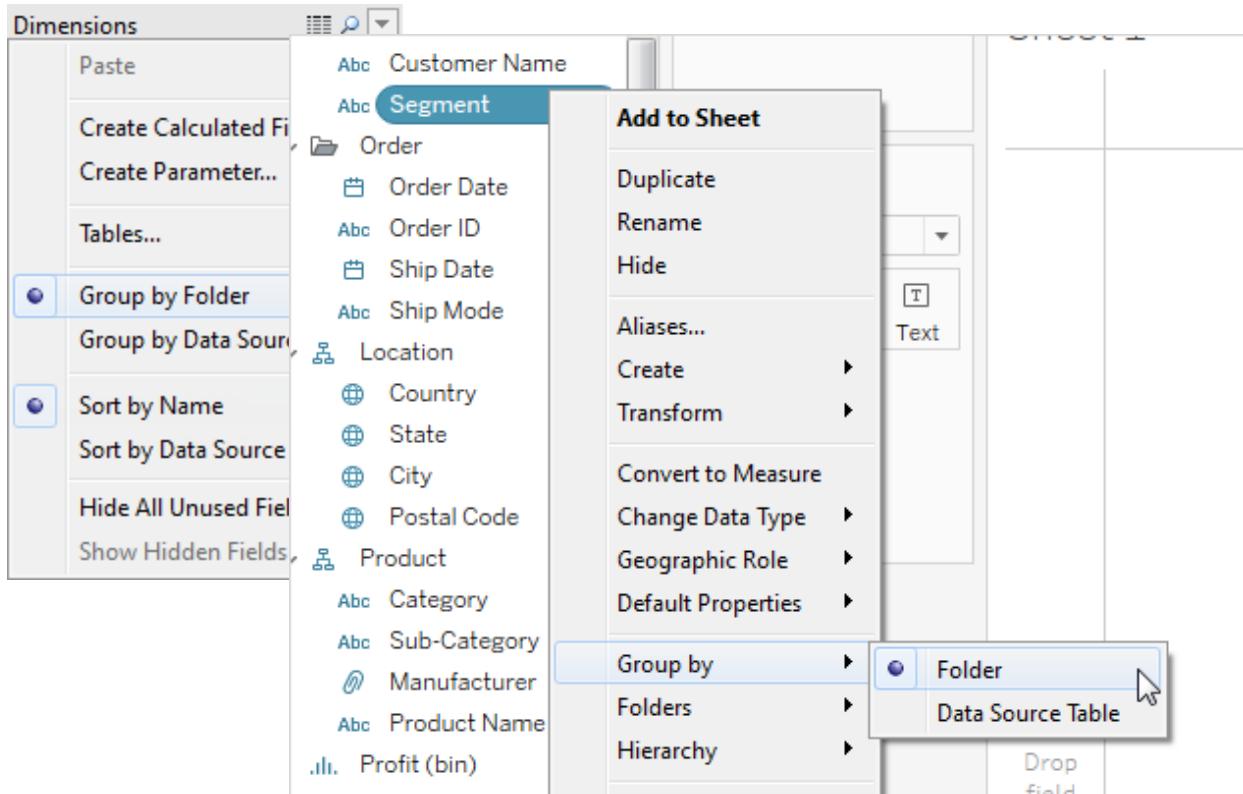
The Data pane has many features and functions to help you organize your data fields, find specific fields, and hide others.

[Organize the Data Pane](#)

You can reorganize the items in the Data pane from its default layout using folders or through sorting.

[Folders](#)

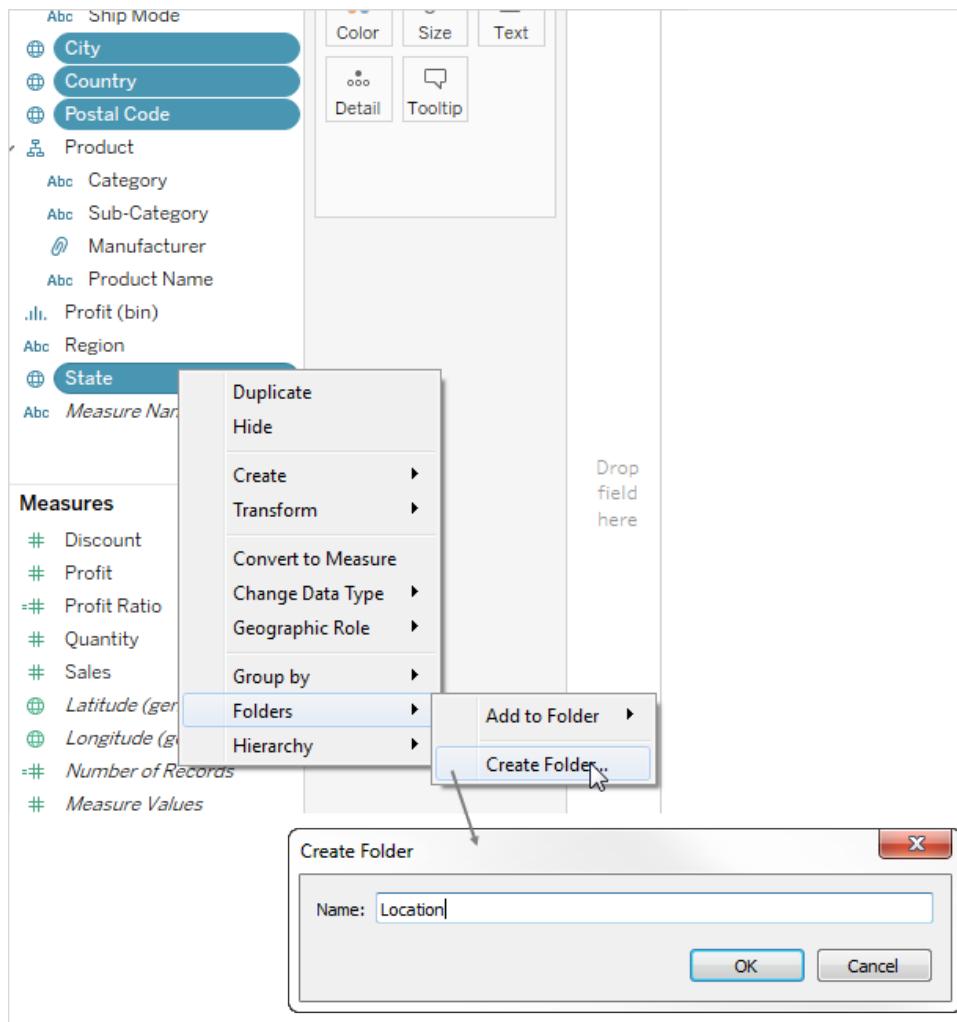
To make data sources with many fields easier to work with, you can organize the Data pane items into folders. Items like fields, parameters, and sets can be grouped into folders. The **Group by Folder** option can be accessed from the Data pane menu. Alternatively, you can also access this option from the field context menu.



When you connect to a single table in your data source, grouping by folder is enabled by default. When you connect to a data source with multiple tables, grouping by table is enabled. When the **Group by Data Source Table** option is selected, the dimensions and measures are grouped according to the database table they belong to. This is especially useful when you have several joined tables.

To group fields into folders:

1. In the Data pane, select the fields you want to group together or right-click (control-click on Mac) an empty area in the Data pane.
2. Select **Create Folder**.
3. When prompted, name the new folder.



After you create a folder structure, you can drag fields from one folder to another or duplicate a field you want to have available in more than one folder.

To add a field to a folder:

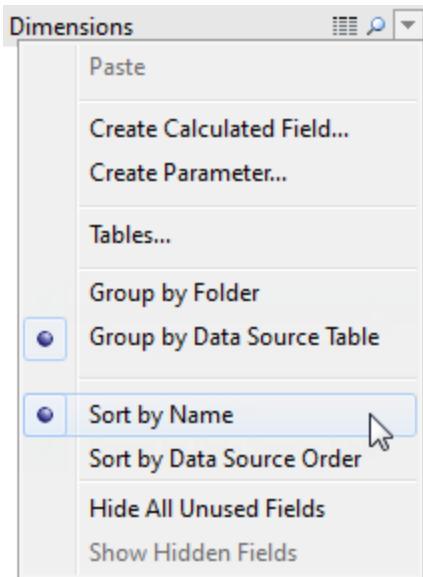
- Drag a field on top of the folder name to add the field to the folder. If the folder is expanded, you can drag the field into the general area of the folder.

The screenshot shows the Tableau Data pane titled "Dimensions". It lists various dimensions and measures organized into folders:

- Customer**:
 - Customer Name
 - Segment
- Location**:
 - Region
 - City
 - Country
 - Postal Code
 - State
- Order**:
 - Order Date
 - Order ID
 - Ship Date
 - Ship Mode
- Product**:
 - Category
 - Sub-Category
 - Manufacturer
 - Product Name
 - Profit (bin)
 - Region
 - Measure Names

Sort

When organizing the Data pane with or without folders, you can have Tableau sort the items. These **Sort by** options are also located in the Data pane menu.



You can sort by one of the following options:

- **Sort by Name** – lists the dimensions and measures in alphabetical order according to their field aliases.
- **Sort by Data Source Order** – lists the dimensions and measures in the order they are listed in the underlying data source.

Find Fields

You can search for fields, folders and hierarchies in the Data pane. When there are many fields in your data source it can be difficult to find a specific one like “Date” or “Customer” or “Profit,” or to find all fields that end in “xyz.” To find an item, do the following:

Click the **Find Field** icon at the top of the Data pane (**Ctrl + F** in Windows, **Command-F** on a Mac) and enter the name of the item you want to search for. You can also enter a string of characters, to search for all item names that contain that string.

As you type in the search box, search filters the contents of the Data pane to show all fields, folders or hierarchies that contain the typed string.

The screenshot shows the Tableau Data pane. At the top, there are tabs for 'Data' and 'Analytics'. Below the tabs, it says 'Sample - Superstore'. A search bar contains the text 'Cate'. Under the search bar, there is a section titled 'Dimensions' with a search icon. A dropdown menu is open under 'Dimensions', showing the following hierarchy: 'Product' (selected), 'Category', 'Sub-Category', and 'Sub-Category & Category (C...)'. There are also icons for 'ABC' and 'ABC...' next to some of the dimension names.

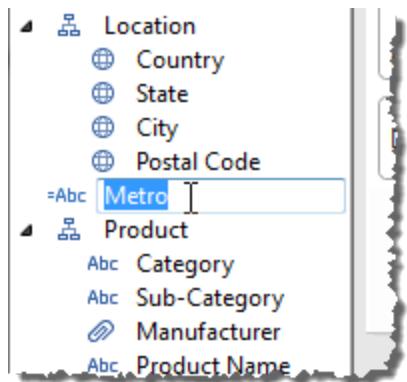
Search remains open until you click the **Find Field** icon or press **Ctrl + F** again.

Rename Fields

You can rename fields in the Data pane. For example, you could rename a field named **Customer Segment** in the data source to be **Business Segment** in Tableau. You can also rename user-created fields. Renaming a field does not change the name of the field in the underlying data source, rather it is given a special name that appears only in Tableau workbooks. The changed field name is saved with the workbook as well as when you export the data source. You can rename any type of field: dimensions, measures, sets, or parameters.

Rename a Field

1. Click field name in the Data pane and hold the mouse button down until the field name is shown in an edit box:



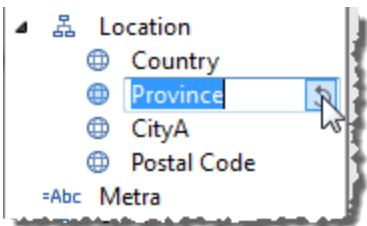
You can also press F2 or Ctrl + Enter to make the field name editable.

2. Type the new name for the field and press Enter.

The field displays with the new name in the Data pane.

Revert to the Default Field Name

If the field you renamed was from the original data source you can click the field name in the Data pane and hold the mouse button down until the field name is shown in the box. At the right of the edit field is a small circular arrow that you can click to restore the original data source field name:



To revert the names of multiple fields that were in the original data source, select them all, right-click, and then choose **Reset Names**.

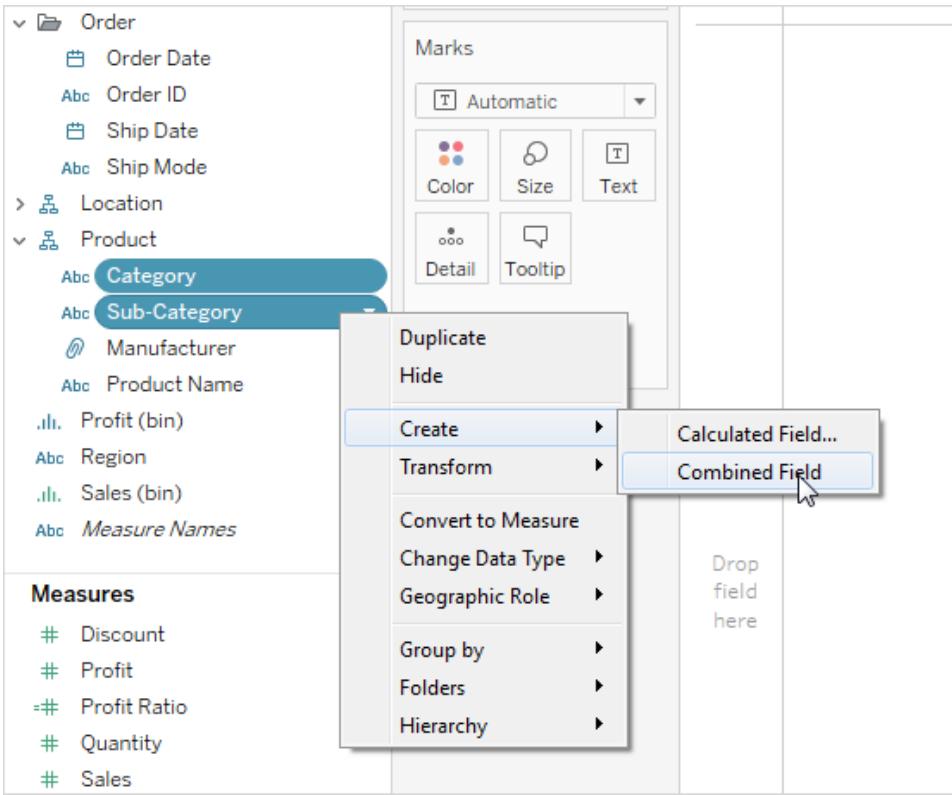
Combine Fields

Combine fields to create a cross product of members from different dimensions. You can combine dimensions if you want to encode a data view using multiple dimensions.

To combine the fields, select multiple dimensions in the Data pane and then right-click (control-click on a Mac) the fields and select **Create > Combined Field**.

Note: For cube (multidimensional) data sources, you must select levels from different hierarchies. In Tableau, cube data sources are supported only in Windows.

For example, the selections shown below will produce a new field that consists of the Category and Sub-Category dimensions.



The two dimensions are combined into a new dimension. The name of the field is automatically created from the names of the original fields. Right-click (control-click on a Mac) the new field and select **Rename** to change the name.

When you use the new field in a view, a header is created for each combination of the two original dimensions. For example, the view below shows the members of the combined Category and Sub-Category fields.

The screenshot shows the Tableau Data pane with the following structure:

- Pages:** Shows "Region" selected under "Columns" and "Sub-Category & Categ.." selected under "Rows".
- Filters:** An empty section.
- Marks:**
 - Automatic (selected)
 - Color
 - Size
 - Text
 - Detail
 - Tooltip
 - SUM(Sales) (selected)
- Sheet 2:** A table titled "Region" with the following data:

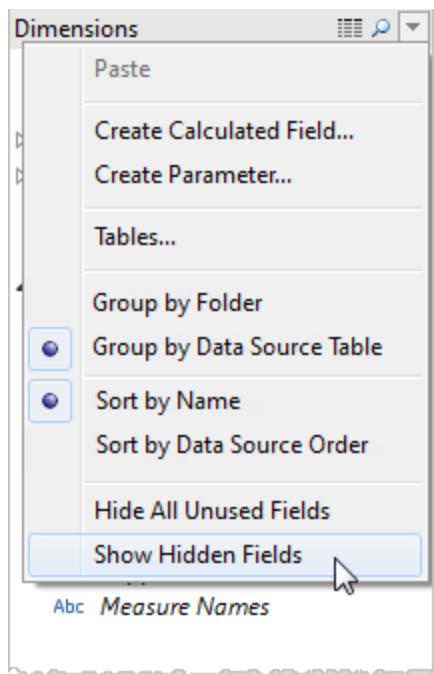
Sub-Category & Category ..	Central	East	South
Accessories, Technology	\$33,956	\$45,033	\$27,277
Appliances, Office Supplies	\$23,582	\$34,188	\$19,525
Art, Office Supplies	\$5,765	\$7,486	\$4,656
Binders, Office Supplies	\$56,923	\$53,498	\$37,030
Bookcases, Furniture	\$24,157	\$43,819	\$10,899
Chairs, Furniture	\$85,231	\$96,261	\$45,176
Copiers, Technology	\$37,260	\$53,219	\$9,300
Envelopes, Office Supplies	\$4,637	\$4,376	\$3,346
Fasteners, Office Supplies	\$778	\$820	\$503
Furnishings, Furniture	\$15,254	\$29,071	\$17,307
Labels, Office Supplies	\$2,451	\$2,603	\$2,353

Note: For cube data sources, to choose to display the fully qualified name, right-click (control-click on a Mac) the combined field in the Data pane and select **Qualify Member Names**.

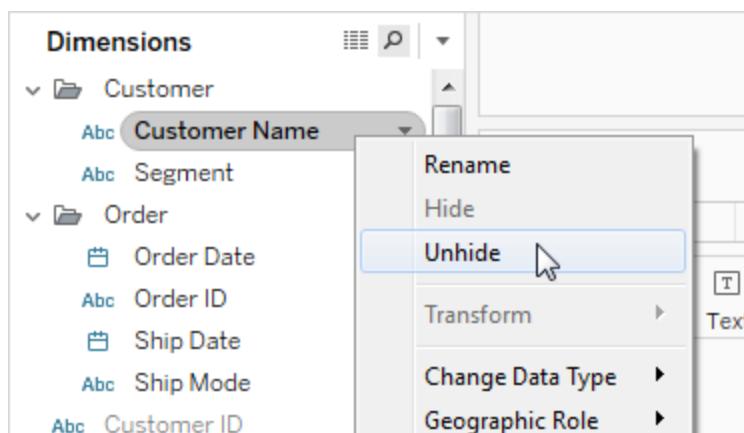
Hide or Unhide Fields

You can selectively hide or show fields in the Data pane. To hide a field, right-click (control-click on a Mac) the field and select **Hide**.

When you want to change your fields from hidden to visible, select **Show Hidden Fields** on the Data pane menu.



The hidden fields are then shown in gray in the Data pane. You can then select one or more hidden fields, right-click (control-click on a Mac) and select **Unhide**.



Select **Hide All Unused Fields** on the Data pane menu to quickly hide all of the fields that are not being used in the workbook.

Note: Hiding fields can be a good way to decrease the size of a data extract file because hidden fields are automatically excluded from the extract.

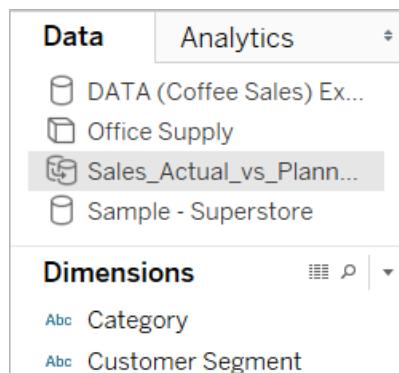
Add Calculated Fields to the Data Pane

You can create calculated fields that appear in the Data pane. These new computed fields can be used like any other field. Select **Create Calculated Field** on the Data pane menu.

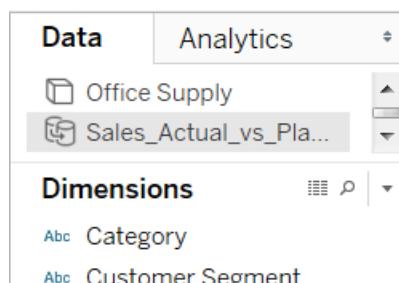
Alternatively, select **Analysis > Create Calculated Field**.

Navigating Data Sources in the Data Pane

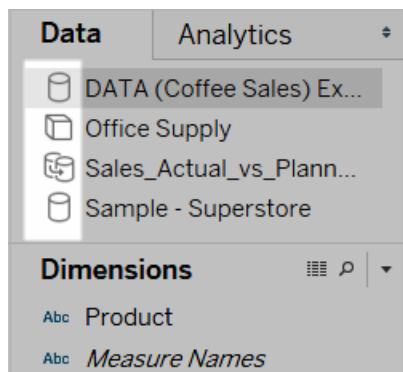
The top of the Data pane lists all of the data sources in a given workbook. Simply select the data source you want to use and the Data pane updates to show the corresponding fields in that data source.



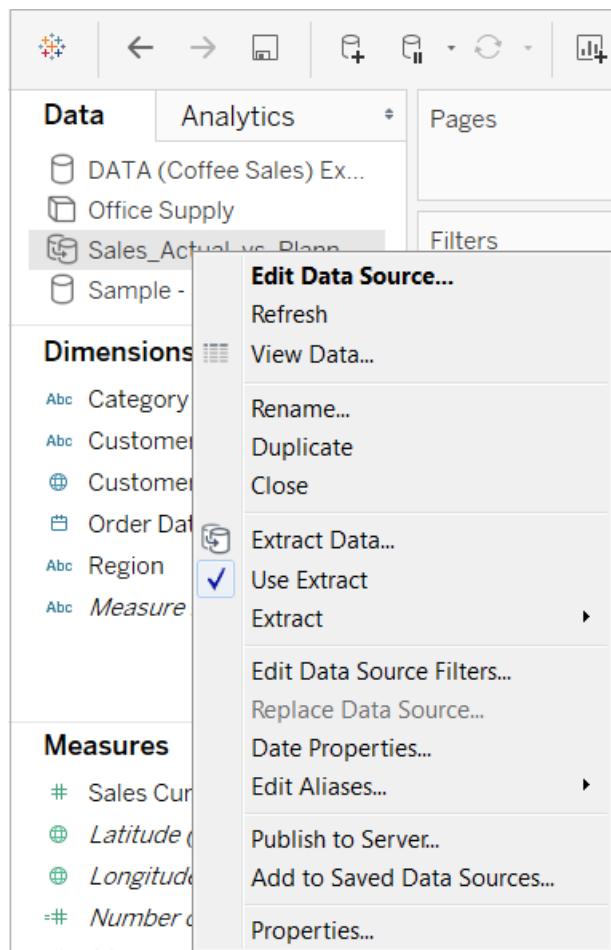
You can resize the data source list area in the Data pane to save space. When you resize to a limited vertical height, you can scroll down to a data source.



Each data source has an icon to indicate its type. For example, the icon can indicate whether the data source is relational, cube (multidimensional), or an extract. In Tableau, cube data sources are supported only in Windows.



You can right-click (control-click on a Mac) the data source to access the commands that are on the Data menu. For example, you can right-click (control-click on a Mac) a data source and rename, export, or close it.



Editing Field Properties

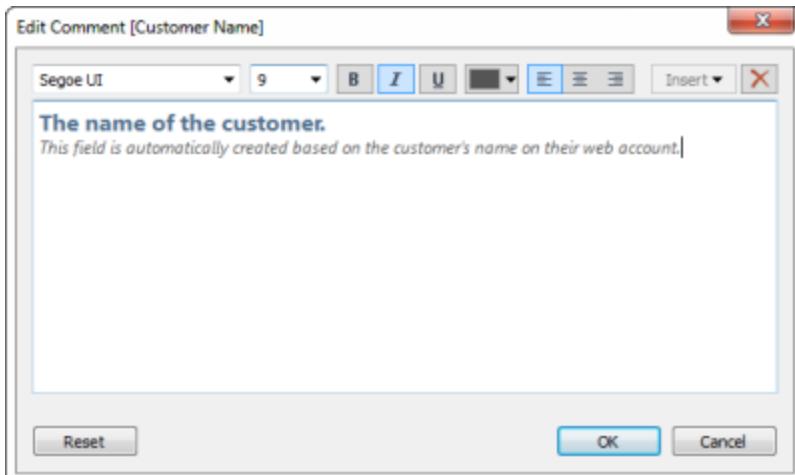
When you drag fields to shelves, the data is represented as marks in the view. You can specify settings for how the marks from each field will be displayed by setting mark properties. For example, when you place a dimension on the color shelf, the marks will be colored by the values within that dimension. You can set the Color property so that anytime you use that dimension on the color shelf your chosen colors are used. Using field properties, you can set the aliases, colors, shapes, default aggregation, and so on.

Comments

Fields can have comments that describe them. The comments display in a tooltip in the Data pane and in the Calculated fields dialog box. Field comments are a good way to give more context to the data in your data source. Comments are especially useful when you are building a workbook for others to use.

Adding a Comment to a Field

1. Right-click (control-click on a Mac) a field in the Data pane and select **Default Properties > Comment**.
2. Write a comment in the subsequent dialog box. Comments support rich text formatting that will be represented in the tooltip.



3. When finished, click **OK**.

Now when you hover the cursor over the field in the Data pane, you see the comment.

The screenshot shows the Power BI Data pane. On the left, under the 'Dimensions' section, there is a tree view with 'Customer' expanded. Under 'Customer', 'Customer Name' is selected and highlighted with a blue background. A tooltip appears over this field, containing the text 'The name of the customer.' and 'This field is automatically created based on the customer's name on their web account.' Below the tooltip, there are three buttons: 'Automatic', 'Color', 'Size', and 'Text'. The 'Text' button is currently selected.

Aliases

Aliases are alternate names for specific values within a dimension. For example, you may want to assign aliases for the values of the “Customer Segment” dimension. Perhaps you want the “Consumer” members of this field to display as “Home Consumer” in all views.

Aliases can be created for the members of most dimensions in the Data pane. You cannot, however, define aliases for continuous dimensions and dates, and they do not apply to measures. The method for creating aliases depends on the type of data source you are using.

Aliases with a Relational Data Source

To create alias for individual dimension values in a relational data source, right-click (control-click on Mac) a field name and select **Aliases**

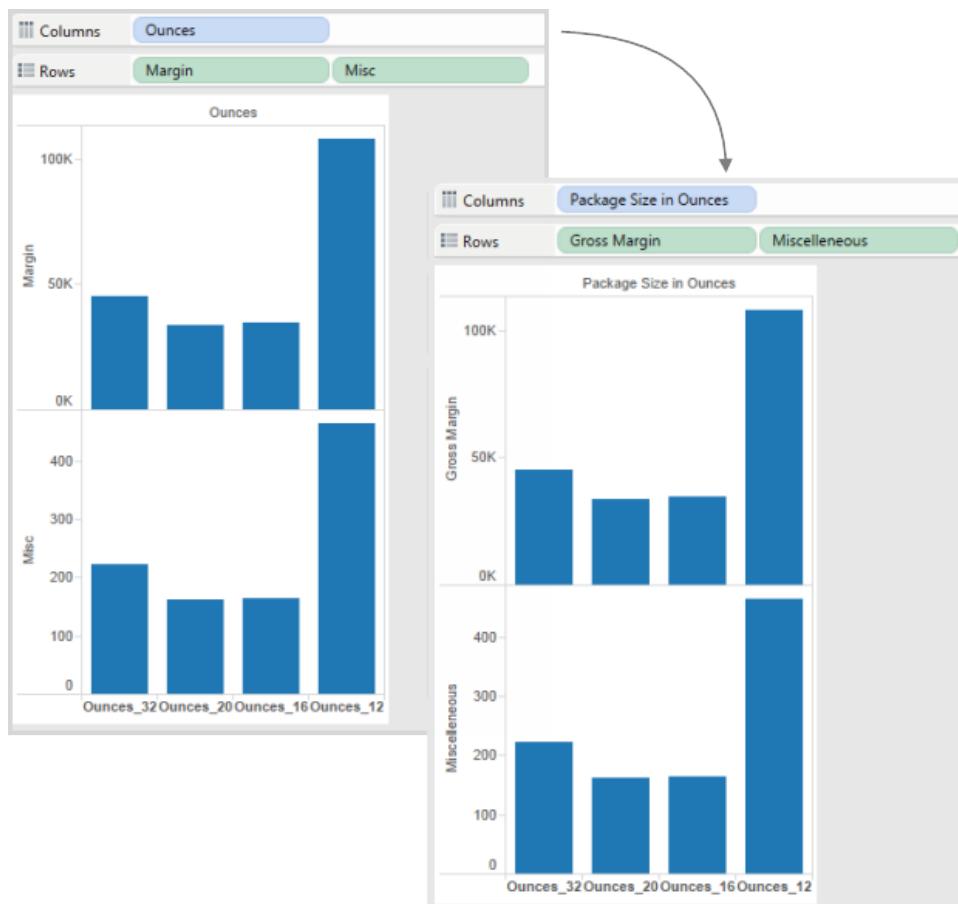
The screenshot shows the Power BI Data pane. On the left, under the 'Dimensions' section, there is a tree view with 'Customer' expanded. Under 'Customer', 'Segment' is selected and highlighted with a blue background. A context menu is open over the 'Segment' field, with the 'Aliases...' option highlighted and a mouse cursor pointing at it. Other options in the menu include 'Add to Sheet', 'Show Filter', 'Duplicate', 'Rename', 'Hide', 'Create', 'Transform', and 'Convert to Measure'.

A dialog box opens allowing you to define aliases for each value within the selected dimension. You can reset the member names back to their original names by clicking the **Clear Aliases** button in the bottom right corner of the Edit Aliases dialog box.

Aliases with a Cube Data Source

Aliases for cube databases are created on the server by the server administrator and can be activated in Tableau using the **Alias File** option on the **Data** menu. Please talk to your database administrator to find out whether your database has aliases available. Aliases are not supported by Microsoft Analysis Services databases.

By default the alias for every member of every dimension is initially defined to be the original member name. For example, the figure below shows a bar chart built from an Essbase database. By default, the original member names are displayed (example on the left). As you can see, these names are not very intuitive. By selecting **Data > Aliases File** and selecting an appropriate alias file set up by the database administrator, meaningful names are displayed in the headers.



Note: In Tableau, cube (multidimensional) data sources are supported only in Windows.

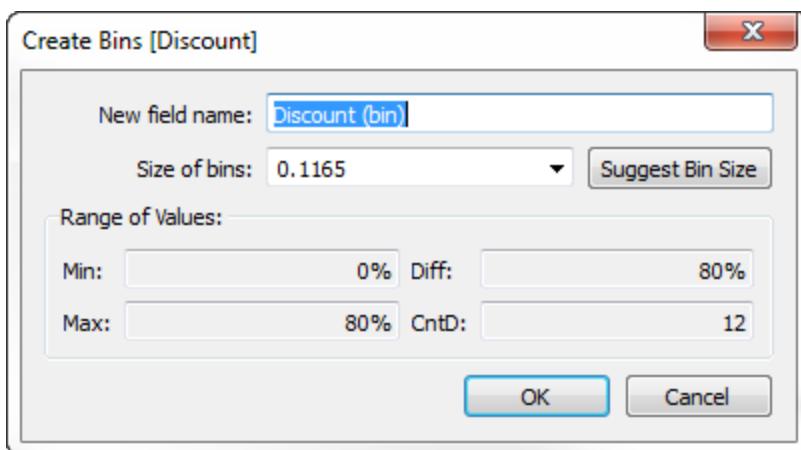
Example – Editing Aliases

This example shows how you can use aliases to rename specific field values within a dimension. You might want to do this to show labels in your view that are more relevant to your users or your analysis than what Tableau shows by default.

In this example, suppose you have a data source that contains a measure called **Discount**. **Discount** contains values from 0 to .25 that you want to analyze by grouping them into three categories: low discount, medium discount, and high discount. To create the categories, you can use bins.

For more information about how to create bins, see [Create Bins from a Continuous Measure on page 597](#).

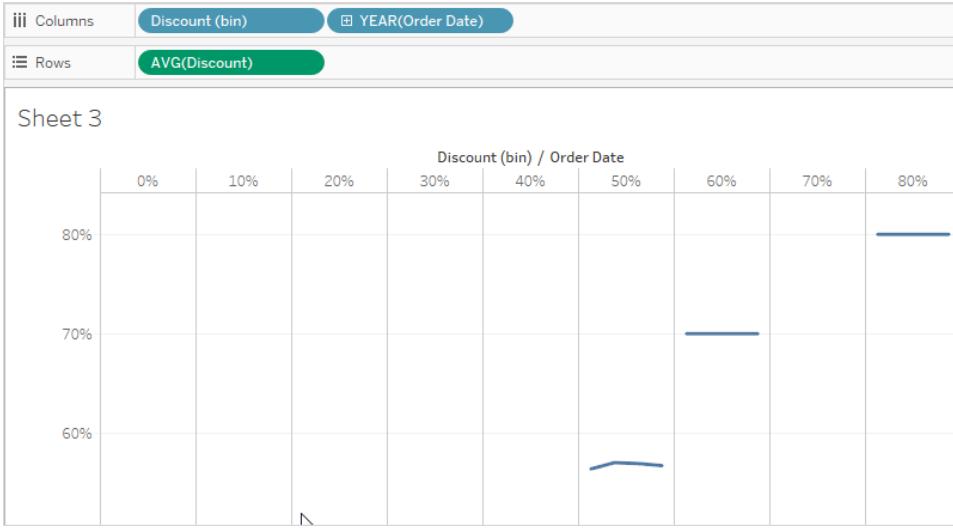
1. In a worksheet, under **Measures**, right-click (control-click on a Mac) **Discount**, and then select **Create > Bins**.



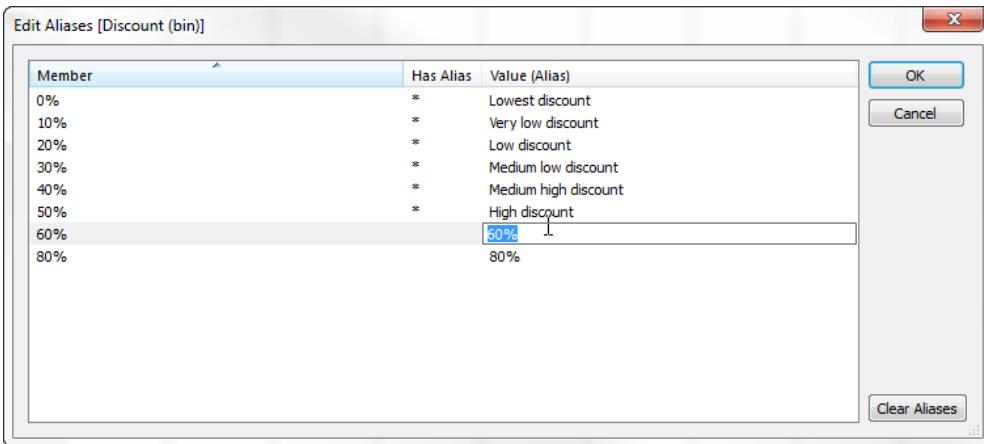
2. In the Create Bins dialog box, set **Size of bins** to **0.1**.

The new binned field is named **Discount (bin)** and appears in the **Dimensions** area of the **Data** pane.

3. Place **Discount (bin)** on the **Rows** or **Columns** shelf. The default aliases for the bins are defined by the lower limit of the bin's numerical range as shown below.



4. To make this view easier to understand, you can define aliases such as "Lowest discount," "Very low discount," and "Low discount" to replace the default column headings "0.0%," "10.0%," and "20.0%," and so on. To do so, right-click **Discount (bin)** in the **Data** pane, and select **Aliases**.
5. In the Edit Aliases dialog box, assign an alias to every member of **Discount (bin)**. For example, for the member originally labeled "0.0%," assign a label "Low discount."



6. To change an alias, right-click **Discount (bin)** in the **Data** pane, and then select **Aliases**.

In the Edit Aliases dialog box, click the alias that you want to change, and then enter the new name. Use the Tab key to move from one value to the next.

To restore the original aliases, click **Clear Aliases**. You can also sort the members or their aliases by clicking the appropriate column header.

Colors

When you use a dimension to color encode the view, default colors are assigned to the field's values. Color encodings are shared across multiple worksheets that use the same data source to help you create consistent displays of your data. For example, if you define the Western region to be green, it will automatically be green in all other views in the workbook. To set the default color encodings for a field, right-click (control-click on Mac) the field in the Data pane and select **Default Properties > Color**.

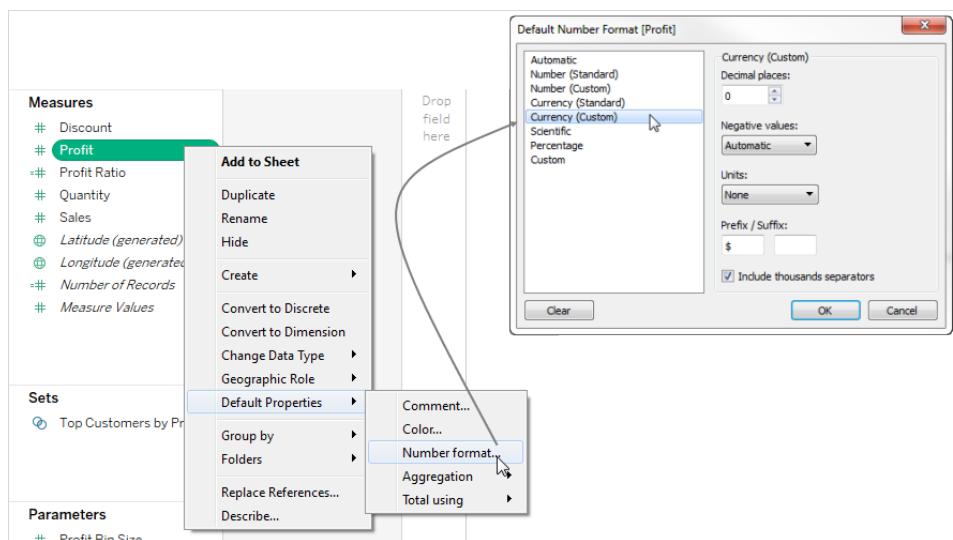
For information about color properties, and how to configure and customize colors in Tableau, see [Color Properties on page 192](#),

Shapes

When you use a dimension to shape encode the view, default shapes are assigned to the field's values. Shape encodings are shared across multiple worksheets that use the same data source to help you create consistent displays of your data. For example, if you define that Furniture products are represented with a square mark, it will automatically be changed to a square mark in all other views in the workbook. To set the default shape encodings for a field, right-click (control-click on Mac) the field in the Data pane and select **Default Properties > Shape**.

Formats

You can set the default number format for date and number fields. For example, you may want to always show the Sales values as currency using the U.S. dollar sign and two decimal places. Or you may want to always show Discount as a percentage. To set the default formats, right-click (control-click on Mac) a date or number field and select either **Date Format** or **Number Format** on the Default Properties menu. A dialog box opens where you can specify a default format.



Sort

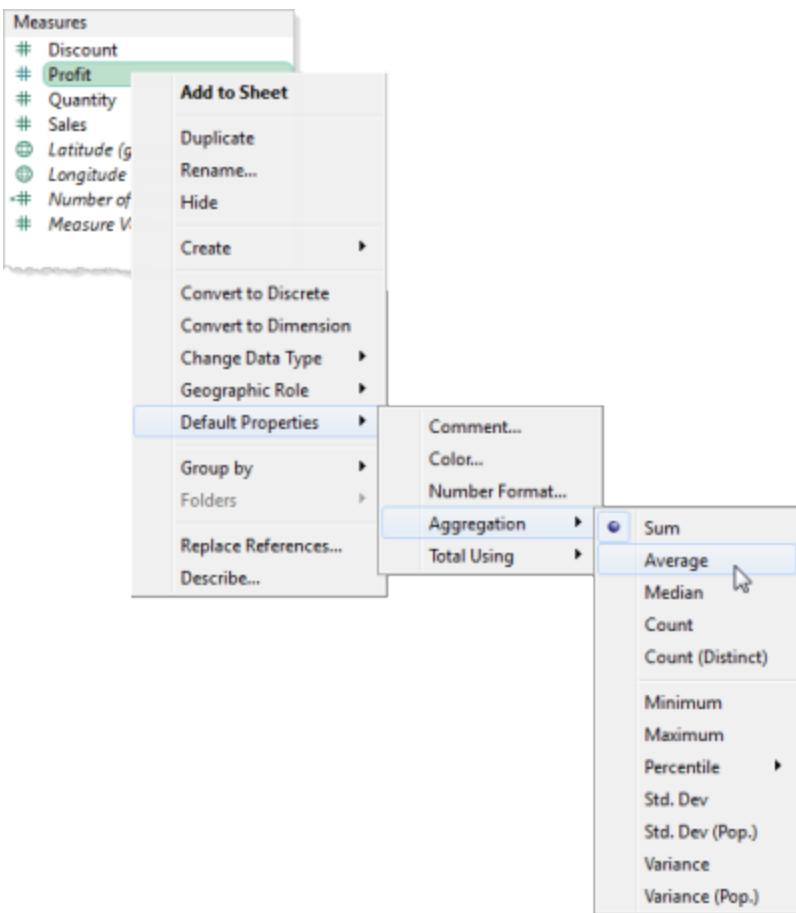
You can set a default sort order for the values within a categorical field so that every time you use the field in the view, the values will be sorted correctly. For example, let's say you have an Order Priority field that contains the values High, Medium, and Low. When you place these in the view, by default they will be listed as High, Low, Medium because they are shown in alphabetical order. You can set a default sort so that these values are always listed correctly. To set the default sort order, right-click (control-click on a Mac) a dimension and select **Default Properties > Sort**. Then use the sort dialog box to specify a sort order.

Note: The default sort order also controls how the field values are listed in a filter in the view.

Aggregation

You can also specify a default aggregation for any measure. The default aggregation will be used automatically when the measure is first totaled in the view.

1. Right-click (control-click on a Mac) any measure in the Data pane and select **Default Properties > Aggregation**.
2. On the Aggregation list, select an aggregation.



Whether you are specifying the aggregation for a field on a shelf or the default aggregation in the Data pane, you can select from several aggregations. See [Aggregations on page 329](#) to learn about each type of aggregation.

Measure Names

When you want to show multiple measures in a view, you can use the **Measure Values** and the **Measure Names** fields. When you add **Measure Names** to a view, all of the measure names appear as row or column headers in the view. The headers include each measure name. This feature becomes useful when you are working with a text table that shows multiple measures. For example, suppose you have a text table containing the aggregated profit of each product category by region.

The screenshot shows a Tableau interface with a single column view. The top shelf has 'Category' selected. The bottom shelf has 'Region' selected. The data table is titled 'Sheet 2' and contains the following data:

Region	Category		
	Furniture	Office Supplies	Technology
Central	-\$2,871	\$8,880	\$33,697
East	\$3,046	\$41,015	\$47,462
South	\$6,771	\$19,986	\$19,992
West	\$11,505	\$52,610	\$44,304

The Marks shelf shows 'Automatic' selected with 'Color', 'Size', and 'Text' options. The Detail and Tooltip buttons are also visible. A green button labeled 'SUM(Profit)' is at the bottom of the Marks shelf.

Now suppose you want to show both the **Profit** and the **Sales** for each category and region. When you add **Sales** to the text table (by dragging it and dropping it in the view), the measures are combined and the **Measure Values** field is added to Text. The **Measure Names** field is automatically added to the **Rows** shelf.

The screenshot shows a Tableau interface with a dual-measure view. The top shelf has 'Category' selected. The middle shelf has 'Region' selected. The bottom shelf has 'Measure Names' selected. The data table is titled 'Sheet 2' and contains the following data:

Region	Category		
	Furniture	Office Supplies	Technology
Central	Profit: -\$2,871	Sales: \$8,880	\$33,697
East	Profit: \$3,046	Sales: \$41,015	\$47,462
South	Profit: \$6,771	Sales: \$19,986	\$19,992
West	Profit: \$11,505	Sales: \$52,610	\$44,304

The Marks shelf shows 'Automatic' selected with 'Color', 'Size', and 'Text' options. The Detail and Tooltip buttons are also visible. A green button labeled 'Measure Values' is at the bottom of the Marks shelf.

The bottom shelf has 'Measure Values' selected, showing 'SUM(Profit)' and 'SUM(Sales)'.

Notice how the header names removed the aggregation label by default. You may want to include the aggregations or call it "Total Profit" and "Total Sales." To change the measure

names, right-click (control-click on Mac) the Measure Names field on the Rows shelf and select **Edit Aliases**. Make the changes and click **OK**.

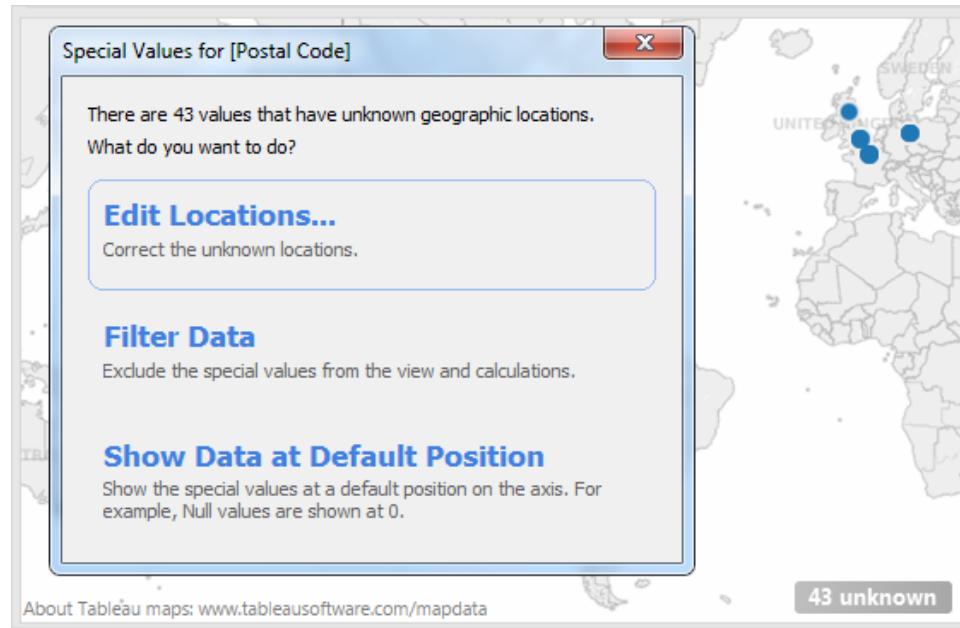
		Category		
Region		Furniture	Office Supplies	Technology
Central	Total Profit	-\$2,871	\$8,880	\$33,697
	Total Sales	\$163,797	\$167,026	\$170,416
East	Total Profit	\$3,046	\$41,015	\$47,462
	Total Sales	\$208,291	\$205,516	\$264,974
South	Total Profit	\$6,771	\$19,986	\$19,992
	Total Sales	\$117,299	\$125,651	\$148,772
West	Total Profit	\$11,505	\$52,610	\$44,304
	Total Sales	\$252,613	\$220,853	\$251,992

Special Values

Some data requires special handling in Tableau. Specifically:

- null values
- unrecognized or ambiguous geographic locations
- negative or zero values when working with a logarithmic scale
- negative or zero values when working with treemaps

When your data contains any of these special values, Tableau cannot plot them in the view. Instead, it displays an indicator in the lower right corner of the view. Click the indicator to see options for how to handle these values.



Null Numbers and Dates and Negative Values on Log Axes

When you drag a measure or continuous date to the view, the values are shown along a continuous axis. If the field contains null values or if there are zeroes or negative values on a logarithmic axis, Tableau cannot plot them. Tableau displays an indicator in the lower right corner of the view. Click the indicator and choose from the following options:

- **Filter Data** - exclude the null values from the view using a filter. When you filter data, the null values are also excluded from any calculations used in the view.
- **Show Data at Default Position** - show the data at a default location on the axis. The null values will still be included in calculations. The default position depends on the data type. The table below defines the defaults.

Type of Data	Default Position
Numbers	0
Dates	12/31/1899
Negative Values on a Log Axis	1
Unknown Geographic Location	(0,0)

Note: See [Logical Functions](#) on page 1571 for functions, such as ISNULL() and IFNULL(), that you can use to handle null values with a calculated field.

Unknown Geographic Locations

When working with maps and geographic fields, unknown or ambiguous locations are identified by the indicator in the lower right corner of the view. Click the indicator and choose from the following options:

- **Edit Locations** - correct the locations by mapping your data to known locations.
- **Filter Data** - exclude the unknown locations from the view using a filter. The locations will not be included in calculations.
- **Show Data at Default Position** - show the values at the default position of (0, 0) on the map.

Zero and Negative Values in Treemaps

When working with treemaps, any null or zero values display in the indicator in the lower right corner of the view. Click the indicator and choose from the following options:

- **Filter Data** - exclude the unknown locations from the view using a filter. The locations will not be included in calculations.
- **Use Absolute Values** - use the absolute value to determine the size of the corresponding area in the view. For example, both values of 5 and -5 are shown as the same size.

If you don't know how to handle the values, you can choose to leave the special values indicator. Generally, you should continue to show the indicator so that you know there is data that is not being shown in the view. However, to hide the indicator, right-click (control-click on a Mac) it and select **Hide Indicator**.

Aggregations

Sometimes it is useful to look at numerical data in an aggregated form such as a summation or an average. The mathematical functions that produce aggregated data are called aggregation functions. Aggregation functions perform a calculation on a set of values and return a single value. For example, a measure that contains the values 1, 2, 3, 3, 4 aggregated as a sum returns a single value: 13. Or if you have 3,000 sales transactions from 50 products in your data source, you might want to view the sum of sales for each product, so that you can decide which products have the highest revenue.

You can use Tableau to set an aggregation only for measures in relational data sources. Multidimensional data sources contain aggregated data only.

Note: Using floating-point values in combination with aggregations can sometimes lead to unexpected results. For details, see [Data Types on page 1549](#).

Tableau provides a set of predefined aggregations that are shown in the table below. You can set the default aggregation for any measure that is not a calculated field that itself contains an aggregation, such as `AVG ([Discount])`. See [How to Set the Default Aggregation for a Measure on page 333](#). You can also set the aggregation for a field already in the view. See [Aggregating Data on page 334](#).

Aggregation	Description	Result for measure that contains 1, 2, 2, 3
Attribute	Returns the value of the given expression if it only has a single value for all rows in the group, otherwise it displays an asterisk (*) character. Null values are ignored. This aggregation is particularly useful when aggregating a dimension. To set a measure in the view to this aggregation, right-click (control-click on Mac) the measure and choose Attribute . The field then changes to show the text ATTR: ATTR(Sales)	N/A
Dimension	Returns all unique values in a measure or dimension.	3 values (1, 2, 3)
Sum	Returns the sum of the numbers in a measure. Null values are ignored.	1 value (8)
Average	Returns the arithmetic mean of the numbers in a	1 value (2)

Aggregation	Description	Result for measure that contains 1, 2, 2, 3
	measure. Null values are ignored.	
Median	<p>Returns the median of the numbers in a measure. Null values are ignored.</p> <p>This aggregation is not available for workbooks created before Tableau Desktop 8.2 or that use legacy connections. It is also not available for connections using any of the following data sources:</p> <ul style="list-style-type: none"> • Access • Amazon Redshift • Cloudera Hadoop • HP Vertica • IBM DB2 • IBM Netezza • Microsoft Excel • Microsoft SQL Server • MySQL • Teradata • Text files <p>If you are connected to a workbook of that uses one of these data sources, Median is unavailable and shows the message "Requires extract." To use this aggregation, extract your data. See Extract Your Data on page 440.</p>	1 value (2)
Count	Returns the number of rows in a measure or a dimension. When applied to a dimension, Tableau creates a new temporary column that is a measure because the result of a count is a number. You can count numbers, dates, booleans, and strings. Null values are ignored in all cases.	1 value (4)
Count (Distinct)	Returns the number of unique values in a measure or dimension. When applied to a dimension, Tableau creates a new temporary column that is a measure because the result of a count is a number. You can count numbers, dates, booleans, and strings. Null values are ignored in all cases.	1 value (3)

Aggregation	Description	Result for measure that contains 1, 2, 2, 3
	<p>This aggregation is not available for the following types of workbooks:</p> <ul style="list-style-type: none"> • Workbooks created before Tableau Desktop 8.2 and that use Microsoft Excel or Text File data sources. • Workbooks that use legacy connections. • Workbooks that use Microsoft Access data sources. <p>If you are connected to a workbook that uses one of these types, Count (Distinct) is unavailable and Tableau shows the message "Requires extract." To use this aggregation, extract your data. See Extract Your Data on page 440.</p>	
Minimum	Returns the smallest number in a measure or continuous dimension. Null values are ignored.	1 value (1)
Maximum	Returns the largest number in a measure or a continuous dimension. Null values are ignored.	1 value (3)
Percentile	<p>Returns the value at the specified percentile for the measure. When you select this aggregation, you must choose from a submenu offering a range of percentile values: 5, 10, 25, 50, 75, 90, 95. When you set this aggregation on a field in the view, the field shows PCT and the percent value assigned. For example:</p> <p>PCT95(BMI)</p> <p>If you want to use a percentage value other than the ones listed, use the PERCENTILE function in a calculation and specify the percentile you want. See Aggregate Functions on page 1580.</p> <p>This aggregation is available for:</p> <ul style="list-style-type: none"> • Non-legacy Microsoft Excel and Text File connections. 	1 value. The value for PCT50 would be 2 for the given data.

Aggregation	Description	Result for measure that contains 1, 2, 2, 3
	<ul style="list-style-type: none"> Extracts and extract-only data source types (for example, Google Analytics, OData, or Salesforce). Sybase IQ 15.1 and later data sources. Oracle 10 and later data sources. Cloudera Hive and Hortonworks Hadoop Hive data sources. EXASolution 4.2 and later data sources. <p>If you are not connected to a workbook that uses one of these data sources, Percentile is unavailable and Tableau shows the message "Requires extract." In this case, if you want to use the Percentile aggregation, consider creating an extract. See Extract Your Data on page 440.</p>	
Std. Dev	Returns the standard deviation of all values in the given expression based on a sample population. Null values are ignored. Returns a Null if there are fewer than 2 members in the sample that are not Null. Use this function if your data represents a sample of the population.	1 value (0.8165)
Std. Dev (Pop.)	Returns the standard deviation of all values in the given expression based on a biased population. Assumes that its arguments consist of the entire population. Use this function for large sample sizes.	1 value (0.7071)
Variance	Returns the variance of all values in the given expression based on a sample. Null values are ignored. Returns a Null if there are fewer than 2 members in the sample that are not Null. Use this function if your data represents a sample of the population.	1 value (0.6667)
Variance (Pop.)	Returns the variance of all values in the given expression based on a biased population. Assumes that its arguments consist of the entire population. Use	1 value (0.5000)

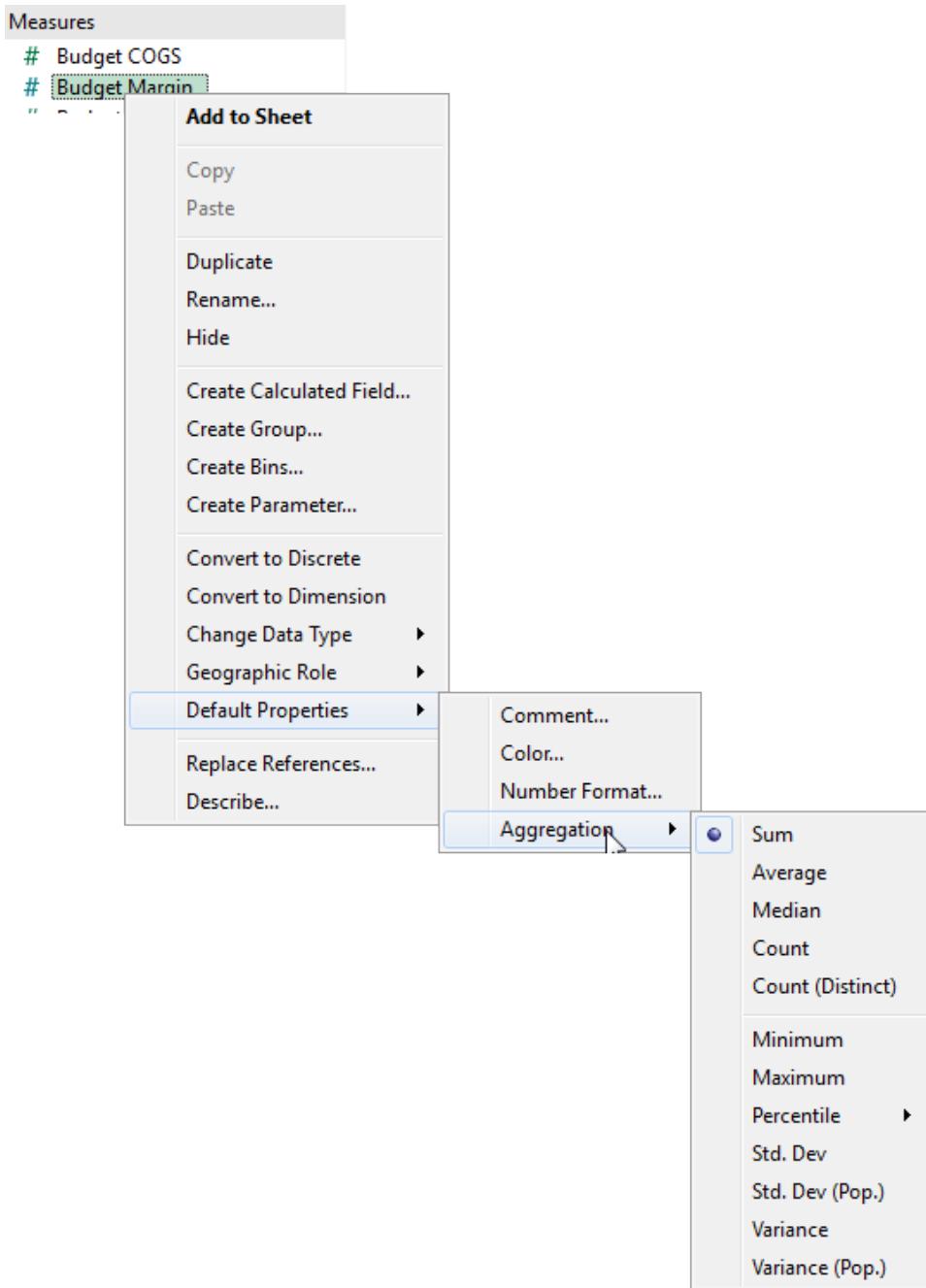
Aggregation	Description	Result for measure that contains 1, 2, 2, 3
	this function for large sample sizes.	
Disaggregate	<p>Returns all records in the underlying data source. To disaggregate all measures in the view, select Aggregate Measures from the Analysis menu (to clear the check mark).</p> <p>Tableau allows you to view data in disaggregated form (relational databases only). When data are disaggregated, you can view all of the individual rows of your data source. For example, after discovering that the sum of sales for rubber bands is \$14,600, you might want to see the distribution of individual sales transactions. To answer this question, you need to create a view that shows individual rows of data. That is, you need to disaggregate the data (see Disaggregating Data on page 338). Another way to look at disaggregated data is to view the underlying data for all or part of a view. See View Data on page 573.</p>	4 values (1, 2, 2, 3)

You can also define custom aggregations as described in [Aggregate Calculations on page 838](#). Depending on the type of data view you create, Tableau will apply these aggregations at the appropriate level of detail. For example, Tableau will apply the aggregation to individual dimension members (the average delivery time in the East region), all members in a given dimension (the average delivery time in the East, West, and Central regions), or groups of dimensions (the sum of sales for all regions and for all markets).

How to Set the Default Aggregation for a Measure

You can set the default aggregation for any measure that is not a calculated field that itself contains an aggregation, such as `AVG([Discount])`. A default aggregation is a preferred calculation for summarizing a continuous or discrete field. The default aggregation is automatically used when you drag a measure to a view. To change the default aggregation, right-click (control-click on Mac) a measure in the Data pane and select **Default Properties > Aggregation** and then select one of the options.

Note: You can use Tableau to aggregate measures only with relational data sources. Multidimensional data sources contain aggregated data only.



Aggregating Data

In Tableau, you can aggregate measures or dimension, though it is more typical to aggregate measures.

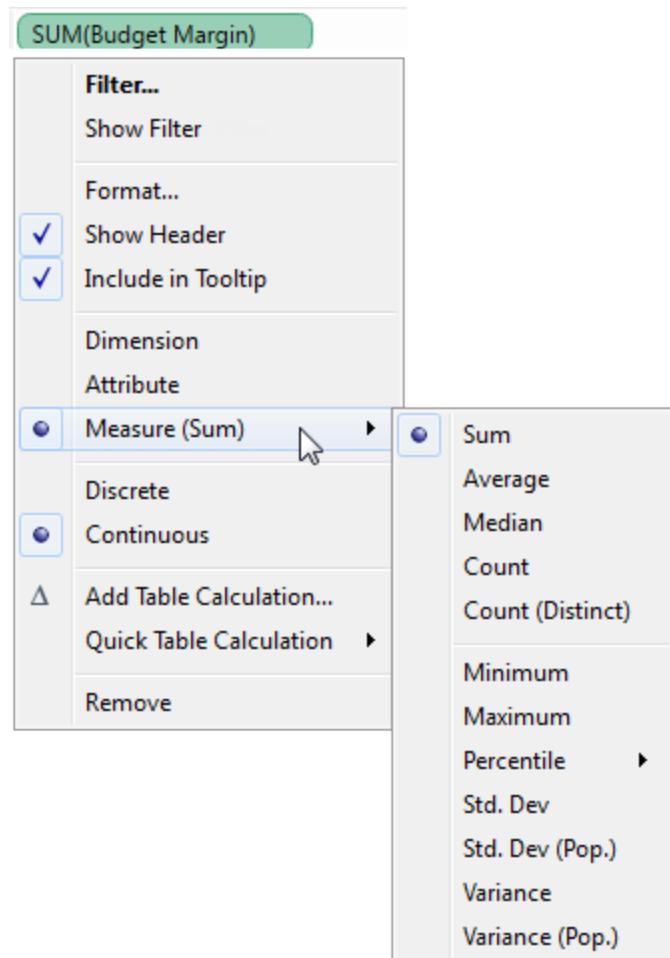
Aggregating Measures

When you add a measure to the view, Tableau automatically aggregates its values. Sum, average, and median are common aggregations; for a complete list, see [Aggregations](#) on

page 329. The current aggregation appears as part of the measure's name in the view. For example, **Sales** becomes **SUM(Sales)**. Every measure has a default aggregation which is set by Tableau when you connect to a data source. You can view or change the default aggregation for a measure—see [How to Set the Default Aggregation for a Measure](#) on page 333.

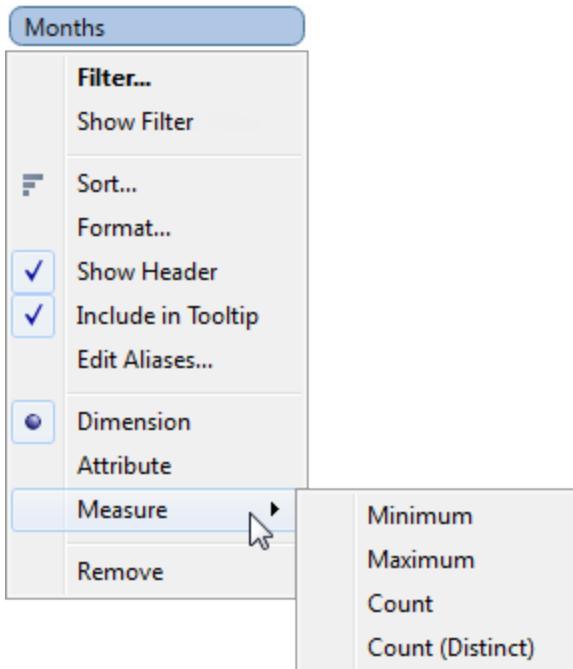
You can aggregate measures using Tableau only for relational data sources. Multidimensional data sources contain data that is already aggregated. In Tableau, multidimensional data sources are supported only in Windows.

You can change the aggregation for a measure in the view from its context menu:



Aggregating Dimensions

You can aggregate a dimension in the view as **Minimum**, **Maximum**, **Count**, or **Count (Distinct)**. When you aggregate a dimension, you create a new temporary measure column, so the dimension actually takes on the characteristics of a measure.



The **Count (Distinct)** aggregation is not supported for Microsoft Access, Microsoft Excel, and Text File data sources. If you are connected to one of these types of data sources, the **Count (Distinct)** aggregation is unavailable and shows the remark "Requires extract." This means that if you save the data source as an extract, you will be able to use the **Count (Distinct)** aggregation.

Another way to view a dimension is to treat it as an Attribute. Do this by choosing **Attribute** from the context menu for the dimension. The **Attribute** aggregation has several uses:

- It can ensure a consistent level of detail when blending multiple data sources.
- It can provide a way to aggregate dimensions when computing table calculations, which require an aggregate expression.
- It can improve query performance because it is computed locally.

Tableau computes Attribute using the following formula:

```
IF MIN([dimension]) = MAX([dimension]) THEN MIN([dimension]) ELSE
"*" END
```

The formula is computed in Tableau after the data is retrieved from the initial query. The asterisk (*) is actually a visual indicator of a special type of Null value that occurs when there are multiple values. See [Troubleshoot Data Blending on page 382](#) to learn more about the asterisk.

Below is an example of using Attribute in a table calculation. The table shows sales by market, market size, and state. Suppose you wanted to compute the percent of total sales each state contributed to the market. When you add a Percent of Total [Quick Table Calculations on](#)

page 849 that computes along State, the calculation computes within the red area shown below. This is because the Market Size dimension is partitioning the data.

Columns		Measure Names		
Rows		Market	Market Size	State
Market	Market Size	State	% of Total Sales along State	
			Sales	
Central	Major Market	Colorado	\$48,179	31.58%
		Illinois	\$69,883	45.80%
		Ohio	\$34,517	22.62%
	Small Market	Iowa	\$54,750	48.68%
		Missouri	\$24,647	21.92%
		Wisconsin	\$33,069	29.40%
East	Major Market	Florida	\$37,443	27.08%
		Massachusetts	\$29,965	21.67%
		New York	\$70,852	51.25%
	Small Market	Connecticut	\$25,429	63.07%
		New Hampshire	\$14,887	36.93%
		Tennessee	\$17,469	44.29%
South	Major Market	Texas	\$37,410	100.00%
	Small Market	Louisiana	\$23,161	34.82%
		New Mexico	\$15,892	23.89%

When you aggregate Market Size as an Attribute, the calculation is computed within the Market (East, in the following image), and the Market Size information is used purely as a label in the display.

Columns		Measure Names		
Rows	Market	ATTR(Market Size)	State	
Market	Market Size	State	Sales	% of Total Sales along State
Central	Major Market	Colorado	\$48,179	18.18%
		Illinois	\$69,883	26.37%
		Ohio	\$34,517	13.02%
	Small Market	Iowa	\$54,750	20.66%
		Missouri	\$24,647	9.30%
		Wisconsin	\$33,069	12.48%
East	Major Market	Florida	\$37,443	20.97%
		Massachusetts	\$29,965	16.78%
		New York	\$70,852	39.68%
	Small Market	Connecticut	\$25,429	14.24%
		New Hampshire	\$14,887	8.34%
South	Major Market	Texas	\$37,410	36.00%
	Small Market	Louisiana	\$23,161	22.29%
		New Mexico	\$15,892	15.29%

Disaggregating Data

To disaggregate all measures in the view, select **Analysis > Aggregate Measures**.

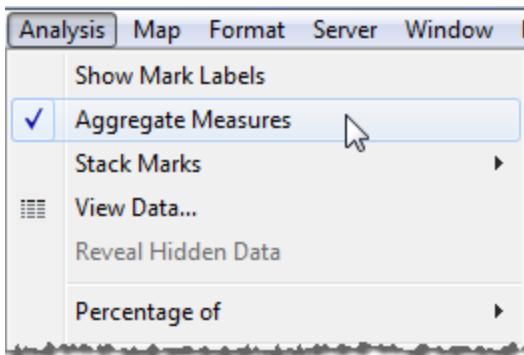


Tableau typically aggregates measures in your view, which means that it collects individual row values from your data source into a single value (which becomes a single mark) adjusted to the level of detail in your view. The different aggregations available for a measure determine how the individual values are collected: they can be added (SUM), averaged (AVG), or set to the maximum (MAX) or minimum (MIN) value from the individual row values. For a complete list of the available aggregations, [Aggregations on page 329](#). The level of detail is determined by the dimensions in your view—for information about the concept of level of detail, see [Dimensions and the Level of Detail on page 277](#).

Disaggregating your data means that Tableau will display a separate mark for every row data value in your data source. This can be useful for analyzing measures that you may want to use both independently and dependently in the view. For example, you may be analyzing the results from a product satisfaction survey with the Age of participants along one axis. You can aggregate the **Age** field to determine the average age of participants or disaggregate the data to determine at what age participants were most satisfied with the product.

Disaggregating data can be useful when you are viewing data as a scatter plot. See [Example – Scatter Plots and Aggregation](#) below.

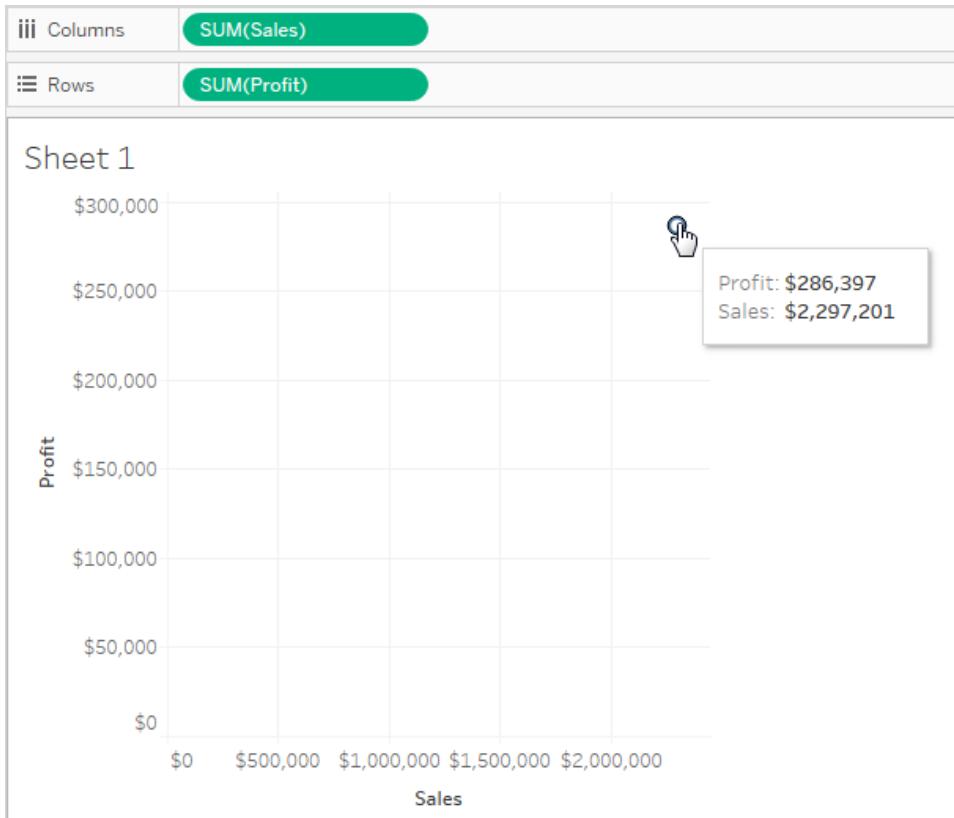
Note: If your data source is very large, disaggregating the data can result in a significant performance degradation.

Example – Scatter Plots and Aggregation

If you place one measure on the **Rows** shelf and another measure on the **Columns** shelf, you are asking Tableau to compare two numerical values. Typically, Tableau chooses a scatterplot as the default visualization in such cases. The initial view can be disappointing—a single mark, showing the sum for all values for the two measures. There are various ways to add detail to a basic scatterplot: **you can use dimensions to add detail**, you can add additional measures and/or dimensions to the Rows and Columns shelves to create multiple one-mark scatter plots in the view, or you can **disaggregate the data**. (Or you can use any combination of these options.) This topic looks at these alternatives, and uses the **Sample-Superstore** data source. To create the initial view, follow these steps:

1. Place the **Sales** measure on the **Columns** shelf.
2. Place the **Profit** measure on the **Rows** shelf.

The measures are automatically aggregated as sums. The default aggregation (SUM) is indicated in the field names. The values shown in the tooltip show the sum of sales and profit values across every row in the data source.



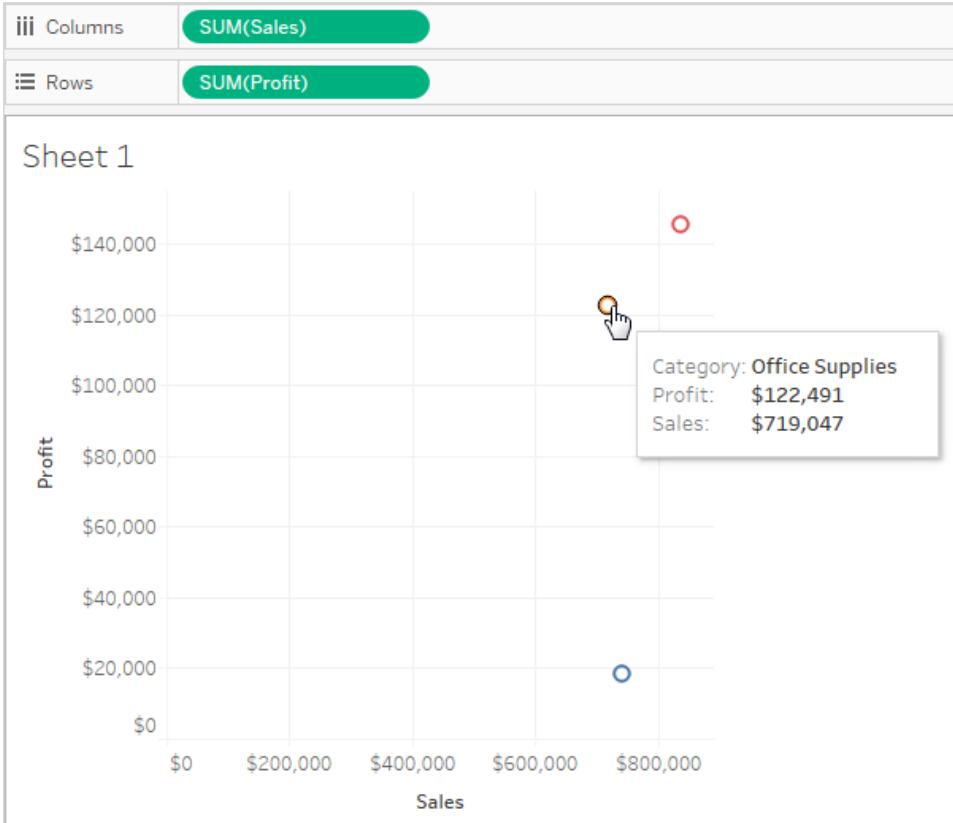
Follow the steps below to use dimensions to add detail to the view and to disaggregate data.

Using Dimensions to Add Detail

Follow these steps to develop the scatter plot view you created above by adding dimensions to show additional levels of detail.

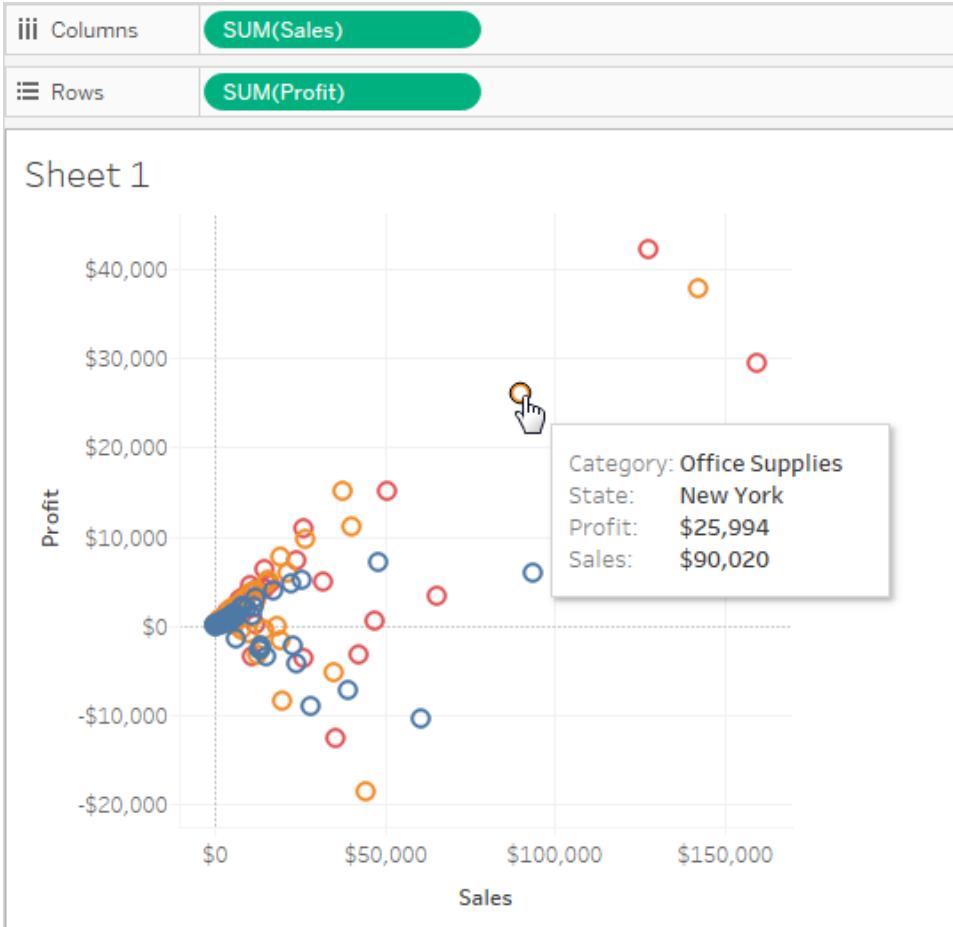
1. Drag the **Category** dimension to **Color** on the Marks card.

This separates the data into three marks—one for each dimension member—and encodes the marks using color.



2. Drag the **State** dimension to **Detail** on the Marks card.

Now there are many more marks in the view. The number of marks is equal to the number of distinct states in the data source multiplied by the number of categories.



Although more marks are now displayed, the measures are still aggregated. So regardless of whether there is one row in the data source where State = North Dakota and Category= Furniture, or 100 such rows, the result is always a single mark.

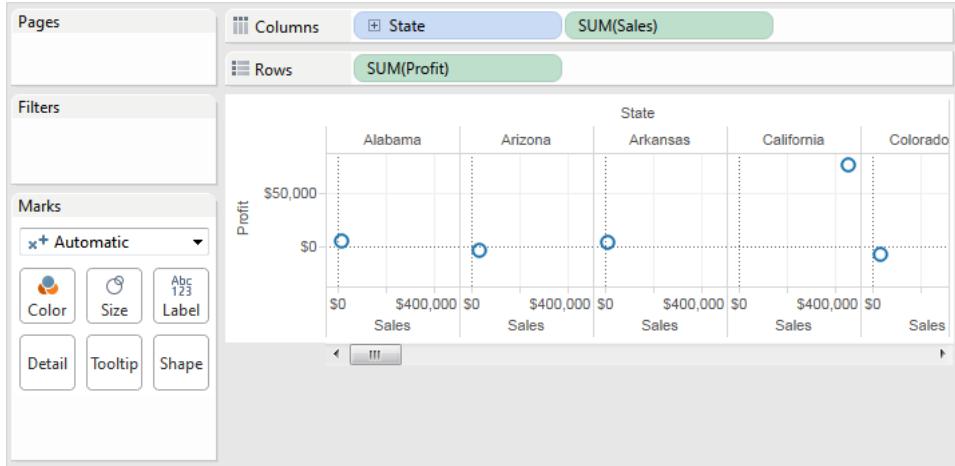
Maybe this process is developing the view in a direction you find useful, or maybe you prefer to go in a different direction—for example, by adding a time dimension to the view, or by introducing trend lines or forecasting. You decide what questions to ask.

Adding More Fields to the Rows and Columns Shelves

Revert to the original one-mark view and follow these steps to develop the scatter plot view by adding fields to the **Rows** and **Columns** shelves.

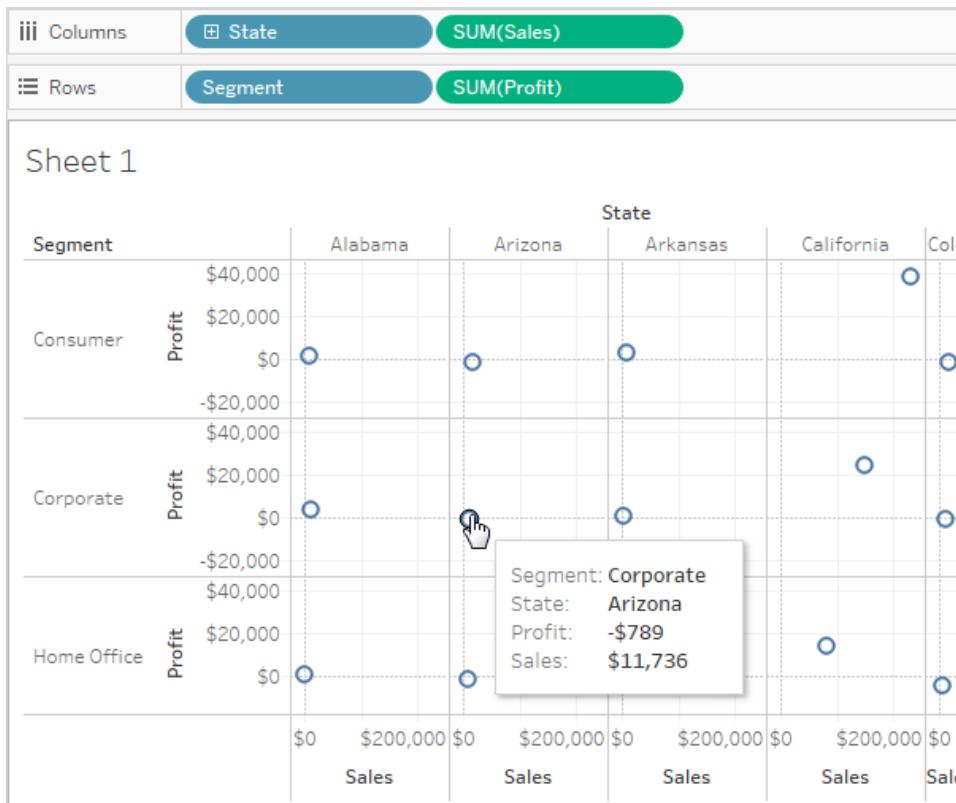
1. Drag the **State** dimension to the **Columns** shelf.

Even if you drop **Continent** to the right of **SUM(Sales)**, Tableau moves it to the left of **SUM(Sales)**. This is because you cannot insert a dimension within a continuous axis. Instead, your view shows a separate axis for each member of the dimension.



2. Drag the **Segment** dimension to the **Rows** shelf.

You now have a view that provides an overview of Sales and Profit across states and customer segments. It can be interesting to hover over the marks in the view to see tooltip data for various segments:

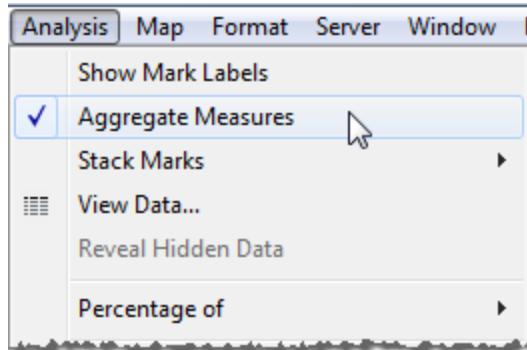


Disaggregate Data

Another way to modify your original one-mark scatter plot to display more marks is by

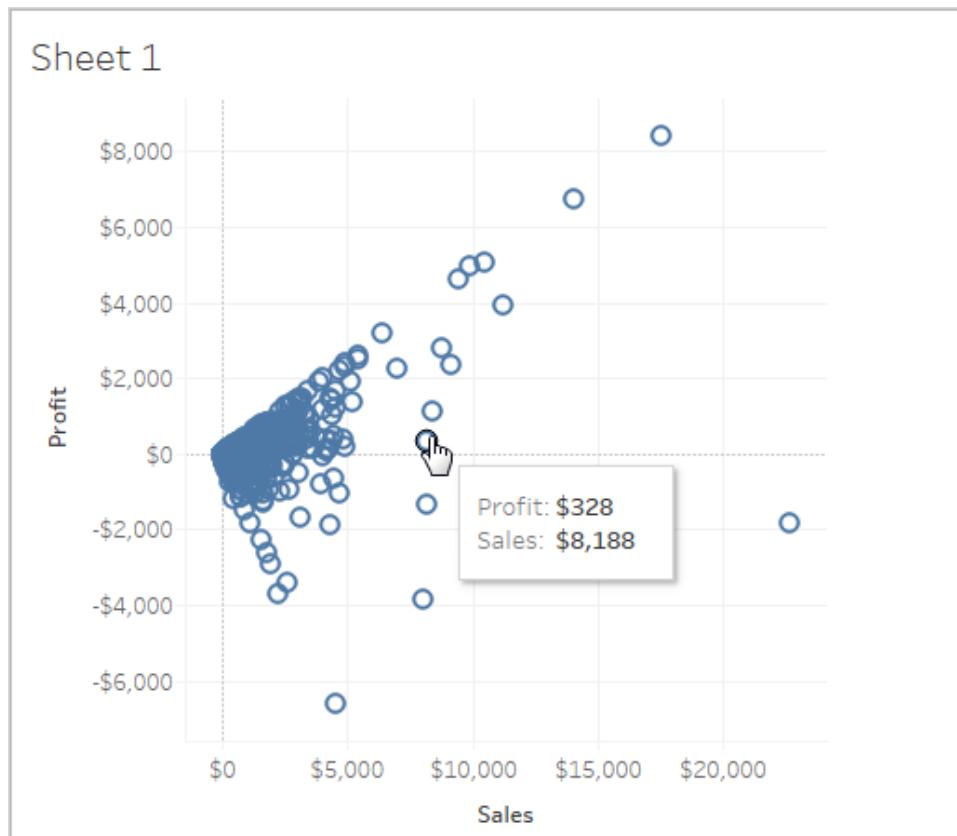
disaggregating the data.

Select **Analysis >Aggregate Measures**.



What you have actually done is to dis-aggregate the data, because this command is a toggle that was originally selected (check mark present). Tableau aggregates data in your view by default.

Now you see a lot of marks—one for each row in your original data source:



When you disaggregate measures, you no longer are looking at the average or sum for the values in the rows in the data source. Instead, the view shows a mark for every row in the data

source. Disaggregating data is a way to look at the entire surface area of the data. It's a quick way to understand the shape of your data and to identify outliers. In this case, the disaggregated data shows that for many rows in the data, there is a consistent relationship between sales income and profit—this is indicated by the line of marks aligned at a forty-five degree angle.

Connect to and Prepare Data

Before you begin your analysis, you must connect to your data and then set up the data source. There are many optional configurations that you can make on the Data Source page that Tableau uses to interpret and interact with your data.

The topics in this section describe how to connect to your data and use these configurations in order to optimize your data source for analysis.

Connect to Your Data

Before you can build a view and analyze your data, you must first connect Tableau to your data. Tableau Desktop supports connecting to a wide variety of data, stored in a variety of places. For example, your data might be stored on your computer in a spreadsheet or a text file, or in a big data, relational, or cube (multidimensional) database on a server in your enterprise. Or, you might connect to public domain data available on the web such as U.S. Census Bureau information, or to a cloud database source, such as Google Analytics, Amazon Redshift, or Salesforce.

When you launch Tableau Desktop, the data connections that are available to you are listed on the **Connect** pane, which is the left pane on the **Start** page. File types are listed first, then common server types, or servers that you've recently connected to. Click **More Servers** to see the complete list of data connections you can use. The data connections supported by your copy of Tableau Desktop are determined by the version you purchased. For more information, see the list of [Data Connections](#) on the Tableau website.

After you've connected to data sources, you can save the connections to have them show up under the **Saved data sources** section on the **Connect** pane.

You supply different information for each data connection that you want to make. For example, for most data connections, you'll need to supply a server name and your sign-in information. With some data connections, you can [Run Initial SQL](#) on page 349 statements, and SSL-enabled servers require that you select the **Require SSL** check box when you connect. The following sections discuss the specific information you need to provide for each data source you want to connect to.

Connect to your specific data source

Select from the following topics for information about how to connect to your data.

Quick Starts - short, simple introductory guides that introduce key concepts or list basic steps to accomplish a task

[Quick Start: Connect to Your Data](#) on page 1658

[Quick Start: Connect to Google Analytics](#) on page 1653

[**Quick Start: Query All Data from Google Analytics**](#) on page 1704

[**Quick Start: Connect to Salesforce**](#) on page 1656

[**Quick Start: SAP HANA Single Sign-On**](#) on page 1684 (Windows only)

[**Quick Start: Use Tableau Server Data Sources**](#) on page 1700

Data source connection reference

Connectors are listed in the order that they appear on the **Connect** pane.

[**Excel**](#) on page 1294

[**Text File**](#) on page 1303

[**Access**](#) on page 1306

[**JSON File**](#) on page 1308

[**Statistical File**](#) on page 1314

[**Other Files**](#) on page 1319 (such as Tableau .tde, .tds, .twbx)

[**Tableau Server or Tableau Online**](#) on page 1320

[**Actian Matrix**](#) on page 1322

[**Actian Vector**](#) on page 1325

[**Amazon Aurora**](#) on page 1329

[**Amazon EMR**](#) on page 1332

[**Amazon Redshift**](#) on page 1337

[**Anaplan**](#) on page 1340

[**Aster Database**](#) on page 1346

[**Cisco Information Server**](#) on page 1349

[**Cloudera Hadoop**](#) on page 1352

[**DataStax Enterprise**](#) on page 1358

[**EXASolution**](#) on page 1361

[**Firebird**](#) on page 1363

[**Google Analytics**](#) on page 1367

[**Google BigQuery**](#) on page 1371

[**Google Cloud SQL**](#) on page 1379

[**Google Sheets**](#) on page 1382

[**Hortonworks Hadoop Hive**](#) on page 1388

[**HP Vertica**](#) on page 1394

IBM BigInsights on page 1397
IBM DB2 on page 1400
IBM PDA (Netezza) on page 1404
Kognitio on page 1407
MapR Hadoop Hive on page 1410
Marketo on page 1415
MarkLogic on page 1419
MemSQL on page 1422
Microsoft Analysis Services on page 1426
Microsoft PowerPivot on page 1428
Microsoft SQL Server on page 1430
MonetDB on page 1435
MySQL on page 1438
OData on page 1441
Oracle on page 1443
Oracle Eloqua on page 1447
Oracle Essbase on page 1451
Pivotal Greenplum Database on page 1454
PostgreSQL on page 1458
Presto on page 1461
Progress OpenEdge on page 1464
QuickBooks Online on page 1467
Salesforce on page 1474
SAP HANA on page 1478
SAP NetWeaver Business Warehouse on page 1485
SAP Sybase ASE on page 1491
SAP Sybase IQ on page 1494
Snowflake on page 1497
Spark SQL on page 1501
Splunk on page 1477
Teradata on page 1504

[Teradata OLAP Connector](#) on page 1511

[Web Data Connector](#) on page 1513

[Other Databases \(ODBC\)](#) on page 1517

Tip: You can quickly create a data source in Tableau by copying and pasting data using the clipboard. For more information, see [Create a Data Source or Add a New Connection with Clipboard Data](#) on page 352.

Run Initial SQL

When connecting to some databases, you can specify an initial SQL command to run when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. This initial SQL is different than a custom SQL connection, which defines a relation (table) to issue queries against.

You can use this command to:

- Set up temporary tables to use during the session.
- Set up a custom data environment.

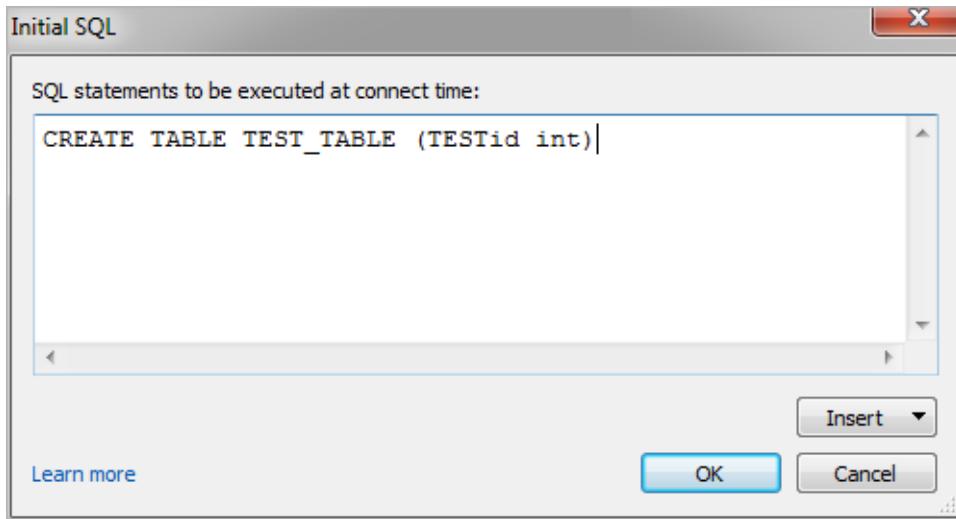
You have the option to add an initial SQL command in the Server Connection dialog box or on the Data Source page.

Note: If your data source supports running an initial SQL statement, an **Initial SQL** link appears in the lower-left corner of the Server Connection dialog box. For information about your data source, see [Connector Examples](#) on page 1294.

To use initial SQL

1. In the Server Connection dialog box, click **Initial SQL**. Or, on the Data Source page, select **Data > Initial SQL** or **Data > Query Banding and Initial SQL** depending on the database you connect to.
2. Enter the SQL command into the Initial SQL dialog box. You can use the **Insert** drop-

down menu to pass parameters to your data source.



Note: Tableau does not examine the statement for errors. This SQL statement is simply sent to the database when you connect.

Your software license may restrict you from using initial SQL with your connection. If you publish to Tableau Server, the server must be configured to allow Initial SQL statements. By default, the server software is configured to allow these statements to run when the workbook is loaded in a web browser. Administrators can disable the functionality on the **Data Connections** tab of the Tableau Server Configuration utility. If the server does not allow initial SQL statements, the workbook opens, but the initial SQL commands are not sent.

Parameters in an initial SQL statement

You can pass parameters to your data source in an initial SQL statement. There are several reasons why this is useful:

- You can configure impersonation using the **TableauServerUser** or **TableauServerUserFull** parameters.
- If your data source supports it, you can set up row-level security (for example, for Oracle VPD or SAP Sybase ASE) to make sure that users see only the data that they are authorized to see.
- You can provide more details in logging, for example, the Tableau version or the workbook name.

The following parameters are supported in an initial SQL statement:

Parameter	Description	Example of
-----------	-------------	------------

		returned value
TableauServerUser	The user name of the current server user. Use when setting up impersonation on the server. Returns an empty string if the user is not signed in to Tableau Server.	asmith
TableauServerUserFull	The user name and domain of the current server user. Use when setting up impersonation on the server. Returns an empty string if the user is not signed in to Tableau Server.	domain.lan\asmith
TableauApp	The name of the Tableau application.	Tableau Desktop Professional Tableau Server
TableauVersion	The version of the Tableau application.	9.3
WorkbookName	The name of the Tableau workbook. Use only in workbooks with an embedded data source.	Financial-Analysis

Examples

The following examples show different ways you can use parameters in an initial SQL statement.

- This example sets the security context on Microsoft SQL Server:

```
EXECUTE AS USER = [TableauServerUser] WITH NO REVERT;
```

- This example shows how, on a DataStax data source, you can use parameters to add detail to logging or to set up a session variable to keep track of the data:

```
SET TABLEAUVERSION [TableauVersion];
```

- This example can be used to help set up row-level security for Oracle VPD:

```
begin
  DBMS_SESSION.SET_IDENTIFIER([TableauServerUser]);
end;
```

Note: Oracle PL/SQL blocks require a trailing semicolon to terminate the block. Consult Oracle documentation for the proper syntax.

Defer execution to the server

You can defer an initial SQL statement so that it is executed only on the server. One reason to defer execution to the server is if you don't have permission to execute the commands that set up impersonation. Use `<ServerOnly></ServerOnly>` tags to enclose the commands to be executed only on the server.

Example:

```
CREATE TEMP TABLE TempTable(x varchar(25)) ;  
INSERT INTO TempTable VALUES (1) ;  
<ServerOnly>INSERT INTO TempTable Values(2);</ServerOnly>
```

Security and impersonation

If you use the **TableauServerUser**, **TableauServerUserFull**, or **WorkbookName** parameter in an initial SQL statement, you will create a dedicated connection that can't be shared with other users. This will also restrict cache sharing, which can enhance security, but may also slow performance.

Create a Data Source or Add a New Connection with Clipboard Data

Sometimes you want to pull in data from an outside source for some quick analysis. Rather than create a whole data source and then connect in Tableau, you can copy and paste the data directly into your workbook. Tableau automatically creates a data source that you can begin analyzing.

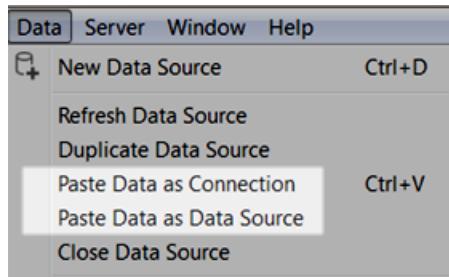
When you paste data on the data source page, Tableau creates a new connection in the existing data source.

When you paste data on the sheet, Tableau creates a new data source that you can begin analyzing. When you paste the data as a data source, the data source is saved as a text file to your Tableau Repository when you save the workbook.

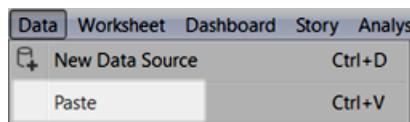
You can copy and paste data from a variety of office applications including Microsoft Excel and Microsoft Word. You can also copy and paste HTML tables from web pages. Tables that are copied as comma separated values or tab delimited can be pasted into Tableau.

Note: Not all applications use these formats when copying.

1. Select the data you want and copy it to the clipboard.
2. Open Tableau Desktop and do one of the following:
 1. On the data source page, select **Data > Paste Data as Connection** or **Paste Data as Data Source**.

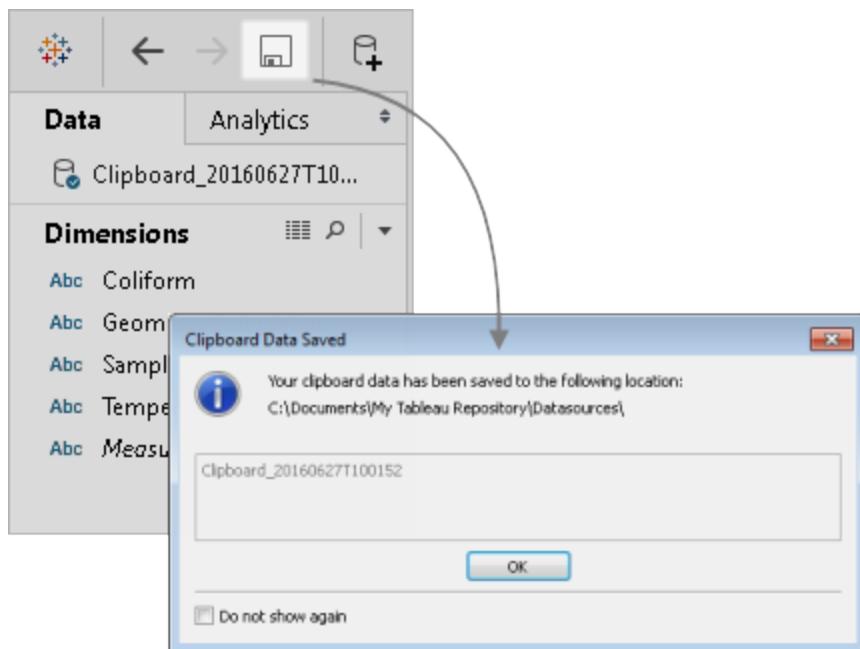


2. On the sheet, select **Data** > **Paste** to paste the data as a data source.



3. Select **File** > **Save** to save the data source.

When you save the workbook, the data source either becomes a part of the existing data source or is added to your repository, depending on which of the methods you choose. If you paste the data as a data source, the data source is saved with the workbook when you save the workbook as a packaged workbook (.twbx).



Set Up Data Sources

After you connect to your data, use the Data Source page to set up the data source and prepare your data for analysis. There are many optional configurations that you can make

before you begin your analysis. The configurations that you make on the Data Source page creates the data source that Tableau uses to interpret and interact with your data.

The topics in this section describe how to use these configurations in order to optimize your data source for analysis.

Join Your Data

In this topic:

- [Over view of join types](#) on the next page
 - [Combine tables from the same database](#) on page 356
 - [Combine tables from different databases](#) on page 359
 - [About working with multi-connection data sources](#) on page 362
 - [Join Your Data](#) above
 - [Review join results in the data grid](#) on page 365
 - [Troubleshoot joins](#) on page 365
-

The data that you analyze in Tableau is often made up of a collection of tables that are related by specific fields (that is, columns). Joining is a method for combining the related data on those common fields. The result of combining data using a join is a table that is typically extended horizontally by adding columns of data.

For example, suppose you are analyzing data for a publisher. The publisher might have two tables. The first table contains ID numbers, first name, last name, and publisher type. The second table contains ID numbers, price, royalty, and title of published books. The related field between the two tables might be **ID**.

Table 1

ID	Fir-st Na-me	Last Na-me	Pub-lisher Type
20-034	Ada-m	Davi-s	Indepen-dent
20-165	Ash-ley	Gar-cia	Big
20-233	Sus-an	Ngu-yen	Small/m-edium

Table 2

Boo-k Title	Pri-ce	Roy-alty	ID
Weat-her in the Alps	19.-99	5,00-0	20-165
My Phys-ics	8.9-9	3,50-0	20-800

The Magic Shoe Lace	15.-99	7,00-0	20-034
---------------------	--------	--------	--------

In order to analyze these two tables together, you can join the tables on **ID** to answer questions like, "How much was paid in royalties for authors from a given publisher?". By combining tables using a join, you can view and use related data from different tables in your analysis.

ID	First Name	Last Name	Publisher Type	Book Title	Price	Royalty
20034	Adam	Davis	Independent	The Magic Shoe Lace	15.99	7,000
20165	Ashley	Garcia	Big	Weather in the Alps	19.99	5,000

Over view of join types

In general, there are four types of joins that you can use to combine your data in Tableau: inner, left, right, and full outer. The tables you can join and the different join types you can use depend on the database or file you connect to. You can tell which join types your data supports by checking the join dialog after you've connected to your data and have at least two tables on the canvas.

Join Type	Result	Description
Inner	When you use an inner join to combine tables, the result is a table that contains values that have matches in both tables.	
Left	When you use a left join to combine tables, the result is a table that contains all values from the left table and corresponding matches from the right table. When a value in the left table doesn't have a corresponding match in the right table, you see a null value in the data grid.	

Right	<p>When you use a right join to combine tables, the result is a table that contains all values from the right table and corresponding matches from the left table.</p> <p>When a value in the right table doesn't have a corresponding match in the left table, you see a null value in the data grid.</p>	
Full outer	<p>When you use a full outer join to combine tables, the result is a table that contains all values from both tables.</p> <p>When a value from either table doesn't have a match with the other table, you see a null value in the data grid.</p>	

Combine tables from the same database

If the tables you need to analyze are from the same database, or workbook (for Excel), or directory (for text) then use the following procedure to combine tables. Combining tables that are from the same database require only a single connection in the data source. Typically, joining tables from the same database yields better performance. This is because querying data that is stored on the same database takes less time and leverages the native capabilities of the database to perform the join.

Note: Depending on the level of detail of the tables you want to combine, you might consider data blending instead. For more information, see [Blend Your Data on page 369](#).

To join tables

1. On the start page, under **Connect**, click a connector to connect to your data. This step creates the first connection in the Tableau data source.
2. Select the file, database, or schema, and then double-click or drag a table to the canvas.

The screenshot shows the Power BI desktop application. On the left, there's a sidebar with 'Connections' (Sample - Superstore, Excel) and 'Sheets' (Orders, People, Returns, New Union). The main area is titled 'Sample - Superstore' and contains a canvas with a red-bordered box around the 'Orders' table. To the right of the box, the text 'Drag sheets here' is displayed. At the bottom of the canvas, there are buttons for 'Sort fields', 'Data source or', 'Show aliases', and 'Show hidden fields'.

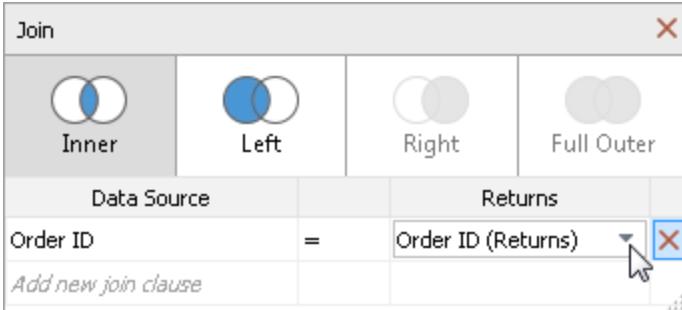
- Double-click or drag another table to the canvas, and then click the join relationship.

The screenshot shows the Power BI desktop application with a data source named 'Orders+ (Sampl...)' (Live connection). It displays two tables: 'Orders' and 'Returns'. A blue line connects the 'Orders' table to the 'Returns' table, representing a join relationship. The 'Returns' table is currently selected. Below the tables, there's a preview pane showing the first few rows of the combined data. The preview includes columns from both tables: Row ID, Order ID, Order Date, Ship Date, and Ship Mode. The data shows two rows where Order ID CA-2011-143336 appears in both tables.

#	Abc	abc	abc	Abc
Orders	Orders	Orders	Orders	Orders
Row ID	Order ID	Order Date	Ship Date	Ship Mode
19	CA-2011-143336	8/27/2012	9/1/2012	Second Class
20	CA-2011-143336	8/27/2012	9/1/2012	Second Class

- Add one or more join conditions by selecting a field from one of the available tables used in the data source, a join operator, and a field from the added table. Inspect the join condition to make sure it reflects how you want to connect the tables.

For example, in a data source that has a table of order information and another for returns information, you could use an inner join to combine the two tables based on the **Order ID** field that exists in both tables.



Note: You can delete an unwanted join condition by clicking the red "x" that displays when you hover over the right side of the condition.

- When you are finished, close the Join dialog box.

After you've created a join, review the data grid to make sure that the join produces the results that you expect. For more information, see [Review join results in the data grid on page 365](#).

Continue to prepare your data source for analysis. You can rename and reset fields, create calculations, clean your Excel data with the Data Interpreter, change the data types of text and Excel-based fields, and so on.

About null values in join keys

In general, joins are performed at the database level. If the columns used to join tables contain null values, most databases return data without the rows that contain the null values. However, if you've set up your single-connection data source to use an Excel, text, or Salesforce connection, Tableau provides an additional option to allow you to join columns that contain null values with other columns that contain null values.

To join on null values

- After you've set up your data source, on the data source page, select **Data > Join null values to null values..**

For example, suppose you have two tables of data that you want to join: **Orders_June** and **Orders_July**.

Orders_June Orders_July

ID	Location
1	New York
2	

ID	Location
1	New York
2	

3	Miami
---	-------

3	Miami
---	-------

If you join on both the **ID** and **Location** columns, most databases return the following table of data:

Join (of Orders_June and Orders_July)

ID	Location	ID(Orders_July)	Location (Orders_July)
1	New York	1	New York
3	Miami	3	Miami

If you are using a single Excel, text, or Salesforce connection in your data source, select **Data > Join null values to null values** to return the following table:

Join (of Orders_June and Orders_July)

ID	Location	ID(Orders_July)	Location (Orders_July)
1	New York	1	New York
2	<i>null</i>	2	<i>null</i>
3	Miami	3	Miami

Note: This option is available for single-connection data sources that use text, Excel, and Salesforce connections. If you add a second connection to a data source that uses this option, the join reverts back to the default behavior of excluding rows with null values.

Combine tables from different databases

Beginning with Tableau version 10.0, if the tables you need to analyze are stored in different databases, or workbooks (for Excel), or directories (for text), use the following procedure to combine tables using a *cross-database join*.

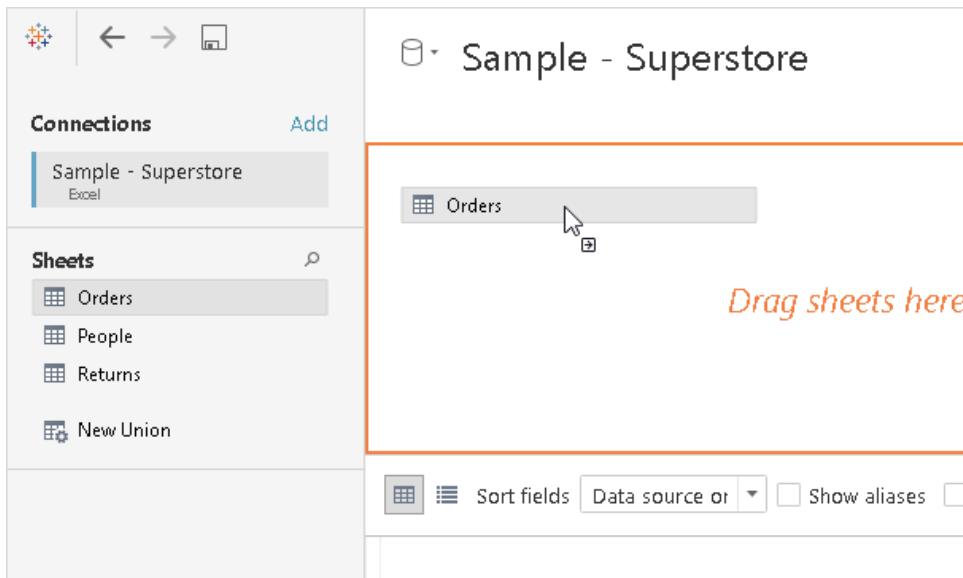
Cross-database joins require that you first set up a multi-connection data source – that is, you create a new connection to each database before you join tables. When you connect to multiple databases, a data source becomes a multi-connection data source. Multi-connection data sources can be advantageous when you need to analyze data for an organization that uses different internal systems or when you need to work with data that is managed separately by both internal and external groups.

Note: In many cases, using a cross-database join is the primary method for combining your data. However, there are some cases that you might need to combine your data using data blending instead. For more information, see [Blend Your Data](#) on page 369.

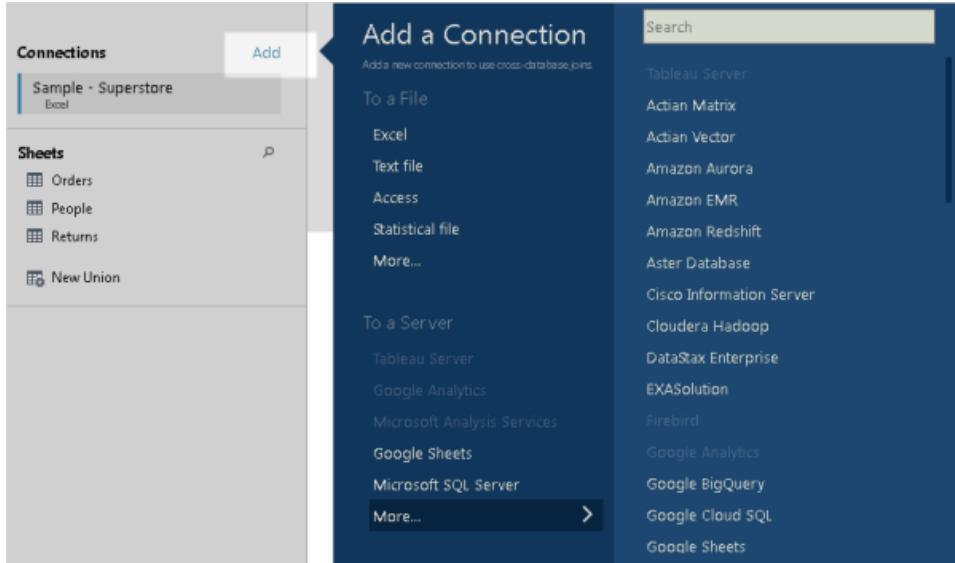
After you've combined tables using a cross-database join, Tableau colors the tables in the canvas and the columns in the data grid to show you which connection the data comes from.

To join tables from different databases

1. On the Start page, under **Connect**, click a connector to connect to your data. This step creates the first connection in the Tableau data source.
2. Select the file, database, or schema, and then double-click or drag a table to the canvas.



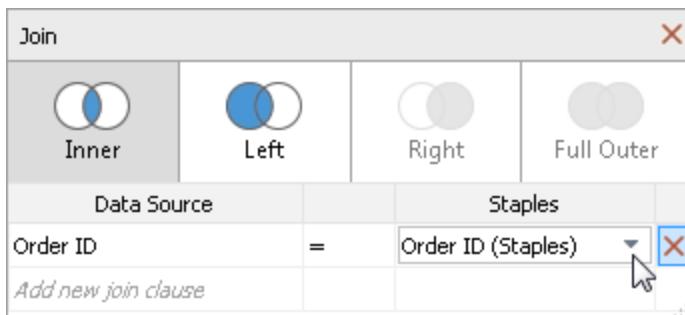
3. In the left pane, under **Connections**, click the **Add** button to add a new connection to the Tableau data source. A new connection is required if you have related data stored in another database.



Note: If the connector you want is not available from the **Connect** list, cross-database joins are not supported for the combination of sources that you want to join. This includes connections to cube data (e.g., Microsoft Analysis Services), extract-only data (e.g., Salesforce, Google Analytics, and OData), and Tableau Server data sources. Instead of joining tables, consider using data blending. For more information, see [Blend Your Data on page 369](#).

4. Add one or more join conditions by selecting a field from one of the available tables used in the data source, a join operator, and a field from the added table. Inspect the join condition to make sure it reflects how you want to connect the tables.

For example, in a data source that has a table of order information and another table of returns information, you could join the two tables based on the **Order ID** field that exists in both tables. Select the type of join.



Note: You can delete an unwanted join condition by clicking the red "x" that displays when you hover over the right-side of the condition.

- When you are finished, close the Join dialog box.

Tables and columns are colored to show you which connection the data comes from.

The screenshot shows the Data Interpreter interface. On the left, there's a sidebar with 'Connections' (Sample - Superstore, Excel, mssql), 'Database' (TestV1), and 'Table' (Election). The main area shows a join between 'Orders' and 'Staples' tables. The 'Orders' table has columns: Sales, Quantity, Discount, Profit. The 'Staples' table has columns: Item Count, Ship Priority. A legend at the bottom indicates colors: blue for 'Orders' and orange for 'Staples'. The 'Connection' dropdown is set to 'Live'.

#	Orders	Staples			
Sales	Quantity	Discount	Profit	Item Count	Ship Priority
\$69	5	80%	-\$124	1	0
\$3	3	80%	-\$4	1	0

After you've created a cross-database join, continue to prepare your multi-connection data source for analysis. You can rename and reset fields, create calculations, clean your Excel data with the Data Interpreter, change the data types of text and Excel-based fields, and so on.

About working with multi-connection data sources

Working with multi-connection data sources is just like working with any other data source, with a few caveats, discussed in this section.

Make extract files the first connection

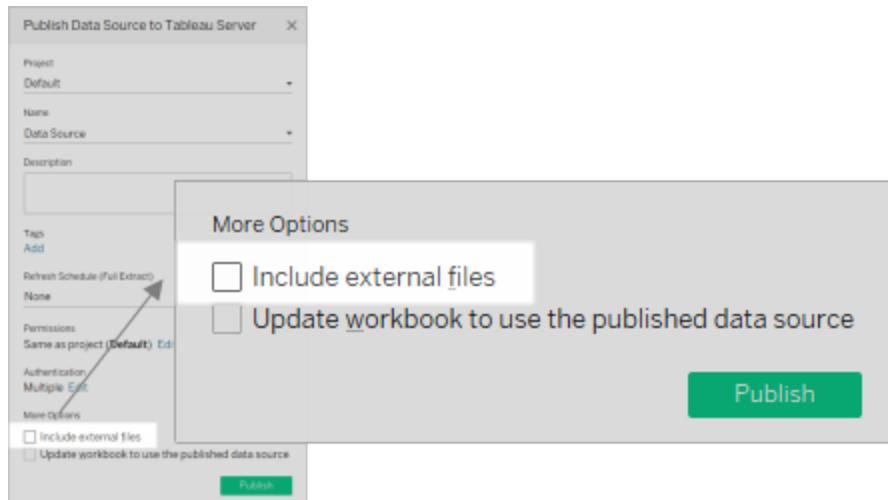
When connecting to extract files in a multi-connection data source, make sure that the connection to the extract (.tde) file is the first connection. This preserves any customizations that might be a part of the extract, including changes to default properties, calculated fields, groups, aliases, etc.

Note: If you need to connect to multiple extract files in your multi-connection data source, only the customizations in the extract in the first connection are preserved.

Extracts of multi-connection data sources that contain connections to file-based data

If you're publishing an extract of a multi-connection data source that contains a connection to file-based data such as Excel, selecting the **Include external files** option puts a copy of the

file-based data on the server as part of the data source. In this case, a copy of your file-based data can be downloaded and its contents accessed by other users. If there is sensitive information in the file-based data that you have intentionally excluded from your extract, *do not* select **Include external files** when you publish the data source.



For more information about publishing data sources, see [Publish a Data Source on page 1234](#).

Calculations and multi-connection data sources

Only a subset of calculations can be used in a multi-connection data source. You can use a specific calculation if it is both:

- Supported by all the connections in the multi-connection data source
- Supported by Tableau extracts.

Stored procedures

Stored procedures are not available for multi-connection data sources.

Pivot data from within a connection

To pivot data, you must use text columns or Excel columns from the same connection. That is, you cannot include columns from different databases in a pivot.

Union data from within a connection

To union data, you must use text tables or Excel tables from the same connection. That is, you cannot union tables from different databases. However, you can union tables across different Excel workbooks and files in different folders. For more information, see the [Union tables using wildcard search on page 395](#).

Collation

Collation refers to the rules of a database that determine how string values should be compared and sorted. In most cases, the collation is handled by the database. However, when you work with cross-database joins, you might join columns that have different collations.

For example, suppose your cross-database join used a join key comprised of a case-sensitive column from SQL Server and a case-insensitive column from Oracle.

In cases like this, Tableau maps certain collations to others to minimize interpreting values incorrectly.

The following rules are used in cross-database joins:

- If a column uses collation standards of the International Components for Unicode (ICU), Tableau uses the collation of the other column.
- If all columns use collation standards of the ICU, Tableau uses the collation of the column of the left table.
- If no columns use collation standards of the ICU, Tableau uses a binary collation. A binary collation means the locale of the database and data type of the columns determine how string values should be compared and sorted.

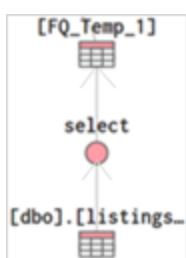
Note: Collation of Japanese characters, that is, Kana-sensitivity, depends on the database that you are connected to.

About queries and cross-database joins

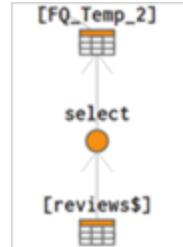
For each connection, Tableau Desktop sends independent queries to the databases in the join. The results are stored in a temporary table, in the format of an extract (.tde) file.

For example, suppose you create connections to two tables, dbo.listings and reviews\$. These tables are stored in two different databases, SQL Server and Excel. Tableau queries the database in each connection independently. the database performs the query and applies customizations such as filters and calculations, and Tableau stores the results for each connection in a temporary table. In this example, FQ_Temp_1 is the temporary table for the connection to SQL Server and FQ_Temp_2 is the temporary table for the connection to Excel.

SQL Server table



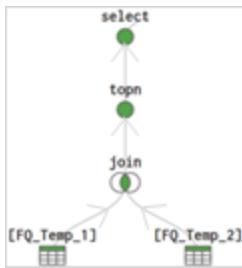
Excel table



When you perform a cross-database join, the temporary tables are joined together by Tableau Desktop. These temporary tables are necessary for Tableau to perform cross-database joins.

After the tables have been joined, "topn" filter is applied to limit the number of values shown in the data grid to the first 1,000 rows. This filter is applied to help maintain responsiveness of the data grid and the overall performance of the Data Source page.

Joined tables



Review join results in the data grid

After you have created a join on the canvas, review the data grid to make sure the join produces the results that you expect. If the data grid displays data that you don't expect, you might need to modify the join.

- **No data** If no data displays in the data grid, you might need to change the join type or a join column used in the join condition.
- **Duplicate data** If you see duplicate data, there are a few things you can do. Consider changing the aggregation of the measure that you use in your analysis, use a calculation, or use data blending instead. For more information about data blending, see [Blend Your Data on page 369](#).
- **Missing data** If some data is missing from the data grid, you might need to change the join type or a join column used in the join condition.
- **Many null values** If you see many null values that you do not expect, you might need to change the join type from the full outer type to the inner type.
- **All null values for one table** If all values for one table are null, there are no matches between the tables that you are joining. If this is not expected, consider changing the join type.

Troubleshoot joins

Over-counting values or duplicated data

When you connect to and join multiple tables together, you set up a *denormalized* version of the data. In some cases, Tableau treats the multiple tables as one table. When multiple tables are treated as one table, after the tables are joined, *all* tables are queried. This can cause values to be over-counted.

For example, suppose you have two tables. The first table contains employee information, such as employee ID and salary. The second table contains organizational information, such as department names.

Table 1

Employee ID	Name	Salary
2010-6	Kim, Michelle	63,000
2010-7	Wilson, Henry	61,000
2010-8	Smith, Andrew	65,000

Table 2

Department	Hire Date	Employee ID	Organization
Development	6/16/2016	2028-9	Strategy
Support	7/28/2015	2010-7	Operations
Support	8/05/2016	2029-9	Operations
Sales	8/25/2016	2010-7	Operations

If you join these tables on **Employee ID**, an employee's salary is counted more than once because the employee is associated with multiple departments.

Employee ID	Name	Salary	Department	Hire Date	Organization
20107	Wilson, Henry	61,000	Support	7/28/2015	Operations
20107	Wilson, Henry	61,000	Sales	8/25/2016	Operations

To help resolve this issue, you can do one of the following:

- **Change the aggregation:** Depending on your analysis, you can use the **MIN** or **AVG** aggregation to remove over-counting.

For example, if you change the aggregation of **Salary** from SUM to MIN or from SUM to AVG.

- **Create a calculated field that removes the duplicate values:** You can create a

calculation that divides the sum of the field being duplicated by the number of instances of the field that is causing the duplication.

For example, the Salary values are being duplicated by the number of instances of Employee ID for Wilson, Henry. In this case, in the view replace Salary with a calculated field that uses the following formula: `SUM ([Salary]) / COUNT ([Employee ID])`.

- **Blend the data:** Instead of creating a join which might duplicate your data, you can blend the data on a common field instead. For more information, see [Blend Your Data on page 369](#).
- **Use custom SQL:** Tableau supports using custom SQL for connecting to multiple tables in a data source. You can pre-aggregate the tables before joining them with a GROUP BY clause. This option requires some knowledge about writing SQL queries and assistance from a database expert if possible. For more information about how to connect to a custom SQL query from Tableau, see [Connect to a Custom SQL Query on page 404](#).

Broken cross-database joins

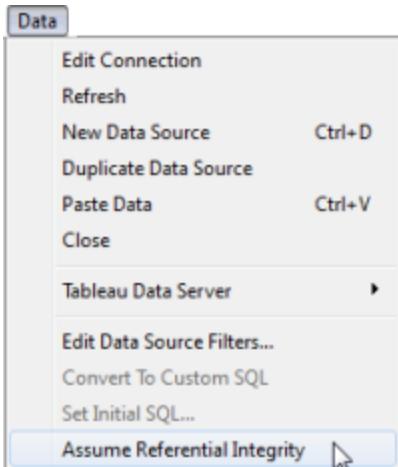
Before you join tables using a cross-database join, make sure that the data types of the join keys match. If the data types of the join keys don't match, the join breaks, which is indicated by a red exclamation point. To fix a broken join, take one of the following steps:

- For text or Excel-based data, modify the data type of one of the text or Excel fields in the join key using the data type menu in the data grid.
- For most connections, you can use the **Convert to custom SQL** option to change the data type of one of the fields in the join key, using a function like `CAST ()`. The **Convert to custom SQL** option is available only when the data source contains only one connection. In this case, remove the second connection and then select **Data > Convert to custom SQL**.

[\(Back to top\)](#)

Assuming Referential Integrity

In some cases, you can improve query performance by selecting the option to **Assume Referential Integrity** from the **Data** menu. When you use this option, Tableau will include the joined table in the query only if it is specifically referenced by fields in the view.



Using this setting is appropriate when you know that your data has referential integrity (see definition below) but your database is not enforcing or cannot enforce referential integrity. If you have the option of configuring referential integrity in your database that is a better option than using this setting because it can improve performance both in the database and in Tableau.

The **Assume Referential Integrity** option in Tableau can only affect performance on Tableau's end. If your data does not have referential integrity and you turn on this setting, query results may not be reliable.

To understand what referential integrity is, imagine connecting to sales data that has two tables: Sales and Product Catalog. These two tables are shown below:

Sales

Product ID (Foreign Key)	Sale Amount	Transaction Date
1	100	10/1/2012
1	2000	10/2/2012
2	50	9/30/2012
3	10	8/21/2012

Product Catalog

Product ID (Primary Key)	Product Name
1	10 Inch Tablet
2	Smart Phone
3	Desk Lamp
4	Memory Stick

Because all products that are sold must have a listing in the Product Catalog, every row in the Sales table has a matching row in the Product Catalog table. When these two tables are joined on Product ID, you end up with a table that looks like this:

Product ID	Product Name	Product ID	Sale Amount	Transaction Date
1	10 Inch Tablet	1	100	10/1/2012
1	10 Inch Tablet	1	2000	10/2/2012
2	Smart Phone	2	50	9/30/2012
3	Desk Lamp	3	10	8/21/2012

Now let's say you build a view to look at Sale Amount by Region. By default, dragging the Sale Amount field to the view may create a query like this:

```
SELECT SUM([Sales Amount]) FROM [Sales] S INNER JOIN [Product Catalog] P ON S.ProductID = P.ProductID
```

By selecting **Assume Referential Integrity**, you tell Tableau that the joined tables have referential integrity. In other words, you are confirming that the Sales table will always have a matching row in the Product Catalog table. Because that is true, Tableau doesn't need any information from the Product Catalog table in order to return these results. When you drag the Sales Amount field into the view, Tableau can simplify the query to:

```
SELECT SUM([Sales Amount]) FROM [Sales]
```

This simplified query can often return quicker results because it removes the join operation. This option impacts only inner joins and does not affect data sources with a single table.

Blend Your Data

In this topic:

- [Prerequisites for data blending](#) on page 371
 - [Differences between joins and data blending](#) on page 371
 - [Blend your data](#) on page 373
 - [Data blending limitations](#) on page 375
-

Data blending is a method for combining data that supplements a table of data from one data source with columns of data from a second data source.

Usually you use joins to perform this kind of data combining, but there are times, depending on factors like the type of data and its granularity, when it's better to use data blending.

For example, suppose you have transactional data stored in Salesforce and quota data stored in an Excel workbook. The data you want to combine is stored in different databases, and the

granularity of the data captured in each table is different in the two data sources, so data blending is the best way to combine this data.

Data blending is useful under the following conditions:

- **You want to combine data from different databases that are not supported by cross-database joins.**

Cross-database joins do not support connections to cubes (for example, Oracle Essbase) or to some extract-only connections (for example, Salesforce). In this case, set up individual data sources for the data you want to analyze, and then use data blending to combine the data sources on a single sheet.

- **Data is at different levels of detail.**

Sometimes one data set captures data using greater or lesser granularity than the other data set.

For example, suppose you are analyzing transactional data and quota data.

Transactional data might capture all transactions. However, quota data might aggregate transactions at the quarter level. Because the transactional values are captured at different levels of detail in each data set, you should use data blending to combine the data.

Use data blending instead of joins under the following conditions:

- **Data needs cleaning.**

If your tables do not match up with each other correctly after a join, set up data sources for each table, make any necessary customizations (that is, rename columns, change column data types, create groups, use calculations, etc.), and then use data blending to combine the data.

- **Joins cause duplicate data.**

Duplicate data after a join is a symptom of data at different levels of detail. If you notice duplicate data, instead of creating a join, use data blending to blend on a common dimension instead.

- **You have lots of data.**

Typically joins are recommended for combining data from the same database. Joins are handled by the database, which allows joins to leverage some of the database's native capabilities. However, if you're working with large sets of data, joins can put a strain on the database and significantly affect performance. In this case, data blending might help. Because Tableau handles combining the data after the data is aggregated, there is less data to combine. When there is less data to combine, generally, performance improves.

Note: When you blend on a field with a high level of granularity, for example, date instead of year, queries can be slow.

Prerequisites for data blending

Your data must meet the following requirements in order for you to use data blending.

Primary and secondary data sources

Data blending requires a primary data source and at least one secondary data source. When you designate a primary data source, it functions as the main table or main data source. Any subsequent data sources that you use on the sheet are treated as a secondary data source. Only columns from the secondary data source that have corresponding matches in the primary data source appear in the view.

Using the same example from above, you designate the transactional data as the primary data source and the quota data as the secondary data source.

Note: Cube (multidimensional) data sources must be used as the primary data source. Cube data sources cannot be used as a secondary data source.

Defined relationship between the primary and secondary data sources

After designating primary and secondary data sources, you must define the common dimension or dimensions between the two data sources. This common dimension is called the *linking field*.

Continuing the example from above, when you blend transactional and quota data, the date field might be the linking field between the primary and secondary data sources.

- If the date field in the primary and secondary data sources have the same name, Tableau creates the relationship between the two fields and shows a link icon () next to the date field in the secondary data source when the field is in the view.
- If the two dimensions don't have the same name, you can define a relationship that creates the correct mapping between the date fields in the primary and secondary data sources.

Differences between joins and data blending

Data blending simulates a traditional left join. The main difference between the two is *when* the join is performed with respect to aggregation.

Left join

When you use a left join to combine data, a query is sent to the database where the join is performed. Using a left join returns all rows from the left table and any rows from the right table

that has a corresponding row match in the left table. The results of the join are then sent back to and aggregated by Tableau.

For example, suppose you have the following tables. If the common columns are **User ID** and **Patron ID**, a left join takes all the data from the left table, as well as all the data from the right table because each row has a corresponding row match in the left table.



User ID	District	Branch	Patron ID	Dist	User ID	District	Level	Branch	Type
1		A001			1	2	3	A001	G
2		B001		2	1	2	3	A001	W
4	5	C001		1	1	2	3	C001	G
1	2	3 W			1	2	3	C001	W
					2	3	4	B001	J
					4	5	6	null	M

Data blending

When you use data blending to combine data, a query is sent to the database for each data source that is used on the sheet. The results of the queries, including the aggregated data, are sent back to and combined by Tableau. The view uses all rows from the primary data source, the left table, and the aggregated rows from the secondary data source, the right table, based on the dimension of the linking fields. Dimension values are aggregated using the ATTR aggregate function, which means the aggregation returns a *single* value for all rows in the secondary data source. If there are multiple values for the rows, an asterisk (*) is shown. Measure values are aggregated based on how the field is aggregated in the view.

You can change the linking field or add more linking fields to include different or additional rows of data from the secondary data source in the blend, changing the aggregated values.

For example, suppose you have the following tables. If the linking fields are **User ID** and **Patron ID**, blending your data takes all of the data from the left table, and supplements the left table with the data from the right table. In this case, not all values can be a part of the resulting table because of the following:

- A row in the left table does not have a corresponding row match in the right table, as indicated by the null value.
- There are multiple corresponding values in the rows in the right table, as indicated by the asterisk (*).

User ID	District	Level	Type	Branch	Patron ID	Dist User ID	District	Level	Branch	Type
1	2	3	G	A001			1	2	3 *	G
2		5	B001		2		2	3	4 B001	J
4			C001		1		4	5	6 null	M
1	2						1	2	3 *	W

Suppose you have the same tables as above, but the secondary data source contains a new field called **Fines**. Again, if the linking fields are **User ID** and **Patron ID**, blending your data takes all of the data from the left table, and supplements it with data from the right table. In this case, you see the same null value and asterisks in the previous example in addition to the following:

- Because the **Fines** field is a measure, you see the row values for the **Fines** field aggregated before the data in the right table is combined with the data in the left table.
- As with the previous example, a row in the left table does not have corresponding row for the **Fines** field, as indicated by the second null value.

User ID	District	Branch	Patron ID	Dist User ID	District	Level	Type	Branch	Fines
1	G	A001			1	2	3 G	*	40
2		B001	2		2	3	4 J	B001	20
4		C001	1		4	5	6 M	null	null
1					1	2	3 W	*	40

↓

Branch	Patron ID	District	Level	Fines
*	1	2	3	40.00
B001	2	3	4	20.00
*	1	2	3	40.00

Blend your data

You can use data blending when you have data in separate data sources that you want to analyze together on a single sheet. The following example demonstrates how to blend data from two data sources: an Excel data source and an SQL Server data source.

To blend your data

Step 1: Connect to your data and set up the data sources

- Connect to a set of data and set up the data source on the data source page. This

example uses the **Sample - Superstore** data source.

2. Select **Data > New data source**, connect to the second set of data, and then set up the data source. This example uses a SQL Server data source that contains information about forecasted sales, called Sales Plan.
3. Click the sheet tab to start building your view.

Step 2: Designate a primary data source

Drag at least one field from your primary data source into the view to designate it as the primary data source.

1. In the **Data** pane, click the data source that you want to designate as the primary data source. In this example, **Sample - Superstore** is selected.
2. Drag the fields you want to use from the data source into the view. In this example, a view is created that shows Sales by Segment and Category.

Step 3: Designate a secondary data source

Any fields used in the view from data sources that are not the primary data source or active links automatically designate subsequent data sources as the secondary data source.

1. In the **Data** pane, click the data source that you want to designate as the secondary data source. In this example, the Sales Plan data source is selected.

When you complete this step, an orange bar displays down the left side of the Data pane. The orange bar indicates the secondary data source. In addition to the orange bar, broken link icons display next to potential linking fields.
2. Click a broken link icon () to establish a relationship between the secondary and primary data sources. This is the field that determines the level of detail that Tableau should aggregate to. In this example, Segment is the linking field.

If a broken link icon does not appear next to the field that should be the linking field or no broken links appear, see **Step 4: (Optional) Define or edit relationships** below.
3. Drag the fields you want to use from the secondary data source into the view. In this example, Sales Plan field is used on the **Detail** card to change the level of detail of the view.

Step 4: (Optional) Define or edit relationships

Tableau detects when a field from the primary data source also exists in a secondary data source, and indicates that the fields are potential linking fields by marking them with a broken link icon in the **Data** pane. You click a broken link icon to establish a relationship between the primary and secondary data sources, and have Tableau blend data from both data sources on a single sheet.

You must have at least one linked field in order to use data from the secondary data source.

1. Select **Data > Edit Relationships**.
2. In the Relationships dialog box, verify that the primary data source is selected from the **Primary data source** drop-down list. In this example, the **Sample - Superstore** data source is selected.
3. Select the secondary data source in the **Secondary data source** pane, select **Custom** in the relationships list, and then click the **Add** button. In this example, the Sales Plan data source is selected.
4. In the Add/Edit Field Mapping dialog box, do the following:
 1. Select a field from the primary data source.
 2. Select a field from the secondary data source to establish the linking field or the relationship between the data sources even though the fields do not have the same name.
 3. Click **OK**.

In this example, a relationship between **Segment** field in the Superstore – Sample data source is selected and **Customer Segment** field in the Sales Plan data source is selected. You can map these two fields to create a relationship even though they don't have the same name.

5. (Optional) Continue to add and remove as many relationships as necessary and then click **OK**.

The related fields are shown in the secondary data source as potential linking fields.

6. Click the broken link icon () next to these fields in the data pane to make the relationship between the primary and secondary data sources active. In this case, the broken link icon next to Customer Segment is made into an active link icon ().

If the related field from the primary data source is used in the view, the link becomes active automatically.

When data blending, the relationship matches values based on the member aliases. You can fix fields that don't match by editing the aliases. For example, when you map a **Segment** field in the primary data source to the **Segment** field in the secondary data source, "Small Business" will not map correctly to "S. Business". You must edit the aliases in one of the data sources. For more information, see [Example – Editing Aliases on page 320](#).

You can also use a secondary data source to re-alias the field values in a primary data source. For more information, see [Alias Field Values Using Data Blending on the next page](#).

Data blending limitations

There are some data blending limitations around non-additive aggregates, such as COUNTD, MEDIAN, and RAWSQLAGG. For more information, see [Troubleshoot Data Blending on](#)

Alias Field Values Using Data Blending

Data blending is a method for combining data. Data blending works by supplementing the data in the primary data source with the data in the secondary data source.

Aliasing is the alternative name that you can assign to a value in a dimension field. You can use aliases to rename specific values within a dimension. This can be useful when you want to show more relevant or descriptive dimension values in your view than what the original data provides.

You can use data blending as a method for re-aliasing values in a data source by using field values from another data source. To do this, there must be a field in the secondary data source that contains aliases for a field in the primary data source.

For example, suppose you have a primary data source that contains information on fruit, including fruit names and code values that the fruit are associated with. A secondary data source contains a field with more descriptive code values.

Primary data source

Fruit	Fruit ID
Apple	A
Banana	B

Secondary data source

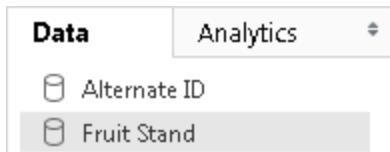
Alternate ID
APP
BAN

When you re-alias a field in your primary data source, you can create a view that replaces the original code values with the more descriptive code values.

Note: After you have used the secondary data source to provide aliases for the primary data source, you no longer need the secondary data source and can close it.

To alias field values

1. Set up the primary and secondary data sources and establish a relationship between the data sources. For more information, see [Step 4: \(Optional\) Define or edit relationships](#) on page 374. The data sources used in this example are Fruit Stand and Alternate ID.



2. Select the primary data source in the **Data** pane, and then drag the field that you want to

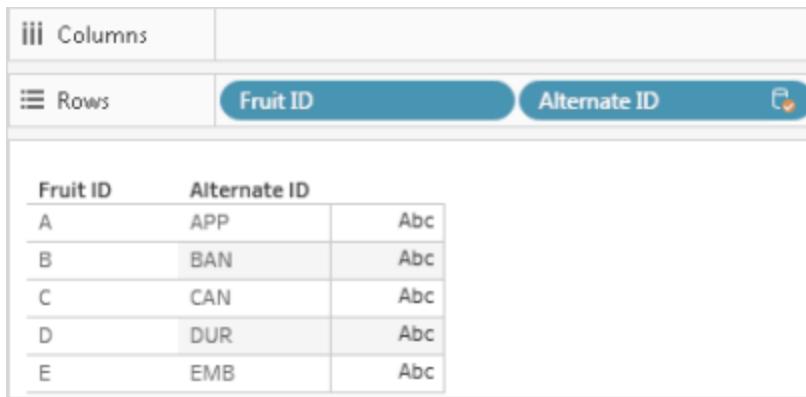
alias to the view.

For this example, Fruit Stand is the primary data source.

3. Select the secondary data source in the **Data** pane, and drag the field that contains the alias information to the view.

For this example, Alternate ID is the secondary data source. **Note:** The field you are using to alias the field values in the primary data source cannot be the linking field you are using to link the primary and secondary data sources.

In this example, Alternate ID field in the secondary data source contains the aliasing information needed by the Fruit Stand field in the primary data source.



Fruit ID	Alternate ID
A	APP
B	BAN
C	CAN
D	DUR
E	EMB

4. Right-click (control-click on a Mac) the field from the secondary data source and select **Edit Primary Aliases**.

Values in the primary data source must have corresponding alias values in the secondary data source. An asterisk under **Has Alias** confirms that the data sources have corresponding alias values .

Edit Aliases [Fruit ID]

Member	Has Alias	Value (Alias)
A	*	APP
B	*	BAN
C	*	CAN
D	*	DUR
E	*	EMB

OK Cancel Clear Aliases

Note: If asterisks don't appear under **Has Alias**, then either the secondary data source doesn't have the corresponding aliases to match the field values in the primary data source, or there is more than one field value in the primary data source that has the same alias in the secondary data source. In cases like this, aliasing field values in the primary data source with data from the secondary data source is not possible.

- Click **OK**. The alias values in the secondary data source replace the field values in the primary data source. In this example, the more descriptive code values from Alternate ID field replaces the values in the Fruit ID field.

Columns		
Rows	Fruit ID	Alternate ID
Fruit ID	APP	APP
	BAN	BAN
	CAN	CAN
	DUR	DUR
	EMB	EMB

- (Optional) Select the secondary data source from the **Data** menu, and select **Close**.

To watch a video that demonstrates aliasing field values using data blending, see [Cleaning Data by Bulk Re-Aliasing](#).

Bring a Field into the Primary Data Source

Data blending is a method for combining data. Data blending works by supplementing the data in the primary data source with the data in the secondary data source.

In cases where you might only need just *some* of the data in the secondary data source, you can create a primary group to bring only the fields that you need into the primary data source. This eliminates the need for the secondary data source, which reduces the size of the workbook and can improve workbook performance.

For example, suppose you have two data sources: Superstore and Population. The Superstore data source contains lots of data, but most importantly, it contains data about states and the region each of the states is associated with. The Population data source contains state and population data, but doesn't contain any data about regions. In this case, you can blend the Population and Superstore data sources to create a primary group. Then so that you can pull in the Region field from the Superstore data source into the Population data source, thereby eliminating the need for the Superstore data source all together.

To blend two data sources and create a primary group

1. Download and open the following workbook from Tableau Public.
 1. Go to
<https://public.tableau.com/profile/tableau.docs.team#!/vizhome/BringaFieldintothPrimaryDataSource/Sheet1>.
 2. Click **Download Workbook** in the upper-right corner and then open the workbook.

In the workbook, the **State** dimension is already on the **Rows** shelf, making it the first field in the view and establishing the Population data source as the primary data source for the sheet. The Superstore data source is also included in the workbook and functions as the secondary data source.

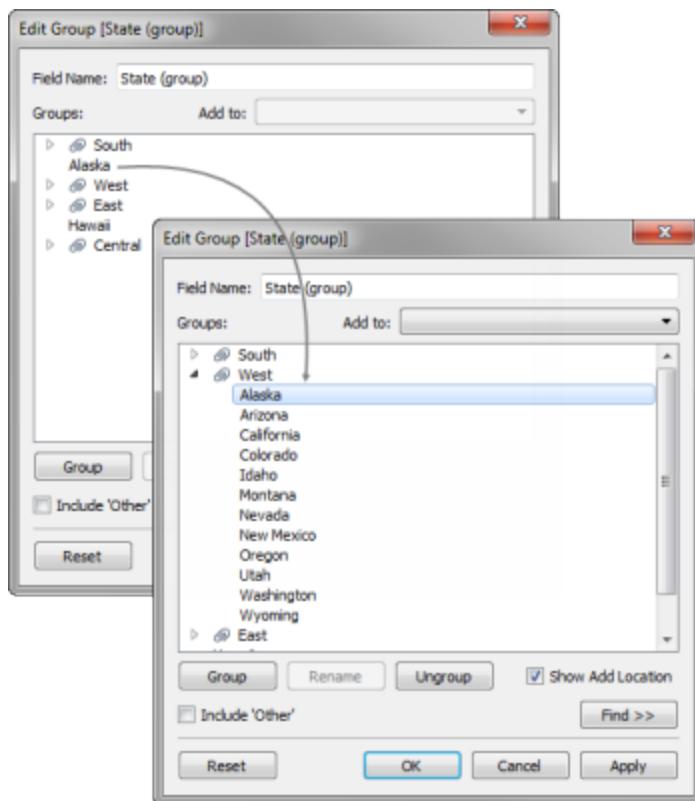
2. In the **Data** pane, select Superstore, and then drag the **Region** dimension to the **Rows** shelf and place it in front of **State**.

The view shows the regions from the Superstore data source (secondary) and the states in the Population data source (primary) that are associated with those regions. There are two states, Alaska and Hawaii, from Population data source (primary) that are not associated with a specific region. These two states are assigned to a "null" region.

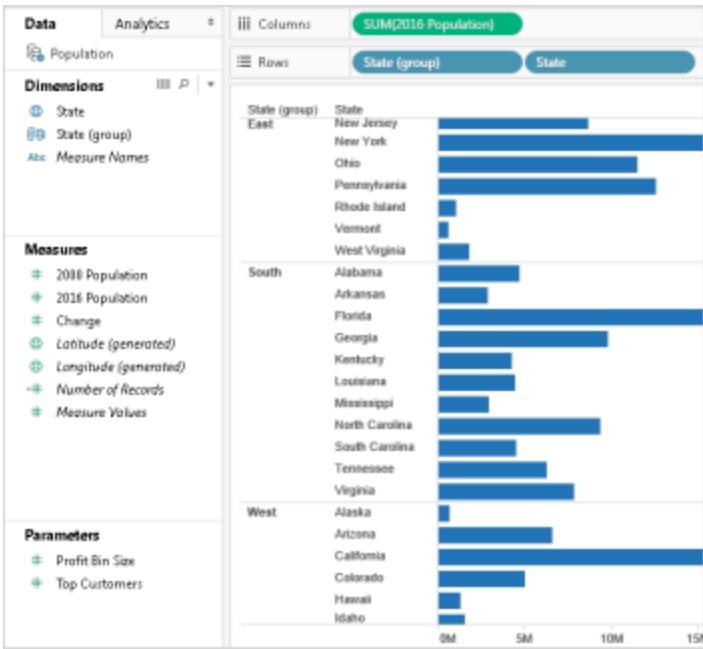
iii Columns

Region	State	
Null	Alaska	Abc
	Hawaii	Abc
Central	Illinois	Abc
	Indiana	Abc
	Iowa	Abc
	Kansas	Abc
	Michigan	Abc
	Minnesota	Abc
	Missouri	Abc
	Nebraska	Abc
	North Dakota	Abc
	Oklahoma	Abc
East	South Dakota	Abc
	Texas	Abc
	Wisconsin	Abc
East	Connecticut	Abc
	Pennsylvania	Abc

3. Right-click the **Region** field in the **Rows** shelf from the Superstore data source (secondary), and select **Create Primary Group**.
4. In the Edit Group dialog box, drag Alaska and Hawaii (from the "null" region) to the **West** region. This task assigns the states in the Population data source (primary) in the "null" region into a region.



5. Click **OK** to close the Edit Group dialog box.
6. In the **Data** pane, select the State data source (secondary). Now you see a new group field listed in the Population data source (primary) called **State (group)**.
7. Drag the **State (group)** dimension over the Region dimension on the **Rows** shelf to replace it.
8. Drag the **2016 Population** measure to the **Columns** shelf to see Population based on the regions from the Superstore (secondary) data source.
9. (Optional) You can close the Superstore data source (secondary), and then continue with your analysis. You can also publish the data source or workbook without the secondary data source. This is because the field you were using from the secondary data source, **Region**, is replaced by the primary group field that you created in the primary data source.



To watch a video that demonstrates another example of the Create a Primary Group functionality, see [Cleaning Data by Bulk Re-Aliasing](#).

Troubleshoot Data Blending

In this topic:

- [Common warnings and errors when blending data sources](#)
 - [Asterisks show in the sheet](#)
 - [Null values appear after blending data sources](#)
 - [Blending issues after publishing data sources](#)
 - [Blending with a cube \(multidimensional\) data source](#)
 - [Other data blending issues](#)
-

When you use data blending to combine your data from different data sources, there are a few common issues that might arise. This topic lists them and describes how you can respond to each situation.

Common warnings and errors when blending data sources

No relationship to the primary data source

When you drag a field from a secondary data source to the view, you might see a warning that says:

Fields cannot be used from the [name of secondary data source] data source, because there is no relationship to the primary data source. In the Data pane, switch to the

[name of secondary data source] data source, and click at least one link icon to blend these data sources.

This warning occurs when you have no active links in the secondary data source

For example, suppose you have two data sources that are related by the **State** and **Date** fields.

At least one of these fields must have the active link icon () next to it in the secondary data source. You can make the link active by clicking the link icon in the **Data** pane or by using the related field from the primary data source in the view.

The secondary data source may not have any relationships to the primary data source. Look in the **Data** pane for the link icon. Tableau automatically links fields that have the same name. If your fields do not have the same name, you must define a relationship between them. For more information, see [Step 4: \(Optional\) Define or edit relationships](#) on page 374.

[Primary and secondary connections are from tables in the same data source](#)

When you drag a field from a secondary data source to the view, you might see a warning that says:

*The primary and secondary connections are from tables in the same data source.
Instead of linking the connections, use the Data menu to join the data. Joins can integrate data from many tables and may improve performance and filtering.*

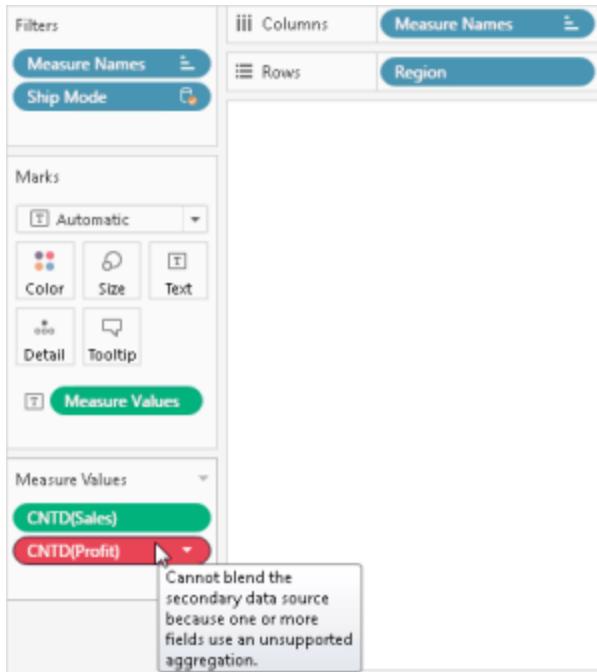
This warning occurs when the workbook contains separate data sources that connect to the same database. Though you can combine data in this way, Tableau recommends that you use a join to combine data from the same database instead. Joins are typically handled by the database, which means that joins leverage some of the database's native performance capabilities.

[Cannot blend the secondary data source because one or more fields use an unsupported aggregation](#)

Data blending has some limitations regarding non-additive aggregates such as COUNTD, MEDIAN, and RAWSQLAGG. Non-additive aggregates are aggregate functions that produce results that cannot be aggregated along a dimension. Instead, the values have to be calculated individually.

These limitations cause certain fields in the view to become invalid under certain circumstances. If you hover your mouse cursor over one of these invalid fields, you see the following error:

Cannot blend the secondary data source because one or more fields use an unsupported aggregation.



This error can occur for one of the following reasons:

- **Groups in the primary data source:** If a non-additive aggregation from the primary data source is in the view, you cannot use a group created in the primary data source. To work around this limitation, convert the group to a calculated field.
- **Non-additive aggregates from the primary data source:** Non-additive aggregates are only supported in the primary data source if the data in the data source comes from a relational database that allows the use of temporary tables. To work around this issue, consider creating an extract of your data source. Extracts support temporary tables.
- **Non-additive aggregates from a multi-connection data source that uses a live connection:** Multi-connection data sources that connect to data using a live connection do not support temporary tables. Therefore, using a multi-connection data source that connects to data using a live connection prohibits the use of blending functionality with non-additive aggregates. To work around this issue, consider creating an extract of your multi-connection data source. Extracts support temporary tables.
- **Linking field in the view before the use of an LOD expression:** This error can also appear when you use a level of detail expression in a view that uses data blending. To resolve the error, make sure the linking field in the primary data source is in the view before you use an LOD expression from the secondary data source.
- **Published data sources as the primary data source:** Because certain versions of Tableau Server does not support temporary tables, there are some limitations around non-additive aggregates.

- For Tableau 8.3 and earlier, Tableau Server does not support temporary tables. Therefore, using a published data source as your primary data source prohibits the use of the blending functionality with non-additive aggregates.
- For Tableau 9.0 and later, you can use COUNTD and MEDIAN with blending functionality in a published data source that is used as the primary data source. However, the other limitations listed above still apply.

Asterisks show in the sheet

When you blend data, make sure that there is only *one* matching value in the secondary data source for each mark in the primary data source. If there are multiple matching values, you see an asterisk in the view that results after you blend data.

For example, suppose you have two data sources: Population and Superstore. The primary data source, Population, has a field called **State**. The secondary data source, Superstore, contains fields called **State** and **Segment**. In the secondary data source, each state has multiple segments. Alabama, for example, has Consumer, Corporate, and Home Office segments.

Primary data source

The screenshot shows the Tableau interface for the primary data source. It includes the Data pane with Population selected, the Analytics pane, Dimensions pane (with State selected), and Measures pane (with 2010 Population selected). The view is set to Pages.

Secondary data source

The screenshot shows the Tableau interface for the secondary data source. It includes the Data pane with Population and Superstore selected, the Analytics pane, Dimensions pane (with Customer and Order selected), and Marks pane (with Automatic selected). The view is set to Columns. A preview table shows State and Segment columns for various states like Alabama, Arizona, Arkansas, California, and Colorado, with multiple segment values listed for each state.

When you blend the two data sources on the **State** field, you create a relationship where individual state values (in the primary data source) can have multiple segment values (in the secondary data source). Multiple values for segments in the secondary data source for each corresponding state value in the primary data source cause asterisks to appear in the view, as demonstrated by the images below.

Avoid this by making sure that there is only *one* matching value in the secondary data source for each mark in the primary data source.

Blended data

Region	State	Segment
Null	Alaska	Null
Null	Hawaii	Null
Central	Illinois	Abc
	Indiana	Abc
	Iowa	Abc
	Kansas	Abc
	Michigan	Abc
	Minnesota	Abc
	Missouri	Abc
	Nebraska	Abc
	North Dakota	Abc
	Oklahoma	Abc
	South Dakota	Abc
	Texas	Abc
	Wisconsin	Abc

Null values appear after blending data sources

Null values can sometimes appear in place of the data you want in the view when you use data blending. Null values can appear for a few reasons:

- The secondary data source does not contain values for the corresponding values in the primary data source.
- The data types of the fields you are blending on are different.
- The values in the primary and secondary data sources use different casing.

Data blending works by supplementing the data in the primary data source with data from the secondary data source based on the linking field. This means Tableau takes all the data in the primary data source, and only the corresponding matches in the secondary data source. If Tableau cannot identify at least one matching value in the secondary data source for the corresponding value in the primary data source, null values appear.

You can resolve this issue by doing the following:

- **Insert data in the secondary data source:** Insert missing data in the secondary data source so that all records in the primary data source have a match. For example, suppose your primary data source contains values for all 50 U.S. states, but your secondary data source only has values for 30 U.S. states. To resolve this issue, you can add values for the remaining 20 U.S. states to the underlying data set of the secondary data source. Alternatively, you might be able to [Bring a Field into the Primary Data Source](#) on page 379 or [Alias Field Values Using Data Blending](#) on page 376 to work around null values if you cannot modify the underlying data.
- **Verify data types in the primary and secondary data sources match:** For example, both data sources should use string data for the fields you are blending on, or both fields should contain numbers. You can't blend fields where one field contains numbers and the other contains strings.

- **Verify the casing of the values in the primary and secondary data sources**
match: If one data source uses a mix of upper- and lowercase names, and the other uses only uppercase, you'll see null values. For more information, see [Nulls Show When Data Blending](#).

Blending issues after publishing data sources

When you make a field-level customization that blends two data sources, the customization is based on one of the data sources, the primary data source. Then, when the primary and secondary data sources are published to Tableau Server or Tableau Online, the primary data source, which contains the customization (i.e., the reference to the secondary data source) no longer works. The customization no longer works because the relationship between the primary and secondary data sources is lost. This is because the relationship between the primary and secondary data sources is defined at the workbook level.

For example, suppose you have two data sources: Store - Main and Store - West. You create a field-level customization, like a calculation, that combines the sales totals from the primary data source, Store - Main, with the sales totals from the secondary data source, Store - West. As soon as you publish the data sources to Tableau Server or Tableau Online, the calculation in Store - Main no longer works. This is because the calculation refers to the secondary data source, Store - West, whose location is now undetermined.

To work around this scenario, do the following:

1. Before creating any field-level customizations, publish each data source first.
2. Connect to each data source.
3. Create a workbook that contains the field-level customizations that you need.
4. Publish the workbook to Tableau Server.

You can use this workbook as a "template" that can be shared and downloaded by you and others.

Blending with a cube (multidimensional) data source

Cube data sources can only be used as the primary data source for blending data in Tableau. They cannot be used as secondary data sources.

Other data blending issues

For other issues that emerge while blending data, see the following links.

Sort is not available

- For issues sorting on a calculated field, see [Sorting by Fields is Unavailable for Data Blended Measures](#).
- For issues with a computed sort, see [Sort Options Not Available from Toolbar When Data Blending](#).

Actions do not behave as expected

- Fields from the secondary data source cannot be added to a URL action. See [Fields from Blended Data Source Unavailable for URL Actions](#).
- Action filters are not behaving as expected. See [Action Filters with Blended Data Not Working as Expected](#).

Unexpected values and field changes

- Invalid fields when using COUNTD, MEDIAN, and RAWSQLLAGG. See [COUNTD Invalid in Published Data Sources When Blending](#).
- Duplicate totals after every date value in the view. See [Issues with Blending on Date Fields](#).
- Latitude and longitude fields are grayed out. See [Error "Invalid field formula" Creating Map](#).
- Underlying data shows different values than blended data. See [Underlying Data Not Consistent with Blended Data](#).

Blend on Summary Data

Data blending is a method for combining data. Data blending works by supplementing the data in the primary data source with the data in the secondary data source.

When one of the data sources that you're working with is large, you can reduce query time by using data blending to combine data in the primary data source with summary data from the secondary (large data source).

For example, suppose you have two data sources: Store - North and Store - South. Store - North is the primary data source and Store - South is the large secondary data source. To reduce query times when working with a large data source like Store - South, you can use data blending to blend only on the summary data of the large secondary data source (Store - South) with the data in the smaller and presumably faster primary data source (Store - North).

To blend on summary data, you can do the following:

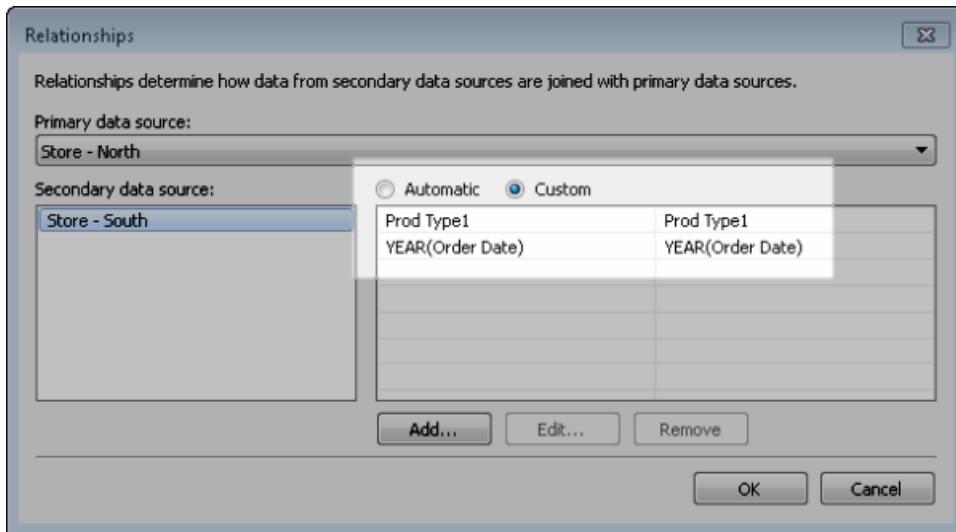
1. Create a relationship between data sources.
2. Bring summary data from the secondary data source into the primary data source.
3. Compare primary data source data to summary data in the secondary data source.

Step 1: Create a relationship between the primary and secondary data sources

1. Download and open the following from Tableau Public.
 1. Go to
<https://public.tableau.com/profile/tableau.docs.team#/vizhome/BlendonSummaryData/Sheet1>.
 2. Click **Download Workbook** in the upper-right corner.

In the workbook, the Order Date dimension is already on the Columns shelf, making it the first field in the view and establishing *Store - North* as the primary data source for the sheet. The *Store - South* data source is also included in the workbook and functions as the large secondary data source.

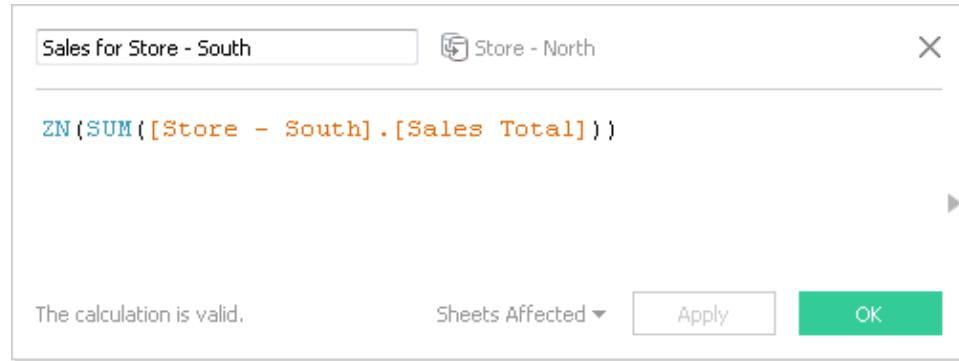
2. Select **Data > Edit Relationships**.
3. In the Relationships dialog box, select **Custom**, and then click **Add**.
4. In the Add/Edit Field Mapping dialog box, in both lists, click the arrow next to Order Date, select **YEAR(Order Date)**, and then click **OK**.
5. In the Relationships dialog box, click **Add** again.
6. In the Add/Edit Field Mapping dialog box, in both lists, click **Prod Type 1**, and then click **OK**.



Step 2: Bring summary data from the secondary data source into the primary data source

1. In the Data pane, select the Store - North data source.
2. Select **Analysis > Create Calculated Field**.
3. In the calculation dialog box, do the following:
 1. In the name text box, enter **Sales for Store - South**.
 2. In the formula text box, type the ZN function, the SUM function, and the Sales Totals field from the Store - South data source.
Your formula should look like the following: `ZN(SUM([Store - South].[Sales Total]))`

This formula pulls in just the sales data from the large secondary data source.



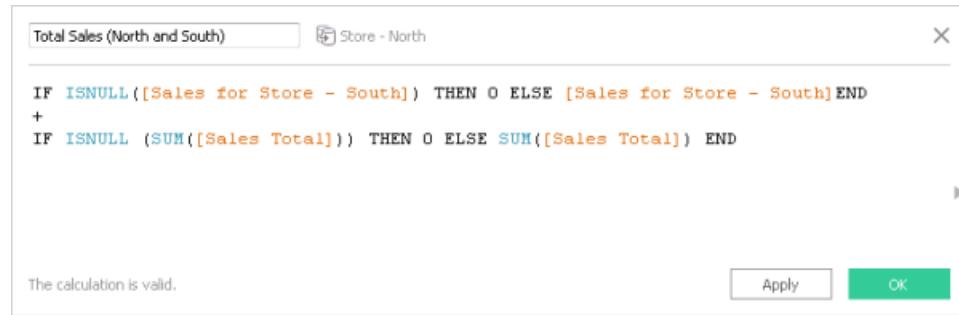
4. Select **Analysis > Create Calculated Field**.

5. In the calculation dialog box, do the following:

1. In the name text box, enter **Total Sales (North and South)**.
2. In the formula text box, type the following:

```
IF ISNULL([Sales for Store - South]) THEN 0 ELSE [Sales  
for Store - South]END  
+  
IF ISNULL (SUM([Sales Total])) THEN 0 ELSE SUM([Sales  
Total]) END
```

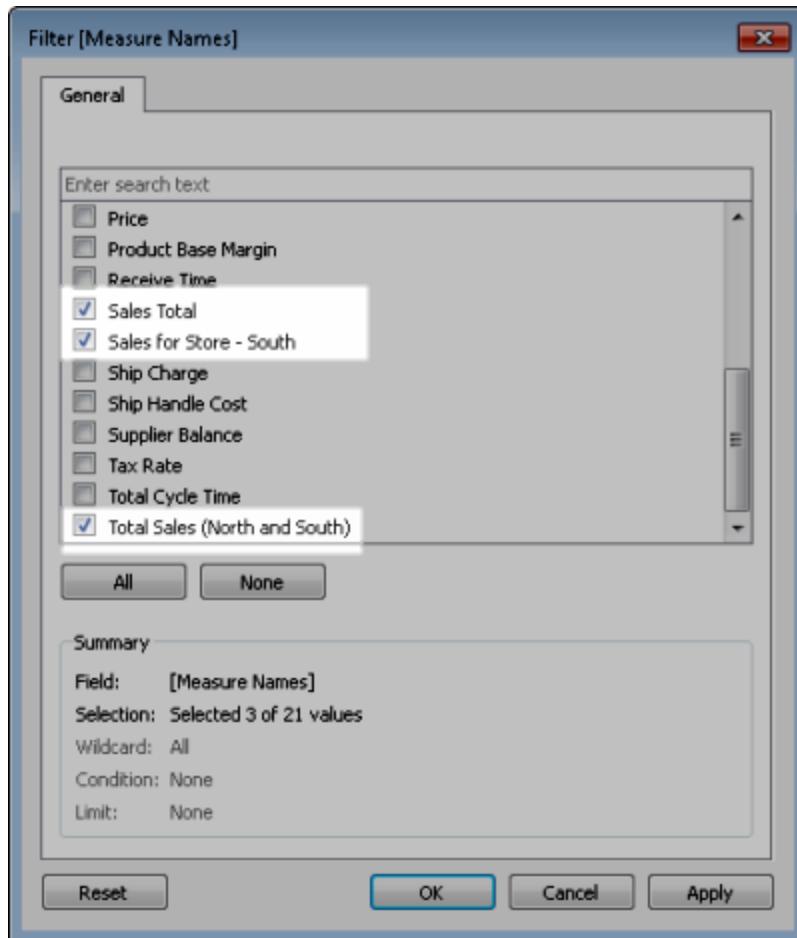
This formula adds the sales data from the large secondary data source to the sales data in the primary data source to get combined sales for Store - South and Store - North.



Step 3: Compare primary data source data to summary data of secondary data source

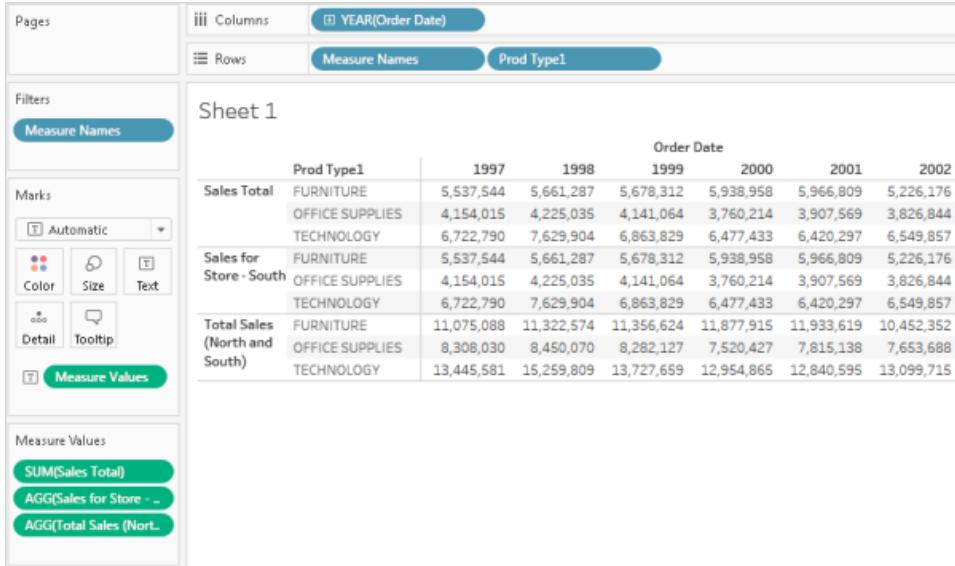
1. From the Data pane, drag **Order Date** to the Columns shelf and **Prod 1** to Rows shelf.
2. Drag **Measure Names** to the Rows shelf, and then to the Filters shelf.
3. In the Filter dialog box, click **None**, and then select **Sales Total, Sales for Store -**

South, Total Sales (North and South) check boxes.



4. From the Data pane, drag **Measure Values** to Text.

The final result is a view that summarizes primary data source and secondary data source data on the Order Year and product category.



Note: The example sales data in both Store - North and Store - South are identical and therefore yield same results for both **Sales Total** and **Sales for Store - South** fields.

Union Your Data

In this topic:

- [Union tables manually](#) on page 394
- [Union tables using wildcard search](#) on page 395
 - [Expand search to find more tables](#) on page 397
- [Rename, modify, or remove unions](#) on page 399
- [Matching field names or field ordering](#) on page 400
- [Metadata about unions](#) on page 401
- [Merge mismatched fields in the union](#) on page 402
- [At a glance: Working with unions](#) on page 403

You can union your data to combine two or more tables from an Excel or Google Sheets workbook or text file data. The workbooks or text files must be in the same folder and come from the same connection.

You union tables by appending values (rows) from one table to another.

For best results, the tables that you combine must have the same structure. That is, each table must have the same number of fields, and related fields must have matching field names and data types.

For example, suppose you have the following customer purchase information stored in three tables, separated by month. The table names are "May2016," "June2016," and "July2016."

May2016

D-day	Customer	Purchases	Type
4	Lane	5	Credit
10	Chris	6	Credit
28	Juan	1	Credit

June2016

D-day	Customer	Purchases	Type
1	Lisa	3	Credit
2-8	Isaac	4	Cash
2-8	Sam	2	Credit

July2016

D-day	Customer	Purchases	Type
2	Mario	2	Credit
1-5	Wei	1	Cash
2-1	Jim	7	Cash

A union of these tables creates the following single table that contains all rows from all tables.

Union

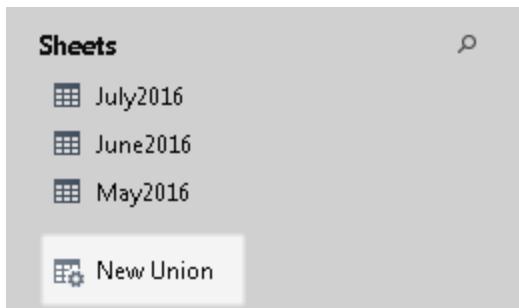
Day	Customer	Purchases	Type
4	Lane	5	Credit
10	Chris	6	Credit
28	Juan	1	Credit
1	Lisa	3	Credit
28	Isaac	4	Cash
28	Sam	2	Credit
2	Mario	2	Credit
15	Wei	1	Cash
21	Jim	7	Cash

Union tables manually

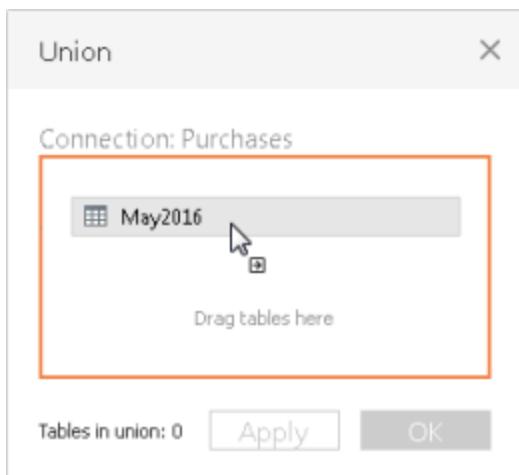
Use this method to manually union distinct text files, Excel sheets, or Google Sheets worksheets. This method allows you to drag individual tables from the left pane of the Data Source page and into the Union dialog box.

To union tables manually

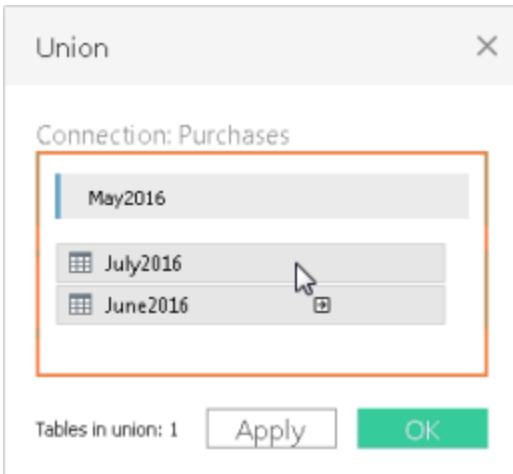
1. On the data source page, double-click **New Union** to set up the union.



2. Drag a table from the left pane to the Union dialog box.



3. Select another table from the left pane and drag it directly below the first table.



Tip: To add multiple tables to a union at the same time, press **Shift** or **Ctrl (Shift)** or **Command** on a Mac, select the tables you want to union in the left pane, and then drag them directly below the first table.

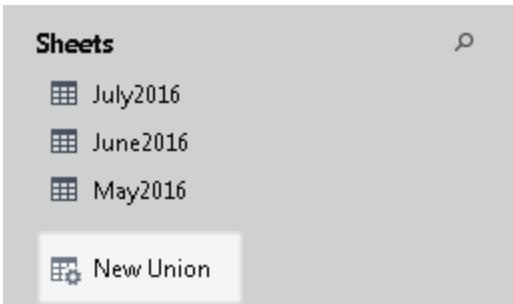
Union tables using wildcard search

Use this method to set up search criteria to automatically include tables in your union. Use the wildcard character, which is an asterisk (*), to match a sequence or pattern of characters in the text file, Google Sheets workbook and worksheet names, or Excel workbook and worksheet names. Use this method to union files across folders, and worksheets across workbooks.

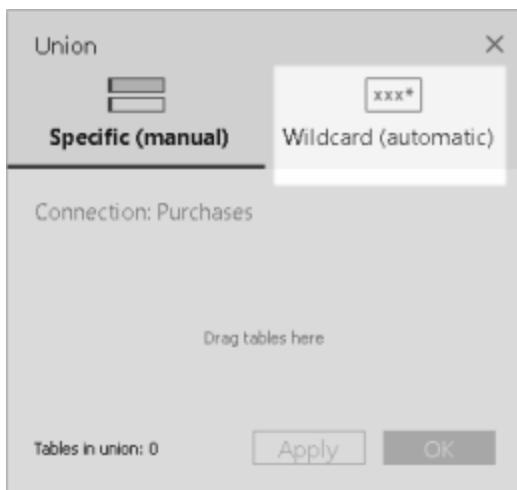
Note: Search is scoped to the selected connection to the text, Google Sheets, or Excel file.

To union tables using wildcard search

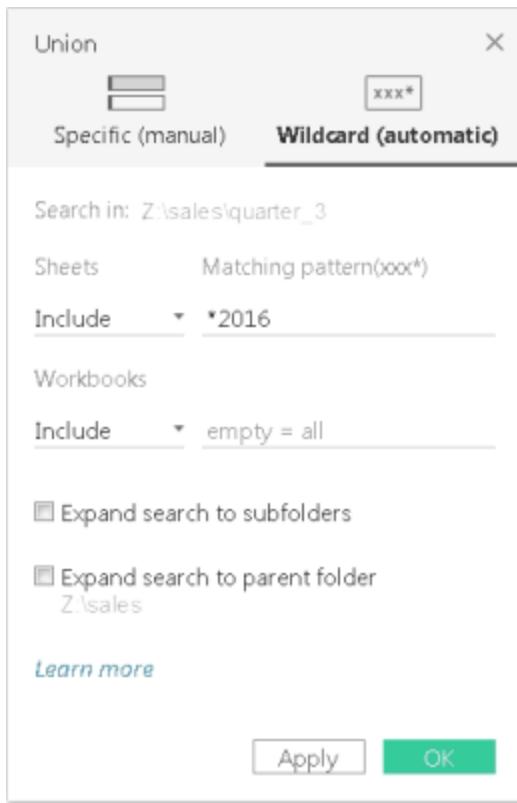
1. On the data source page, double-click **New Union** to set up the union.



2. Click **Wildcard (automatic)** in the Union dialog box.



3. Enter the search criteria that you want Tableau to use to find tables to include in the union.



For example, you can enter ***2016** in the **Include** text box to union tables in Excel worksheets that end with "2016" in their names. Search criteria like this will result in the union of May2016, June2016, and July2016 tables (Excel worksheets), from the selected connection. In this case, the connection is called Sales, and the connection

made to the Excel workbook containing the worksheets you wanted was in the quarter_3 folder in the sales directory (e.g., Z:\sales\quarter_3).

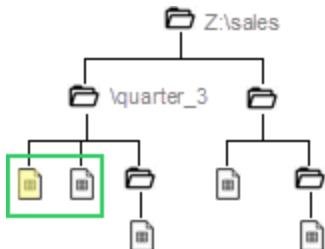
Expand search to find more tables

The tables initially available to union are scoped to the connection you've selected. If you want to union more tables that are located outside of the current folder (for text, Google Sheets, or Excel files) or in a different workbook (for Google Sheets or Excel worksheets), select one or both check boxes in the Union dialog box to expand your search.

For example, suppose you want to union all Excel worksheets that end with "2016" in its name outside of the current folder. The initial connection is made to an Excel workbook located in the same directory in the above example, Z:\sales\quarter_3.

- **Include:** If you enter *2016 in the **Include** text box and leave the remaining search criteria of the dialog as is, Tableau looks for all Excel worksheets that end with "2016" in its name inside the current folder.

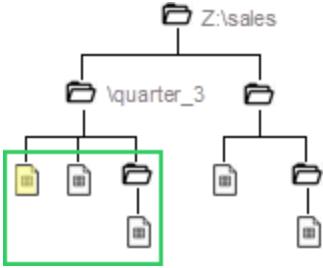
In the diagram below, the yellow highlighted item represents the current location, that is, the Excel workbook that you created a connection to in the "quarter_3". The green box represents the tables belonging to workbooks and sheets that are unioned as result of this search criteria.



- **Include + Expand search to subfolders:** If you enter *2016 in the **Include** text box and select the **Expand search to subfolders** check box, Tableau does the following:

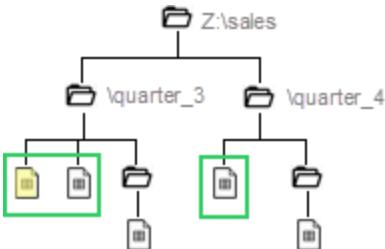
- Looks for all Excel worksheets that end with "2016" in their names inside the current folder.
- Looks for additional Excel worksheets that end with "2016" in their names that are located in Excel workbooks in subfolders of the "quarter_3" folder.

In the diagram below, the yellow highlighted item represents the current location, that is, the Excel workbook that you created a connection to in the "quarter_3" folder. The green box represents the tables belonging to workbooks and worksheets that are unioned as a result of this search criteria.



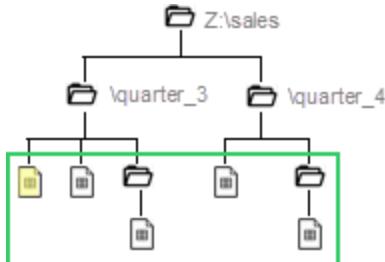
- **Include + Expand search to parent folder:** If you enter *2016 in the **Include** text box and select the **Expand search to parent folder** check box, Tableau does the following:
 - Looks for all Excel worksheets that end with "2016" in their names inside the current folder, "quarter_3."
 - Looks for additional Excel worksheets that end with "2016" in their names that are located in parallel folders of the "quarter_3" folder. In this example, "quarter_4" is the parallel folder.

In the diagram below, the yellow highlighted item represents current location, that is, the Excel workbook that you created a connection to in the "quarter_3" folder. The green boxes represent the tables belonging to the workbook and worksheets that are unioned as a result of this search criteria.



- **Include + Expand search to subfolders + Expand search to parent folder:** If you enter *2016 in the **Include** text box and select both the **Expand search to subfolders** and **Expand search to parent folder** check boxes, Tableau does the following:
 - Looks for all Excel worksheets that end with "2016" in their names inside the current folder, "quarter_3."
 - Looks for additional Excel workbooks that are located in the subfolders of the current folder, "quarter_3."
 - Looks for additional Excel workbooks that are located in parallel folders and subfolders of the "quarter_3" folder. In this example, "quarter_4" is the parallel folder.

In the diagram below, the yellow highlighted item represents the current location, that is, the Excel workbook that you created a connection to. The green box represents the tables belonging to the workbook and worksheets that are unioned as a result of this search criteria.



Note: When working with Excel data, wildcard search includes named ranges but excludes tables found by Data Interpreter.

Rename, modify, or remove unions

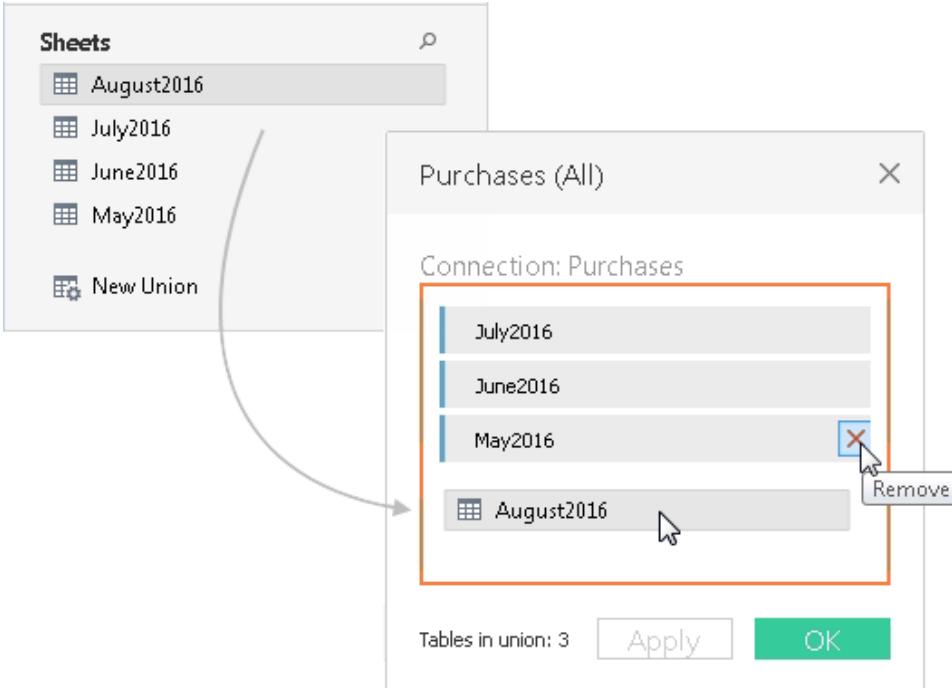
To rename a union

1. Double-click the union table on the canvas.
2. Enter a new name for the union.



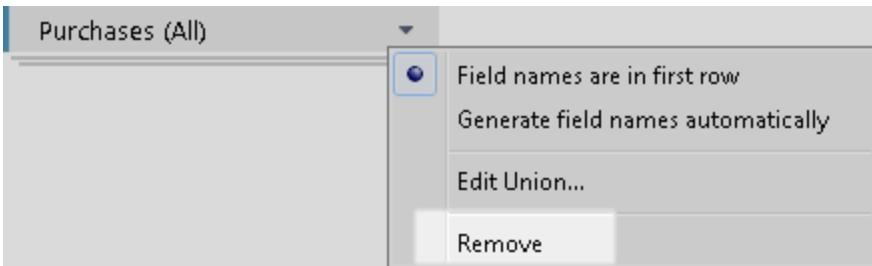
To add or remove tables in the union

1. Click the union drop-down arrow and then select **Edit Union**.
2. You can drag additional tables that you want to union from the left pane, or hover over a table until the remove icon displays and then click the icon to remove the table.



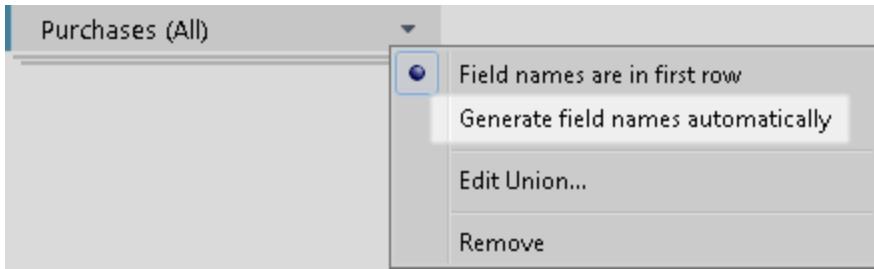
To remove a union

- Click the union drop-down arrow and select **Remove**.



Matching field names or field ordering

Tables in a union are combined by matching field names. If there are no matching field names (or your tables do not contain column headers), you can tell Tableau to combine tables based on the order of the fields in the underlying data by creating the union and then selecting **Generate field names automatically** option from the union drop-down menu.



Metadata about unions

After you create a union, additional fields about the union are generated and added to the grid. The new fields provide information about where the original values in the union come from, including the sheet and table names. These fields are useful when unique information that is critical to your analysis is embedded in the sheet or table name.

For example, the tables used in the example above have unique month and year information stored in the table name instead of in the data itself. In this case, you can use the **Table name** field that is generated by the union to access this information and use it in your analysis.

Abc Purchases (A.I) Customer	Purchases (A.B) Purchases	Abc Purchases (A.B) Type	Abc Purchases (A.B) Path	Abc Purchases (A.B) Sheet name	Abc Purchases (A.B) Table name
Max	2	Credit	Purchases_Metadata.xlsx	July2016	July2016
Wendy	1	Cash	Purchases_Metadata.xlsx	July2016	July2016
Jim	7	Cash	Purchases_Metadata.xlsx	July2016	July2016
Lisa	3	Credit	Purchases_Metadata.xlsx	June2016	June2016
Isaac	4	Cash	Purchases_Metadata.xlsx		
Sam	2	Credit	Purchases_Metadata.xlsx		
Arnold	5	Credit	Purchases_Metadata.xlsx		
Lane	5	Credit	Purchases_Metadata.xlsx		
Chris	6	Credit	Purchases_Metadata.xlsx		
Juan	1	Credit	Purchases_Metadata.xlsx		

Abc Purchases (A.I) Sheet name	Abc Purchases (A.I) Table name
July2016	July2016
July2016	July2016
July2016	July2016
June2016	June2016
... - 2016	... - 2016

If a named range is used in a union, null values display under the **Sheet name** field.

Note: You can use the fields generated by a union, **Sheet name** and **Table name**, as join keys. You can use a unioned table in a join with another table or unioned table.

Merge mismatched fields in the union

When field names in the union do not match, fields in the union contain null values. You can merge the non-matching fields into a single field using the merge option to remove the null values. When you use the merge option, the original fields are replaced by a new field that displays the first non-null value for each row in the non-matching fields.

You can also create your own calculation or modify the underlying Excel, Google Sheets, or text data to combine the non-matching fields.

For example, suppose a fourth table, "August2016", is added to the underlying data. Instead of the standard "Customer" field name, it contains an abbreviated version called "Cust."

August2016

Day	Cust.	Purchases	Type
7	Maria	2	Credit
9	Kathy	1	Credit
18	Vijay	7	Cash

A union of these tables creates a single table that contains all rows from tables, with several null values. You can use the merge option to combine the related customer fields into a single field.

Union (with null values)

Da-y	Cus-tomer	Purchas-es	Typ-e	Cus-t.
4	Lane	5	Cre-dit	null
10	Chris	6	Cre-dit	null
28	Juan	1	Cre-dit	null
1	Lisa	3	Cre-dit	null
28	Isaac	4	Cas-h	null
28	Sam	2	Cre-	null

Union (with columns that have been merged)

Da-y	Purchas-es	Typ-e	Cus-tomer, Cust.
4	5	Cre-dit	Lane
10	6	Cre-dit	Chris
28	1	Cre-dit	Juan
1	3	Cre-dit	Lisa
28	4	Cas-h	Isaac

Da-y	Cus-tomer	Purchas-es	Typ-e	Cus-t.
			dit	
2	Mario	2	Cre-dit	null
15	Wei	1	Cas-h	null
21	Jim	7	Cas-h	null
7	null	2	Cre-dit	Mari-a
9	null	1	Cre-dit	Kat-hy
18	null	7	Cas-h	Vija-y

Da-y	Purchas-es	Typ-e	Cus-tomer, Cust.
		h	
28	2	Cre-dit	Sam
2	2	Cre-dit	Mario
15	1	Cas-h	Wei
21	7	Cas-h	Jim
7	2	Cre-dit	Maria
9	1	Cre-dit	Kathy
18	7	Cas-h	Vijay

After you merge fields, you can use the field generated from the merge in a pivot or split, or use the field as a join key. You can also change the data type of the field generated from a merge.

To merge mismatched fields

1. Select two or more columns in the grid.
2. Click the column drop-down arrow, and then select **Merge mismatched fields**.

To remove a merge

- Click the column drop-down arrow of the merged field and select **Remove merge**.

At a glance: Working with unions

- A unioned table can be used in a join.
- A unioned table can be used in a join with another unioned table.
- The fields generated by a union, **Sheet name** and **Table name**, can be used as the join key.
- When working with Excel data, wildcard search includes named ranges but excludes

tables found by Data Interpreter.

- If a named range is used in a union, null values display under the **Sheet name** field.
- The field generated from a merge can be used in a pivot or split.
- The field generated from a merge can be used as a join key.
- The data type of the field generated from a merge can be changed.
- Union tables from within the same connection. That is, you cannot union tables from different databases.

Connect to a Custom SQL Query

In this topic:

- [Connect to a custom SQL query](#) below
 - [Examples of custom SQL queries](#) on the next page
 - [Edit a custom SQL query](#) on page 410
 - [Use parameters in a custom SQL query](#) on page 411
-

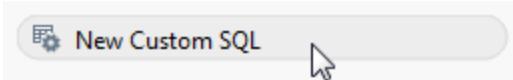
For most databases, you can connect to a specific query rather than the entire data set. Because databases have slightly different SQL syntax from each other, the custom SQL you might use to connect to one database might be different from the custom SQL you might use to connect to another. However, using custom SQL can be useful when you know exactly the information you need and understand how to write SQL queries.

Though there are several common uses for custom SQL, you might use custom SQL to union your data across tables, recast fields to perform cross-database joins, restructure or reduce the size of your data for analysis, etc.

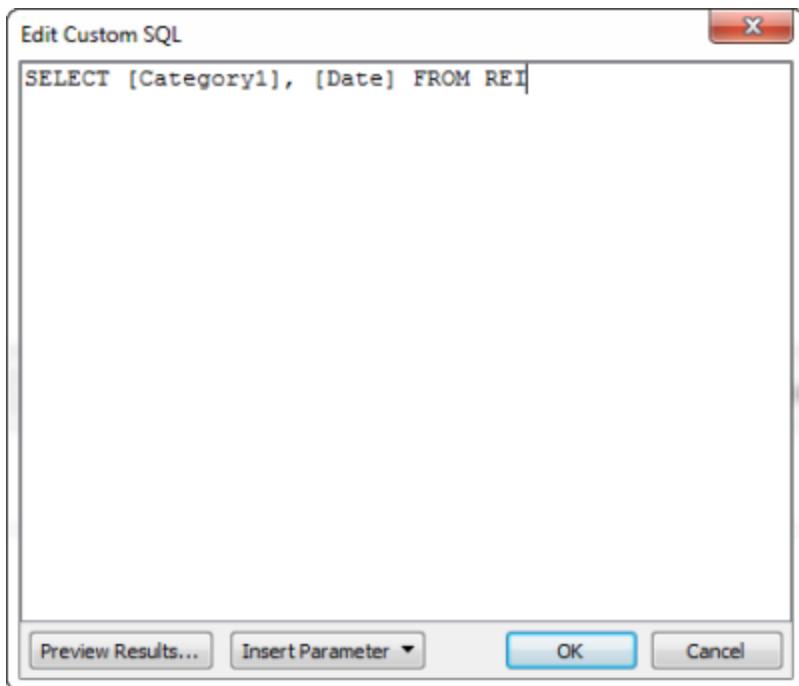
Note: For Excel and text file data sources, this option is available only in workbooks that were created before Tableau Desktop 8.2 or when using Tableau Desktop on Windows with the legacy connection. To connect to Excel or text files using the legacy connection, connect to the file, and in the Open dialog box, click the **Open** drop-down menu, and then select **Open with Legacy Connection**.

Connect to a custom SQL query

1. After connecting to your data, double-click the **New Custom SQL** option on the Data Source page.



2. Type or paste the query into the text box.



3. When finished, click **OK**.

When you finish making the connection to your data, only relevant fields display in the data grid of the Data Source page.

Examples of custom SQL queries

Combine your tables vertically (union)

If you need to append data to each other, you can use the union option directly in Tableau. In some cases your database does not support this option, so you can use custom SQL instead.

For example, suppose you have the following two tables: November and December.

November

Guest ID	Party Size
3005	4
3006	2
3007	2
3008	2

December

Guest ID	Party Size
3009	2
3010	2
3011	4
3012	5
3013	3
3014	4
3015	2

You can use the following custom SQL query to append the second table, December, to the first table, November:

```
SELECT * FROM November UNION ALL SELECT * FROM December
```

The result of the query looks like this in the data grid:

Abc Custom SQL Query Guest ID	# Custom SQL Query Party Size
3005	4.00000
3006	2.00000
3007	2.00000
3008	2.00000
3009	2.00000
3010	2.00000
3011	4.00000
3012	5.00000
3013	3.00000
3014	4.00000
3015	2.00000

For more information about the union option, see [Union Your Data](#) on page 392.

Change data type of a field to do a cross-database join

When you want to perform a join between two tables, the data type of the fields you join on must be the same. In cases when the data type of the fields are not the same, you can use custom SQL to change the data type (cast) the field before performing the join.

For example, suppose you want to join two tables, Main and Sub, using the Root and ID fields, respectively. The Root field is a number type and the ID field is a string type. You can use the following custom SQL to change the data type of Root from a number to a string so that you can join the Main and Sub tables using the Root and ID fields.

```
SELECT [Main].[Root] AS [Root_Number]
CAST([Main].[Root] AS INT) AS [Root_String]
FROM [Main]
```

The result of this query shows the original Root field and the Root field cast as a string.

#	Abc
Root_Number	Custom SQL Query
	Root_String
7,981.00	7981
740.00	740
741.00	741
742.00	742
1,760.00	1760

For more information about joins and cross-database joins, see [Join Your Data on page 354](#).

Reduce the size of your data

When working with very large data sets, sometimes you can save time while working with your data if you reduce its size first.

For example, suppose you have a large table called summary_tests. You can use the following custom SQL query to retrieve the specified columns and records thereby reducing the size of the data set that you connect to from Tableau.

```
SELECT [tests_allup].[id] AS [ID], [tests_allup].[version] AS
[Version], [tests_allup].[type] AS [Type]
FROM [tests_allup]
WHERE [tests_allup].[version] >=9.0.0
GROUPBY [tests_allup].[id]
```

Restructure your data (pivot)

In some cases, you might be working with a table that needs to be restructured before analysis. Though this type of task can be done directly in Tableau by using options like pivot, your database might not support it. In this case, you can use custom SQL instead.

For example, suppose you have the following table:

Season ID	Items - Don't like	Items - Defective	Items - Too small	Items - Too big
R000151493		1	3	20
R000151493		2	44	1
R000151495		0	0	0
R000151495		0	17	1
R000151497		1	0	0
R000151497		0	15	0
R000151789		0	0	0
R000151789		0	12	4
R000151813	3	0	0	3
R000151813	3	0	53	1
R000151815	0	0	0	0
R000151815	0	0	0	0
R000151855	0	0	0	3
R000151855	59	0	22	0
R000151857	4	0	14	18
R000151857	0	0	0	33
R000153013	0	0	21	0
R000153013	9	1	19	0

To change its structure and optimize your data for analysis in Tableau, you can use the following custom SQL query:

```

SELECT Table1.Season ID AS [Season ID],
Table1.Items - Don't like AS [Quantity],
"Don't Like" AS [Reason]
FROM Table1
UNION ALL
SELECT Table1.Season ID AS [Season ID],
Table1.Items - Defective AS [Quantity],
"Defective" AS [Reason]
FROM Table1
UNION ALL
SELECT Table1.Season ID AS [Season ID],
Table1.Items - Too big AS [Quantity],
"Too Big" AS [Reason]
FROM Table1
UNION ALL
SELECT Table1.Season ID AS Season ID,
Table1.Items - Too small AS [Quantity]
"Too Small" AS [Reason]
FROM Table1

```

The result of the query looks like this in the data grid:

Abc Custom SQL Query Season ID	Abc Custom SQL Query Quantity	Abc Custom SQL Query Reason
R000151493	<i>null</i>	Don't Like
R000151493	<i>null</i>	Don't Like
R000151495	<i>null</i>	Don't Like
R000151495	<i>null</i>	Don't Like
R000151497	<i>null</i>	Don't Like
R000151497	<i>null</i>	Don't Like
R000151789	<i>null</i>	Don't Like
R000151789	<i>null</i>	Don't Like
R000151813	3	Don't Like
R000151813	3	Don't Like
R000151815	0	Don't Like
R000151815	0	Don't Like
R000151855	0	Don't Like
R000151855	59	Don't Like

For more information about the pivot option, see [Pivot Data from Columns to Rows](#) on page 423.

Combine (join) and aggregate your data

If you need to combine tables and aggregate your data, you can use both a join and default aggregation type options directly in Tableau. In some cases you might need to use custom SQL instead.

For example, suppose you have the following two tables: Orders and Vendors.

Orders

Order	CustomerID	VendorID
10248	32	1
10249	90	3
10250	81	2

Vendors

VendorID	Name
1	Tminus Shipping
2	Packing You
3	ShipWise Express

You can use the following custom SQL query to find a count on the number of orders and do a left join on the Orders and Vendors tables:

```
SELECT Vendors.Name,COUNT(Orders.Order) AS Number Of Orders
FROM Orders
LEFT JOIN Vendors
ON Orders.VendorID=Vendors.VendorID
GROUP BY Name;
```

The result of the query looks like this:

Abc	#
Custom SQL Query	Custom SQL Query
Name	Number Of Orders
Packing You	1
ShipWise Express	1
Tminus Shipping	1

For more information about joins, see [Join Your Data on page 354](#).

Errors when duplicate columns are referenced

If your custom SQL query references duplicate columns, you may get errors when trying to use one of the columns in your analysis in Tableau. This will happen even if the query is valid. For example, consider the following query:

```
SELECT * FROM authors, titleauthor WHERE authors.au_id =
titleauthor.au_id
```

The query is valid, but the **au_id** field is ambiguous because it exists in both the “authors” table and the “titleauthor” table. Tableau will connect to the query but you will get an error anytime you try to use the **au_id** field. This is because Tableau doesn’t know which table you are referring to.

Edit a custom SQL query

To edit a custom SQL query

1. On the data source page, in the canvas, hover over the custom SQL table until the edit button displays.
2. Click the edit button.
3. In the dialog box, edit the custom SQL query.

Use parameters in a custom SQL query

You can use parameters in a custom SQL query statement to replace a constant value with a dynamic value. You can then update the parameter in the workbook to modify the connection. For example, you may connect to a custom SQL query that provides web traffic data for a particular page that is specified by a pageID. Instead of using a constant value for the pageID value in the SQL query, you can insert a parameter. Then after finishing the connection, you can show a parameter control in the workbook. Use the parameter control to switch out the pageID and pull in data for each page of interest without having to edit or duplicate the connection.

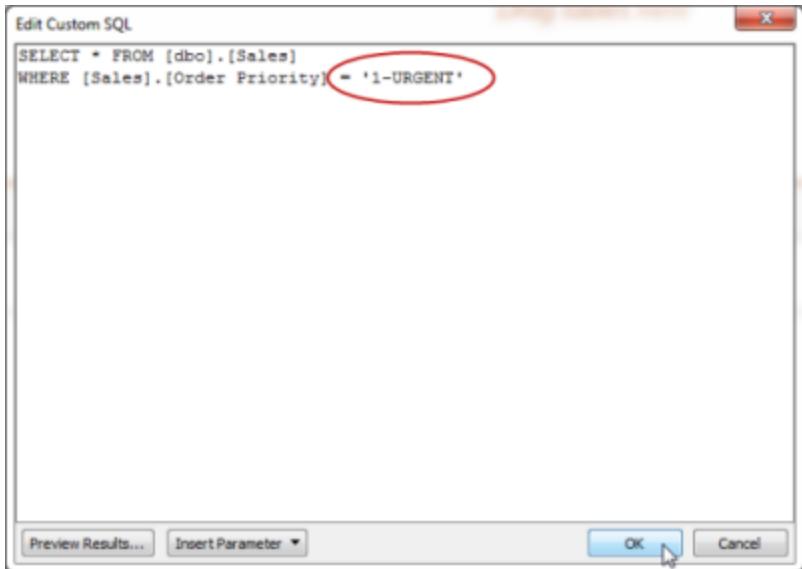
You can create a parameter directly from the Custom SQL dialog box or use any parameters that are part of the workbook. If you create a new parameter, it becomes available for use in the workbook just like any other parameter. See [Parameters on page 989](#) to learn more.

To add a parameter to a custom SQL query

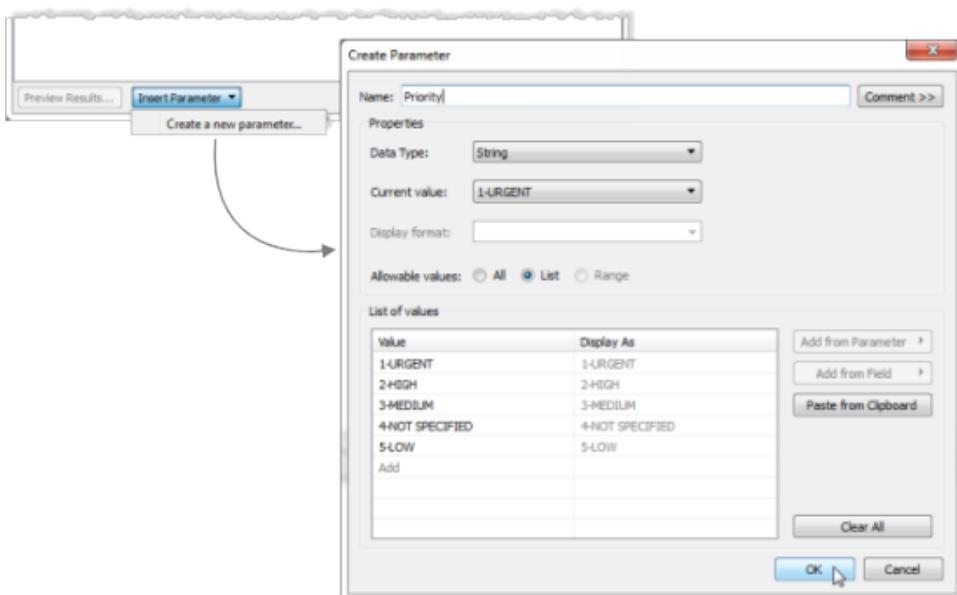
1. On the data source page, in the canvas, hover over the table until the edit icon displays, and then click the edit button.
2. At the bottom of the dialog box, click **Insert Parameter**.
3. Select a constant value in the SQL statement and then, from the **Insert Parameter** drop-down menu select the parameter you want to use instead. If you have not created a parameter yet, select **Create a new parameter**. Follow the instructions in [Creating Parameters on page 989](#) to create a parameter.

Note: Parameters can only replace literal values. They cannot replace expressions or identifiers such as table names.

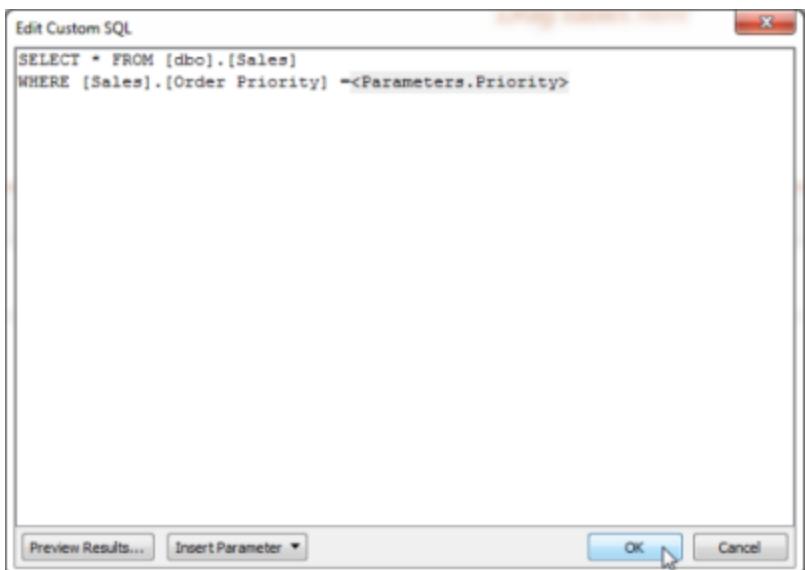
In the example below, the custom SQL query returns all orders that are marked as Urgent priority. In the custom SQL statement, the order priority is the constant value. If you want to change the connection to see the High priority orders, you would have to edit the data source.



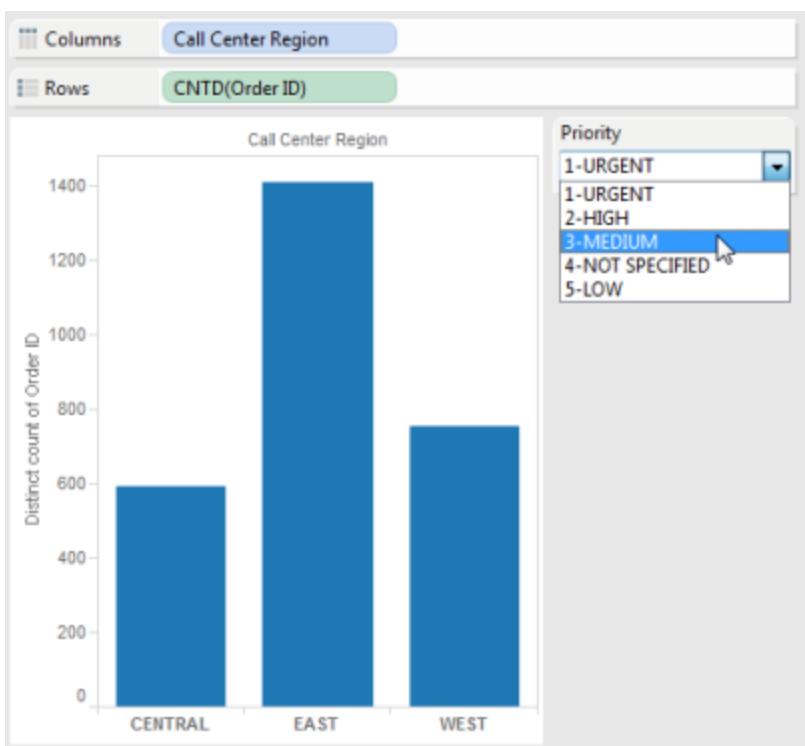
Instead of creating and maintaining many variations of the same query, you can replace the constant order priority value with a parameter. The parameter should contain all of the possible values for Order Priority.



After you create a parameter, you can insert it into the SQL statement to replace the constant value.



After you finish editing the connection, the new parameter is listed in the Parameters area at the bottom of the Data pane and the parameter control displays on the right side of the view. As you select different values, the connection updates.



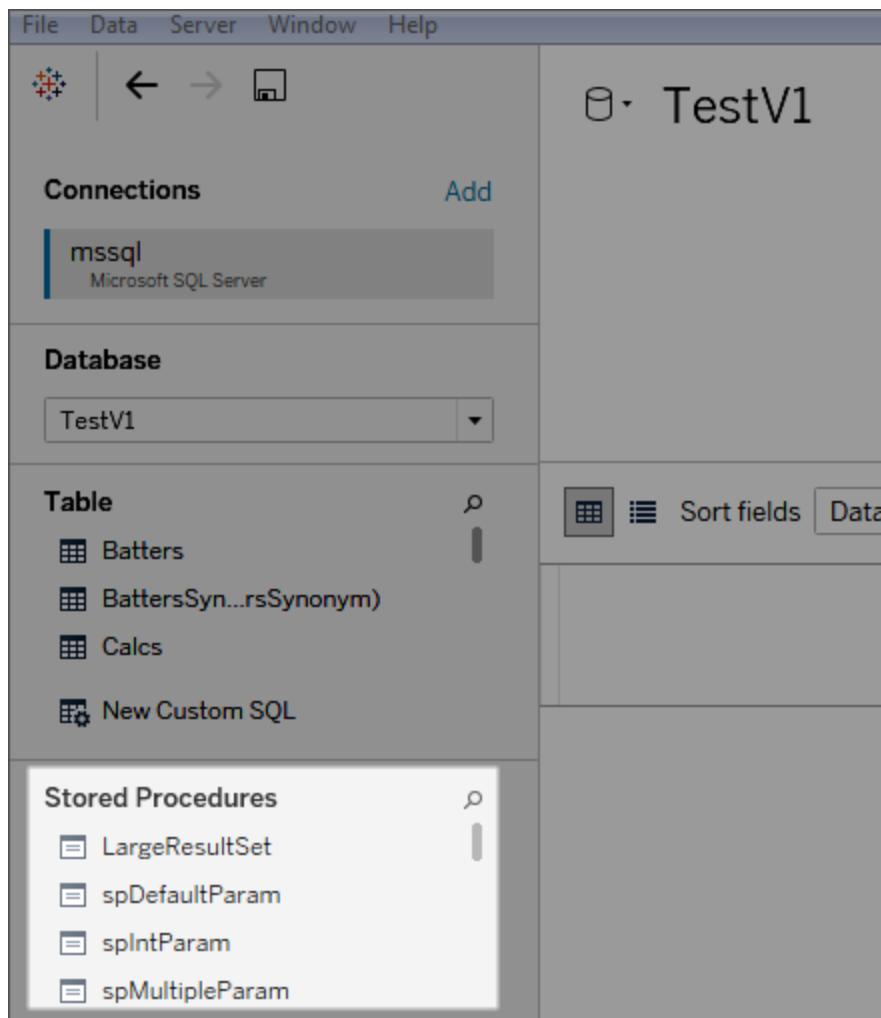
Note: If you are using an extract, you must refresh the extract in order to reflect changes to the parameter. Publishing a data source that uses Custom SQL parameters includes the parameters. The parameters are transferred to any workbooks that connect to the data source.

Use a Stored Procedure

A stored procedure is a subroutine available to applications that access a relational database system. When you connect to a SAP Sybase ASE, Microsoft SQL Server, or Teradata database with Tableau, you can use a stored procedure to define the connection.

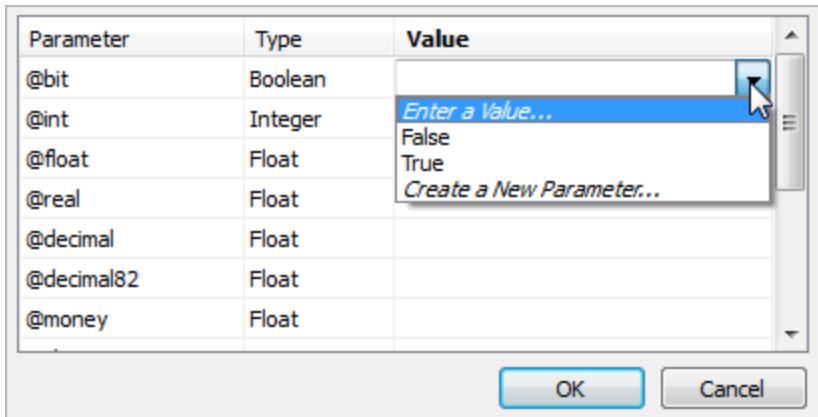
For Oracle, you can use a table function to define the connection. Oracle table functions act similarly to stored procedures and are listed under **Stored Procedures** on the Data Source page.

When you create a data source using one of these data types, available procedures are listed under **Stored Procedures**, as shown in the Microsoft SQL Server example:



From the left pane, drag the procedure to the canvas or double-click one of the listed procedures. If parameters are available in the procedure, the Parameters dialog box automatically displays.

Instead of entering a value, you can use an existing Tableau parameter, or create a new Tableau parameter for the value:



If you then expose the Tableau parameter in the view, users are able to change the value of the parameter in the procedure interactively.

Notes on Stored Procedures

If you use stored procedures to define a data source for Tableau, keep the following in mind:

- If a stored procedure returns more than one result set, Tableau reads the first one and ignores the rest.
- If a stored procedure has output parameters, Tableau filters out the stored procedure.
- Stored procedures that have parameters of a non-scalar type are excluded.
- Result set columns that don't have matching types in Tableau (such as varbinary, geometry, and hierarchyid) are logged. If all result set columns map to unknown data types, Tableau displays a message:
"The result set... has no usable columns."
- Stored procedures that return no result sets are listed on the data source page but fail if selected.
- If no value is provided for a parameter that the stored procedure requires, an error occurs. Tableau cannot determine in advance whether parameters are required.
- Tableau does not perform any transaction management for stored procedures. That is, stored procedure writers must not depend on Tableau to start transactions before invoking stored procedures, or to commit them afterward.
- Column names must be unique for stored procedures to work. If two columns have the same name, or if no name is provided, the procedure can result in an error message:
"InsertData: unbound column error"
- If there are multiple queries in a stored procedure (for example, to read values from another table or to hold temporary combinations) each of the queries must return the same sets of columns in the same order (same names and data types). To ensure that

column order and names match in the query results, you may need to explicitly `CAST` to ensure the data type is correct, for example `CAST (Username as VARCHAR(20))`, and explicitly name the columns. If a stored procedure does not follow these guidelines, an error message can result:

"InsertData: unbound column error"

- If there are multiple queries in a stored procedure (for example, to read values from another table or to hold temporary combinations) and the procedure is generating an error, try adding `SET NOCOUNT ON` to the top of the procedure. This prevents the message which shows the count of number of rows affected by a Transact-SQL statement from being returned as part of the result set for a query.

In addition, the following constraints apply for specific databases.

[Stored Procedure Constraints for Teradata Databases](#)

The following constraints apply for stored procedures on Teradata databases.

- Values must be provided for every parameter. If the user does not provide a value for one or more parameters, Tableau displays a Teradata database error stating there are too few values provided for the stored procedure.

[Stored Procedure Constraints for SQL Server Databases](#)

The following constraints apply for stored procedures on SQL Server databases.

- If the result set for a stored procedure contains columns of type `IMAGE` or `TEXT`, the stored procedure will fail with an "Incorrect syntax" error message.
- If the total width of the result set (number of bytes in each row) exceeds 8060, the stored procedure fails. This can occur with very wide tables (hundreds of columns) or with tables having large text columns, intended to hold thousands of characters of text.
- Tableau does not display stored procedures from schema "sys".
- If the user does not provide a value for one or more parameters that the procedure requires, Tableau displays a SQL Server database error in the form "The procedure requires a value for parameter @x but one was not provided."
- Stored procedures that contain multiple queries should follow the guidelines listed in Notes on Stored Procedures (above).
- Tableau Desktop does not support the Microsoft SQL Server `TIME` data type. When fields of this type are included in a stored procedure on a Microsoft SQL Server database Tableau Desktop will not import them.

[Stored Procedure Constraints for SAP Sybase ASE Databases](#)

The following constraints apply for stored procedures on SAP Sybase ASE databases (Windows only).

- The database must have a properly configured remote server.
- If the user does not provide a value for one or more parameters that the procedure requires, Tableau displays a Sybase ASE database error in the form "The procedure requires a value for parameter @x but one was not provided."

Convert Strings to Date Fields

Dates in your data can be represented in many different ways. And because different databases (and files) do not handle dates in a uniform way, Tableau cannot always interpret the date fields in your data as you might expect.

For example, suppose you are connected to a table in a SQL Server database. The date field in your table contains values that are concatenated like *03Jan2017*, which causes Tableau to interpret it as a string instead of a date or datetime type like you might have expected. In cases like this, you can do one of two things to convert your string values into a date or datetime type.

- **Create a calculation using the DATEPARSE function below**
- **Create a calculation using the DATE function** on page 421

Create a calculation using the DATEPARSE function

The DATEPARSE function converts a string to a datetime type. When you create a calculation that uses the DATEPARSE function, Tableau creates a new field in your Tableau data source that allows you to interact with your date data as a datetime type. To successfully produce date values from a string, the DATEPARSE function requires you to provide two parameters: format and string.

Note: This function is available for non-legacy Excel and text file connections, Google Sheets, MySQL, Oracle, PostgreSQL, and Tableau extracts.

DATEPARSE (format, string)

- **Format:** The *format* parameter is a combination of symbols that represent the format of your date in your data. The syntax for these symbols is defined by the International Components for Unicode (ICU) formatting language. A complete list of the symbols can be found on the Formatting Date and Times Overview page (<http://userguide.icu-project.org/formatparse/datetime>) on the ICU website.

Note: Some date formats are not supported by all databases and file types.

- **String:** The *string* parameter is the date or field in your data that you want to convert to a datetime type.

For example, suppose the table you're working with contains a column of date data that is called "Original Date." The "Original Date" field is a string type.

Original Date
03Jan2017
05Jan2017
07Mar2017
19Mar2017
30Apr2017

In this case, you can create a calculated field called "New Date" that uses the DATEPARSE function to convert the string values in the "Original Date" field into datetime values:

```
DATEPARSE ("ddMMMyyyy" , [Original Date])
```

The "New Date" calculation produces the following column:

New Date
1/3/2017 12:00:00 AM
1/5/2017 12:00:00 AM
3/7/2017 12:00:00 AM
3/19/2017 12:00:00 AM
4/30/2017 12:00:00 AM

In this example, the new date values are based on an English locale and default formatting.

Locale considerations

The DATEPARSE function relies on the locale specified by your computer settings to interpret and then display the strings that you want to convert. More specifically, the locale will affect whether a certain format can be recognized. This means that if a format is not supported by the locale, then you might see a null value or no value returned. For example, suppose you have the following string in your data:

12Sep2016:9:8:8.6546

The values returned from the DATEPARSE function for this string are different based on locale. In the English locale you will get a certain value, but for the Japanese locale you will get no value. In this case, no value is returned because the Japanese locale does not recognize "Sep."

Datetime value

English locale	Japanese locale
#9/12/2016 9:08:09 AM#	-

As with all dates, after you have used the DATEPARSE function to convert a string to a datetime type, by default Tableau will display the datetime value in the default format of your locale. If the locale changes, the result of the DATEPARSE function might display your new datetime value in a different format.

Note: Because of a Jet limitation, the DATEPARSE function cannot correctly identify locale information specified by your computer settings in extracts created from Access data sources. To resolve this issue, consider exporting your Access data to Excel.

Commonly used formats

To get a better idea of how to format your string data, review the table below for examples of the DATEPARSE function for some commonly used date strings. Note: To avoid formatting issues related to locale, use the ISO formatted date literal as a best practice.

Date string	DATEPARSE function	Equivalent date literal in ISO 8061	Display value returned (Display value is based on English locale and default formatting. To change how the new datetime value displays in the view, right-click your DATEPARSE calculation in the Data pane and select Default Properties > Date Format.)
16.8.97	DATEPARSE ("d.M.yy", "16.8.97")	#1997-08-16#	8/16/1997 12:00:00 AM
7/12/91	DATEPARSE ("d/M/yy", "7/12/91")	#1991-12-07#	12/7/1991 12:00:00 AM

Date string	DATEPARSE function	Equivalent date literal in ISO 8061	Display value returned (Display value is based on English locale and default formatting. To change how the new datetime value displays in the view, right-click your DATEPARSE calculation in the Data pane and select Default Properties > Date Format.)
12Sep2012:9:8:8.6546	DATEPARSE ("ddMMMyyyy:H:m:s.SSSSSS", "12Sep2012:9:8:8.6546") Note: Fractional seconds, in excess of three digits, are truncated to the millisecond.	#2012-09-12T09:08:09#	9/12/2012 9:08:09 AM
5:09	DATEPARSE ("h:m", "5:09")	#05:09:00#	5:09:00 AM
97.11	DATEPARSE ("yy.MM", "97.11")	#1997-11#	11/1/1997 12:00:00 AM
9743	DATEPARSE ("yyMd", "9743")	#1997-04-03#	4/3/1997 12:00:00 AM
97-8-14 3:2:1	DATEPARSE ("yy-MM-dd h:m:s", "97-8-14 3:2:1")	#1997-08-14T03:02:01#	8/14/1997 3:02:01 AM
19977345	DATEPARSE ("yyyyMdHm", "19977345")	#1997-07-03T04:05#	7/3/1997 4:05:00 AM

Create a calculation using the DATE function

If the DATEPARSE function is not available for the data that you're working with, you can use the DATE function instead.

The DATE function converts a number, string or date expression to a date type. When you create a calculation that uses the DATE function, Tableau creates a new field in your Tableau data source that allows you to interact with your date data as a date. To successfully produce date values from a number, string, or date expression using the DATE function, Tableau needs to be able to interpret the components of the string into date parts. After the components of the date are identified, Tableau uses the computer locale to determine the default format of the date.

For example, suppose the table you're working with contains a column of date data that is called "Original Date." The "Original Date" field is a string type.

Original Date
03Jan2017
05Jan2017
07Mar2017
19Mar2017
30Apr2017

In this case, you can create a calculated field called "New Date" that uses an expression in a DATE function to convert the string values in the "Original Date" field into date values.

You can create a calculated field called "New Date" that uses a date expression inside a DATE function to convert the string values in the "Original Date" field into date values. For this example, the date expression is comprised of the LEFT function to isolate the day component, the MID function to isolate the month component, and the RIGHT function to isolate the year component.

```
DATE (LEFT([Original Date], 2) + "/" + MID([Original Date],3,3) +
      "/" + RIGHT([Original Date],4))
```

The "New Date" calculation produces the following column:

New Date
1/3/2017
1/5/2017
3/7/2017
3/19/2017
4/30/2017

In this example, the new date values are based on an English locale and default formatting.

Pivot Data from Columns to Rows

You can pivot the data in your Microsoft Excel, text file, and Google Sheets data sources from crosstab format into columnar format to more easily perform analysis.

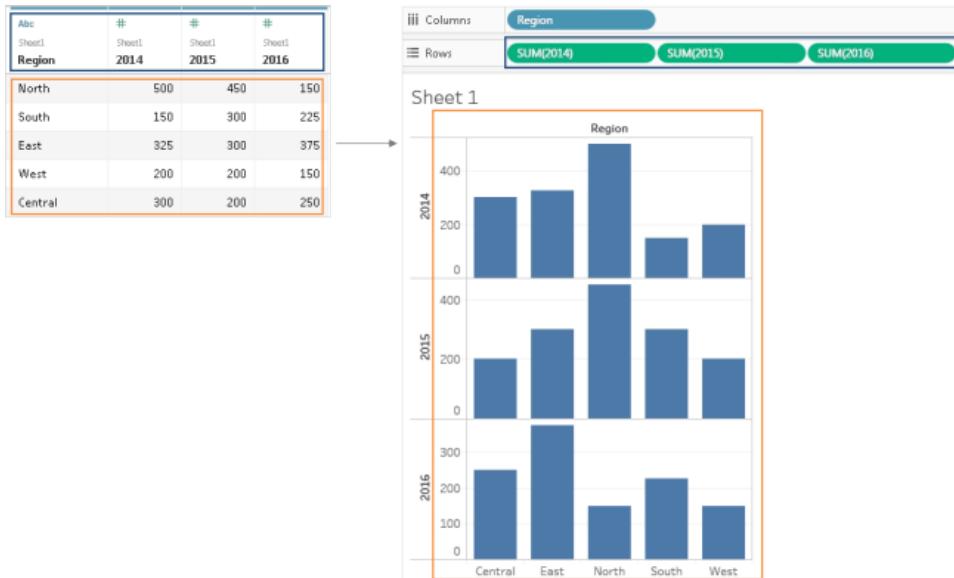
Or, if you are not working with the types of data listed, you can use custom SQL as an alternative way to pivot your data in Tableau.

Pivot Excel, text, and Google Sheets data

For example, suppose your data source contains sales values by year:

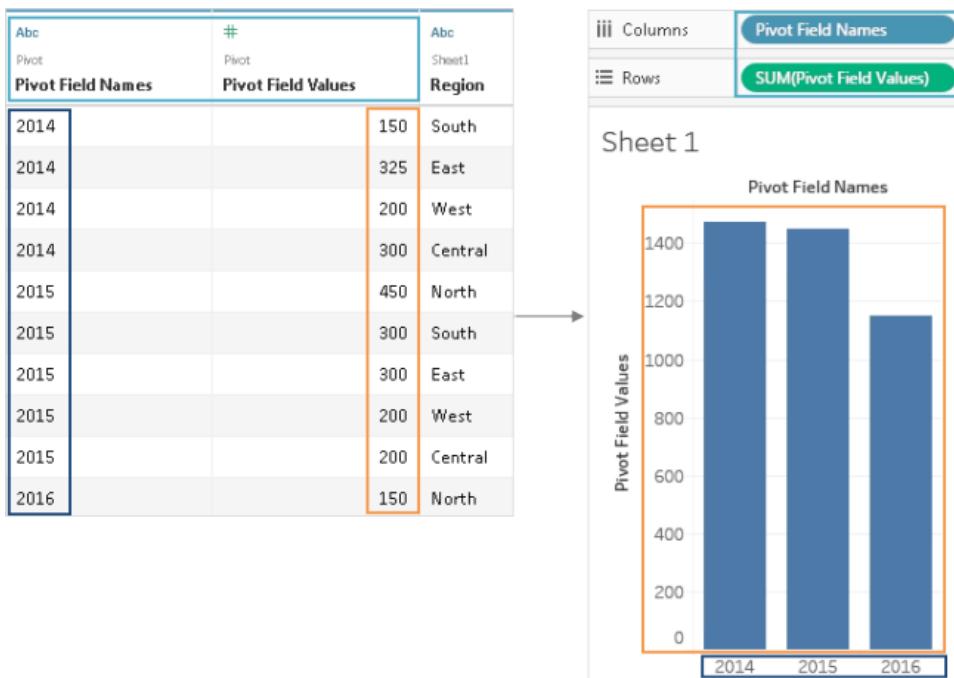
Region	2014	2015	2016
North	500	450	150
East	150	300	225
South	325	300	375
West	200	200	150
Central	300	200	250

The following image shows the example data source in the grid (left) and a view that you can create with the example data source (right).



Although you see a breakdown of sales by year, suppose you want to show all sales values by year, irrespective of region. To do this with the example data source, use pivot to change the format of the existing fields so that all year values are contained in one field, and all sales values are contained in another.

The following image shows the example data source in the grid after using pivot (left) and a view that you can create with the pivot fields (right).



To pivot fields

1. On the data source page, in the grid, select two or more fields to pivot.
2. Click the drop-down arrow next to the field name, and then select **Pivot**.

Region	2014	2015	2016
North	500	450	1
South	150	300	2
East	325	300	3
West	200	200	1
Central	300	200	250

The original fields in the data source are replaced with new fields called “Pivot field names” and “Pivot field values.” You can always rename the new pivot fields. If you decide that using pivot does not help, you can undo the changes or remove the pivot. Alternatively, you can change the data type of the pivot field to adjust how the data is interpreted.

To add more fields to a pivot

1. On the data source page, in the grid, click the drop-down arrow next to the field name.
2. Select **Add Data to Pivot**.

To remove a pivot

1. On the data source page, in the grid, click the drop-down arrow next to the field name.
2. Select **Remove Pivot**. All new pivot fields are removed and the original fields are restored.

At a glance: Working with pivots

- The Pivot option is available from the grid and metadata grid.
- All fields in the pivot must be from the same connection.
- Only one pivot is allowed per data source.
- Pivot fields can be used as the join key.
- The Pivot option cannot be used in calculated fields.

Troubleshooting pivots

- **Red fields in the view and fields with exclamation points in the Data pane:** Because the original fields are replaced with new pivot fields, any references to the original fields in the view will no longer work. They cause fields to become red in the view or show a red exclamation point next to the field in the **Data** pane.
- **Null values in the grid:** If all of the original fields used in the pivot are removed, for example in an extract refresh, null values display in the pivot fields.

Pivot using custom SQL

You can also use custom SQL to pivot your data. When you use the UNION ALL operator in a custom SQL query, you can take values from distinct columns and put them into a new column.

For example, suppose you have a table called **Contest**.

Contest

Runner	Start Time	End Time
Amanda	9/3/2016 3:04 PM	9/3/2016 3:25 PM
Oscar	9/3/2016 3:04 PM	9/3/2016 3:21 PM
William	9/3/2016 3:04 PM	9/3/2016 3:16 PM

To optimize your analysis of this data in Tableau, you can use the following custom SQL query to pivot the "Start Time" and "End Time" columns so that their values are in a single column.

```
Select [Runner]
, 'Start' as [Action]
, [Start Time] as [Time]
From [Contest]
Union ALL
Select [Runner]
, 'End' as [Action]
, [End Time] as [Time]
From [Contest]
```

The above custom SQL query does the following:

- Pivots the **Start Time** column header into a string value called **Start** and adds that value to a new column called **Action**.
- Pivots the **End Time** column header into a string value called **End** and adds that value to a new column called **Action**.
- Pivots the **Start Time** and **End Time** columns so that their values are in a new column called **Time**.

The following table shows the results of this custom SQL query.

Runner	Action	Time
Amanda	Start	9/3/2016 3:04 PM
Oscar	Start	9/3/2016 3:04 PM
William	Start	9/3/2016 3:04 PM
Amanda	End	9/3/2016 3:25 PM
Oscar	End	9/3/2016 3:21 PM
William	End	9/3/2016 3:16 PM

To pivot data using custom SQL

1. Connect to your data.
2. Double-click the **New Custom SQL** option in the left pane. For more information, see [Connect to a Custom SQL Query on page 404](#).
3. In the **Edit Custom SQL** dialog box, copy and paste the following custom SQL query and replace the contents with information about your table:

```
Select [Static Column]  
  
, 'New Value (from Column Header 1)' as [New Column Header]  
  
, [Pivot Column Values 1] as [New Values]  
  
From [Table]  
  
Union ALL  
  
Select [Static Column]  
  
, 'New Value (from Column Header 2)' as [New Column Header]  
  
, [Pivot Column Values 2] as [New Values]  
  
From [Table]  
  
Union ALL  
  
Select [Static Column]  
  
, 'New Value (from Column Header 3)' as [New Column Header]  
  
, [Pivot Column Values 3] as [New Values]  
  
From [Table]
```

Where the following is true:

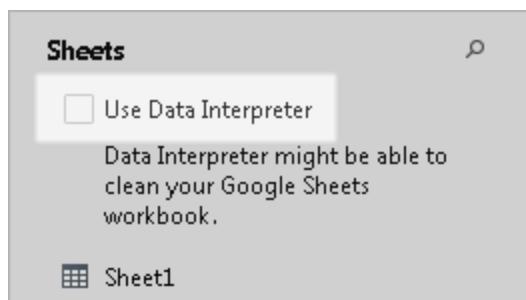
- **Static Column:** A comma-delimited list of the columns from *Table*, both dimensions and measures, which should not be included in the pivot.
- **New Value (from Column Header 1-3):** New names that you give to the original column headers, which are used as row values in the pivot.
- **Pivot Column Values 1-3:** The columns whose values need to be pivoted into a single column.
- **New Column Header:** The name you give the new column that contains the new row values from *New Value (from Column Header 1-3)*.

- *New Values*: The name given to the new column that contains the original values from *Pivot Column Values 1-3*.
 - *Table*: The table that you connected to.
4. Click **OK**.

Clean Your Data with Data Interpreter

Sometimes, the format of your Google Sheets or Microsoft Excel data makes it difficult to analyze in Tableau. For example, your data might include additional tables, sub-tables, hierarchical headers, extraneous headers and footers, or empty rows and columns. Data Interpreter detects these sub-tables so that you can work with a subset of your data independently of the other data. It also removes the extraneous information to help prepare your data source for analysis.

After you set up the data source, if Tableau detects sub-tables, unique formatting, or that the data contains some extraneous information, it prompts you to use Data Interpreter.



Note: When you clean your data with Data Interpreter, Data Interpreter cleans all the data associated with a connection in the data source. Data Interpreter does not change the underlying data.

Turn on Data Interpreter and review results

1. After you have connected to your data and set up your data source, on the data source page, select the **Clean with Data Interpreter** check box.
2. Click **Review data**.

A copy of your data source opens in Excel on the **Key for the Data Interpreter** tab.

3. Review the annotation key to find out how to read the results.
4. Click the subsequent tabs to review how Data Interpreter interpreted the data source.

You can also review Data Interpreter results directly in the grid below Data Interpreter.

If Data Interpreter does not provide the expected results, you can clear the **Clean with Data Interpreter** check box to use the original data source.

5. If Data Interpreter detects additional tables in your data, you can replace the current table with the found table by dragging it to the canvas.

If Data interpreter has misidentified the range of the found table, click the table dropdown arrow on the canvas, and then select **Edit Found Table** to adjust the corners of the found table (the top-left cell and bottom-right cell of the table).

When Data Interpreter is not available

The Data Interpreter option might not be available for the following reasons:

- **The data source is already in a format that Tableau can interpret:** If Tableau Desktop doesn't need extra help from Data Interpreter to handle unique formatting or extraneous information, Data Interpreter option is not available.
- **Many rows or many columns:** Data Interpreter option is not be available when your data has the following attributes:
 - Data contains more than 2000 columns.
 - Data contains more than 3000 rows and more than 150 columns.
- **The data source is not supported:** Data Interpreter is only available for Google Sheets and Excel data sources. Your Excel data must be in the XLS and XLSX formats. Excel data in CSV formats is not supported.

Split a Field into Multiple Fields

If you have string fields in your data that contain multiple units of information, for example, the first and last name of a customer, it might be easier to analyze the data if you split the values in that field into separate fields. You can use split or custom split options in Tableau to separate the values based on a separator or a repeated pattern of values present in each row of the field. In this example, the common separator is a space character ().

The diagram illustrates a data transformation process. On the left, a data source page shows a single column named "Customer Name". This column contains eight rows with values: Claire Gute, Claire Gute, Darrin Van Huff, Sean O'Donnell, Sean O'Donnell, Brosina Hoffman, Brosina Hoffman, and Brosina Hoffman. An arrow points from this source to a second table on the right. The second table, titled "Customer Name - ...", has two columns: "First Name" and "Last Name". It contains the same eight rows as the source, with the values split into two fields: "Claire" and "Gute" for the first two rows, "Darrin" and "Van Huff" for the third, and "Sean" and "O'Donnell" for the fourth. The last four rows show "Brosina" and "Hoffman" respectively.

	=Abc	=Abc
	Calculation	Calculation
	Customer Name - ...	Customer Name - ...
Claire Gute	Claire	Gute
Claire Gute	Claire	Gute
Darrin Van Huff	Darrin	Van Huff
Sean O'Donnell	Sean	O'Donnell
Sean O'Donnell	Sean	O'Donnell
Brosina Hoffman	Brosina	Hoffman
Brosina Hoffman	Brosina	Hoffman
Brosina Hoffman	Brosina	Hoffman

The new fields created by the split or custom split are added to the data source as calculated fields.

Splits and custom splits are based on the `SPLIT` string function. For more information, see [String Functions on page 1560](#).

You can tell if your data supports the `SPLIT` function by checking the field drop-down menu:

- On the data source page, check the menu for **Split** and **Custom Split**.
- On the Data pane in the worksheet, check the menu for **Transform > Split and Custom Split**.

Because different connection types support different functionality in Tableau, you may need to check both places to determine if your data supports the split and custom split options.

Split fields automatically

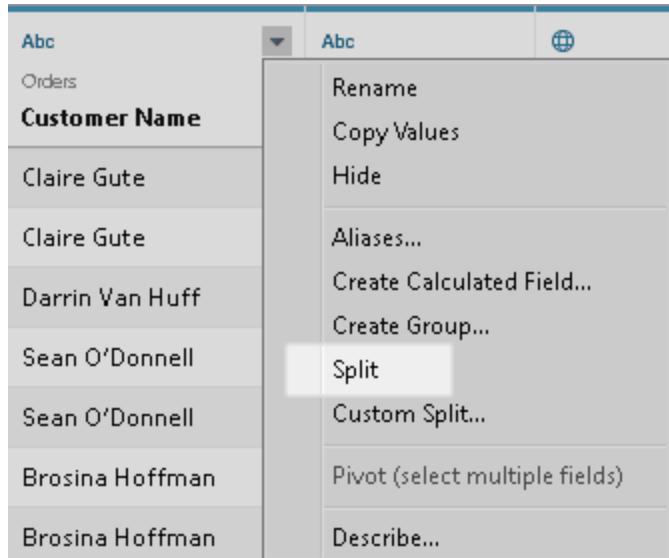
A string field can be split automatically based on a common separator that Tableau detects in the field.

Depending on the connection type, a split can automatically separate a field's values in up to ten new fields. The data type of the new fields generated by the split can vary depending on the pattern combination that Tableau detects in the separator that is used to split the original field.

Note: In some cases, if the split takes too long to generate new fields or Tableau cannot find a common separator, a custom split dialog box displays. For more information, see the [Custom split on page 432](#) section.

To split a field automatically

1. On the Data Source page, in the grid, click the drop-down arrow next to the field name.
2. Select **Split**.



Note: You can also use the split option from the Data pane in the worksheet. In the Data pane, right-click the field you want to split, and then select **Transform > Split**.

If you do not like the results of the split, you can go to the Data pane and edit the calculated fields that are created by the split. Alternatively, you can click undo in the Tableau Desktop toolbar or remove the split.

To remove fields created by a split

1. On the Data Source page, in the grid, click the drop-down arrow next to the field name.
2. Select **Delete**.
3. Repeat steps 1-2 to remove all the fields created by the split.

Alternatives to automatic split

Sometimes, using automatic split is not the best option. The following are example of cases when you should not split fields automatically:

- **Values contain different number of separators:** Fields cannot be split automatically if the number of separators varies from value to value. For example, suppose a field has the following values:

jsmith| accounting | north

dnguyen | humanresources

lscott | recruiting| west

karnold |recruiting |west

In cases like this, consider using a custom split. For more information, see the [Custom split](#) below section.

- **Values contain mixed separators:** Fields cannot be split automatically if the separator types are different. For example, suppose a field contains the following values:

smith.accounting

dnguyen-humanresources

lscott_recruiting

karnold_recruiting

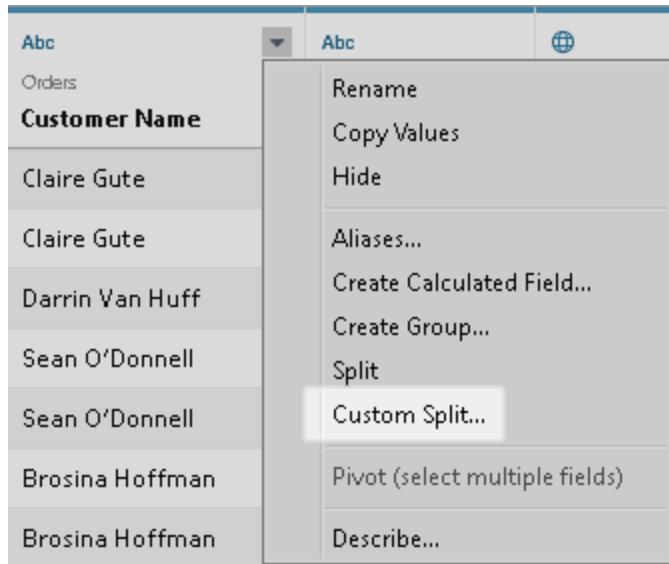
In cases like this, consider using regular expressions to create new fields. For more information, see [Additional Functions](#) on page 1603.

Custom split

You can use the custom split option to specify a common separator for the split. Like the split option, a custom split can separate a field's values in up to ten new fields. In addition, you can choose to split the values at the first n occurrences of the separator, the last n occurrences of the separator, or at all occurrences of the separator. The data type of the new fields generated by the custom split always results in a string data type.

To use a custom split

1. On the Data Source page, in the grid, click the drop-down arrow next to the field name.
2. Select **Custom Split**.



Note: You can also access the custom split option from the Data pane. In the Data pane, right-click the field you want to split, and then select **Transform > Custom Split**.

If you do not like the results of the split, you can go to the Data pane and edit the calculated fields that are created by the split. Alternatively, you can click undo in the toolbar or remove the split.

To remove fields created by a custom split

1. On the Data Source page, in the grid, click the drop-down arrow next to the field name.
2. Select **Delete**.
3. Repeat steps 1-2 to remove all the fields created by the custom split.

At a glance: Working with splits and custom splits

The following is a list of additional notes you might need to know about splits and custom splits.

- New fields generated from a split or custom split cannot be used as keys to join tables, but can be used to blend data sources.
- New fields generated from a split or custom split cannot be used in a pivot.
- Split and custom split options are not supported for sets, groups, parameters, and bins.
- Microsoft SQL Server only allows up to four split fields.
- To generate more than ten new fields, consider using a split or custom split on the field that was previously generated by a split or custom split.

Troubleshooting splits and custom splits

The following is a list of issues you might experience when using splits and custom splits:

- **Split and custom split options missing for a supported data source type:** Split and custom split options are available only for fields that are a string data type.
- **Null values or empty cells:** After creating a split or a custom split, new fields might contain null values or no values at all. Null values or empty cells occur when there are no values for all of the expected new fields.
- **Data has been removed:** Tableau might use portions of the field's values as a separator. If a portion of a field's values is used as a separator, those values no longer appear in the new fields. For example, suppose a field contains the following values:

ZIP-98102

ZIP-98109

ZIP-98119

ZIP-98195

In this case, the split will create a new field with the following values:

98102

98109

98119

98195

The split will not create a separate field for "ZIP-" because the split uses it as a separator.

Filter Data from Data Sources

You can create filters on a data source, thereby reducing the amount of data in the data source.

If you create an extract from a data source that already has data source filters in place, those filters are automatically recommended as extract filters, and will appear in the Extract dialog. Those recommended filters are not required to be part of the Extract filter list, and can safely be removed without affecting the existing set of data source filters.

Data source filters can be useful for restricting the data users can see when you publish a workbook or data source. When you publish a data source to Tableau Server, the data source and any associated files or extracts are transported in entirety to the Server. As you publish a data source you can define access permissions for downloading or modifying the data source, and you can also choose the users and groups who can remotely issue queries through

Tableau Server against that data source. When users have query permission and no download permission, you can share a rich data model having calculated fields, aliases, groups, sets and more—but only for querying.

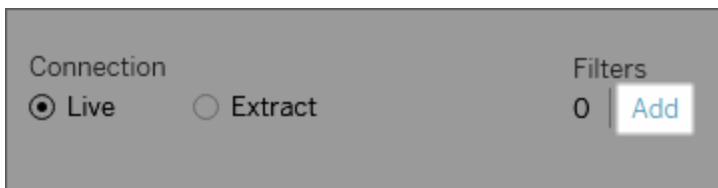
Furthermore, users who query published data source will never be able to see or modify any data source filters present on the originally published data source, but all of the users' queries will be subject to those data source filters. This is a great way to offer a restricted subset of your data, for example by filtering dimensions for specific users and groups, or by defining data source filters based on a fixed or relative date range. This is often useful for data security, and it also allows you to manage performance of the remote database which Tableau Server will ultimately query on a user's behalf. For systems that rely heavily on partitions or indexing, data source filters may yield tremendous control over the performance of queries issued by Tableau.

Create a data source filter

The primary way to create a data source filter is from the data source page.

To create a data source filter

1. On the data source page, click **Add** in the Filters section in the upper-right corner of the page.



To create a data source filter on a worksheet, right-click (control-click on a Mac) the data source and choose **Edit Data Source Filters**.

Whether you start from the Data Source page or from a worksheet, you see an Edit Data Source Filter dialog box, listing any existing data source filters.

2. Click **Add** to open an Add Filter dialog box listing all fields in the data source.
3. Click to select a field to filter; then specify how the field should be filtered, just as you would for a field on the Filters shelf.

To add an additional data source filter, repeat this procedure.

Understand Field Type Detection and Naming Improvements

Tableau uses a collection of commonly used patterns to detect and transform your data to make it easier for you to interact with. For example, when you work with fields in the Data pane, Tableau takes the data from your data source, detects its type, and divides it into dimensions and measures.

Tableau can also detect fields that should be treated as dimensions and clean up field names by making them more readable. You see these improvements only when the field names, which come from column headers in your data source, meet the patterns or conditions described below.

Note: Tableau never changes your underlying data.

If you prefer to maintain the field types and field names from the underlying data, see the [Undo changes made by Tableau](#) section at the bottom of this article.

Fields that are treated as a dimension

Field names that contain certain keywords are treated as dimensions, even if the values for those fields are numeric.

Keywords Code, Key, and ID

Field names that contain the following keywords and meet the conditions listed are treated as dimensions instead of measures.

Conditions:

- Contains keywords Code, ID, or Key.
- Keywords are either separated from other text in the field name by non-letter characters, all capitalized, or the first letter is capitalized in a field name that otherwise has mixed casing.
- Keywords are at the beginning or end of the field name, with leading or trailing non-letter characters. In traditional Chinese, Japanese, and Korean, the key word must be located at the end of the field name.

Language	Keywords
English	Code, Id, Key
Chinese (Traditional)	Id
French	Cle, Clé, Identificação, Identificador
German	Chiffre, Kennung, Kennnummer, Id, Identität, Schlüssel
Japanese	Id
Korean	Id
Portuguese	Chave, ID
Spanish	Clave, Id, Identificación

Keywords Number, Num, and Nbr

Field names that end with Number, Num, or Nbr are treated as a dimension instead of a measure. For example, “Record Number” is treated as a dimension but “Number of Records” is not. In addition, the Korean field name must be four or fewer characters long.

Language	Keywords
English	Number, Num, Nbr
Chinese (Traditional)	-
French	Nombre, Num, Nb
German	Nr, Num, Nummer, Zahl
Japanese	番号
Korean	번 호
Portuguese	Número, Num, Nº
Spanish	Núm, N.º, Nro, Número

Keywords related to dates

Field names containing keywords that are recognized as date parts are treated as dimensions. These field names can only contain only one additional word unrelated to dates to qualify. For example, “Fiscal Year” is treated as a dimension but “Fiscal Year Information” is not. In addition, traditional Chinese field names must be four or fewer characters long and cannot contain digits. Japanese and Korean field names must be four or fewer characters long.

Language	Keywords
English	Year, Yr, Day, Day of Week, Week, Wk, Month, Quarter, Fy
Chinese (Traditional)	年, 月
French	An, AF, Annee, Année, Jour de la semaine, Jour, Mois, Semaine, Trimestre
German	BJ, FJ, EJ, GJ, Jahr, Jr, Quart, Quartal, Monat, Tag de Woche, Tag, Viertel, Woche, Wo
Japanese	年度

Korean	년도
Portuguese	Ano, AF, Dia da semana, Dia, Mês, Meses, Semana, Trimestre
Spanish	AF, Año, Día de la semana, Día de semana, Día, Mes, Samana, Trim, Trimestre

Field name clean-up

Field names that contain specific characters or capitalized in a certain way are renamed.

Field names with underscore and space characters

Non-leading and trailing underscore (_) characters in field names are converted into space () characters. Carriage return or line feed characters in a field name are removed. However, field names that contain leading underscore characters remain unchanged. In addition, leading and trailing spaces in field names are removed. This type of field name clean up applies only to English, French, German, Portuguese, and Spanish languages.

Example 1: The field name “Country_Name” is converted to “Country Name”.

Example 2: The field name “_Days_On_Market” is converted to “_Days On Market”.

Field names with multiple underscore characters in a row remain unchanged. For example, “Country__Name” remains as “Country__Name”.

Field names and capitalization

Field names that use all capital letters with non-letter characters are converted to all lower-case letters except for the characters immediately after the non-letter character.

Additionally, field names that use all lower-case letters are converted so the first letter in the field name is capitalized.

This type of field name clean up applies only to English, French, German, Portuguese, and Spanish languages.

Example 1: The field name “PC1” is converted to “Pc1.” The field name “COUNTRY_NAME” is converted to “Country Name”. However, “Budget COGS” remains as “Budget COGS” .

Example 2: The field name “FDA” does not change. However, the field name “FDA_Sales” is converted to “Fda Sales”.

Example 3: The field name “age” is converted to “Age”.

Space characters are added to the case boundaries of field names that contain mixed casing with non-letter characters.

Example: The field name “ThisCase” is converted to “This Case”.

Field names that are two or three letters long

Field names that are two or three letters long that are part of a multiple word phrase are converted so that each letter in the word is capitalized, unless the two or three letter word contains a vowel (i.e., a, e, i, o, or u).

Example: The field name "Unit Qty" is converted to "Unit QTY". However, the field name "Sales Amt" remains as "Sales Amt".

Several three-letter acronyms are converted to all uppercase. Examples of three-letter acronyms include the following:

CIF	FDA	MPI
DMA	FOB	MSA
DOB	FTE	SKU
EIN	KPI	UPC
ESP	LOB	URL
		USD

The following field names are converted to all lowercase letters unless the name occurs at the beginning of the field name:

1st	as	down	of	th (suffix following a number)
2nd	at	for	on	than
3rd	but	in	off	the
a	by	inclu.	or	to
an	de	into	over	vs
and	da	la	per	with

Note: The casing of field names that are one letter long remain unchanged.

Undo changes made by Tableau

If you prefer to maintain the field types and field names specified by the underlying data, you can undo changes made by Tableau either manually, at the data source level, or automatically, at the application level.

To undo changes manually

1. Open your workbook and go to the Data Source page.
2. Click the metadata grid button.
3. Press **Ctrl+Click** (Windows) or **Command+Click** (Mac) to select the rows with changes you want to undo.
4. Click the drop-down arrow for one of the rows selected, and select **Reset name**.

To automatically undo changes made by Tableau at the application level, follow the procedure in the [Overriding Automatic Field Renaming](#) article. This method affects all data sources that you access from Tableau Desktop.

Extract Your Data

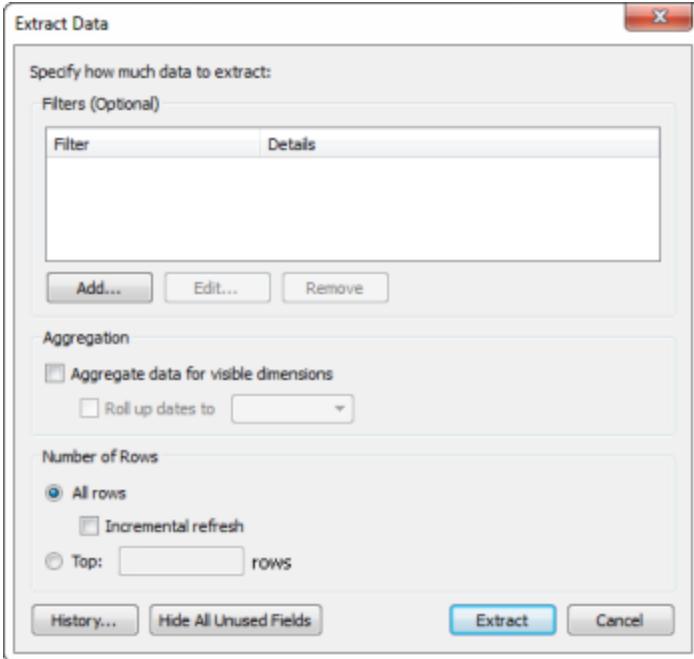
Extracts are saved subsets of a data source that you can use to improve performance or to take advantage of Tableau functionality that is not available in the underlying data. When you extract your data to create an extract, you can reduce the total amount of data by defining filters and limits. After you create an extract you can refresh it with data from the original data. When refreshing the data, you have the option to either fully refresh the data, which replaces all of the extract contents, or you can incrementally refresh the extract, which only adds rows that are new since the previous refresh.

Extracts can:

- Improve performance. For file based data sources such as Excel or Access, a full extract takes advantage of the Tableau data engine. For large data sources, a filtered extract can limit the load on the server when you only need a subset of data.
- Take advantage of Tableau functionality that is not available in the original data source, such as the ability to compute Count Distinct.
- Provide offline access to your data. If your data source is not available (for example, because you are traveling), you can extract the data to a local data source.

To create an extract

1. On the Data Source page, select a data source on the **Data** menu and then select **Extract Data** to open the Extract Data dialog box.

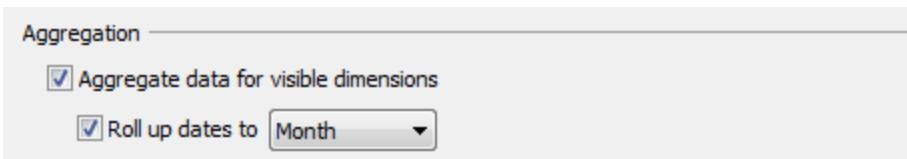


2. Optionally define filters to limit the data that will be extracted. Any fields that are hidden in the Data pane will be automatically excluded from the extract. Click the **Hide All Unused Fields** button to quickly remove them from the extract.

To add filters, click the **Add** button under the Filters list.

3. Specify whether to **Aggregate data for visible dimensions**. When you select this option the measures are aggregated using their default aggregation. Aggregating the data can minimize the size of the extract file and increase performance.

When you choose to aggregate the data you can also choose to **Roll up dates** to a specified date level such as Year, Month, etc.



The examples below show how the data will be extracted for each aggregation option.

Original Data	<table border="1"> <thead> <tr> <th></th><th>A</th><th>B</th><th>C</th><th>D</th></tr> </thead> <tbody> <tr><td>1</td><td>Date</td><td>Region</td><td>Sales</td><td></td></tr> <tr><td>2</td><td>1/1/2009</td><td>South</td><td>\$500</td><td></td></tr> <tr><td>3</td><td>1/1/2009</td><td>West</td><td>\$200</td><td></td></tr> <tr><td>4</td><td>1/1/2009</td><td>West</td><td>\$100</td><td></td></tr> <tr><td>5</td><td>1/1/2009</td><td>East</td><td>\$300</td><td></td></tr> <tr><td>6</td><td>1/2/2009</td><td>South</td><td>\$600</td><td></td></tr> <tr><td>7</td><td>1/2/2009</td><td>South</td><td>\$400</td><td></td></tr> <tr><td>8</td><td>1/2/2009</td><td>East</td><td>\$100</td><td></td></tr> <tr><td>9</td><td></td><td></td><td></td><td></td></tr> </tbody> </table>		A	B	C	D	1	Date	Region	Sales		2	1/1/2009	South	\$500		3	1/1/2009	West	\$200		4	1/1/2009	West	\$100		5	1/1/2009	East	\$300		6	1/2/2009	South	\$600		7	1/2/2009	South	\$400		8	1/2/2009	East	\$100		9					Each record is shown as a separate row. There are 7 rows in the data source.
	A	B	C	D																																																
1	Date	Region	Sales																																																	
2	1/1/2009	South	\$500																																																	
3	1/1/2009	West	\$200																																																	
4	1/1/2009	West	\$100																																																	
5	1/1/2009	East	\$300																																																	
6	1/2/2009	South	\$600																																																	
7	1/2/2009	South	\$400																																																	
8	1/2/2009	East	\$100																																																	
9																																																				
Aggregate Data <i>(no roll up)</i>	<table border="1"> <thead> <tr> <th></th><th>A</th><th>B</th><th>C</th><th>D</th></tr> </thead> <tbody> <tr><td>1</td><td>Date</td><td>Region</td><td>Sales</td><td></td></tr> <tr><td>2</td><td>1/1/2009</td><td>East</td><td>\$300</td><td></td></tr> <tr><td>3</td><td>1/1/2009</td><td>South</td><td>\$500</td><td></td></tr> <tr><td>4</td><td>1/1/2009</td><td>West</td><td>\$300</td><td></td></tr> <tr><td>5</td><td>1/2/2009</td><td>East</td><td>\$100</td><td></td></tr> <tr><td>6</td><td>1/2/2009</td><td>South</td><td>\$1,000</td><td></td></tr> <tr><td>7</td><td></td><td></td><td></td><td></td></tr> </tbody> </table>		A	B	C	D	1	Date	Region	Sales		2	1/1/2009	East	\$300		3	1/1/2009	South	\$500		4	1/1/2009	West	\$300		5	1/2/2009	East	\$100		6	1/2/2009	South	\$1,000		7					Records with the same date and region have been aggregated into a single row. There are 5 rows in the data source.										
	A	B	C	D																																																
1	Date	Region	Sales																																																	
2	1/1/2009	East	\$300																																																	
3	1/1/2009	South	\$500																																																	
4	1/1/2009	West	\$300																																																	
5	1/2/2009	East	\$100																																																	
6	1/2/2009	South	\$1,000																																																	
7																																																				
Aggregate Data <i>(roll up dates to Month)</i>	<table border="1"> <thead> <tr> <th></th><th>A</th><th>B</th><th>C</th><th>D</th></tr> </thead> <tbody> <tr><td>1</td><td>Date</td><td>Region</td><td>Sales</td><td></td></tr> <tr><td>2</td><td>1/1/2009</td><td>East</td><td>\$400</td><td></td></tr> <tr><td>3</td><td>1/1/2009</td><td>South</td><td>\$1,500</td><td></td></tr> <tr><td>4</td><td>1/1/2009</td><td>West</td><td>\$300</td><td></td></tr> <tr><td>5</td><td></td><td></td><td></td><td></td></tr> </tbody> </table>		A	B	C	D	1	Date	Region	Sales		2	1/1/2009	East	\$400		3	1/1/2009	South	\$1,500		4	1/1/2009	West	\$300		5					Dates have been rolled up to the Month level and records with the same region have been aggregated into a single row. There are 3 rows in the data																				
	A	B	C	D																																																
1	Date	Region	Sales																																																	
2	1/1/2009	East	\$400																																																	
3	1/1/2009	South	\$1,500																																																	
4	1/1/2009	West	\$300																																																	
5																																																				

		source.
--	--	---------

4. Select the number of rows you want to extract. You can extract all or the top N rows. Tableau first applies any filters and aggregation and then extracts the number of rows from the filtered and aggregated results.
The number of rows options depend on the type of data source you are extracting from. For example, not all data sources support sampling so that option is not always available.
5. When finished, click **Extract**.
6. In the subsequent dialog box, select a location to save the extract into and give the file a name. Then click **Save**.

Note: If the Save dialog does not display, see the **Troubleshooting extracts** section, below.

Troubleshooting extracts

- **Creating an extract takes a long time:** Depending on the size of your data set, extracting the data can take a long time. However, after you have extracted the data and saved it to your computer, performance will improve.
- **Extract is not created:** If your data set contains a really large number of columns (e.g., in the thousands), in some cases Tableau might not be able to create the extract. If you encounter problems, consider extracting fewer columns or restructuring the underlying data.
- **Save dialog does not display or extract is not created from a .twbx:** If you follow the above procedure to extract data from a packaged workbook, the Save dialog does not display. When an extract is created from a packaged workbook (.twbx), the extract file is automatically stored in the package of files associated with the packaged workbook. To access the extract file that you created from the packaged workbook, you must unpack the workbook. For more information, see [Packaged Workbooks on page 1267](#) to unpack the packaged workbook.

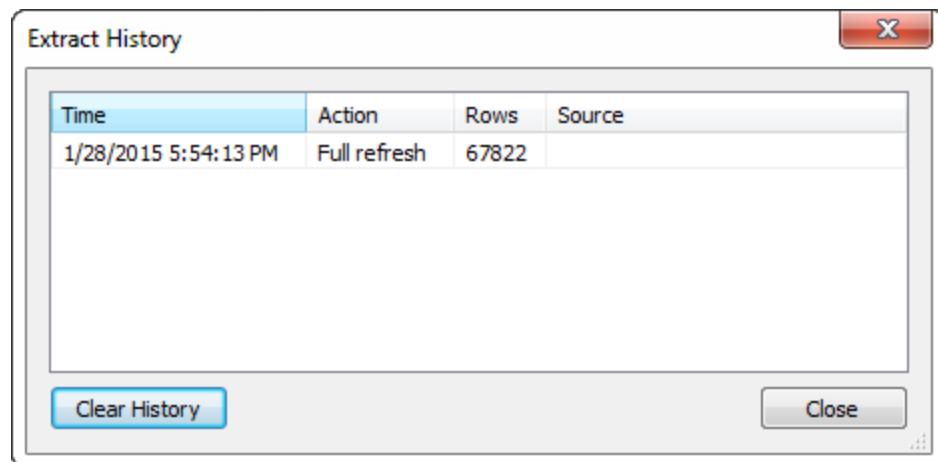
Use Extracts

After you create an extract, the current workbook begins using the extract. However, the extract connection is not saved with the workbook until the next time you save. That means, if you close the workbook without saving first, the workbook will connect to the original data source the next time you open it.

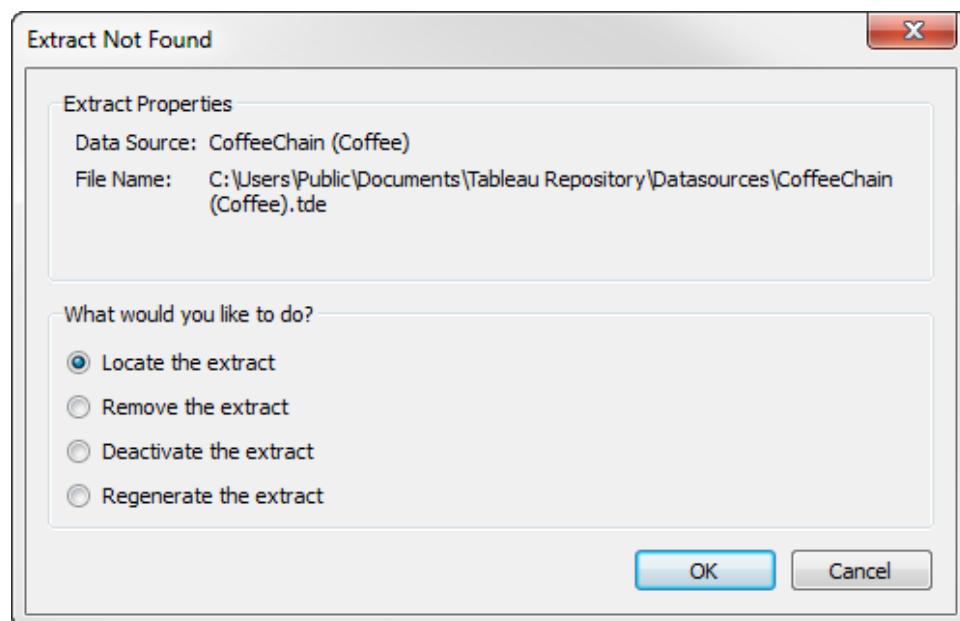
You may want to create an extract with a sample of the data so you can set up the view and then switch to the whole data source, thus avoiding long queries every time you place a field on the shelf. You can toggle between using the extract and using the entire data source by selecting a data source on the **Data** menu and then selecting **Use Extract**.

You can remove an extract at anytime by selecting a data source on the **Data** menu and then selecting **Extract > Remove**. When you remove an extract you can choose to Remove the extract from the workbook only or Remove and delete the extract file, which will delete the extract from your hard drive.

You can see when the extract was last updated and other details by selecting a data source on the **Data** menu and then selecting **Extract > History**.



If you open a workbook that was saved with an extract and the relevant extract file (.tde) is missing, Tableau prompts you for what to do next:



The options are:

- **Locate the extract**

Select this option if the extract exists but not in the location where Tableau originally saved it. Click **OK** to open an Open File dialog box where you can specify the new location for the extract file.

- **Remove the extract**

Select this option if you have no further need for the extract. This is equivalent to closing the data source. All open worksheets that reference the data source are deleted.

- **Deactivate the extract**

Use the original data source from which the extract was created, instead of the extract.

- **Regenerate the extract**

Recreates the extract. All filters and other customizations you specified when you originally created the extract are automatically applied.

Refresh Extracts

When the underlying data changes, you can refresh the extract by selecting a data source on the **Data** menu and then selecting **Extract > Refresh**. Extracts can be configured to be fully refreshed, replacing all of the data with what's in the underlying data source, or incrementally refreshed, adding just the new rows since the last refresh.

Full Refresh

By default, extracts are fully refreshed. That means that every time you refresh the extract, all of the rows are replaced with the data in the underlying data source. While this kind of refresh ensures you have an exact copy of what is in the underlying data source, it can sometimes take a long time and be expensive on the database depending on how big the extract is.

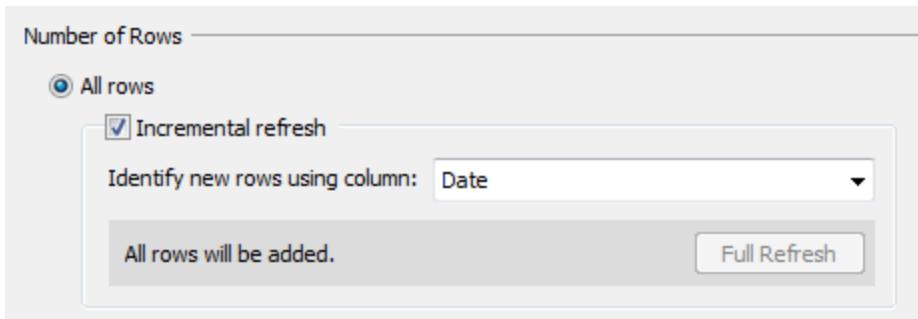
If the extract is not set up for incremental extract, selecting to refresh the extract will fully refresh the extract. If you're publishing the data source to Tableau Server, you can specify the type of refresh in the Scheduling & Passwords dialog box. Most data sources support an incremental refresh.

Incremental Refresh

Rather than refreshing the entire extract, you can set it up to only add the rows that are new since the last time you extracted data. For example, you may have a data source that is updated daily with new sales transactions. Rather than rebuild the entire extract each day, you can just add the new transactions that occurred that day. Then once a week you may want to do a full refresh just to be sure you have the most up to date data.

Follow the steps below to set up an extract to be incrementally refreshed.

1. Select a data source on the **Data** menu and then select **Extract**.
2. In the Extract Data dialog box, select **All rows** as the number of Rows to extract. Incremental refresh can only be defined when you are extracting all rows in the database. You cannot increment a sample extract.
3. Select **Incremental refresh** and then specify a column in the database that will be used to identify new rows. For example, if you select a Date field, refreshing will add all rows whose date is after that last time you refreshed. Alternatively, you can use an ID column that increases as rows are added to the database.



Note: Tableau Data Engine stores time values with a precision of up to 3 decimal places. If you specify a datetime or timestamp column for **Identify new rows using column**, and your database uses a higher precision than Tableau, you can end up with duplicate rows after an incremental refresh. For example, if the database has two rows, one with a datetime value of 2015-03-13 17:30:56.502352 and one with a datetime value of 2015-03-13 17:30:56.502852, Tableau will store both rows using a datetime value of 2015-03-13 17:30:56.502 thereby creating duplicate rows.

4. When finished, click **Extract**.

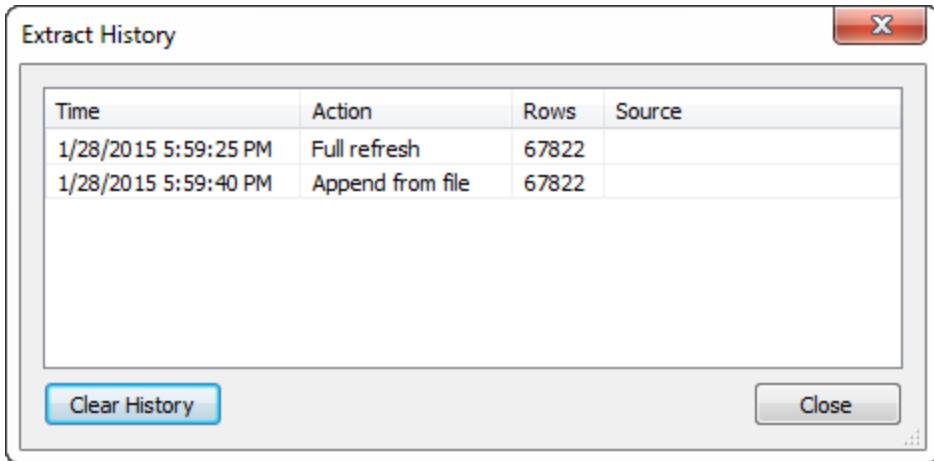
The steps above can be used to define a new extract or configure an existing extract for incremental refresh. If you are editing an existing extract, the last refresh is shown so you can be sure you are updating the extract with the correct data.

If you publish the data source to Tableau Server you can specify a schedule for incremental refresh as well as full refresh in the Schedules & Passwords dialog box.

Extract History

You can see a history of when the extract was refreshed by selecting a data source on the **Data** menu and then select **Extract > History**.

The Extract History dialog box shows the date and time for each refresh, whether it was full or incremental, and the number of rows that were added. If the refresh was from a file, it also shows the source file name.



Add Data to Extracts

There are two ways you can add new data to an extract: from a file or from a data source. However, to add new data, you must connect to your data and create a new data source, and the columns in the file or data source must match the columns in the extract.

Add data from a file

You can add new data to an extract from a file-based data source. Use this option when the file type of the extract is the same as the file type of the data that you want to add. Alternatively, you can add data from a Tableau data extract (.tde) file. For example, you may have text files that are generated for a task that is performed every day. To add each day's worth of information to your extract whose original data source is also a text file, use the **Append Data from File** command.

To add data from a file

1. On the **Data** menu, select a data source, and then select **Extract > Append Data from File**.
2. Browse to and select the file that has the new data.

Note: By default, the file format of the extract's original data source is used. To add data from a Tableau data extract, click the file format drop-down list, and then select **Tableau Data Extract (*.tde)**.

3. When finished, click **OK**.

Add data from a data source

You can also add new data to an extract from another data source that's already open in the workbook. Use this option when the data from the data source that you want to add matches the extract data source before it was extracted. For example, you created an extract from a SQL Server database that has the past ten years worth of data. However, the maintained

version of that data is stored on a different SQL Server database. You can add new data to the extract by using the **Append Data from Data Source** command.

Note: Joins or custom SQL should be specified in the data source before adding data to the extract.

To add data from a data source

1. On the **Data** menu, select a data source, and then select **Extract > Append Data from Data Source**.
2. In the dialog box, select the data source that you want to append.
3. When finished, click **OK**.

Using either option will add new rows to the extract. To see a summary of the number of rows that were added, select a data source on the **Data** menu and then select **Extract > History**.

Note: When you refresh this extract, the data will be replaced with the data from the original data source.

Optimize Extracts

To improve performance when working with extracts you can optimize the extract. Optimizing an extract creates secondary structure in the extract that speed up future queries.

Optimize the extract by selecting a data source on the **Data** menu and then selecting **Extract > Optimize**.

The following types of optimizations are made:

Materialized Calculated Fields

Calculated fields are computed in advance and stored in the extract. In future queries, Tableau can look up the already computed value rather than running the computation again. The following types of calculated fields ARE NOT materialized:

- Calculations that use unstable functions such as NOW() and TODAY()
- Calculations that use external functions such as RAWSQL and R
- Table calculations

In addition, if the formula for a materialized calculation changes or the calculation is deleted from the data source, the materialized calculation is dropped from the extract until the extract is optimized again.

Acceleration Views

When a workbook contains filters that are set to show only relevant values, computing the available values for that filter can be an expensive query. For these filters, Tableau must

evaluate the other filters in the workbook first and then compute the relevant values based on their filter selections. To speed up these queries, a view can be created that computes the possible filter values and caches them for faster lookup later.

Update Server Data Sources that Are Using Extracts

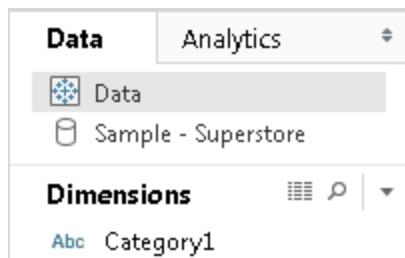
You have the following options for updating data sources that are using extracts that have been published to Tableau Server or Tableau Online:

- You can add the data source or a workbook that connects to it to a refresh schedule in Tableau Server or Tableau Online (cloud-based data sources only).
- You can update the data source in Tableau Desktop and then republish it.
- You can add to or refresh the data source on Tableau Server or Tableau Online, from Tableau Desktop, without first adding to or refreshing the extract in Tableau Desktop.

The remainder of this topic describes the third option.

Update Server Extracts From Tableau Desktop

Before you attempt to update a data source on Tableau Server or Tableau Online that is using an extract, verify that Tableau Desktop is connected to the published data source on Tableau Server, as indicated by the Tableau Server icon next to the data source name in the Data pane:



To update the server data source, right-click (control-click on a Mac) the data source in the Tableau Desktop Data pane, select **Tableau Data Server**, and choose one of the following options:

- **Edit Server and Site Path**

If you have changed the location of the data source in Tableau Server or Tableau Online, choose this option to point Tableau Desktop to the new, correct location.

- **Refresh from Source**

Refreshes the extract (full or incremental) using the data in the original data source.

This command is available only for extracts that include a connection to the original data source. If you connected directly to a Tableau Data Extract file (.tde) and then published

it, the connection to the original data source is not included and this option is therefore unavailable.

- **Append from Data Source**

Updates the extract from another open (and compatible) data source. If you connected directly to a Tableau Data Extract file (.tde) and then published it, the connection to the original data source is not included and this option is therefore unavailable.

Note: If you see the Tableau Data Server option, but the commands in the submenu are unavailable, the data source exists on the server, but is not an extract.

It is also possible to update an extract on Tableau Server using a command-line utility. See [Tableau Data Extract Command-Line Utility](#) below.

Tableau Data Extract Command-Line Utility

You can automate extract refresh tasks using the Tableau Data Extract Utility. This is a command-line utility that comes with Tableau Desktop, through which you can refresh published data sources or append data to them from a file.

Note: The Tableau Data Extract Utility is available only with Tableau Desktop on Windows.

Run the utility

1. Open the Command Prompt as an administrator and change to the Tableau Desktop bin directory. For example:
`cd C:\Program Files\Tableau\Tableau 10.1\bin`
2. Use either of the following commands, adding parameters described in the tables below.
 - `tableau refreshextract`
 - `tableau addfiletoextract`

Note: When using the utility, always specify `tableau` on the command line or in scripts, never `tableau.exe`.

Syntax and parameters for the `tableau refreshextract` command

Use `tableau refreshextract` to refresh an extract on Tableau Server or Tableau Online. Refreshing an extract updates an existing extract with any modifications that have been made to the data source since the last refresh.

To see help for this command, at the Windows command prompt, type the following command:

```
tableau refreshextract --help
```

Using parameters

- All options have a full form that you use with a double hyphen (for example, `--server`).
- Some options also have a short form that you use with a single hyphen (for example, `-s`).
- If the value for an option contains spaces, enclose it in quotation marks.

tableau refreshextract command options

Short Form	Full Form	Description
	<code>--source-username <user name></code>	A valid user name for the data source connection. Use this option with <code>--source-password</code> , or use <code>--original-file</code> instead of the user name and password options. Note: You must provide the user name and password when refreshing a published extract, even if the data source was originally published with embedded credentials.
	<code>--source-password "<password>"</code>	The password for the data source user.
	<code>--original-file <path and file name></code> or <code>--original-file <path and folder name></code>	Path and file name for the data source to be refreshed on the server. For example: <code>--original-file c:\folder\file.csv</code> To refresh a multi-file data source, pass the path to a folder that contains the data files. For example: <code>--original-file c:\folder</code> . If the file is on a network share, use the UNC format for the path: <code>\server\path\filename.csv</code>

	--force-- full-- refresh	If the data source is set up for incremental refreshes, use this option to force a full extract refresh. If this option is not included, an incremental refresh is performed. Not all data sources support incremental refresh.
-s <server http address>	--server <URL>	The URL for the Tableau server on which the data is published. For Tableau Online, specify https://online.tableau.com .
-t <site id>	--site <siteid>	In a multiple-site environment, specifies the site to which the command applies. For Tableau Online, use this argument if your user name is associated with more than one site. For Tableau Server, if you do not specify a site, the default site is assumed. The site id is independent of the site name, and it is indicated in the URL when you view the site in a browser. For example, if the URL for the page you see after signing in to Tableau Online is https://on-line.tableau.com/t/vernazza/views the site id is <i>vernazza</i> .
	--data-- source <data-- source>	The name of the data source, as published to Tableau Server or Tableau Online.
	--project <pro-- jectname>	The project to which the data source belongs. If this option is not included, the default project is assumed.
-u <user name>	--username <username>	Valid Tableau Server or Tableau Online user.
-p "<pass-- word>"	--password "<pass-- word>"	The password for the specified Tableau Server or Tableau Online user.
	--proxy-- username <username>	The user name for a proxy server.

	--proxy-password " <i><password></i> "	The password for a proxy server.
-c "<path and file name>"	--config-file "<path and file name>"	Path and file name information for a file containing configuration options for the command. Always enclose the path in double quotation marks. For more information, see Using a config file on page 456 below.

Sample tableau refreshextract command

The following command refreshes an extract named CurrentYrOverYrStats that has been published to Tableau Server on-premises. This command specifies the following:

- The name of your Tableau Server.
- Server user name and password.
- Project name.
- The name of the data source to refresh, along with the data source username and password.

```
C:\Program Files\Tableau\Tableau 10.1\bin>tableau
refreshextract --server https://our_server_name --username
OurServerSignIn --password "OurServerPwd" --project "New
Animations" --datasource "CurrentYrOverYrStats" --source-
username OurDatabaseSignIn --source-password
"OurDatabasePassword"
```

The following command refreshes an extract named CurrentYrOverYrStats that has been published to Tableau Online. This command specifies the following:

- Tableau Online user and password.
- Tableau Online site and project names.
- The data source, which in this case is hosted by a cloud-based data source provider (for example, Salesforce.com), and the username and password to sign in to the hosted data source.

```
C:\Program Files\Tableau\Tableau 10.1\bin>tableau
refreshextract --server https://online.tableau.com --
username email@domain.com --password "OurServerPwd" --site
vernazza --project "New Animations" --datasource
"CurrentYrOverYrStats" --source-username database_
```

```
user@hosted_datasource_provider.com --source-password "db_password"
```

To refresh an extract of file-based data source, provide the path to the original file from which you created the extract. If the file is on a network share, use the UNC format instead of a mapped drive.

```
C:\Program Files\Tableau\Tableau 10.1\bin>tableau  
refreshextract --server https://online.tableau.com --  
username email@domain.com --password "OurServerPwd" --site  
vernazza --project "New Animations" --datasource  
"CurrentYrOverYrStats" --original-file  
"\\"server\path\filename.csv"
```

Syntax for `tableau addfiletoextract`

Use `tableau addfiletoextract` to append file content to an extract that has been published to Tableau Server or Tableau Online. This command combines the two files.

If you want simply to update an existing extract with the latest changes, use the `refreshextract` command instead. Using `addfiletoextract` to update an existing extract will duplicate data instead.

To see help for this command, at the Windows command prompt, type the following command:

```
tableau addfiletoextract --help
```

All options have a full form that you use with a double hyphen (for example, `--server`). Some options also have a short form that you use with a single hyphen (for example, `-s`). If the value for an option contains spaces, enclose it in quotation marks.

`tableau addfiletoextract` command options

Short Form	Full Form	Description
	<code>--file <path and file name></code>	Path and file name information for the data file containing data to append. The file can be from Excel, Access, a Tableau data extract, or a delimited text file. It cannot be password protected. Use UNC format if the file is on a network share. For example, <code>\\"server\path\filename.csv</code>

<code>-s <server http address></code>	<code>--server <URL></code>	The URL for the Tableau server on which the data is published. For Tableau Online, specify <code>https://online.tableau.com</code> .
<code>-t <site id></code>	<code>--site <site id></code>	In a multiple-site environment, specifies the site to which the command applies. For Tableau Online, you must include this argument if your user name is associated with more than one site. For Tableau Server, if you do not specify a site, the default site is assumed.
	<code>--datasource <data-source></code>	The name of the data source, as published to Tableau Server or Tableau Online.
	<code>--project <projectname></code>	The project to which the data source belongs. If this option is not included, the default project is assumed.
<code>-u <username></code>	<code>--username <username></code>	Valid Tableau Server or Tableau Online user.
<code>-p "<password>"</code>	<code>--password "<password>"</code>	The password for the specified Tableau Server or Tableau Online user.
	<code>--proxy-username <username></code>	The user name for a proxy server.
	<code>--proxy-password "<password>"</code>	The password for a proxy server.
<code>-c "<path and filename>"</code>	<code>--config-file "<path and filename>"</code>	Path and file name information for a file containing configuration options for the command. Always enclose the path in double quotation marks. For more information, see Using a config file on the next page below.

Sample tableau addfiletoextract command

```
C:\Program Files\Tableau\Tableau 10.1\bin>tableau  
addfiletoextract --server https://our_server_name --username  
OurServerSignIn --password "OurServerPwd" --project "New  
Animations" --datasource "CurrentYrOverYrStats" --file  
"C:\Users\user1\Documents\DataUploadFiles\AprMay.csv"
```

```
C:\Program Files\Tableau\Tableau 10.1\bin>tableau  
addfiletoextract --server https://online.tableau.com --  
username email@domain.com --password "OurServerPwd" --site  
vernazza --project "New Animations" --datasource  
"CurrentYrOverYrStats" --file  
"C:\Users\user2\Documents\DataUploadFiles\AprMay.csv"
```

Using a config file

You can use a plain text editor, such as Notepad or Text Edit, to create a config (configuration) file that you can use with `tableau refreshextract` or `tableau addfiletoextract`. A config file can be useful if you expect to update the same data source regularly over time. Instead of having to type the same options each time you run a command, you specify the config file. A config file also has the advantage of not exposing user names and passwords on the command line.

Create the config file

For example, say you created a file called `config.txt` and saved it to your Documents folder. And in the file, you included the parameter information shown below.

For an extract published to Tableau Server:

```
server=https://our_server_name  
username=OurServerSignIn  
password=OurServerPwd  
project>New Animations  
datasource=CurrentYrOverYrStats
```

For an extract from a hosted data source, published to Tableau Online, where **server** is `https://online.tableau.com`:

```
server=https://online.tableau.com  
username=email@domain.com  
password=OurPassword
```

```
project=New Animations  
datasource=CurrentYrOverYrStats  
source-username=database_user@hosted_datasource_provider.com  
  
source-password=db_password
```

Reference the Config File from the Command Line

After you create the config file, you run the `tableau refreshextract` or `tableau addfiletoextract` command, pointing to the config file as the only option you use on the command line, and enclosing the config file's path in double quotation marks. The syntax is as follows:

```
tableau refreshextract --config-file "<path>"
```

For example, to refresh the extract specified in the sample in the [Create the config file](#) on the previous page section, you would run the following command (making sure that you are working in the bin directory for your version of Tableau Desktop):

```
C:\Program Files\Tableau\Tableau 10.1\bin>tableau  
refreshextract --config-file  
"C:\Users\user1\Documents\config.txt"
```

Syntax Differences for Config Files

The syntax for specifying options inside a config file differs from the syntax you use on the command line in the following ways:

- Option names do not begin with dashes or hyphens.
- You use an equals sign (with no spaces) to separate option names from option values.
- Quotation marks are not necessary (or allowed) around values, even when they include spaces (as for the `project` option in the example shown earlier).

Use Windows Task Scheduler to Refresh Extracts

You can use Windows Task Scheduler, in combination with the Tableau Data Extract Command-Line Utility, to automate regular updates to Tableau Desktop data sources from within your corporate firewall. You can configure a task to occur once per day, week, or month, or after a specific system event. For example, run the task when the computer starts.

To learn more, see the [Task Scheduler How To...](#) page in the Microsoft TechNet library.

Tableau Data Extract API

Starting in Tableau 9.1, the Tableau Data Extract API is incorporated into the Tableau SDK. You can use the Tableau SDK to create an extract and publish it to Tableau Server. The SDK supports Windows, Linux, and the Mac, using C, C++, Java, and Python.

For more information, see the [Tableau SDK documentation](#).

Manage Data Sources

The topics in this section describe the tasks you can perform after you have set up data sources.

Edit Data Sources

At anytime during your analysis you can edit the data source used in the workbook. You might want to edit the data source to:

- **Specify a new location for the data source.** For example, suppose the name or location of a data source you were using has changed and is no longer available using the previous connection information. In this case, you can direct the workbook to the correct location without losing your work.
- **Apply analyses created using one data source to another data source.** Suppose you create a workbook containing several views involving markets, products, sales, and profits and you want to apply the analyses to a new data source. Instead of recreating each view, you can edit the original data connection and specify a new data source.
- **Add and remove tables.**
- **Create custom SQL queries for supported data sources.**
- **Use stored procedures for supported data sources.**

To edit the data source

1. On the **Data** menu, select a data source, and then select **Edit Data Source**.
2. On the data source page, make the changes to the data source.

For example, if you are using a **Microsoft Excel** data source, you can specify a new file by clicking the Excel file name under **Workbook**. Alternatively, you can select a different table to analyze.

Replace Field References

When you successfully connect to a new data source, all worksheets in the workbook that previously referred to the original data source now refer to the new data source. If the new data source does not have the same field names as the original workbook, the fields become invalid

and are marked with an exclamation point . You can quickly resolve the problem by replacing the field's references.

For example, say you have a workbook connected to a data source that contains a Product Category field. Then you edit the data source to point to a new data source that has all the

same data but instead of Product Category, the field name has been changed to Product Type. The Product Category field remains in the Data pane but is marked as invalid. To make the field valid, you can replace the references, which means you can map the invalid field to a valid field in the new data source (for example, Product Category corresponds to Product Type).

To replace field references

1. Right-click (control-click on a Mac) the invalid field in the Data pane and then select **Replace References**.
2. In the Replace References dialog box, select a field from the new data source that corresponds to the invalid field.

Duplicate the Data Source

Sometimes you'll want to make changes to a data source such as add more tables, hide and show fields, set field defaults, and so on. When you make these changes it affects all sheets that use the data source. You can duplicate the data source so that you can make the changes without affecting the existing sheets.

To duplicate a data source

- On the **Data** menu, select a data source , and then select **Duplicate**.

When you duplicate a data source, the duplicate source name has “(copy)” appended to the end.

Rename the Data Source

When you connect to a data source, you are given the option to give it a name for use in Tableau Desktop.

To rename the data source

- On the **Data** menu, select **Rename**.

Naming a connection is useful when you have a single workbook connected to many data sources. The name you assign can help you keep track of the specifics of the connection. You can also review the connection properties by selecting a data source on the **Data** menu, and then selecting **Properties**.

Refresh Data Sources

If the underlying data changes—for example, if new fields or rows are added, data values or field names are changed, or data is deleted, Tableau will reflect those changes the next time you connect to the data source. However, because Tableau Desktop queries the data and does not import the data, you can immediately update Tableau to reflect the data modifications without disconnecting, provided the changes have been saved in the underlying data first.

If you are connected to a data source that has been modified, you can immediately update Tableau Desktop with the changes by selecting a data source on the **Data** menu and then selecting **Refresh**.

If a field that is used in a Tableau worksheet is removed from the underlying data of the data source and then the data source is refreshed, a warning message displays indicating that the field will be removed from the view and the worksheet will not display correctly because of the missing field.

Replace Data Sources

There are times when you may want to update a workbook or sheet to use a different data source. Rather than rebuild your workbook using the new connection, you can replace the data source.

Note: When you publish a data source to Tableau Server or Tableau Online, the current workbook in Tableau Desktop will automatically connect to the published data source, and automatically close the local data source. So it is not necessary to replace the data source when you publish a data source.

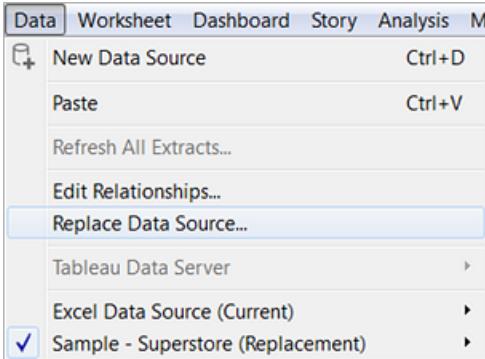
You can only replace one data source with another when both data sources are relational data sources, or when both data sources are cube (multidimensional) data sources. In Tableau, cube data sources are supported only in Windows.

The two data sources do not have to be identical, however, any differences will affect the sheets in the workbook. Any fields, groups, sets, and calculated fields that don't exist in the new data source (or have a different name) are removed from the Data pane. If these fields were used by any sheets in the workbook, they will remain in the Data pane but will be marked invalid. Additionally, you may see changes to custom sets, groups, and calculated fields that depend on the missing fields. For information about how to replace field references to correct invalid fields, see [Edit Data Sources on page 458](#).

Replacing a data source does not merge or edit the data sources. Rather, replacing a data source simply redirects fields in the worksheet to map to corresponding fields in the new data source. To successfully replace a data source, any fields in the original data source (for example, calculated, groups, sets, and parameters), should also exist in the new data source. If not, then you may want to manually copy and paste these fields to the new data source before replacing the old one.

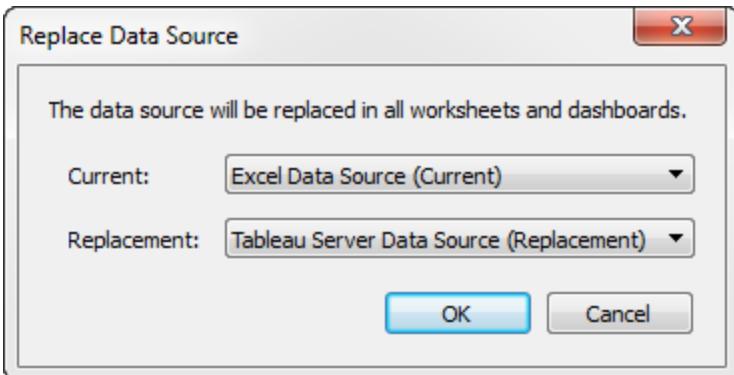
Follow the steps below to replace a data source:

1. Open a workbook that connects to the *old* data source.
2. Select **Data > New Data Source** and then connect to the *new* data source.
3. Select **Data > Replace Data Source**.



You will only be able to see the **Replace Data Source** command when you are on a worksheet tab—not when you are on the Data Source page. You will only be able to select it when there is at least one field in the view.

4. In the Replace Data Source dialog box, select the **Current** data source and the **Replacement** data source.



5. When finished, click **OK**.

All worksheets, dashboards, and stories that used the original data source are updated to use the new data source. Click **Undo** on the toolbar to revert the change and return to your original data source.

Export Data Sources

At any time while connected to a data source, you can export data source information as a shortcut that allows you to quickly connect to the data source in the future. You might want to do this if you often connect to the same data source multiple times or if you've added joined tables, default properties, or custom fields, such as groups, sets, calculated fields, and binned fields, to the Data pane.

Note: You can also save custom fields by saving the workbook or by creating a bookmark file.

Export Data Source in Tableau Desktop on Windows

You can export the data source on a Windows computer in either of the following two formats:



Data Source (.tds) - contains just the information you need to connect to the data sources such as data source type, location, and custom fields. If you connect to local file data sources (Excel, Access, text, extracts), the file path is stored in the data source file.

Data source files contain the following types of information:

- data source type
- data source connection information specified in the data source page (for example, server, port, location of local files, and tables)
- groups
- sets
- calculated fields
- bins
- default field properties (for example, number formats, aggregation, and sort order)



Packaged Data Source (.tdsx) - contains all the information in the Data Source (.tds) file as well as any local file data sources (Excel, Access, text, and extracts). This file type is a single zipped file and is good for sharing a data source with people who may not have access to the original data that is stored locally on your computer.

Export Data Source in Tableau Desktop on a Mac

You can export the data source on a Mac computer in either of the following two formats:



Data Source (.tds) - contains just the information needed to connect to the data sources such as data source type, location, and custom fields. If you connect to local file data sources (Excel, text, and extracts), the file path is stored in the data source file.

Data source files contain the following types of information:

- data source type
- data source connection information specified in the data source page (for example, server, port, location of local files, and tables)
- groups
- sets
- calculated fields

- bins
- default field properties (for example, number formats, aggregation, and sort order)



Packaged Data Source (.tdsx) - contains all the information in the Data Source (.tds) file as well as any local file data sources (Excel, text, and extracts). This file type is a single zipped file and is good for sharing a data source with people who may not have access to the original data that is stored locally on your computer.

After you export, the data sources are available on the **Connect** pane.

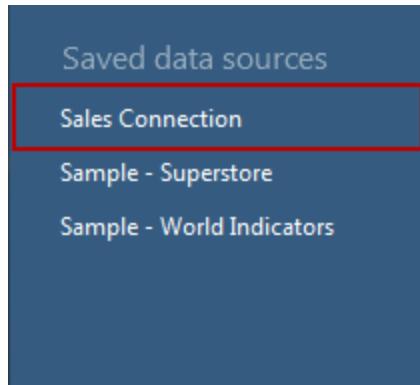
By default, the data source files are stored in the Datasources folder of the Tableau Repository. Data source files stored in another location do not display on the **Connect** pane. You can connect to data source files by selecting **File > Open** and navigating to the file. You can also connect by dragging the data source file onto Tableau Desktop icon or onto the running application.

Note: If you move a local file data source that is referenced by a .tds file, you will be prompted to locate or replace the original data source when you try to open the data source file. To avoid saving a specific file path, save the data source as a TDSX file, which packages a copy of the original local file data source with the .tds file. If you choose to replace the original data source, the replacement data source must be of the same type (for example, Excel or MySQL) as the original.

To export a data source

1. On the **Data** menu, select a data source, and then select **Add to Saved Data Sources**.
2. Complete the **Add to Saved Data Sources** dialog box by specifying a file name and selecting the type of data source file.

The new .tds or .tdsx file is listed in the Saved data sources section of the **Connect** pane.



Upgrade Data Sources

If you have workbooks that were created before Tableau Desktop 8.2 that use Microsoft Excel or text file data sources, or you are using the Excel or text file legacy connection option, you have the option to upgrade the data sources in your workbook. By upgrading your Excel and text file data sources you can take advantage of 1.) better data interpretation, and 2.) compatibility on the Mac. If you are considering upgrading an existing workbook to use the new connection, review the tables below to compare how your data is processed and displayed between the legacy and default connection types.

Note: If there are certain capabilities of the Jet driver that your data relies on, you might need to use the legacy connection to get the expected results. Workbooks that were created before Tableau Desktop 8.2 that use Excel and text file data sources automatically use the legacy connection. To create a new workbook that uses the legacy connection, in Tableau Desktop, navigate to your Excel or text file data, click the drop-down arrow on **Open**, and then select **Open with Legacy Connection**.

File formats and attributes

The following tables show some examples of the types of Excel and text files, and tables that you can connect to in Tableau when using the legacy and default connections types.

Excel

Your Data	Legacy Connection	Default Connection
.xlsb file format	Allows connections to Excel data in .xlsb file format.	Does not allow connections to Excel data in .xlsb format. Connections to Excel data in .xls or .xlsx file formats can be used instead.
Excel built-in	Allows connections to built-in named ranges.	Built-in named ranges are hidden.

Your Data	Legacy Connection	Default Connection
named ranges		
Hidden sheet	Allows connections to a table in a hidden sheet.	Hides a table in a hidden sheet.
Very hidden sheet	Allows connections to a table in a very hidden sheet.	Hides a table in a very hidden sheet.
Table that contain charts	Allows connections to tables that contain Excel charts. However, the table does not contain any values.	Hides tables that contain Excel charts.
Empty table	Allows connections to an empty table.	Hides an empty table.
Table headers	Treats the first row of a table as a header.	Automatically detects whether the first row of a table is a header.
Table width	Limits the table width to 255 columns.	No table width constraints.
Empty columns and rows	Empty columns and rows are visible.	Empty columns and rows are hidden because they do not contain any values.

Text

Your Data	Legacy Connection	Default Connection
File name length	Does not allow connections to a file whose file name is more than 64 characters long.	No length constraints for file names.
Multiple period in the file name	Does not allow connections to a file whose file name contains multiple periods.	No period constraints for file names.

Your Data	Legacy Connection	Default Connection
Number of columns in the file	Uses the number of fields present in the first row to determine the number of columns in the file.	Scans the entire file and uses the most common number of fields in a row to determine the number of columns in the file.
Headers rows	Does not automatically detect whether the first row in the file is a header. Note: You can manually specify that the first row in the file is a header.	Automatically detects whether the first row in the file is a header. Note: You can manually override the detection.
Empty columns	Empty columns are visible.	Empty columns are hidden because they do not contain any values.

Characters and formatting

The following tables show some examples of how your Excel and text file data might be displayed in Tableau when using the legacy and default connection types.

Excel

Your Data	Legacy Connection	Default Connection
Number of characters in the field name	Field names are truncated to 64 characters.	No constraint on the number of characters in a field name.
Special characters in field names	Special characters are not allowed in field names. For example, special characters like quotation marks and periods are converted to number signs. Brackets are converted to parentheses.	Special characters are allowed in field names.
Leading and trailing spaces in field names	Leading and trailing spaces in field names are allowed.	Leading and trailing spaces in field names are automatically removed from column names.
Duplicate	For duplicate field names, an index number is	For duplicate field

Your Data	Legacy Connection	Default Connection
field names	appended onto the field name. For example, Test Scores1.	names, a space and an index number are appended to the field name. For example, Test Scores 1.
Excel cell formatting	Supports cell formatting configured using the Format Cells option in Excel.	Does not support cell formatting configured using the Format Cells option in Excel.
Precision with currency values	Fields that contain currency values are represented with a maximum precision of four digits	Full precision of currency values is represented.

Text

Your Data	Legacy Connection	Default Connection
Special characters in file names	Special characters are not allowed in file names. For example, special characters like periods are converted to number signs.	Special characters are allowed.
Characters in header field names	Automatically removes leading spaces from header field names.	Automatically removes leading and trailing spaces from header field names.
Field separators	Recognizes only commas as field separators. Note: You can manually override the detection.	Automatically detects the field separator. Note: You can manually override the detection.
Text enclosed with quotation marks	Quotation marks used to enclose text values remain visible.	Quotation marks used to enclose text values are automatically hidden.
ANSI and OEM	ANSI and OEM code pages are	ANSI and OEM code pages are auto-

Your Data	Legacy Connection	Default Connection
code pages	supported.	matically converted to be cross-platform compatible.
BOM code pages	Inconsistently supports BOM code pages.	Supports BOM code pages.

Data type detection

The following tables show some examples of how data type is detected and how certain values are displayed in Tableau when using the legacy and default connection types.

Excel

Data Type Detection	Legacy Connection	Default Connection
Columns	<p>Data type of a column is determined by the first 8 rows.</p> <p>Note: After the data type of a column in the table has been determined, it cannot be changed.</p>	<p>Data type of a column is determined by the 95% of the first 10000 rows.</p> <p>Note: You can manually override the detection.</p>
Date values without time	Date values are assigned a datetime data type.	Date values without time are assigned a date data type. Date values with time are assigned a datetime data type.
Number values	All number values are represented as real numbers.	Number values without decimal points are represented as integers.
Null values	If a column contains a null cell, the data type for the column is automatically designated as a string data type.	Null cells do not affect data type detection.
Reference errors or empty cells	If a column contains cells with reference errors or empty cells, the entire column is interpreted as a string data type.	Reference errors or cells with no values do not affect the data type detection.

Data Type Detection	Legacy Connection	Default Connection
Overriding data type detection	After the data type of a column in the table has been determined, it cannot be changed.	Data type of a column can be changed after the automatic detection.
Time value precision	The smallest measurement of time values is whole seconds.	The smallest measurement of time values is fractional seconds.
DecimalSymbol and CurrencyDecimalSymbol schema.ini values	Both DecimalSymbol and CurrencyDecimalSymbol schema.ini values are recognized.	DecimalSymbol and CurrencyDecimalSymbol schema.ini values are recognized. However if both values are used, DecimalSymbol takes precedence.
Cells formatted as text	The data type of a column is detected as a string data type when cells are formatted as text using the Format Cells option in Excel.	Does not support cell formatting configured using the Format Cells option in Excel.

Text

Data Type Detection	Legacy Connection	Default Connection (8.2 and later)
Columns	Data type of a column is determined by the first 25 rows.	Data type of a column is determined by the first 1024 rows.
Boolean (True/False) values	Boolean values are assigned the string data type.	Boolean values are assigned the boolean data type.
Values that become null	Spaces in a cell, whether enclosed in quotation marks or not, are treated as null values. Columns with null values are detected as the string data type.	Two field separators in a row are treated as a null value. Null values are ignored during data type detection.

Data source connection properties

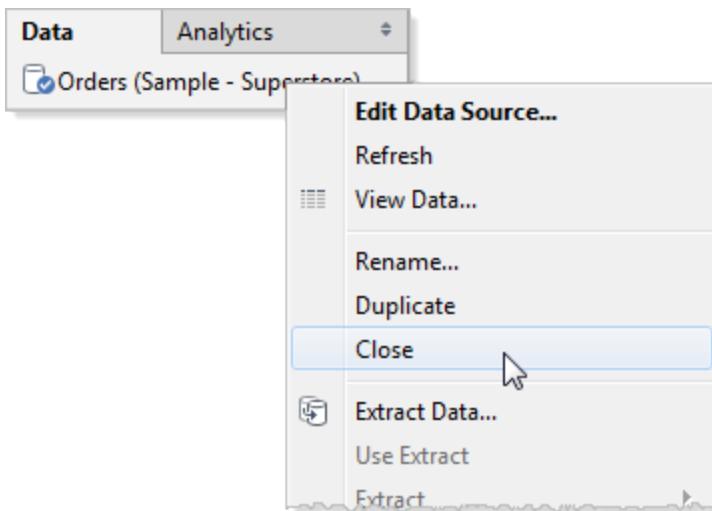
The following table shows examples of the differences in data source connection properties in Tableau when using the legacy and default connection types.

Property	Legacy Connection	Default Connection
Custom SQL	Custom SQL is allowed.	Does not allow the use of Custom SQL.
Join type	Allows left, right, and inner join types.	Allows left and inner join types.
Join operators	Allows equal to (=), greater than (>), greater than or equal to (>=), less than (<), less than or equal to (<=), and not equal to (<>) join operators.	Allows equal to (=) join operators.

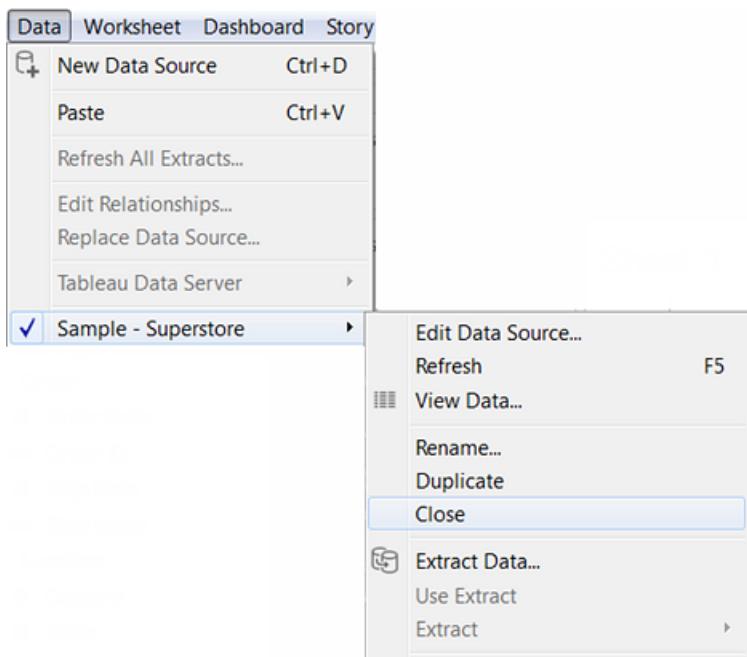
Close Data Sources

You can close a data source at any time. Doing so does not modify the data source. Instead, it disconnects Tableau from the data so that you can no longer query it. Additionally, the data source is cleared from the Data pane and all open worksheets associated with the data source are cleared. If you accidentally close a data source, use the Undo button to reopen it. Close a data source by doing one of the following:

- Right-click (control-click on a Mac) the data source at the top of the Data pane and select **Close**.



- Select a data source on the **Data** menu and then select **Close**.



Design Views and Analyze Data

Tableau Desktop is a kind of laboratory in which you can discover the meaning that lies hidden in your data.

In this section you can discover the various features at your disposal as you build views, and learn the basic skills you need to create elegant, insightful views, dashboards, and stories.

Building Data Views

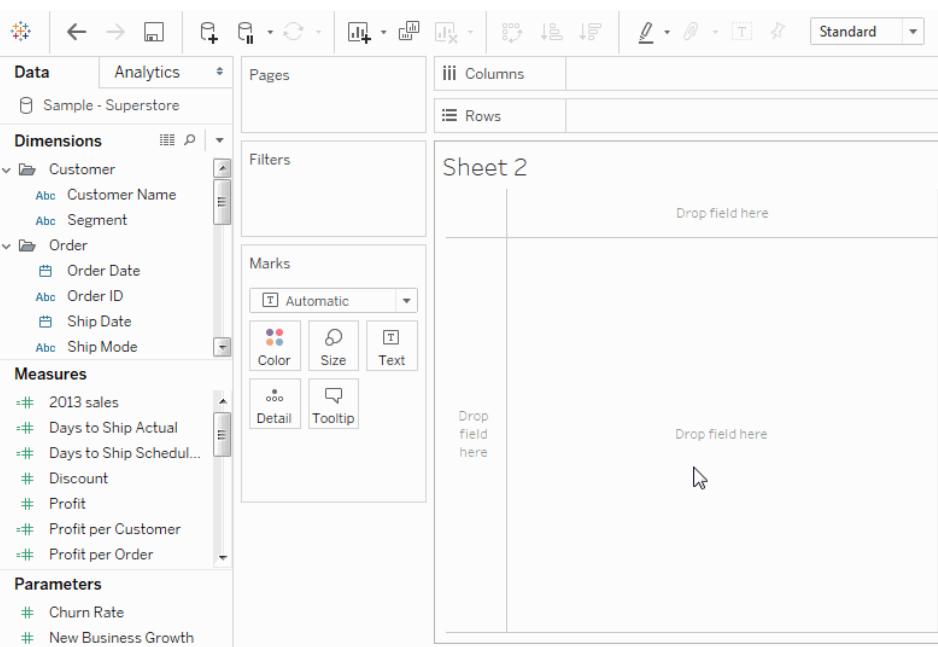
In this section you can discover the various features at your disposal as you build views.

Dragging Fields

You can build views of your data by dragging fields from the Data pane to the view. You can drag fields to a variety of active areas in the view or place them on the shelves and cards that are part of every worksheet.

The Basics

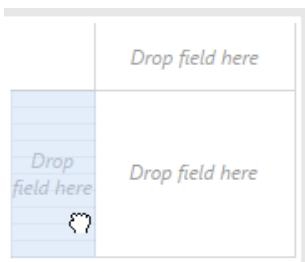
When you begin creating a new data view on a blank worksheet, you can drag a field from the **Data** pane and drop it directly into the view, rather than to a shelf.



While dragging fields you can hover over the different areas in the view to see how the field will be incorporated into the structure of the view. For example, dimensions typically add a row and

column headers to the view, while measures add continuous axes. Below are some examples of how fields can be added to the view.

Dimensions add headers



Measures add axes

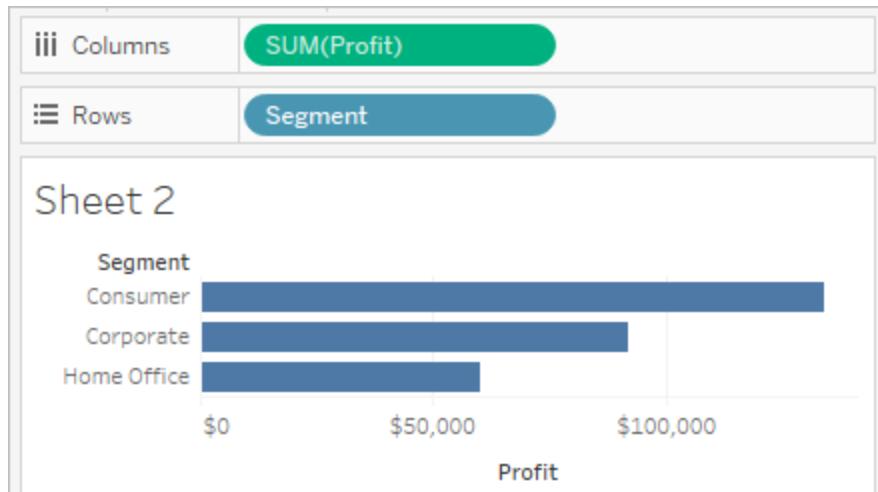


Show Me automatically adds the field according to best practices



For a more advanced discussion of dimensions and measures, see [Dimensions and Measures on page 268](#).

When you drop a field on one of the active areas in the view, the field is added to the view and also to one of the shelves or cards. For example, in the view below the **Segment** dimension was added to the **Rows** shelf and the **Profit** measure was added to the **Columns** shelf—and automatically aggregated so that the values in the data are summed.



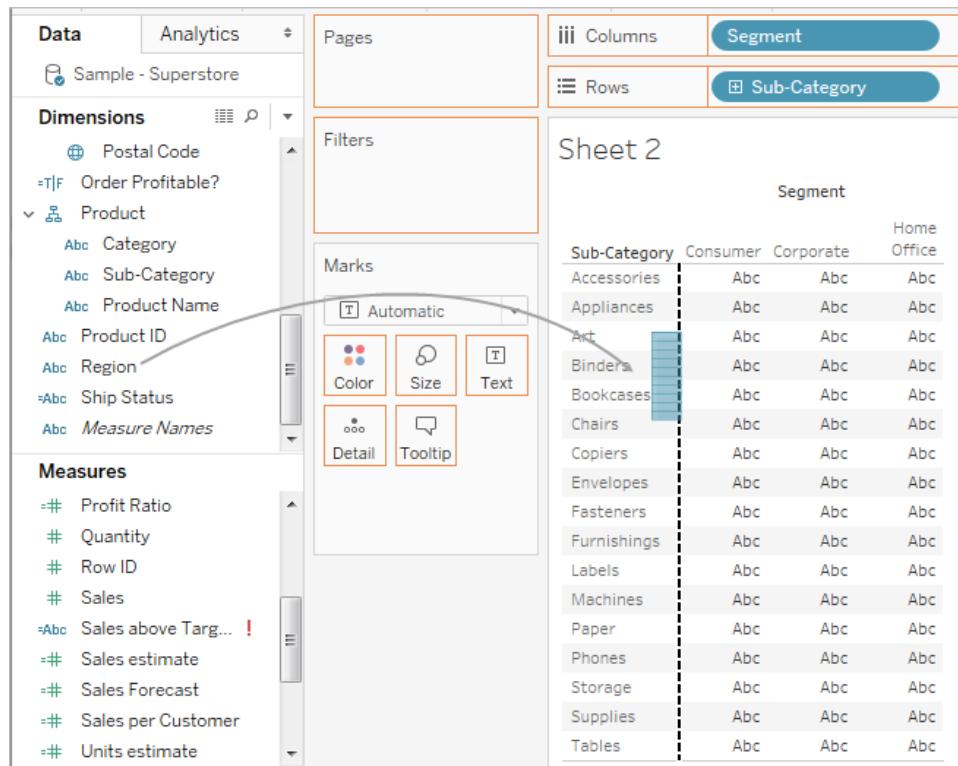
You can, of course, also drag fields directly to the shelves instead of to the active areas in the view. You can also drag fields from one shelf to another shelf. To remove a field from a shelf, drag it off the worksheet or select **Remove** from the field's menu (available when you right-click (control-click on Mac) a field in the view. To quickly remove multiple fields from a shelf, right-click (control-click on Mac) the shelf and select **Clear Shelf**.

Adding More Fields

You can add as many fields as necessary by dragging and dropping them on the different areas of the view. Once there are more fields in the view there are some extra options available. For example you can replace fields by dropping them on existing headers and axes in the view. Or, instead of replacing a field you can add measures to an existing axis. Finally, you can rearrange the rows and columns in the view.

Adding Headers Using Dimensions

You can add headers to a view by dragging a dimension and dropping it to either side of an existing header, or to the left of an axis. For example, in the view below you can add the **Region** dimension by dragging it and dropping it to the right of the Sub-Category names.



As you hover over the view, a dotted black line indicates active areas where you can add headers.

If you drop **Region** there, it adds a column for Region along the top of the table, and slices the data so that you see results for each region within each sub-category.

The screenshot shows a Tableau interface with the following configuration:

- Columns:** Sub-Category, Region, Segment
- Rows:** Accessories, Appliances, Art, Binders
- Segment:** Consumer, Corporate, Home Of..

The data table is as follows:

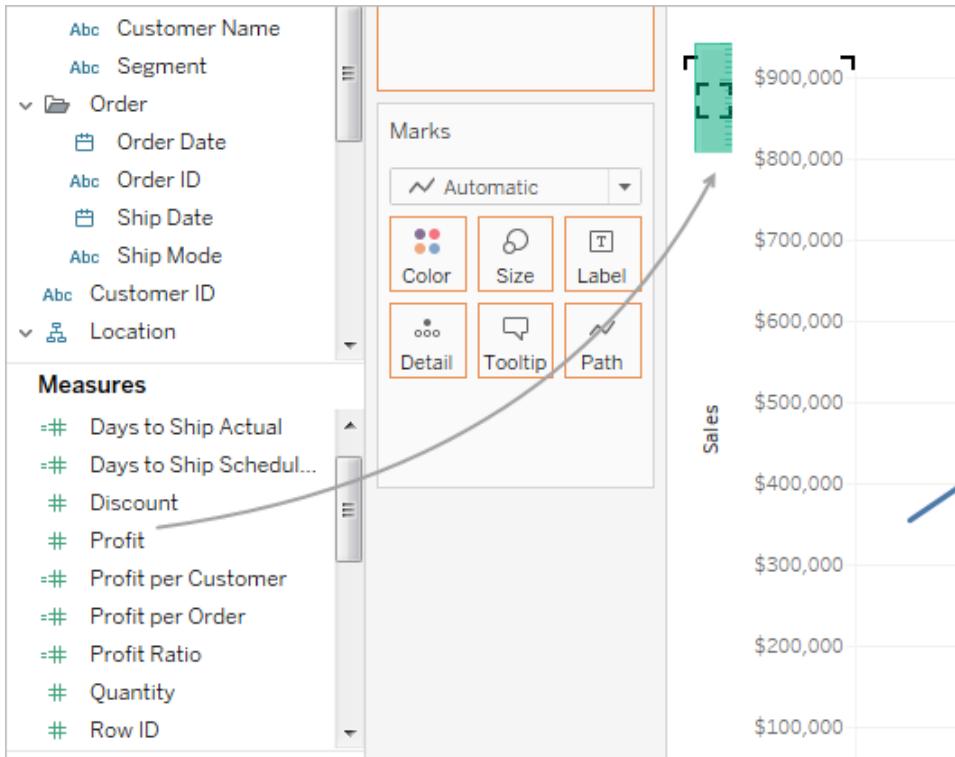
Sub-Category	Region	Consumer	Corporate	Home Of..
Accessories	Central	Abc	Abc	Abc
	East	Abc	Abc	Abc
	South	Abc	Abc	Abc
	West	Abc	Abc	Abc
Appliances	Central	Abc	Abc	Abc
	East	Abc	Abc	Abc
	South	Abc	Abc	Abc
	West	Abc	Abc	Abc
Art	Central	Abc	Abc	Abc
	East	Abc	Abc	Abc
	South	Abc	Abc	Abc
	West	Abc	Abc	Abc
Binders	Central	Abc	Abc	Abc
	East	Abc	Abc	Abc

Adding Axes Using Measures

You can add axes by dragging a measure and dropping it on an active area in the view. If an axis already exists in the view you can replace the existing axis, blend the new measure with the existing axis, or add a secondary axis.

Replace the Existing Axis

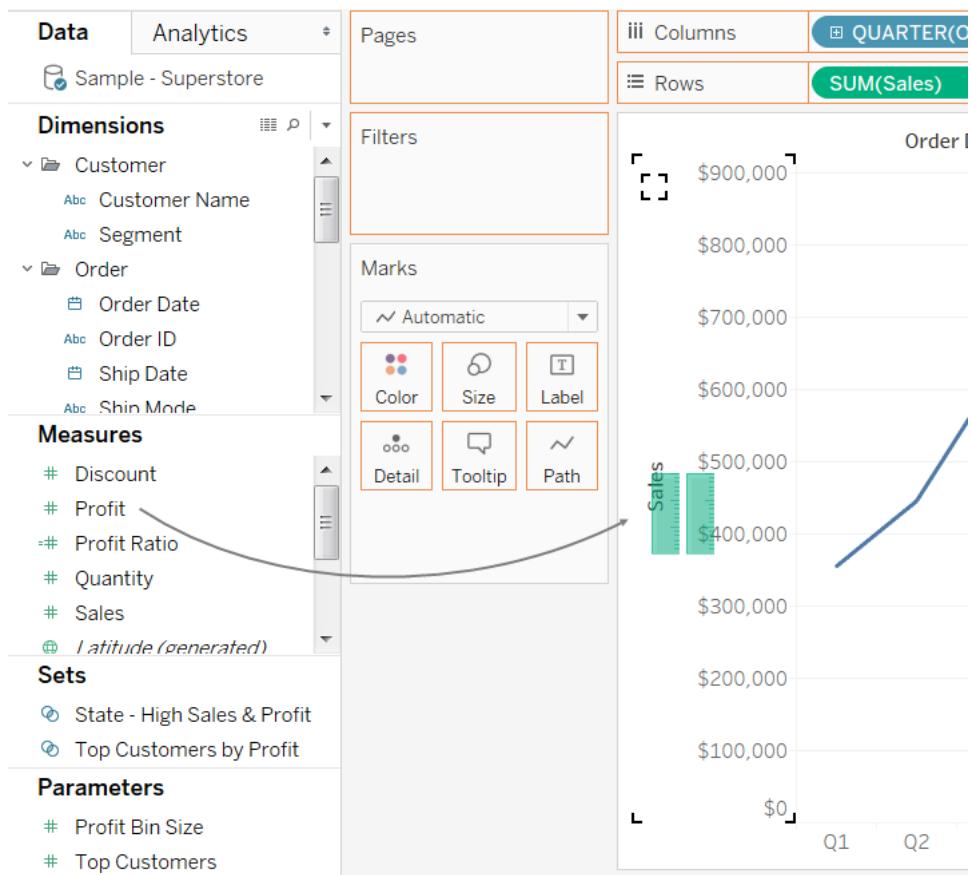
Drag the new measure to the top left portion of the axis in the view. A small square drop zone appears and a single axis icon displays to indicate that a single axis will be left when you drop the measure. In this case you are replacing one measure (**Sales**) with another (**Profit**).



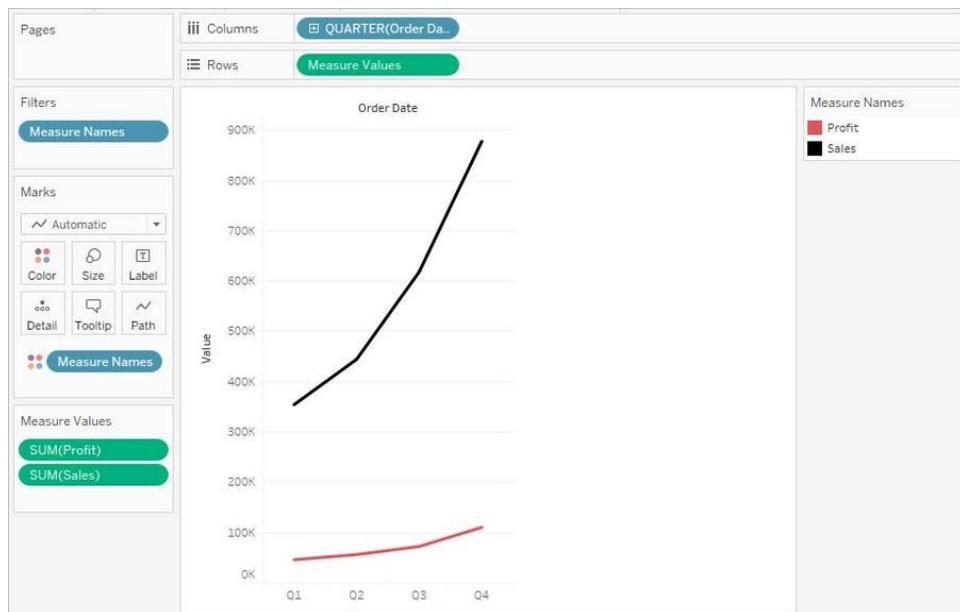
Blend the Measures on Single Axis

You can show multiple measures on a single axis by dragging the new measure directly on top of the existing axis. Blending measures uses the Measure Names and Measure Values fields. For more information, see [Measure Values and Measure Names](#) on page 302.

In the image below you are adding a second measure (**Profit**) to the existing measure (**Sales**) on the axis.

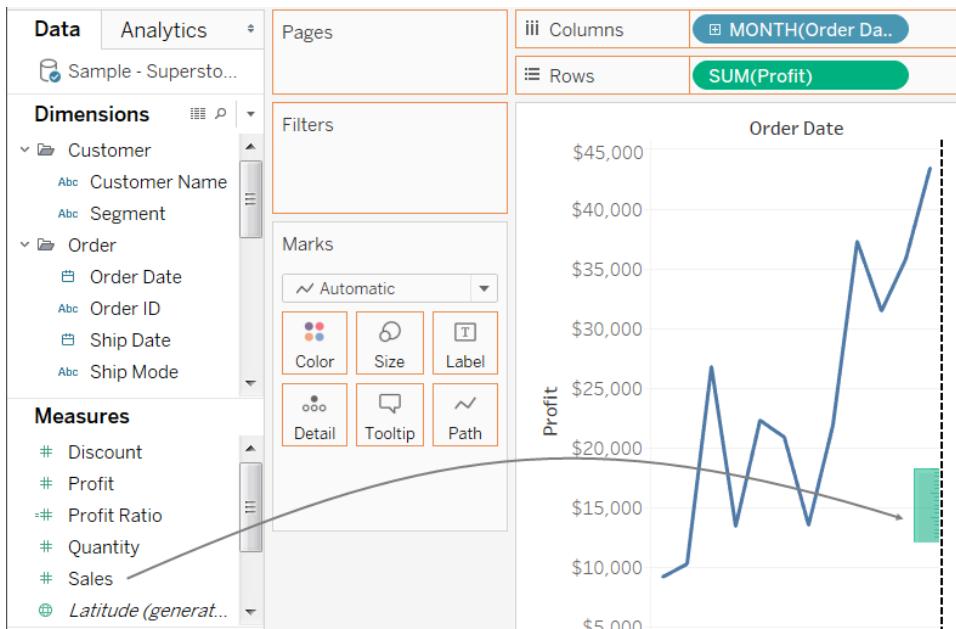


The result is a view with both measures plotted along a single axis. In this case, Sales is shown in orange and Profit is shown in blue.

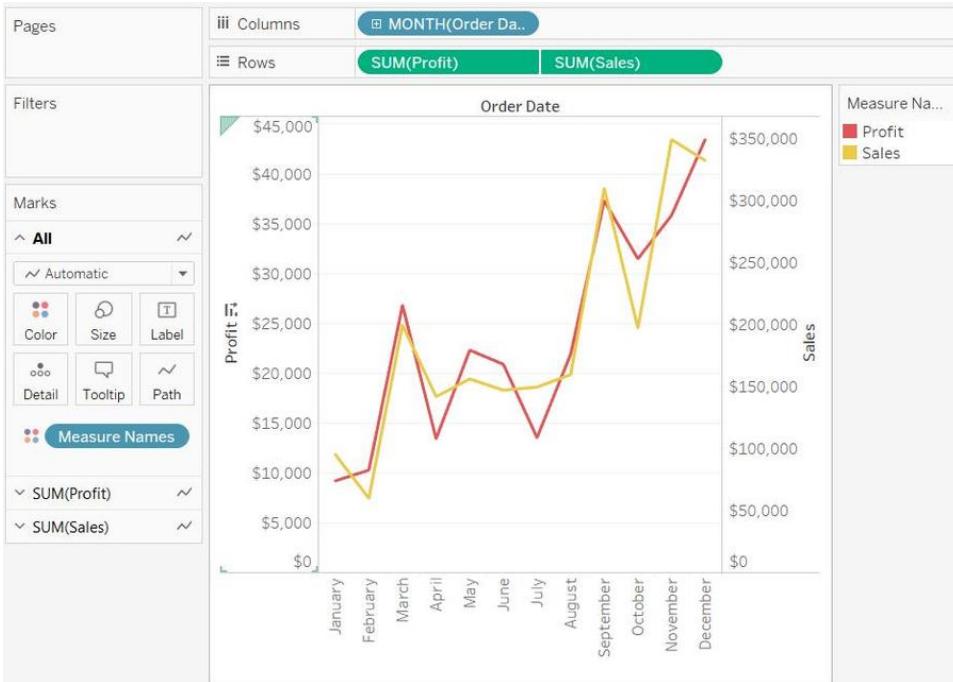


Add a Dual Axis

Drag the field to the right side of the view to add the measure as a dual axis. Dual Axes are useful when you want to compare two fields that have different scales. In this case, blending the two axes would distort the view. Instead you can add a dual axis.



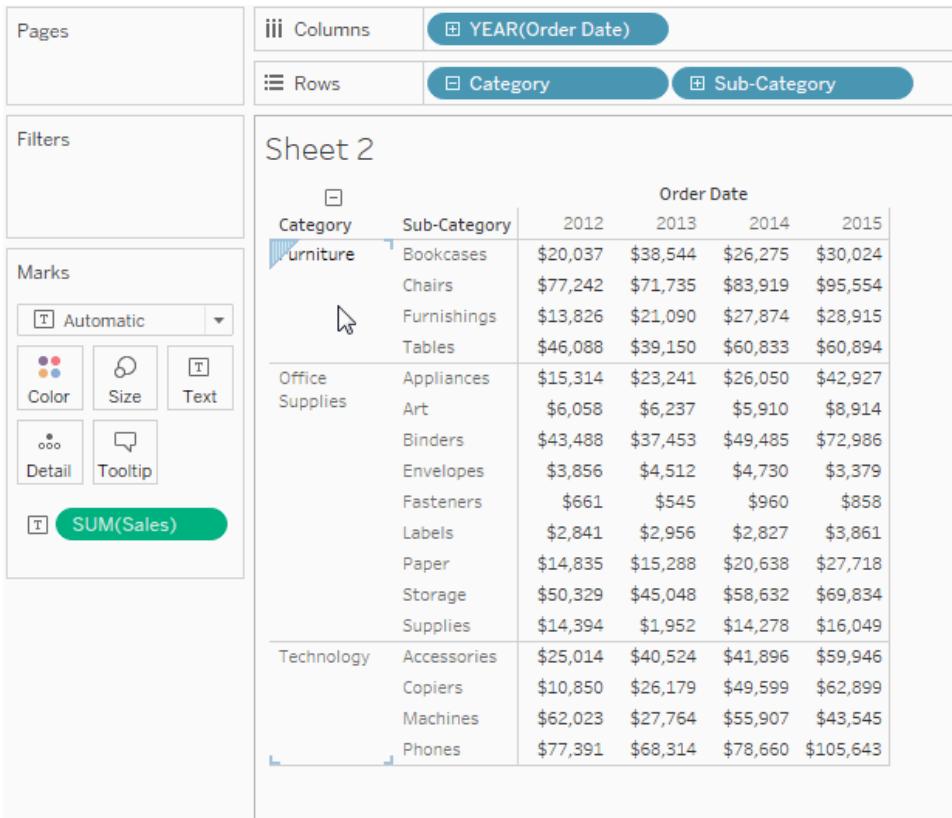
The result is a dual axis view where the Profit axis corresponds to the blue line and the Sales axis corresponds to the orange line.



You can add up to four axes to the view: two on the Columns shelf and two on the Rows shelf. To turn a measure into a dual axis, right-click (control-click on Mac) a field on the **Rows** or **Columns** shelf and select **Dual Axis**.

Rearrange the Rows and Columns

Finally, you can rearrange the rows and columns in the view by dragging the selection border for headers or an axis.



Sorting

In Tableau, sorting a data view means arranging dimension members in a specified order. Tableau supports computed sorting and manual sorting.

Computed Sorting

You might want to sort customers by alphabetical order, or sort a product line from lowest sales to highest sales. Both of these sorts are “computed sorts” because they use programmatic rules that you define to sort the field.

About Computed Sorting

Sorting dimensions in a computed manner follows these rules:

- You can sort any discrete field after it has been placed on a shelf (except the **Filters** shelf).
- Each dimension that appears on a worksheet can be sorted independently of any other dimension.
- The shelf location of the dimension determines the component of the data view that's sorted. For example, if the dimension resides on the Columns shelf, the columns of the

data view are sorted for that field. If the dimension resides on the **Color** shelf, the color encodings are sorted.

- Sorts are computed based on the values of the filters and sets in the view. Refer to [Groups on page 497](#) for more information.

- Sorted fields are identified by a sort icon  on the right side of the field.

Continuous fields are automatically sorted from lowest number to highest number (as indicated by the axes) and you cannot manually change the sort. However, you can reverse the order of an axis using field specific formatting.

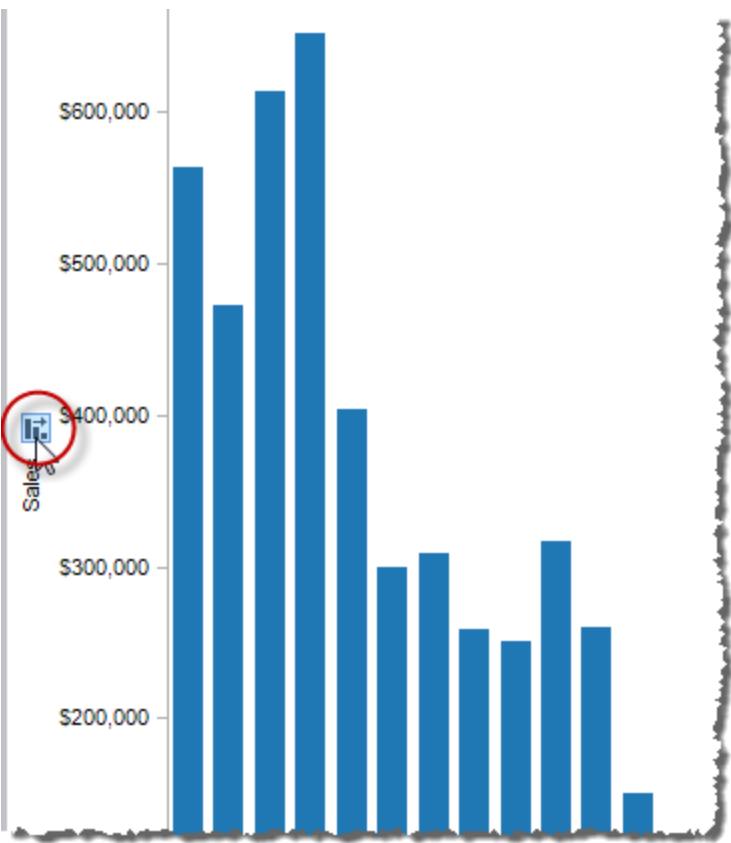
[How to Sort Data \(Computed Sorts\)](#)

Computed sorts can be applied either directly on an axis in the view or using the sort dialog box to apply computed sorts to specific fields in the view.

[Sorting on an Axis](#)

A quick way to create a computed sort is to use the sort buttons on an axis. The sort buttons on an axis will automatically create a sort that makes sense for the view you've created. The sort will be computed, which means it will update correctly if the underlying data changes.

When you hover over an axis with your cursor, a sort icon displays. Click it once to sort Ascending, again to sort Descending, and a third time to clear the sort.

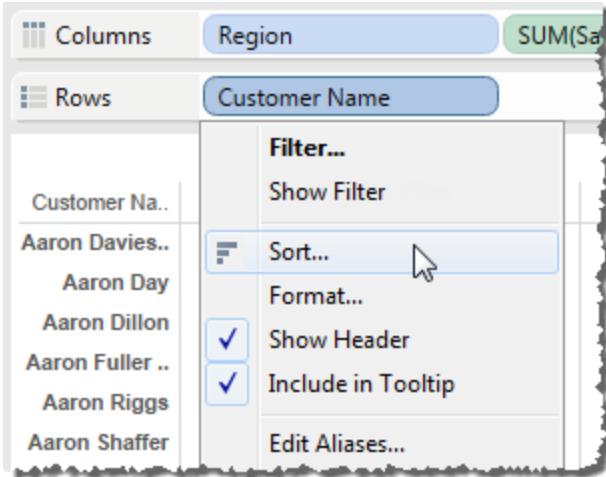


Sorting Specific Fields

You can sort a specific field using the Sort dialog box.

1. Open the Sort dialog box.

Right-click (control-click on Mac) on the field that you want to sort and select **Sort** from the context menu.



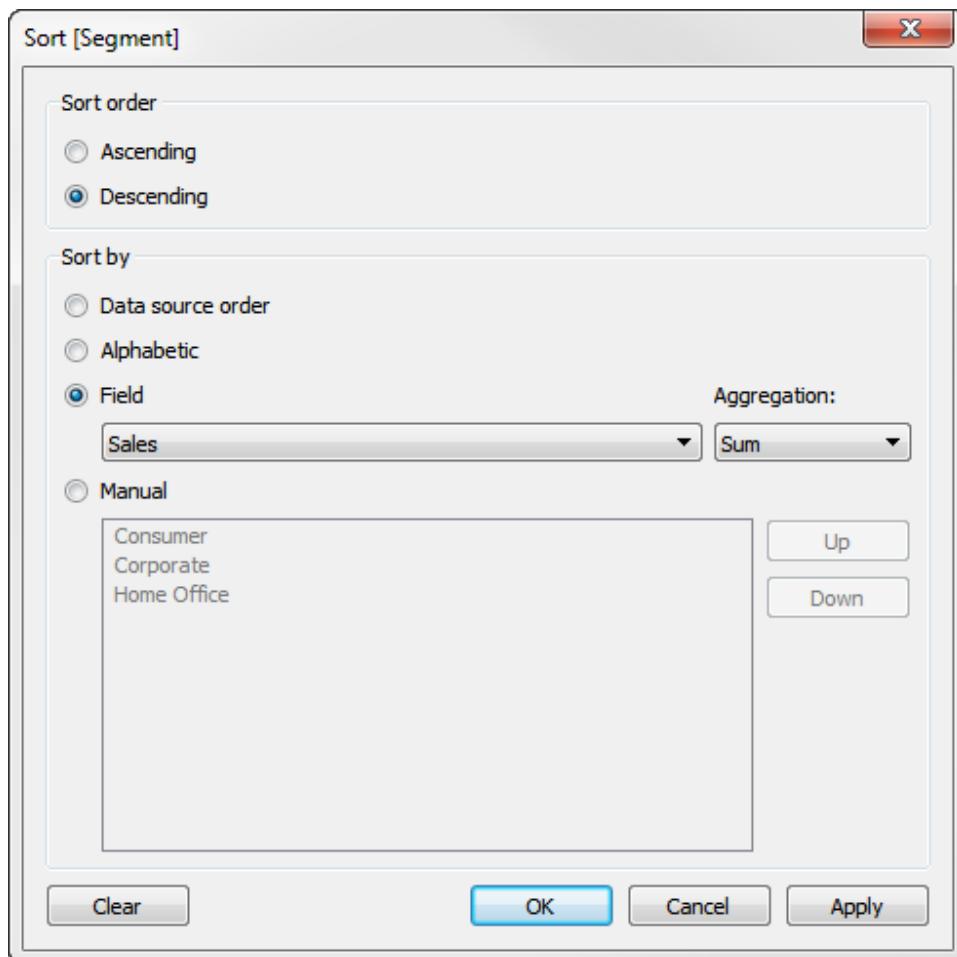
2. Specify the sorting options.

Complete the **Sort** dialog box by specifying the following criteria:

- Sort order - Displays the sort results in ascending or descending order.
- Sort by – Sort by one of these three options:
 - **Data source order** - the order that the data source naturally orders the data. Generally for relational data sources, this tends to be in alphabetical order. If you are using a cube, this order is the defined hierachal order of the members within a dimension.
 - **Alphabetic** - the order of the letters in the alphabet.
 - **Field** - order the data based on the associated values of another field. For example, you could order several products by their total sales values.

When working with a relational data source and sorting by another field, you must also specify the aggregation function to use. This option is not available for multidimensional data sources because aggregations are defined when the cube is created and cannot be modified in Tableau. In Tableau, multidimensional data sources are supported only in Windows.

A typical scenario is to sort one or more dimensions by a measure. For example, the **Sort** dialog box shown below is configured to sort the members of the Customer Segment field in descending order and by the sum of the Sales measure. The results will be displayed so that the member with the highest sales is displayed first, the member with the second highest sales is displayed second, and so on.



You should keep the following rules in mind when interpreting the sort results:

- Tableau computes the sort across the entire table using the specified criteria. Refer to [Example – Sorting a Text Table](#) below for an example.
- Sorts do not break the dimension hierarchy. Sorted fields are always displayed within the ordered context already set forth by the fields on the Rows and Columns shelves. This means that Tableau will not rearrange any of the headers of the fields that appear before (to the left of) the sorted field.

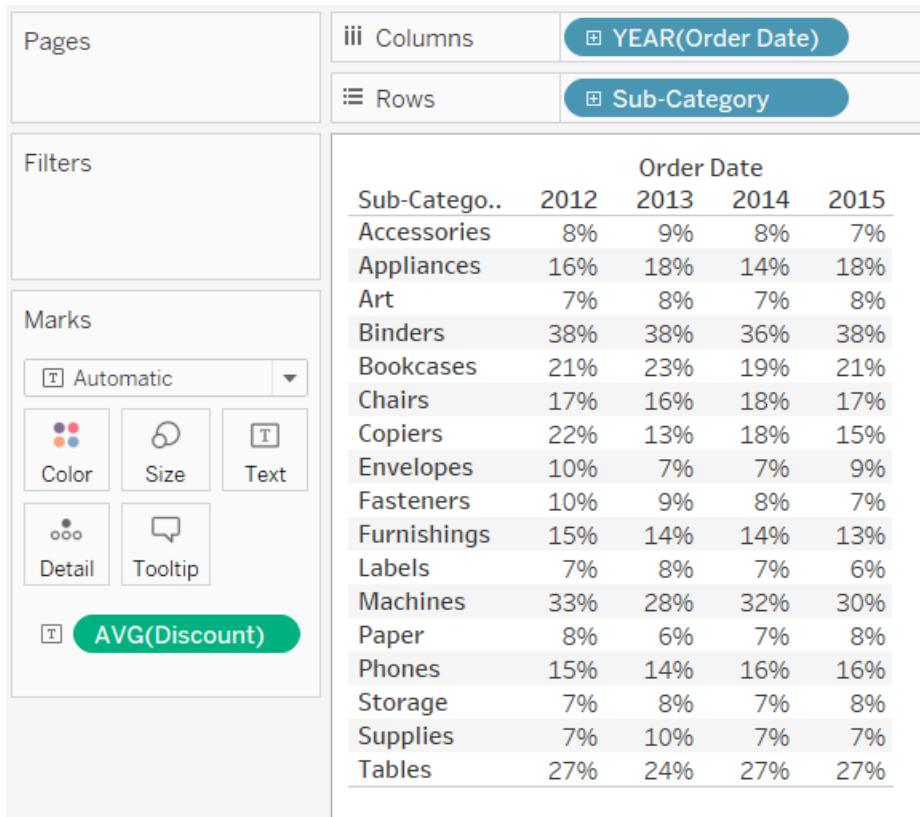
If you want to break the dimension hierarchy when sorting a multidimensional data source, place only the hierarchy level that you want sorted on the Rows or Columns shelf.

[Example – Sorting a Text Table](#)

Using the **Sample-Superstore** data source, this example sorts the rows and columns of a text table to determine which products and years have the highest average discounts. To create the view, follow the steps below:

1. Place the **Order Date** dimension on the **Columns** shelf and the **Sub-Category** dimension on the **Rows** shelf.

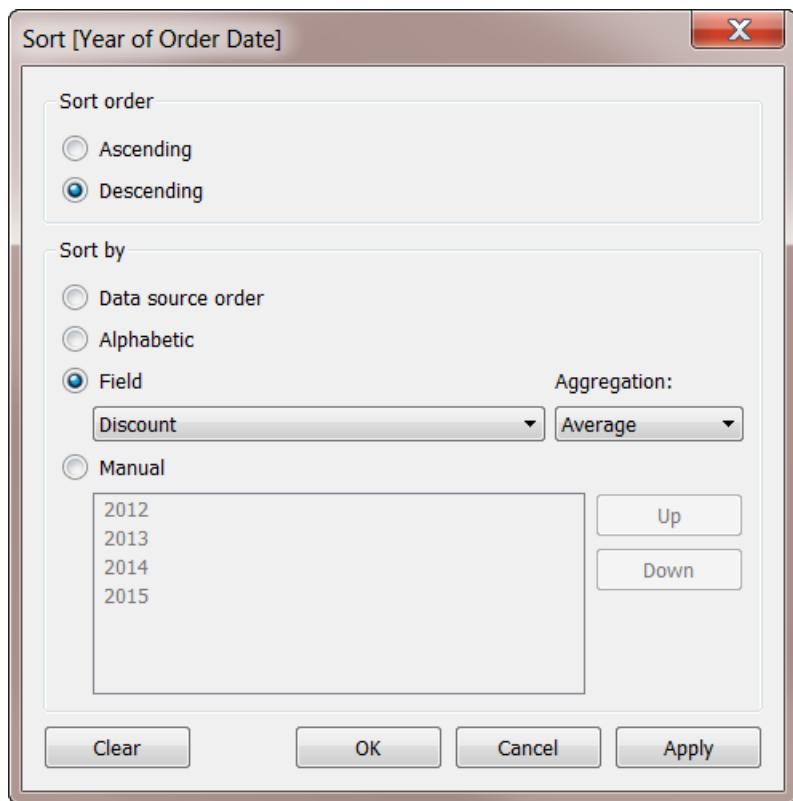
Complete the text table by placing **Discount** on **Text**. By default, the table is sorted in alphabetical order.



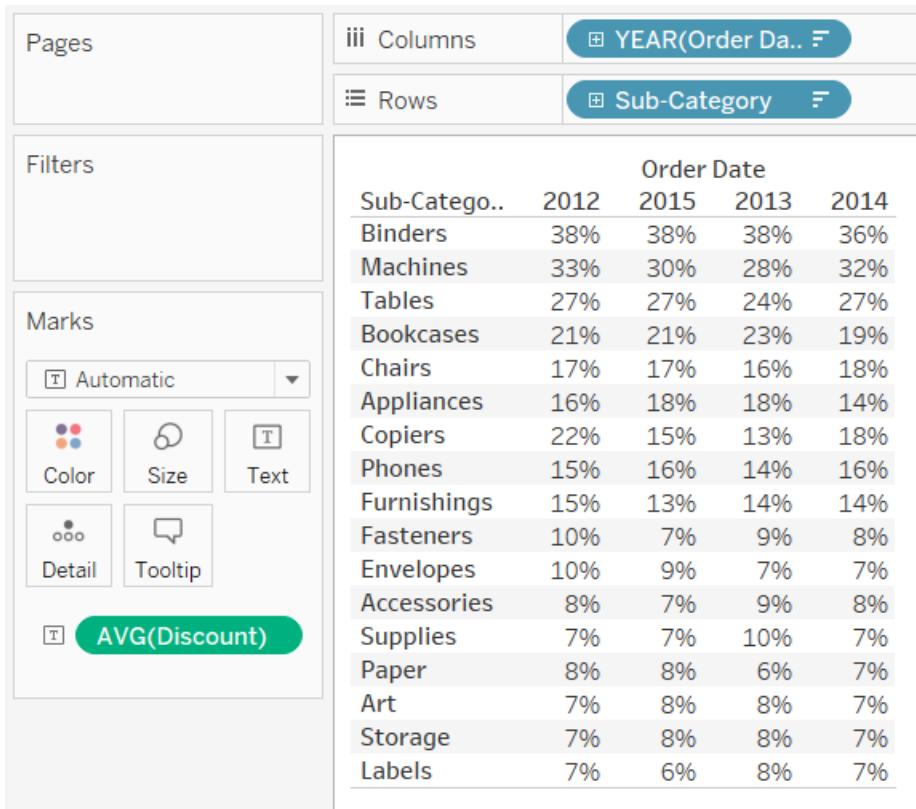
2. Sort the fields.

Right-click (control-click on Mac) **Order Date** on the Columns shelf and select **Sort** from the context menu. In the Sort dialog box, do the following:

- For Sort order, select **Descending**.
- For Sort by, select **Field**, and then select the following:
 - For Field, select **Discount**.
 - For Aggregation, select **Average**.
- Click **OK**, and then apply the same sort to **Sub-Category**.



The view is shown below. Binders are the top row in the table because it has the largest average discount across all years, while Labels are at the bottom in the table because that sub-category has the smallest average discount across all years. Similarly, 2012 is the left most column because it has the largest average discount for all products, while 2014 is the right most column because it has the smallest average discount for all products.



At first glance, it's not clear if the data has been correctly sorted, because Tableau computes the sort across the entire table using the specified criteria. By turning grand totals on for both columns and rows, using the **Analysis > Totals** menu, you can see that the sort was performed correctly. For more information about how to turn on grand totals, see **Totals** on page 976.

The screenshot shows the Tableau Data Editor interface. On the left, there's a sidebar with 'Pages' (selected), 'Filters', and 'Marks' sections. Under 'Marks', 'Automatic' is selected, and 'Color', 'Size', 'Text', 'Detail', and 'Tooltip' are listed. A green button labeled 'AVG(Discount)' is also present. The main area displays a data table with the following columns: Sub-Catego.., 2012, 2015, 2013, 2014, and Grand Total. The table lists various product categories and their sales percentages across four years. The 'Grand Total' column shows the cumulative percentage for each year.

Sub-Catego..	Order Date				
	2012	2015	2013	2014	Grand Total
Binders	38%	38%	38%	36%	37%
Machines	33%	30%	28%	32%	31%
Tables	27%	27%	24%	27%	26%
Bookcases	21%	21%	23%	19%	21%
Chairs	17%	17%	16%	18%	17%
Appliances	16%	18%	18%	14%	17%
Copiers	22%	15%	13%	18%	16%
Phones	15%	16%	14%	16%	15%
Furnishings	15%	13%	14%	14%	14%
Fasteners	10%	7%	9%	8%	8%
Envelopes	10%	9%	7%	7%	8%
Accessories	8%	7%	9%	8%	8%
Supplies	7%	7%	10%	7%	8%
Paper	8%	8%	6%	7%	7%
Art	7%	8%	8%	7%	7%
Storage	7%	8%	8%	7%	7%
Labels	7%	6%	8%	7%	7%
Grand Tot..li.	16%	16%	16%	15%	16%

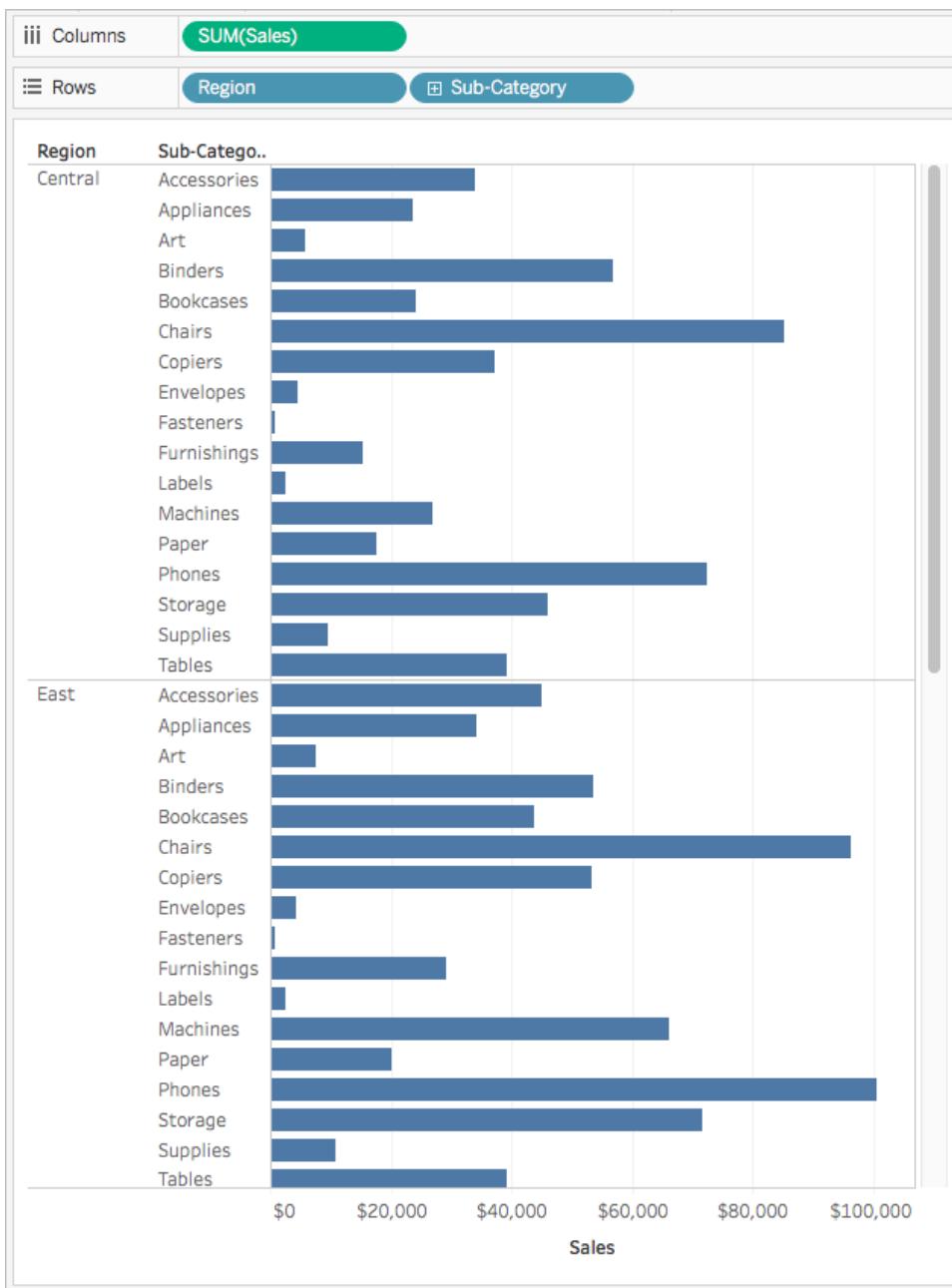
Example—Create a Nested Sort

You can create nested sorts in Tableau.

For example, suppose you are looking at sales sub-categories for all regions in your country. In the view below, you can see that the Central region sells more products in the Chairs subcategory than any other subcategory. However it's harder to compare how Chairs ranked in other regions.

To see which subcategories ranked the highest in each region, you might want to sort the subcategories within each region instead. You can do this with nested sorting.

Follow along with the example below to learn how to create a nested sort in Tableau.



Set up the view

1. In Tableau Desktop, open a new workbook and connect to the **Sample Superstore** data source that comes with Tableau.
2. Navigate to a new worksheet.
3. From Measures, drag **Sales** to the **Columns** shelf.
4. From Dimensions, drag **Region** to the **Rows** shelf.

5. From Dimensions, drag **Sub-Category** to the **Rows** shelf, to the right of the Region field.

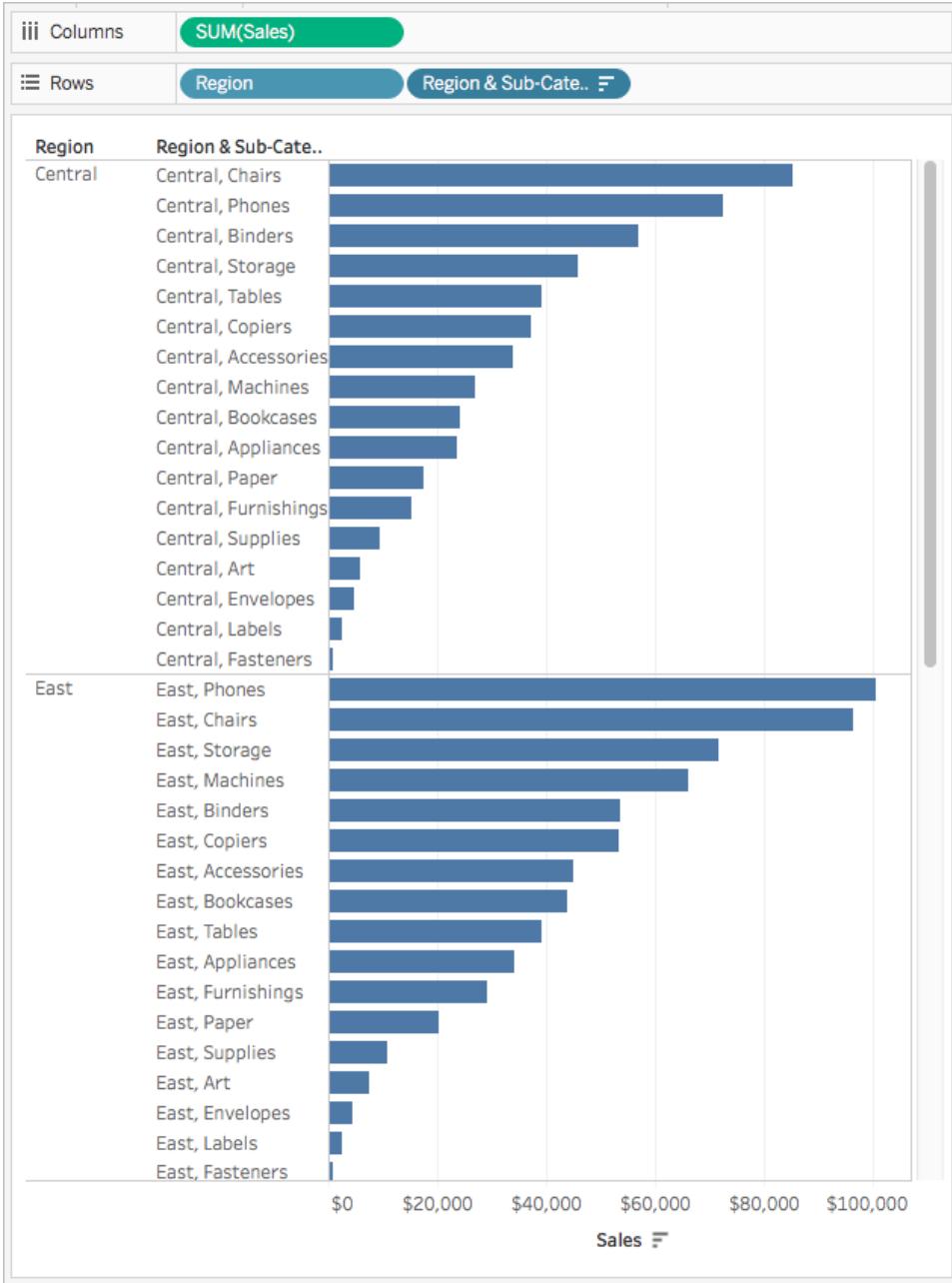
Create a nested sort

1. In the Data pane, select the **Region** and **Sub-Category** fields.

Note: To select multiple fields, press and hold the CTRL key on your keyboard as you select the fields (On Mac, press and hold the Command key).

2. In the Data pane, right-click the selected fields and select **Create > Combined field**.
3. Remove the **Sub-Category** field from the **Rows** shelf.
4. From Dimensions, drag the combined field (**Region & Subcategory**) to the **Rows** shelf.
5. On the Rows shelf, right-click the **Region & Sub-Category** field and select **Sort**.
6. In the Sort dialog box, do the following:
 - For **Sort order**, select **Descending**.
 - For **Sort by**, select **Field**, click the drop-down, and then select **Sales**.
Aggregation should be set to **Sum**.
 - Click **OK**.

Now you have a nested sort where the subcategories are listed in descending order of sum of sales within each region. You can see that Chairs are the highest selling subcategory in the Central region, but in the East region, Phones are the best selling product; Chairs are the second-best selling product.



Note: You might also want to change the formatting so that the row dividers are at the row level (sub-category) instead of on the pane (region) level. To do so:

1. Select **Format > Borders**.
2. In the Format Borders pane, under **Row Dividers**, move the **Level** slider to the right.

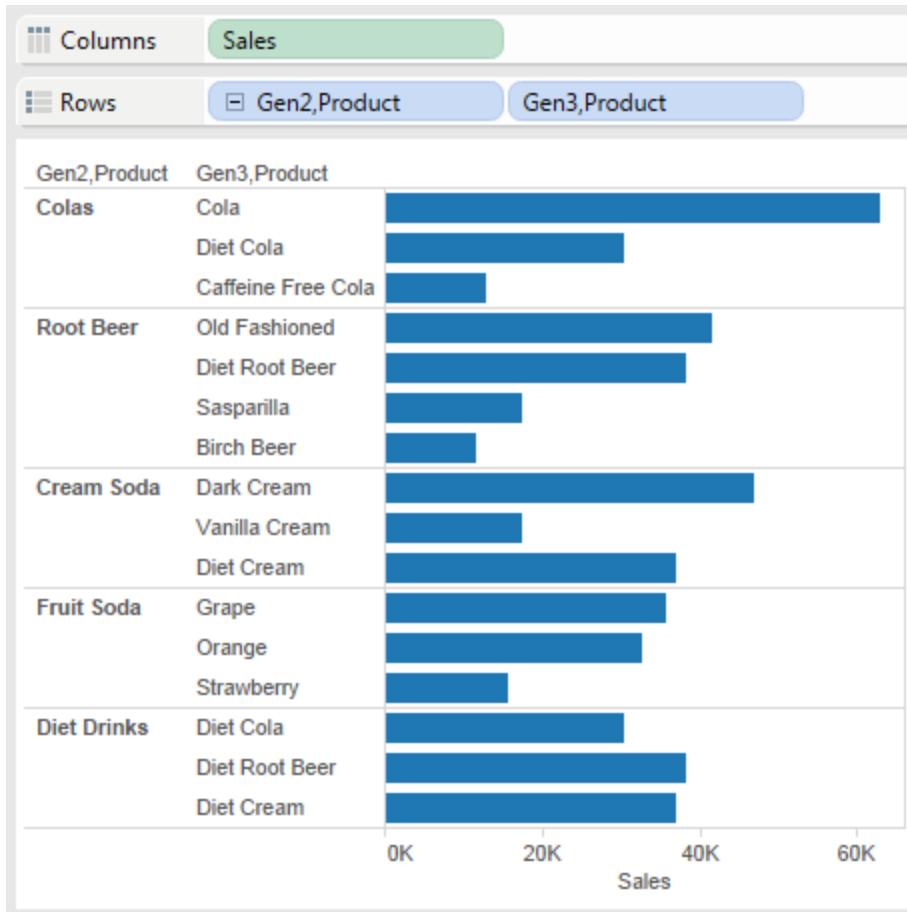
Example – Sorting a Hierarchy

This example uses a multidimensional data source to sort the rows of a bar chart in order to determine which beverages have the highest sales. To create the view, follow the steps below.

Note: In Tableau, multidimensional data sources are supported only in Windows.

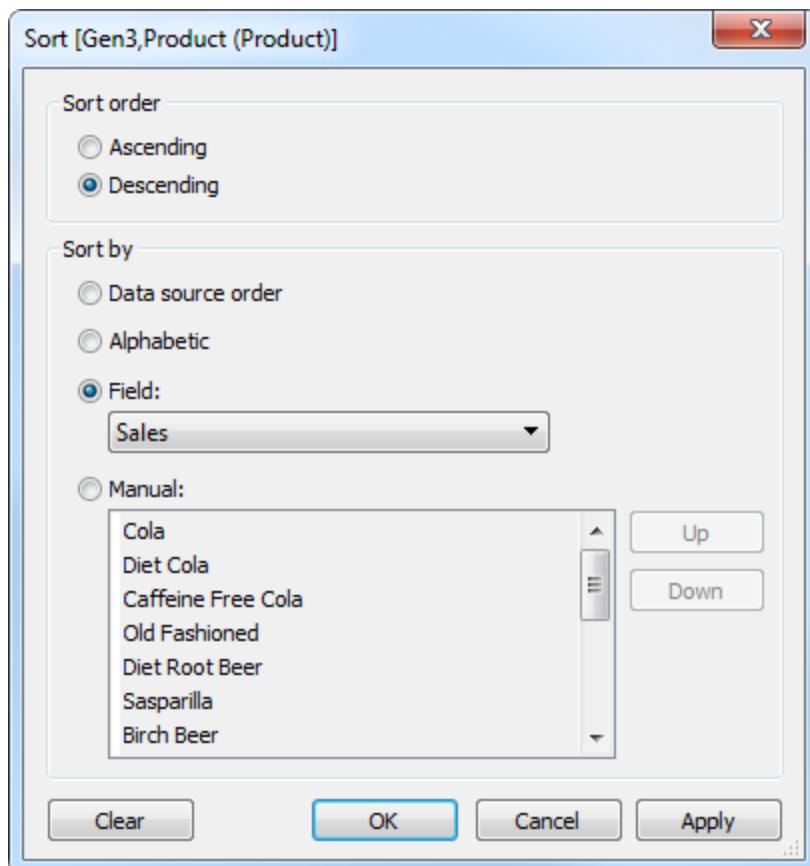
1. Place the **Sales** measure on the Columns shelf and the **Gen2,Product** dimension on the Rows shelf.

Drill down one level in the hierarchy to display **Gen3,Product**.



2. Sort **Gen3,Product** in descending order by the **Sales** measure.

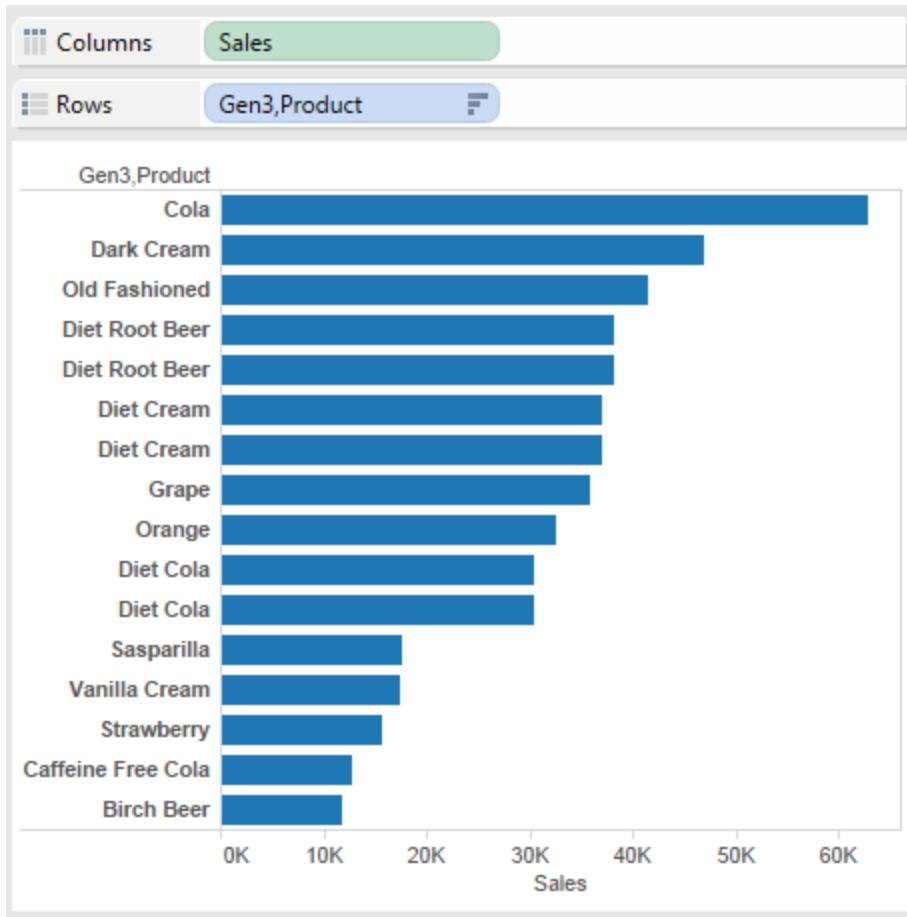
Right-click (Control-click on Mac) on Gen3,Product and select **Sort** from the field's context menu. In the Sort dialog box, select **Descending** as the Sort order and sort by the **Sales** field.



The view is shown below. Notice that the **Gen3,Product** members are sorted within each parent member. For example, Cola, Diet Cola, and Caffeine Free Cola are sorted only within the Colas level. Tableau does not rearrange headers that appear before the sorted field.



3. If you want to order dimension members without regard to its parent, you should remove Gen2,Product from the Rows shelf. The sorted data are shown below.



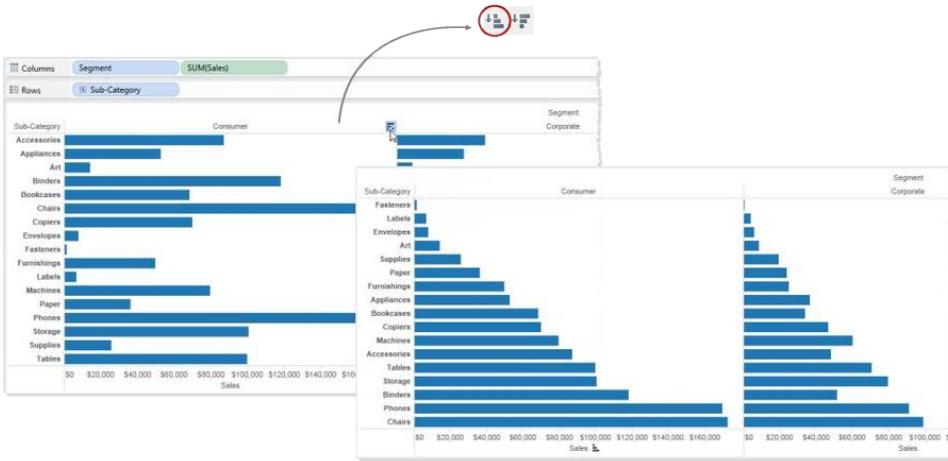
Manual Sorting

Manual sorting allows you to rearrange the order of dimension members in the table by dragging them in an ad-hoc fashion, giving precise control over how items appear next to one another in tables and in legends. It also gives you control over the order in which data is drawn on the screen. This control is useful when comparing specific pieces of data or interpreting overlapping data. Manual sorts can only be applied to discrete fields including a discrete measure.

There are two ways to manually sort the data in a view. You can either select items in the view and use the Sort toolbar buttons or you can drag and drop headers in the view.

Sorting Using the Toolbar and Tooltips

The two sort buttons on the toolbar and in tooltips manually sort a selection either in ascending or descending order based on the other fields in the view. For example, the view below shows sales by product category and customer segment. When you select the Corporate Market column, thus selecting all of the products in that segment, the quick sort buttons sorts the product field by SUM(Sales), which is the measure in the view.

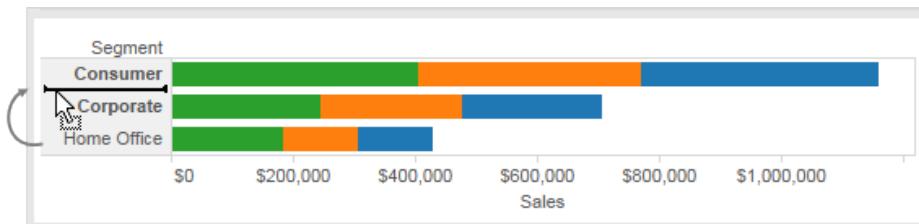


An easy way to anticipate how a selection will be sorted is to make a selection in the view and hover over the ascending or descending sort buttons on the toolbar. The tooltip for each button describes how the selection will be sorted.

Using the quick sort buttons on the toolbar or in the tooltip creates a manual sort which you can always modify using the sort dialog box. Right-click (control-click on Mac) a sorted field (indicated with bold text) and select **Sort** to open the Sort dialog box.

Sort by Drag and Drop

1. Select the dimension member you want to move. This can be any dimension member that appears in a row or column header of a table, or in a legend like the color legend.
2. Drag the member to the desired location within that row, column or legend.

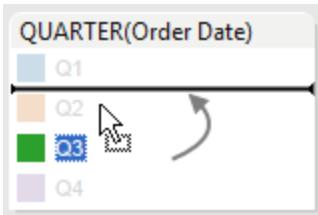


Example- Manually Sorting Drawing Order

Changing the drawing order of a field allows you to see obscured data in your views in cases where data of one color or shape obscure data of another color or shape. For instance, if you can't see red marks in a scatter plot because they are obscured by green marks, you can change the drawing order so that the red points are drawn on top of the green points (and vice versa).

Change the drawing order of a field by re-arranging the order of dimension members in a legend. For instance, if you want to place red items in front of green items in a view, select the red legend entry and move it higher on the list of items shown in the legend. The marks are

drawn in the view according to the order in the legend, from bottom to top. Also you can toggle back and forth between layered field items by dragging any one of the fields from top to bottom or from bottom to top.



Sorting the drawing order is not restricted to color legends. You can reorder shape legends as well. If you have multiple valid legends, the drawing order is defined first by shape, then by color. For example, suppose you have both a shape legend and a color legend. If you have a red circle on top of a green square, moving the green above the red in the color legend will not necessarily move the green square on top of the red circle. It depends on the order in the shape legend first. If circles are above squares in the shape legend, no amount of reordering the color legend will get that square on top of the circle. Instead, move the square shape above the circle shape first and then reorder the color legend.

Groups

A group is a combination of dimension members that make higher level categories. For example, if you are working with a view that shows average test scores by major, you may want to group certain majors together to create major categories. English and History may be combined into a group called Liberal Arts while Biology and Physics may be grouped as Science Majors. Groups are useful for both correcting data errors (e.g., combining CA, Calif., and California into one) as well as answering "what if" type questions (e.g., "What if we combined the East and West regions?").

Creating Groups

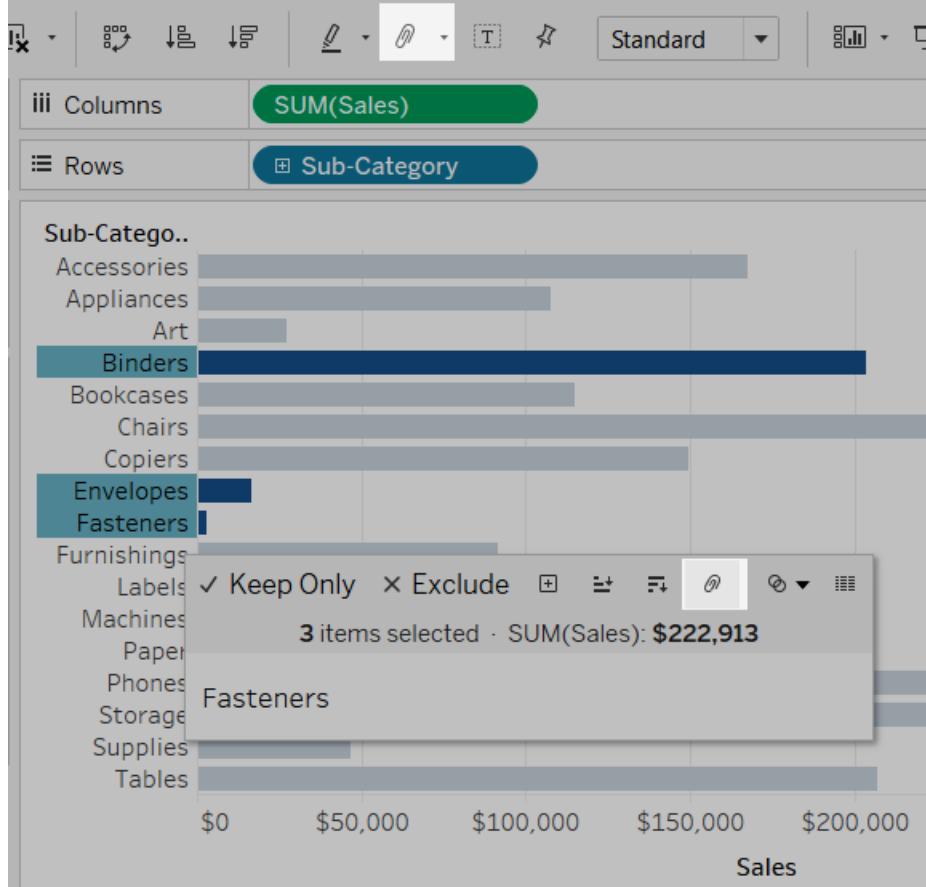
You can create a group by selecting headers in the view or selecting marks. You can also create a group from a dimension in the **Data** pane. Regardless of how you create a group, a new group field is added to the **Data** pane. You can use the group field like other fields in the view, adding it to the Columns or Rows shelves, to the Marks card, or to the Filters shelf. When you create a group by selecting headers or marks in the view, the view is updated to use the grouped field.

Correct data errors or combine dimension members

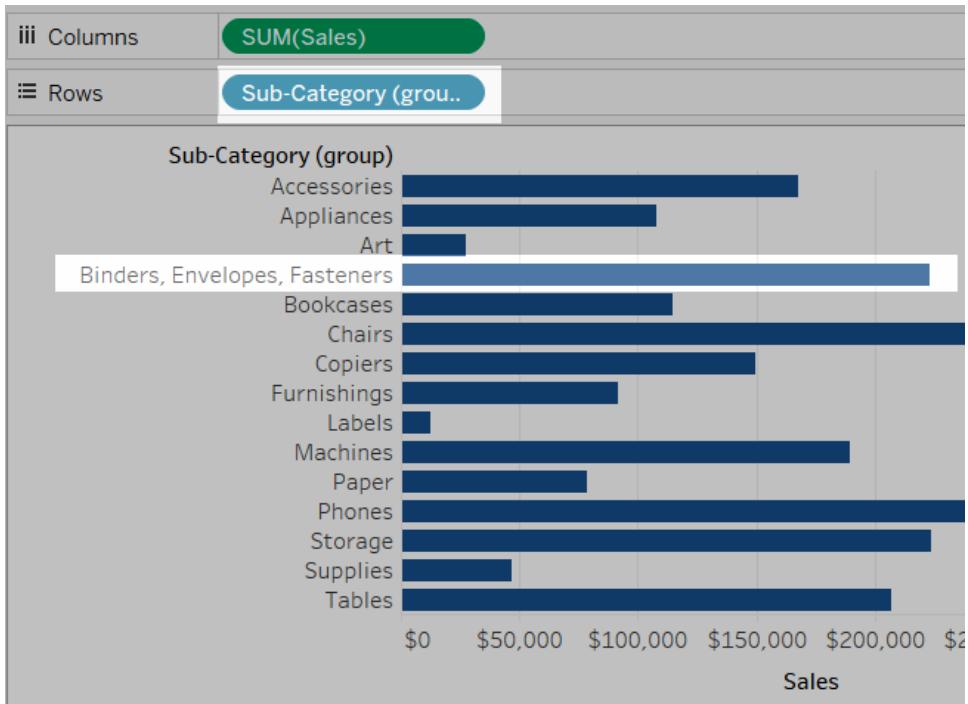
If you are using groups to correct data errors (for example, to combine "CA" and "California") or to combine dimensions members (for example, to combine "East" and "West" regions), the easiest option is to select headers in the view.

1. Press and hold the **CTRL** or **Shift** key on the keyboard to multi-select headers in the view. On a Mac, press and hold the Command key.
2. Click the **Group** button  on the toolbar.

Alternatively, you can right-click and select **Group** or click the **Group** icon on the tooltip.



The selected members are combined into a single member. In this example, the view now shows the SUM(Sales) across all binders, envelopes and fasteners as a single mark. A default is automatically constructed using the combined member names. The dimension on the Rows or Columns shelf is replaced with the new grouped field.

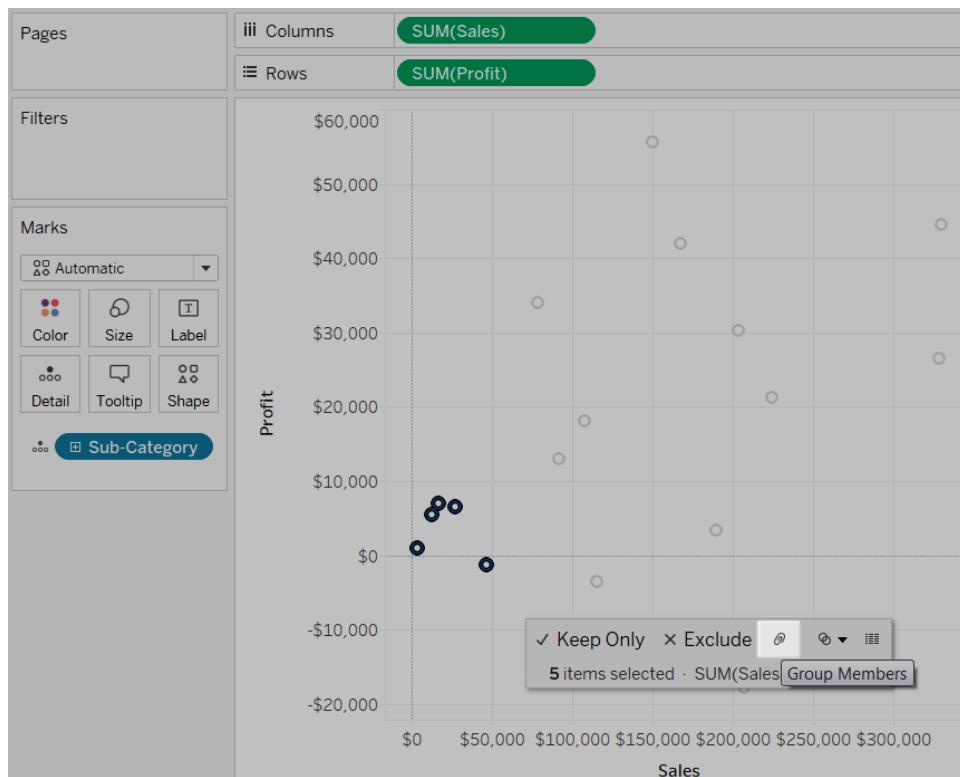


Color a view using groups

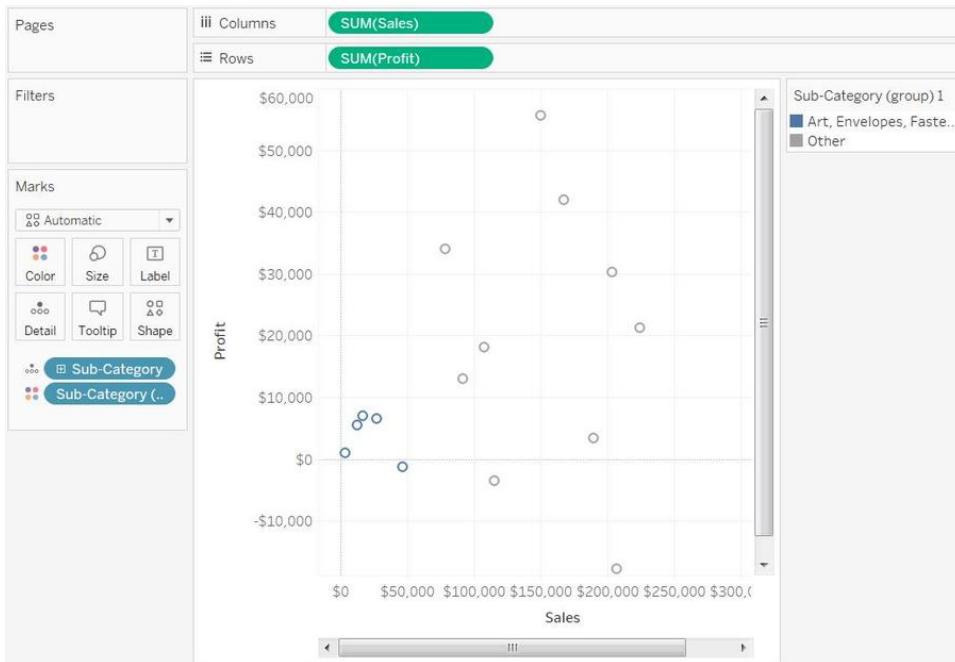
In addition to correcting data errors and combining dimensions, you can visually identify groups of related marks by selecting marks directly. This technique is especially useful when working with a scatter plot or a view that doesn't have headers that you can select when defining the group.

1. Press and hold the **CTRL** or **Shift** key on the keyboard to select one or more marks in the view. On a Mac, use the **Shift** key.
2. Click the **Group** button  on the toolbar.

Alternatively, you can right-click and select **Group** or click the **Group** icon on the tooltip.

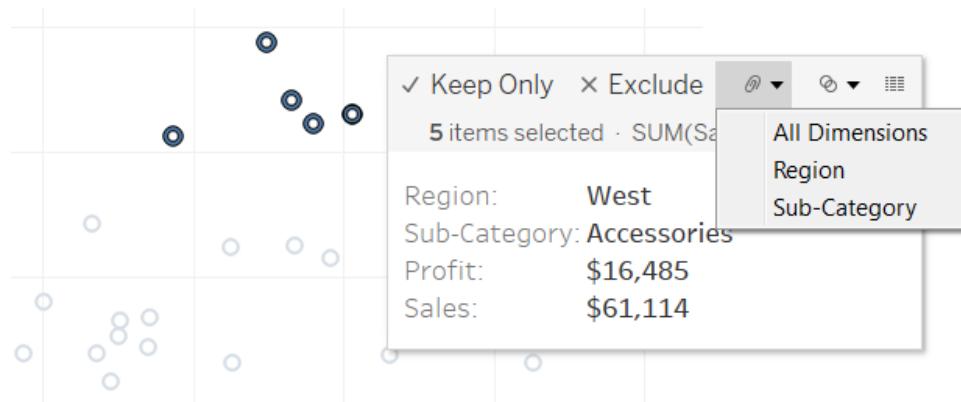


The selected marks are grouped and all other members are combined into an "Other" category. The new group field is automatically added to Color. For more information about grouping fields using Other, see [Including an Other Group on page 507](#)



Note: If there were already fields on **Color**, they are moved to **Detail** and replaced with the group field.

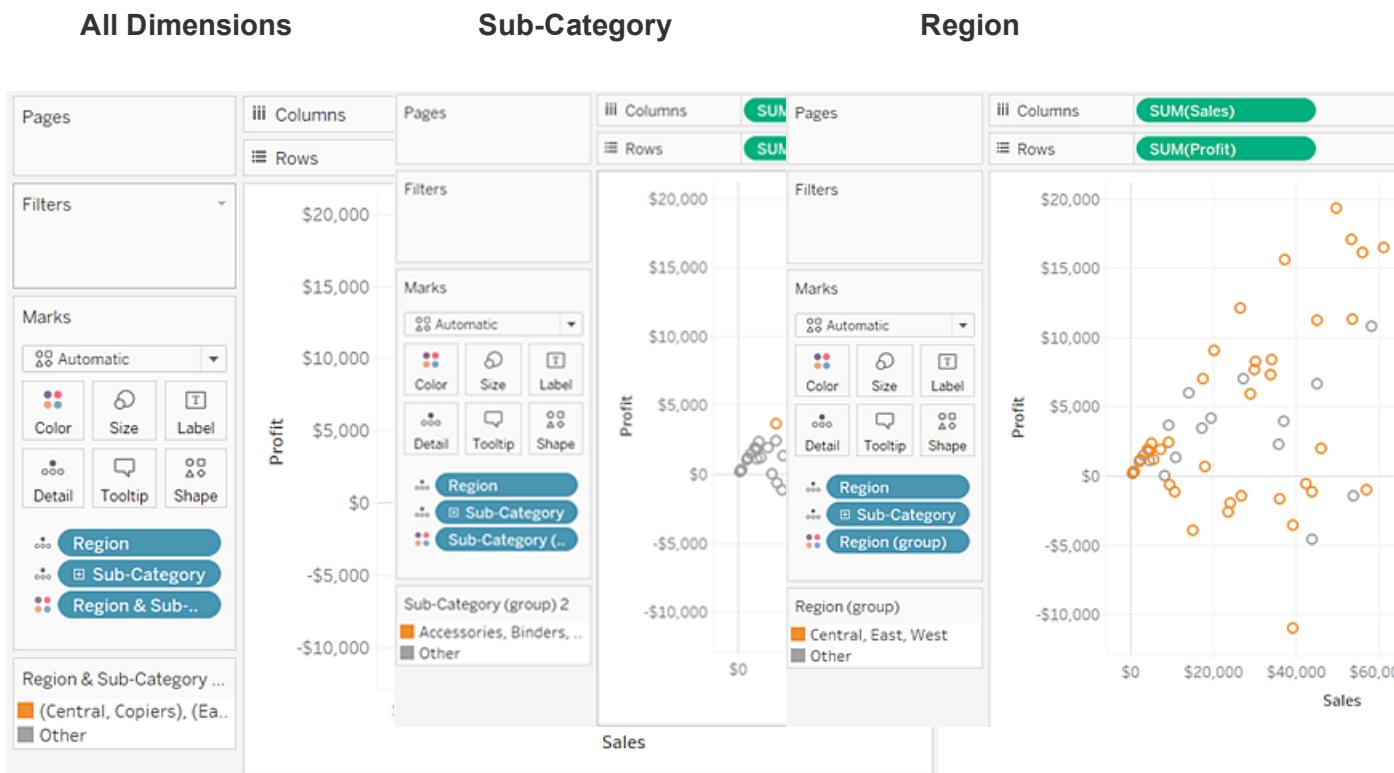
When you create groups by selecting marks, it's possible that the marks will represent multiple dimensions. For example, you may have a scatter plot that shows the Sales vs. Profit by Region and Sub-Category. In this case, a selection of marks will represent members of both the Region and Sub-Category dimensions. When the selection represents multiple dimensions, the **Group** menu lets you choose to group on **All Dimensions** or on a particular dimension.



In the example above, the 5 selected marks represent a region and the following sub-category combinations .

- West, Copiers
- West, Binders
- West, Accessories
- East, Copiers
- Central, Copiers
- South, Art
- South, Envelopes
- South, Labels
- West, Envelopes
- West, Labels

The views below show the results of grouping these measures on All Dimensions, Sub-Category, and Region.



The 5 marks are combined and the rest of the marks are added to an "Other" category.

All marks associated with any of the three sub-categories are combined and everything else is added to an "Other" category.

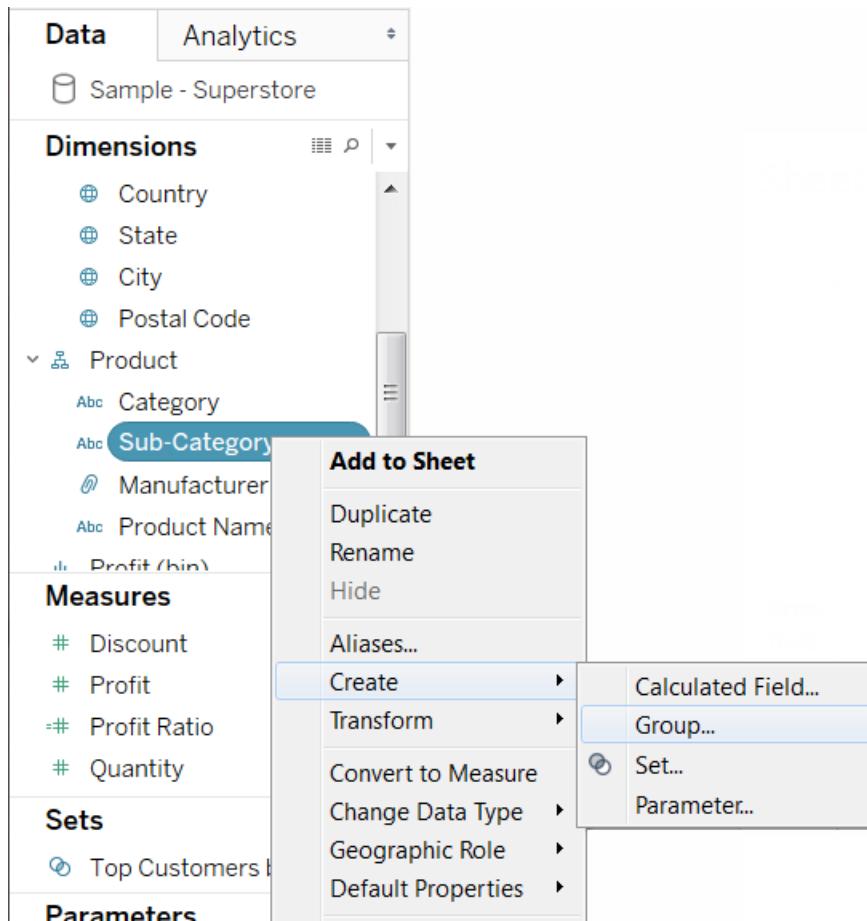
All marks associated with the four regions are combined and everything else is added to an "Other" category.

Create groups from a Dimension in the Data pane

Whenever you create a group, a group field is added to the **Data** pane. When you create a group by selecting headers or marks in the view, the new group field is also added to the view

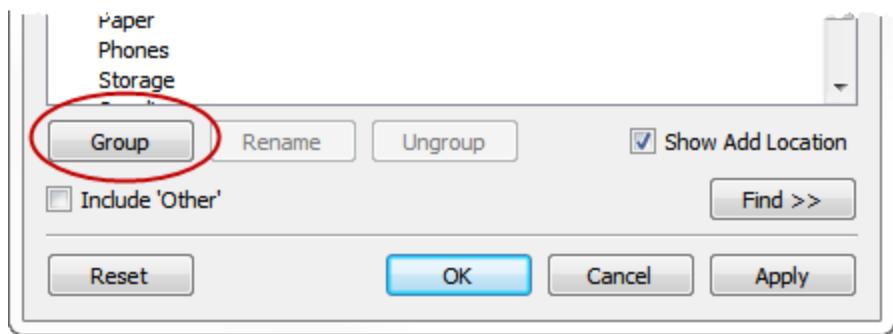
either on the Rows or Columns shelf or to Color on the Marks card. Instead of creating a group by selections in the view, you can create a group directly from a dimension in the **Data** pane.

1. Right-click (control-click on Mac) a dimension in the **Data** pane and select **Create > Group**.

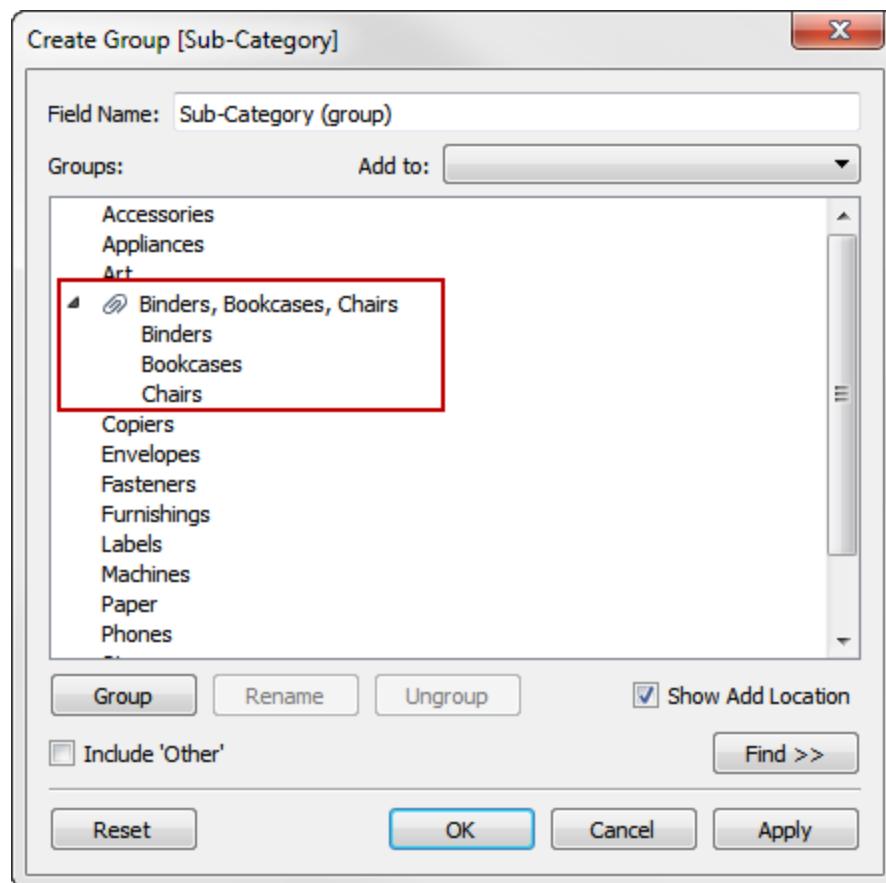


2. In the Create Group dialog box, select several members that you want to group. Press and hold the **CTRL** key (the Command key on a Mac) on your keyboard to select multiple members.

3. Click the **Group** button at the bottom of the dialog box.



The selected members are combined into a single member. A default name is automatically constructed using the combined member names. Rename the group by selecting it in the list and clicking the **Rename** button at the bottom of the dialog box. You can search for members using the Find options. For more information, see [Finding Members in Group Dialog Box on page 510](#).



Note: To add to or remove members from a group, right-click the grouped field in the Data pane and select **Edit Group**. In the Edit Group dialog box you can also change the default name of the group and combine fields into new groups. Refer to [Editing an Existing Group](#) below to learn more. Alternatively, you can ungroup members by

selecting them in the view and clicking the Ungroup button  on the toolbar or tooltip.

Editing an Existing Group

After you have created a grouped field, you can add and remove members from the groups, create new groups, change the default group names, and change the name of the grouped field. You can make some changes directly in the view, and others through the Edit Group dialog box.

Edit groups directly in the view

You can add and remove members from an existing group and create new groups by selecting marks directly in the view. Do one of the following:

- **Add to an existing group** - Select one or more marks you want to add to an existing group. Then hold **SHIFT** or **CTRL** on your keyboard (use the **⌘** key on a Mac) to also select at least one member of the group you want to add to. Click the **Group**  button on the toolbar or tooltip.
- **Remove from an existing group** - Select one or more marks you want to ungroup.

Then click the **Ungroup Members**  button on the toolbar or tooltip. Make sure that the selection only contains members from one group at a time.

- **Create a new group** - Select one or more marks that you want to create a new group with. Then click the Group button on the toolbar or tooltip.

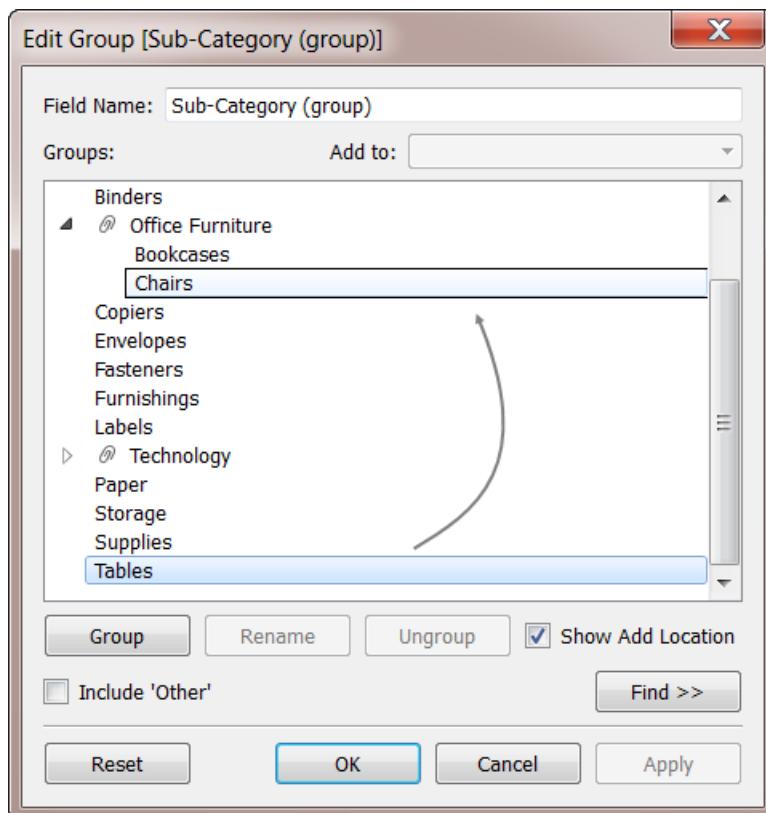
Open the Edit Group dialog box

While editing groups directly in the view is useful when rapidly defining ad hoc groups, you can more finely control the members of each group using the Edit Group dialog box. Right-click (control-click on Mac) the grouped field in the **Data** pane or in the view and select **Edit Group**.

The screenshot shows the 'Dimensions' pane in Tableau. On the left, there are two sections: 'Dimensions' and 'Measures'. Under 'Dimensions', items include Ship Date, Ship Mode, State, Sub-Category, and Measure Names. Under 'Measures', items include Discount, Profit, Quantity, and Sales. A context menu is open over the 'Sub-Category (group)' item, listing options: Add to Sheet, Cut, Copy, Edit Group..., Duplicate, Rename, and Hide. The 'Edit Group...' option is highlighted.

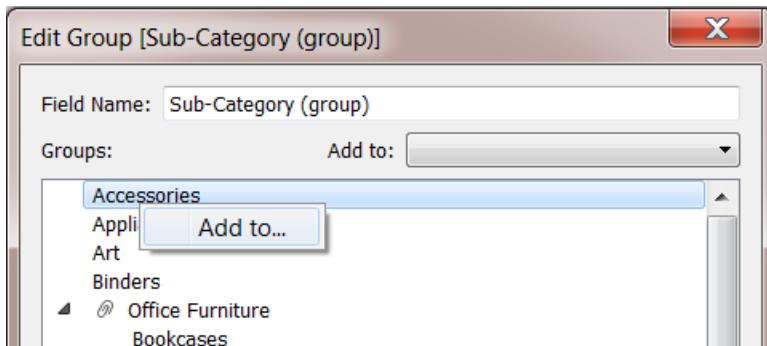
Then do one of the following:

- Select one or more members and drag and drop them into the existing group. This method works best if you are working with a dimension that has few members.

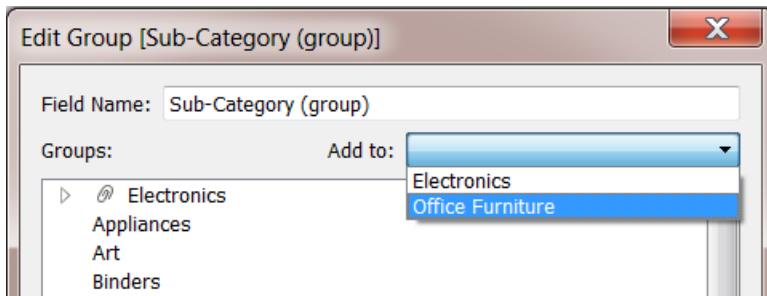


- Select one or more members, right-click (control-click on Mac) and select **Add To**. In the

Add to Group dialog box, select the group you want to add the selected members to and click **OK**.



- Select one or more members and select the group in the **Add to** drop-down list at the top of the dialog box.



Note: Rename a group by selecting the group in the Edit Group dialog box and then click the Rename button.

Including an Other Group

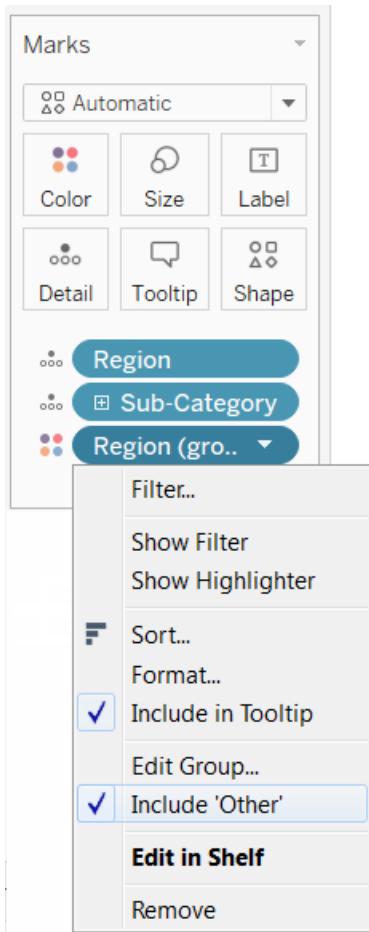
When you create groups in Tableau, you have the option to **Include Other**. When you include an other category, the groups you have defined are shown and all other dimension members are combined into an "Other" category. This option is useful for highlighting certain groups or comparing specific groups against everything else. For example, you may have a grouped field that combines product categories based on their sales performance. You may want to highlight High Performers and Low Performers and group all the other categories into an "Other" category.

Includes Other

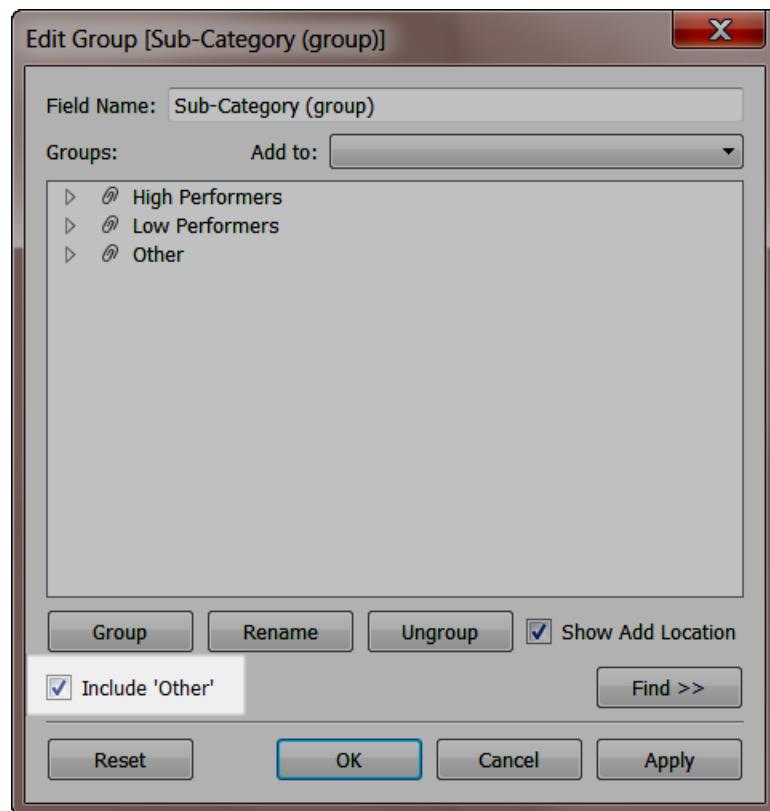
Does not include Other



Define whether to include the other category by selecting the **Include Other** option on the field menu.



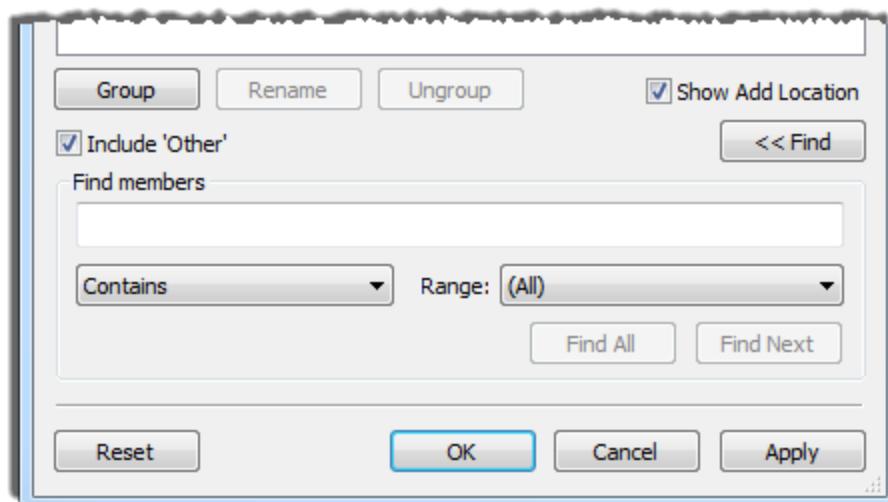
Alternatively, you can right-click (control-click on Mac) the field in the Data pane or on the sheet and select **Edit Group**. Then select the **Include Other** option near the bottom of the Edit Group dialog box.



Finding Members in Group Dialog Box

When you create groups from a large dimension with many members, use the Find option to quickly select the members you are looking for and add them to an existing group.

1. Show the find options by clicking the **Find** button at the bottom of the dialog box.



2. Type all or part of the member name into the text box and select an appropriate result criteria from the drop-down menu. You can select whether to find members that start with, contain, or are an exact match to the search term.
3. Select a Range to search in. You can select to search all members, or within specific groups.
4. Click **Find All** to select all the matching members or select **Find Next** to manually navigate through each of the search results.
5. When you have found and selected the members of interest, you can quickly add them to an existing group by selecting the group on the **Add to** drop-down menu at the top of the dialog box.

Sets

Sets are custom fields that define a subset of data based on some conditions. A set can be based on a computed condition, for example, a set may contain customers with sales over a certain threshold. Computed sets update as your data changes. Alternatively, a set can be based on specific data point in your view. You can use sets to compare and ask questions about a subset of data. For example, in a scatter plot showing product sales, you may select the products with the most sales and add them to a set. You may then create another set that contains the products with the highest return rates. With these two sets you can ask questions like "What percent of my total sales is from high returned products?" Or you may go further and compare the two sets to each other to create a new set that contains only high sales products that are also returned frequently.

Tableau displays sets at the bottom of the Data pane and labels them with the set icon .

Sets

-  State - High Sales & Profit
 -  Top Customers by Profit
-

When you drag a set to the view, you can choose to aggregate the members using In/Out mode or to list all of the individual members of the set. See [Using Sets on page 517](#) to learn more about these modes.

If you are connected to a multidimensional data source, you may see additional sets that were created in your data source. These sets are also listed in the Sets area of the Data pane but are

indicated with the server named set icon  . A server named set can be used in the same way as any other sets you create in Tableau.

Sets can be automatically generated as a result of an action. These sets are indicated by the

action set icon  . See [Actions on page 1116](#).

Finally, when working with workbooks connected to Tableau Server or Tableau Online, User

Filters display in the Sets area of the Data pane using the user filter set icon user  . See [Restrict Access at the Data Row Level on page 1252](#).

If you are connected to a relational data source, creating a set that is based on a continuous date field will use the exact dates instead.

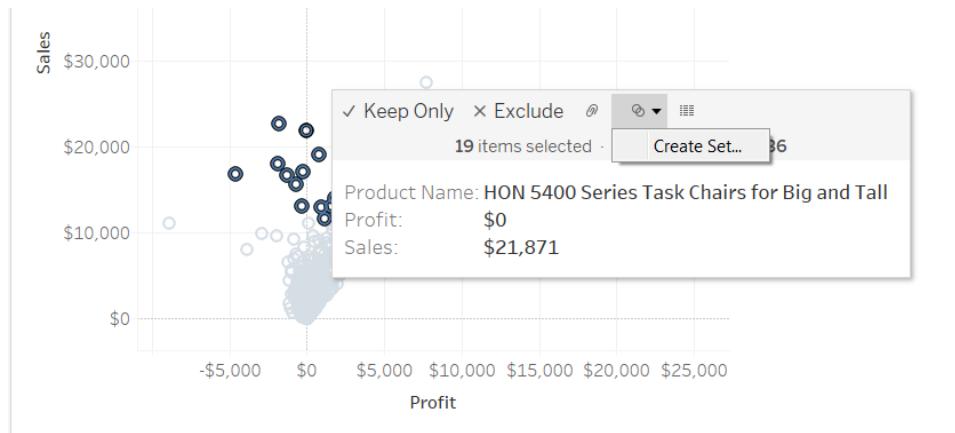
Creating a Set

There are many ways to create and edit sets in Tableau. To create a set, select marks or headers in the view or right-click (control-click on Mac) a field in the **Data** pane. You can also create a set that is based on a filter you've already defined. Regardless of how you create the set, it is either constant or computed. A constant set contains a specified list of members based on one or more dimensions. For example, a constant set might contain the specific names of products that you are keeping an eye on. A computed set is dynamic and the members change as the underlying data changes. For example, a computed set may contain the top 5 products by total sales. As the sales data changes, the companies included in the set will also change.

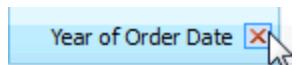
Constant sets

The members of a constant set are fixed and do not change. To create a constant set, you need to select the members you want to include. A constant set can be based on a single dimension or multiple dimensions.

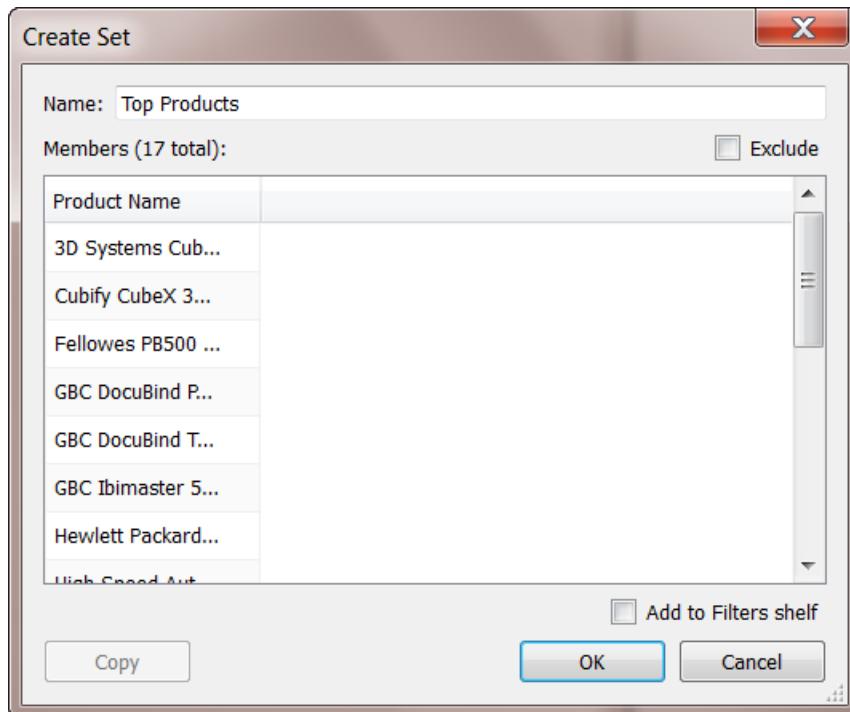
1. Select one or more marks or headers in the view.
2. Right-click (control-click on Mac) and select **Create Set** or click the **Create Set** option on the tooltip.



3. In the Create Set dialog box, type a name for the set.
4. Optionally complete any of the following:
 - By default, the set includes the members listed in the dialog box. You can select the option to **Exclude** these members instead. When you exclude, the set will include all of the members you didn't select.
 - Remove any dimensions that you don't want to be considered by clicking the red "x" icon that displays when you hover over a column heading



- Remove any specific rows that you don't want to include in the set by clicking the red "x" icon that displays when you hover over the row .
- If the marks you selected represent multiple dimensions, each member of the set will be a combination of those dimensions. You can specify the character that separates the dimension values.
- Select **Add to Filters shelf** to automatically move the set to the Filters shelf once it is created.

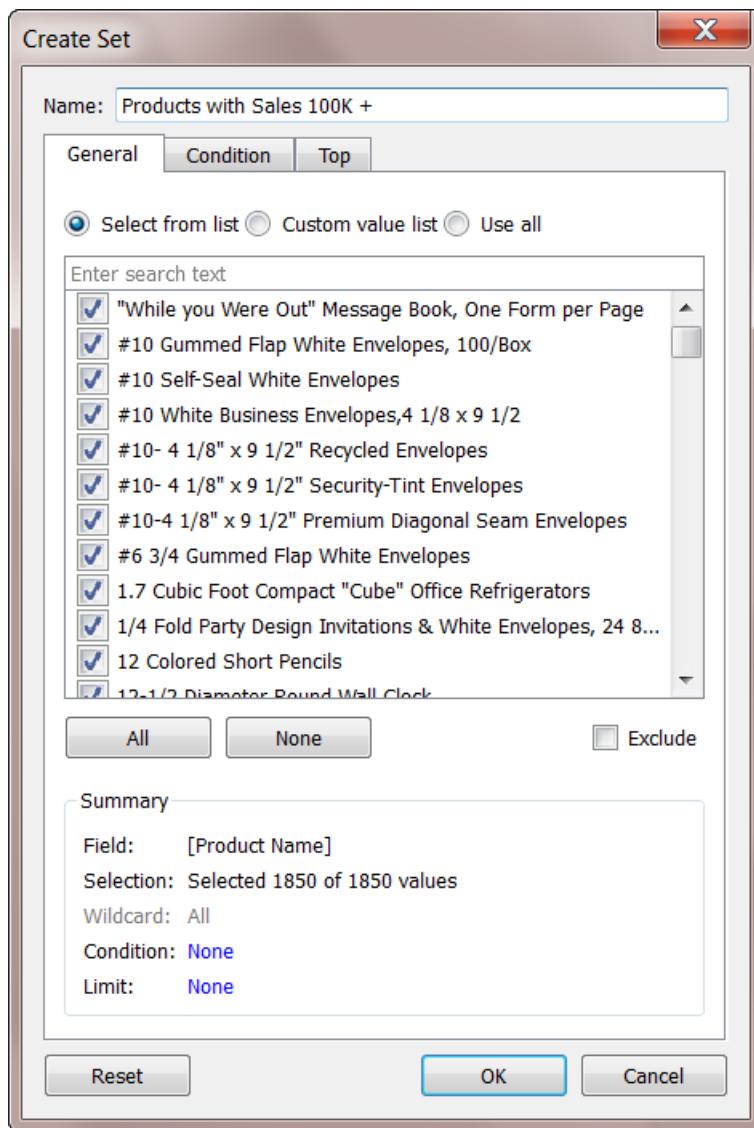


5. Click **OK**.

Computed sets

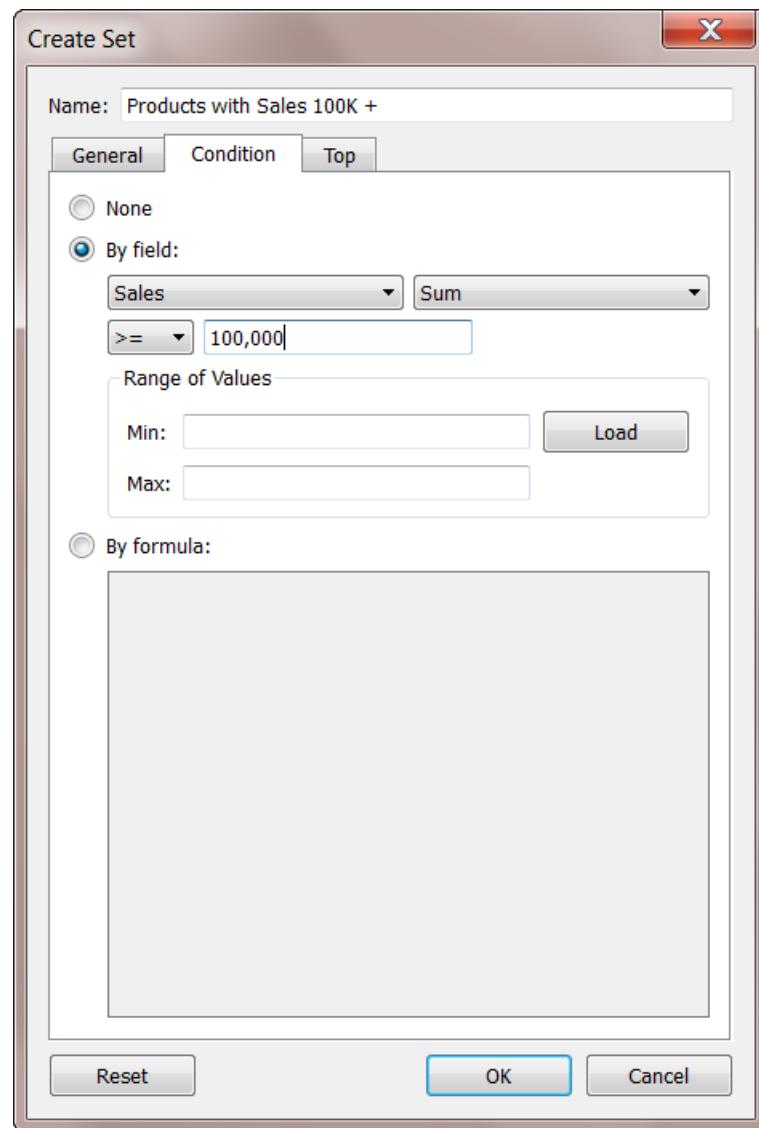
The members of a computed set are dynamic and change when the underlying data changes. Computed sets can only be based on a single dimension.

1. Right-click (control-click on Mac) a dimension in the Data pane.
2. Select **Create > Set**.
3. In the Create Set dialog box, type a name for the set.
4. On the General tab, select one or more values that will be considered when computing the set. You can alternatively select the **Use all** option to always consider all members even when new members are added or removed.



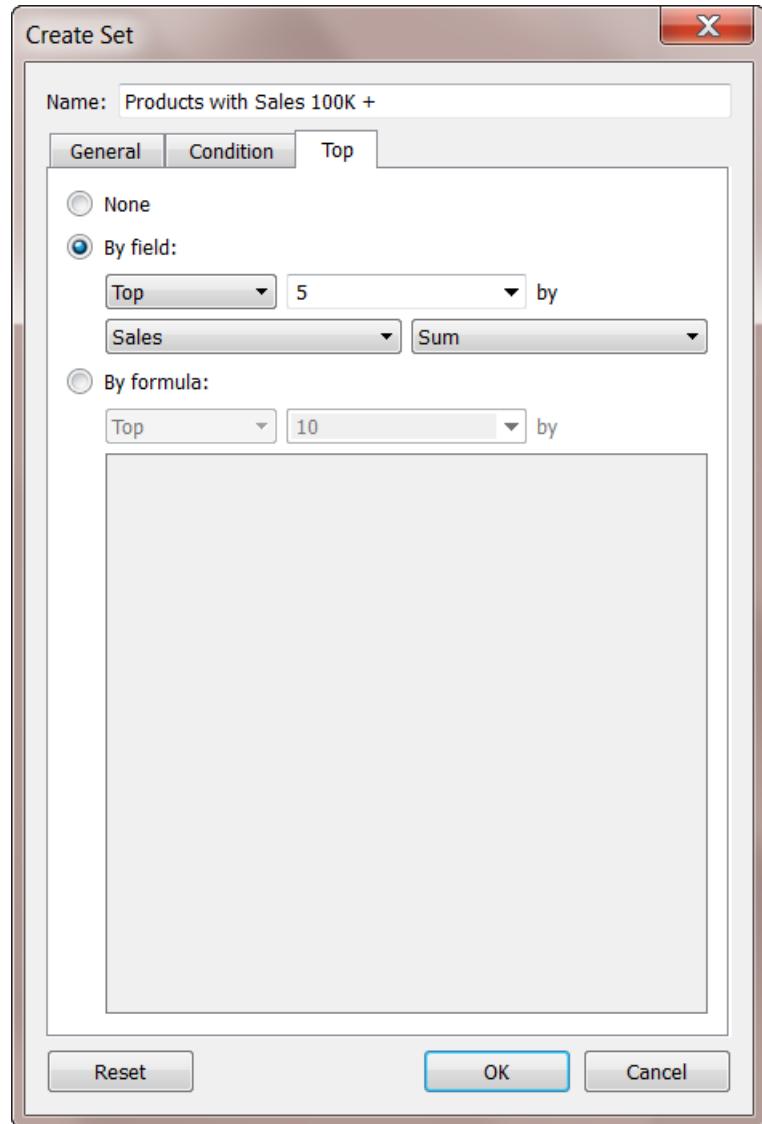
5. Define how to compute the members by adding conditions and limits:

- **Condition:** Use the Condition tab to define rules that determine what members to include in the set. For example, you might specify a condition that is based on total sales that only includes products with sales over \$100,000.



Set conditions work the same as filter conditions. See [Filter Data from Your Views](#) on page 672 to learn more.

- **Top:** Use the Top tab to define limits on what members to include in the set. For example, you might specify a limit that is based on total sales that only includes the top 5 products based on their sales.



Set limits work the same as Filter limits. See [Filter Data from Your Views on page 672](#) to learn more.

6. Click **OK**.

Using Sets

After you create a set, it displays at the bottom of the Data pane in the Sets area. You can then drag sets into the view like any other field. For example, you can drag the set to the Filter shelf to quickly filter the view to only show the members of the set. Alternatively, you can drag the set to the Marks card or the Rows and Columns shelves. When you drag a set to the view, you can choose to show the members of the set or aggregate the members into In/Out categories.

Show In/Out of Set

In most cases, when you drag a set to the view, Tableau displays the set using the In/Out mode. This mode separates the set into two categories: In, which contains the members in the set; and Out, which contains any members that are not part of the set. For example, in a set defined as the top 25 customers, the top customers would be part of the In category and all other customers would be part of the Out category. Using the In/Out mode makes it easy to compare the members in the set to everything else. You can use this mode to answer questions like "What percent of my total revenue comes from members of this set?"

Switch a set to use In/Out mode by selecting **Show In/Out of Set** on the field menu.

When a set is in In/Out mode, the field on the shelf is prefaced by the text, "IN/OUT" followed by the set name.

Note: In/Out mode is not available in workbooks created before version 8.2 that use Microsoft Excel or text file data sources, workbooks that use the legacy connection, or workbooks that use Microsoft Access data sources.

Show Members in Set

As an alternative to showing the set using In/Out mode, you can list the members in the set. Showing the members in the set automatically adds a filter to the view that includes only the members of the set.

Switch a set to list the individual members by selecting **Show Members in Set** on the field menu.

Note: To display the fully qualified member names for cubes, right-click the set in the Data pane and select **Qualify Member Names**.

Combining Sets

You can combine two sets to compare the members. When you combine sets you create a new set containing either the combination of all members, just the members that exist in both, or members that exist in one set but not the other. Combining sets allows you to answer complex questions to understand cohorts of data. For example, to determine the percentage of customers who purchased both last year and this year, you can combine two sets containing the customers from each year and return only the customers that exist in both sets. Another example would be to determine what products sell the most but also have the highest return rate.

To combine two sets, they must be based on the same dimensions. That is, you can combine a set containing the top customers with another set containing the customers that purchased last year. However, you cannot combine the top customers set with a top products set. In the latter

case, the one set is based on the Customer dimension while the other is based on the Product dimension.

1. Select two sets in the Data pane that you want to combine.
2. Right-click (control-click on Mac) the sets and select **Create Combined Set**.
3. In the Create Set dialog box, type a name for the new combined set.
4. Verify that the two sets you want to combine are selected in the two drop-down menus.
5. Select one of the following options for how to combine the sets:
 - **All Members in Both Sets** - the combined set will contain all of the members from both sets.
 - **Shared Members in Both Sets** - the combined set will only contain members that exist in both sets.
 - **Except Shared Members** - the combined set will contain all members from the specified set that don't exist in the second set. These options are equivalent to subtracting one set from another. For example, if the first set contains Apples, Oranges, and Pears and the second set contains Pears and Nuts; combining the first set except the shared members would contain just Apples and Oranges. Pears is removed because it exists in the second set.
6. Optionally specify a character that will separate the members if the sets represent multiple dimensions.
7. When finished, click **OK**.

Note: This functionality is not available in workbooks created before version 8.2 that use Microsoft Excel or text file data sources, workbooks that use the legacy connection, or workbooks that use Microsoft Access data sources.

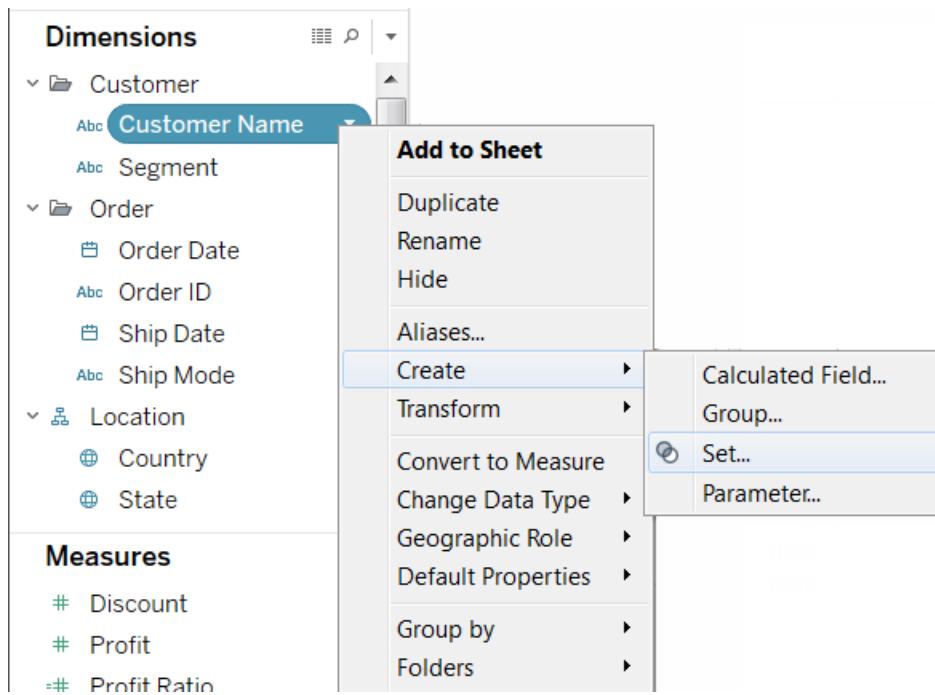
Examples of Sets

There are many ways you can use sets to answer complex questions and compare cohorts of data. Below are some examples of ways you can use sets to define and compare subsets of data.

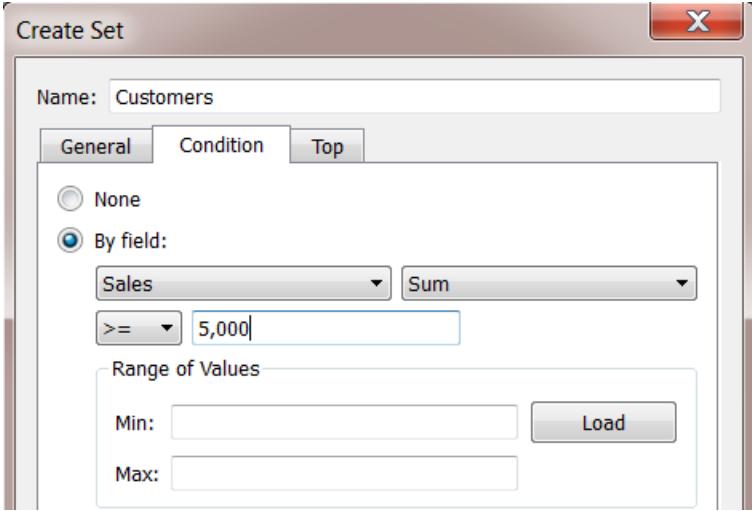
How do members of a set contribute to the total?

You may have all kinds of questions surrounding how the members in a set contribute to the overall total. For example, what percent of total sales come from repeat customers? You can answer these types of questions using the IN/OUT mode for set. The example below uses sales data to calculate the percent of total sales from customers who have purchased at least 20 times.

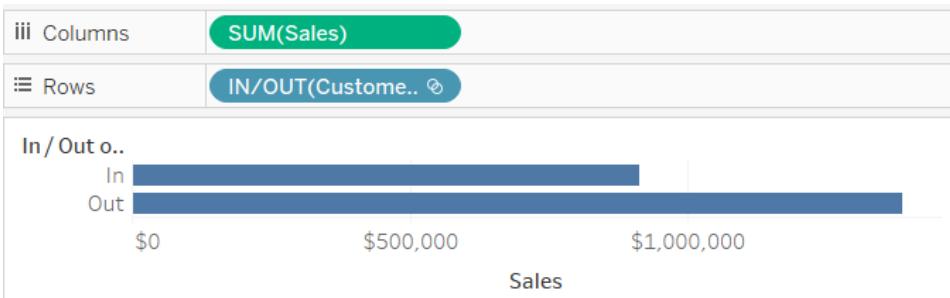
1. Right-click (control-click on Mac) the **Customer Name** dimension in the Data pane and select **Create > Set**.



2. In the Create Set dialog box, type a name for the set. In this example, we'll call the set, "Customers"
3. Select the **Use all** option so the condition always applies to all values even when new customers are added.
4. On the Condition tab, click **By field**, and then define a condition that only includes customers when **Sum of Sales** is greater than or equal to **5,000**.

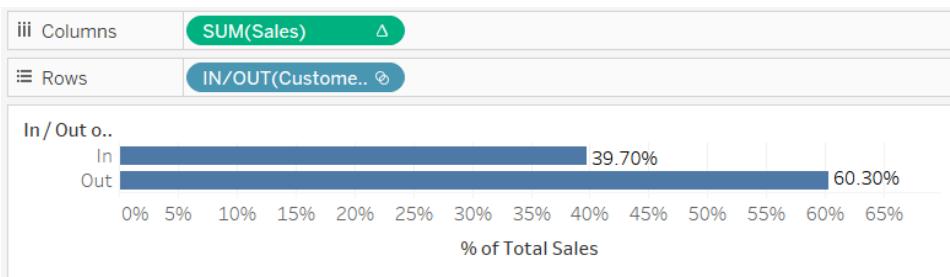


5. Click **OK**.
6. Drag the new set from the **Sets** area at the bottom of the **Data** pane to the **Rows** shelf.
7. From Measures, drag **Sales** to the **Columns** shelf. The view now shows the total sales for customers who have purchased more than \$5,000 dollars of product and the total sales for all other customers.



8. Finally, click on the drop-down arrow on the **Sum (Sales)** field on the Column shelf and select **Quick Table Calculation > Percent of Total** on the context menu.

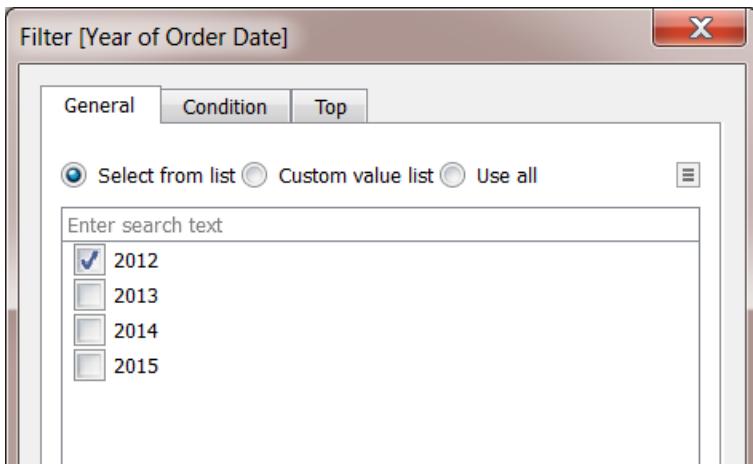
The view now shows that customers with sales greater than or equal to 5,000 make up about 39% of the overall sales.



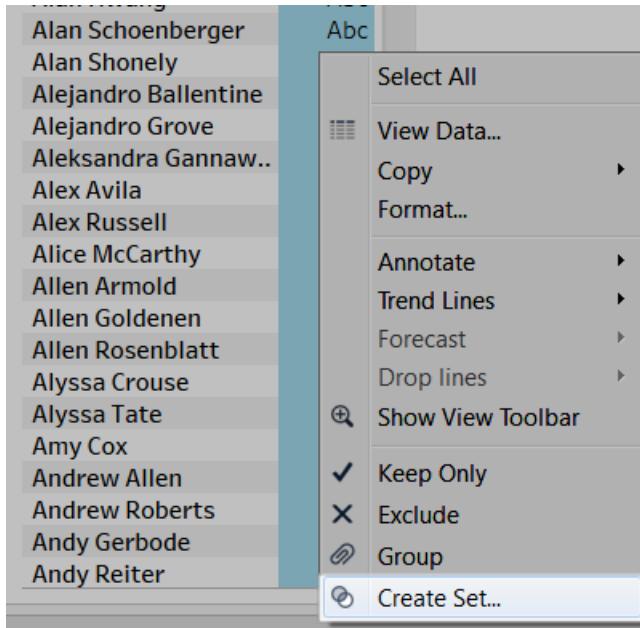
How many members of a set exist in another set?

Another common use of sets is to compare subsets of data or cohorts. For example, you may wonder how many customers that purchased last year also purchased this year. Or if a customer purchased a specific product, what other products did they buy? You can answer these types of questions by creating multiple sets and combining them. The example below uses sales data to determine how many customers who purchased in 2012 also purchased in 2013.

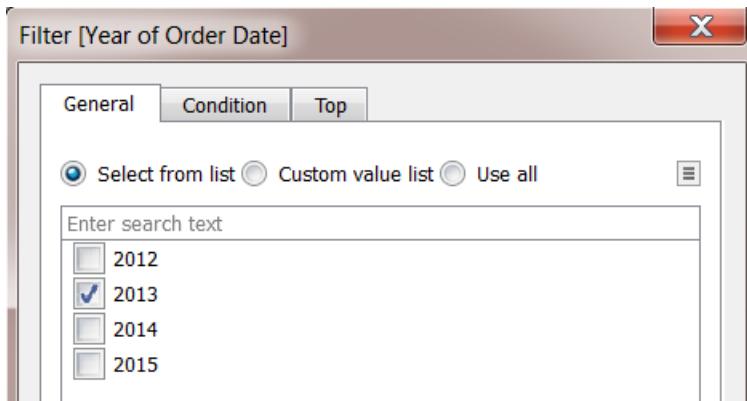
1. Drag the **Customer Name** field to the Rows shelf.
2. Drag the **Order Date** field to the Filters shelf.
3. In the Filter Field dialog box, select **Years** and click **Next**.
4. In the Filter dialog box, select **2012** and click **OK**.



5. Back in the view, press CTRL + A (Command-A on a Mac) on your keyboard to select all of the customers.
6. Right-click (control-click on Mac) the selection and select **Create Set**.



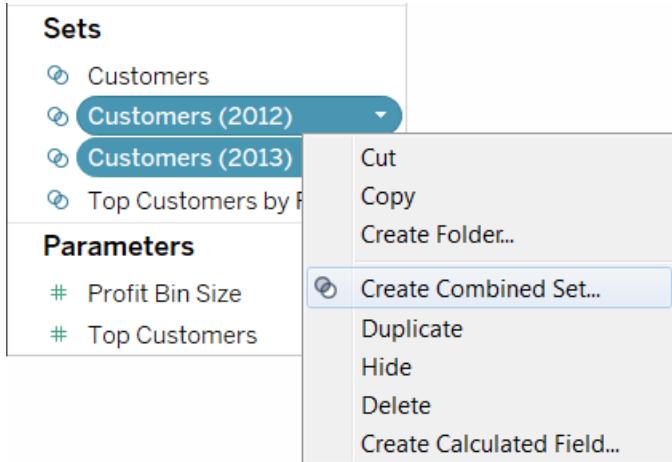
7. In the Create Set dialog box that opens, type a name for the set. In this example, we'll call the set "Customers (2012)".
8. Click **OK**.
9. On the **Filters** shelf, right-click (control-click on Mac) **Order Date** and select **Edit Filter**.
10. In the Filter dialog box, change the filter to only include **2013** instead of **2012**, and then click **OK**.



11. Again, press CTRL + A (Command-A on a Mac) on your keyboard to select all of the customers.
12. In the view, right-click (control-click on Mac) the selection and select **Create Set**.
13. In the Create set dialog box that opens, type a name for the set. This set will be called

"Customers (2013)".

14. Click **OK**.
15. In the **Data** pane, select both the **Customers 2012** and **Customers 2013** by holding the Ctrl key (Command key on a Mac) on your keyboard as you select.
16. Right-click (control-click on Mac) the selection and select **Create Combined Set**.

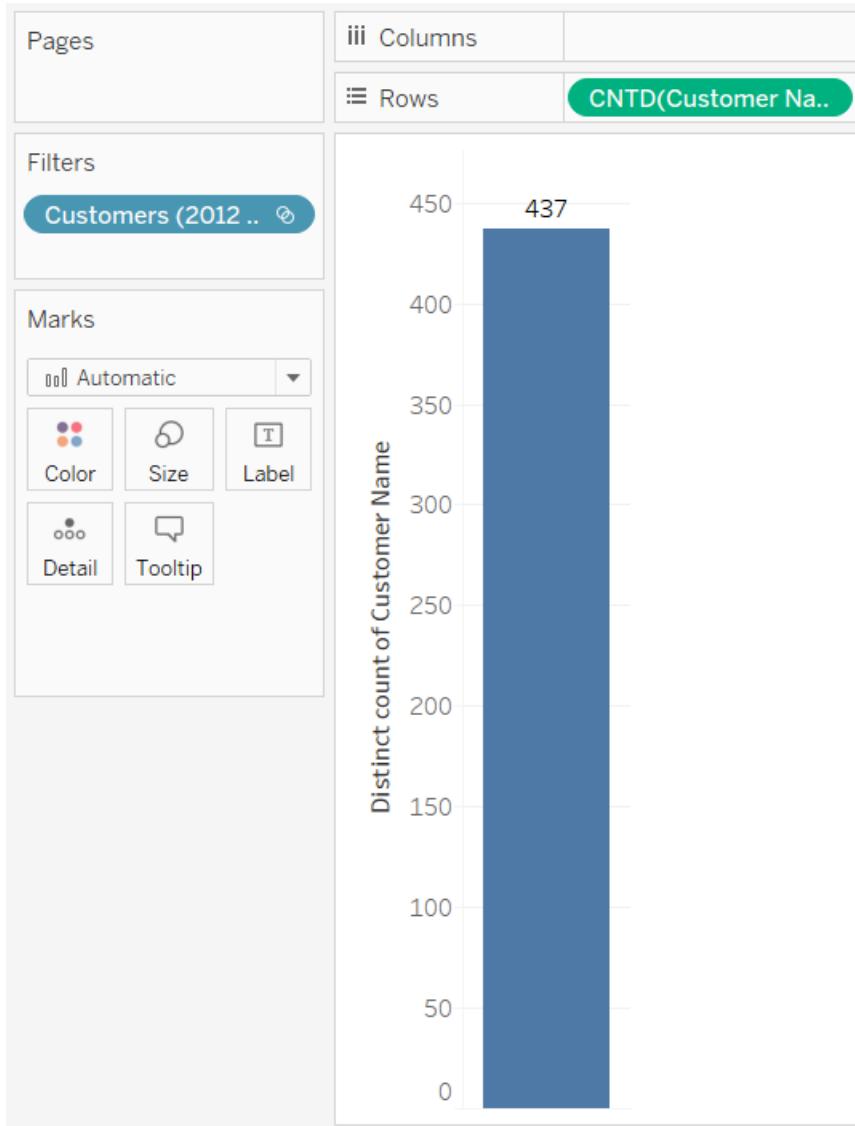


17. In the Create Set dialog box, type a name for the new set. In this example, we'll call the set "Customers (2012 & 2013)".
18. Make sure the correct two sets are selected in the drop-down menus.
19. Select the option to include **Shared Members in Both Sets**.



20. Click **OK**.

21. At the bottom of the workbook, click the New Worksheet  icon.
22. In the new worksheet, drag the **Customer Name** dimension to the **Rows** shelf.
23. Click the drop-down arrow on the Customer Name field on the Rows shelf and select **Measure > Count (Distinct)** from the context menu.
24. Finally, from the **Sets** area of the **Data** pane, drag the **Customers (2012 & 2013)** field to the **Filters** shelf. You can see that 437 customers purchased products in both 2012 and 2013.



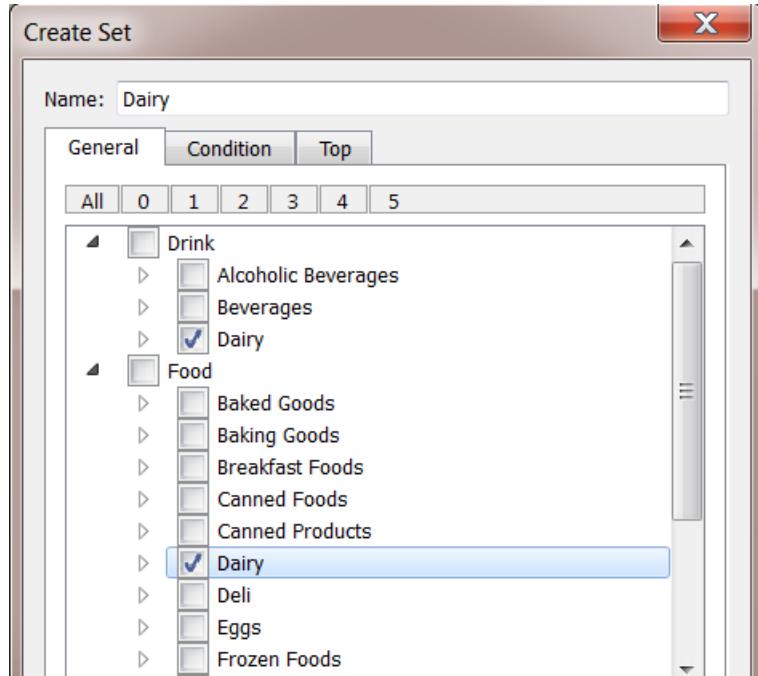
Hierarchical sets and descendants

A hierarchical set filters data to the selected members and all of their descendants. They are

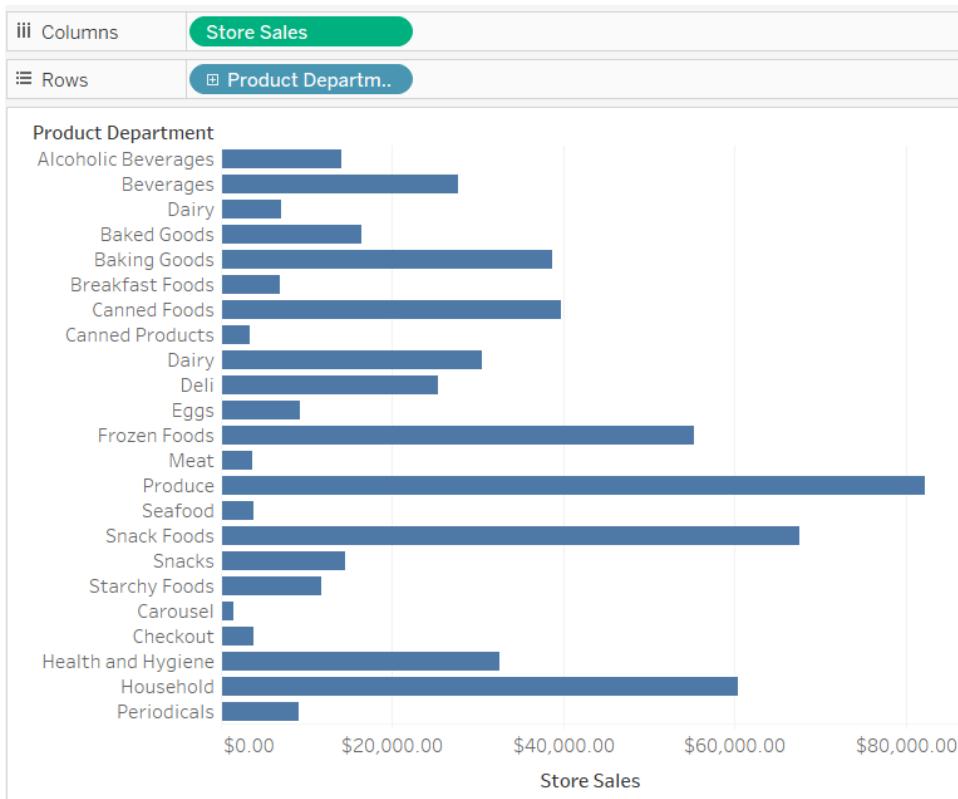
unique to multidimensional (cube) data sources and are defined within the data source prior to connecting to Tableau Desktop.

When you create sets in Tableau from a cube data source, descendants and any hierarchical structures are automatically included with the selected members.

For example, a set named **Dairy** is created from the **Product** hierarchy. As shown below, it includes only the Dairy product department.



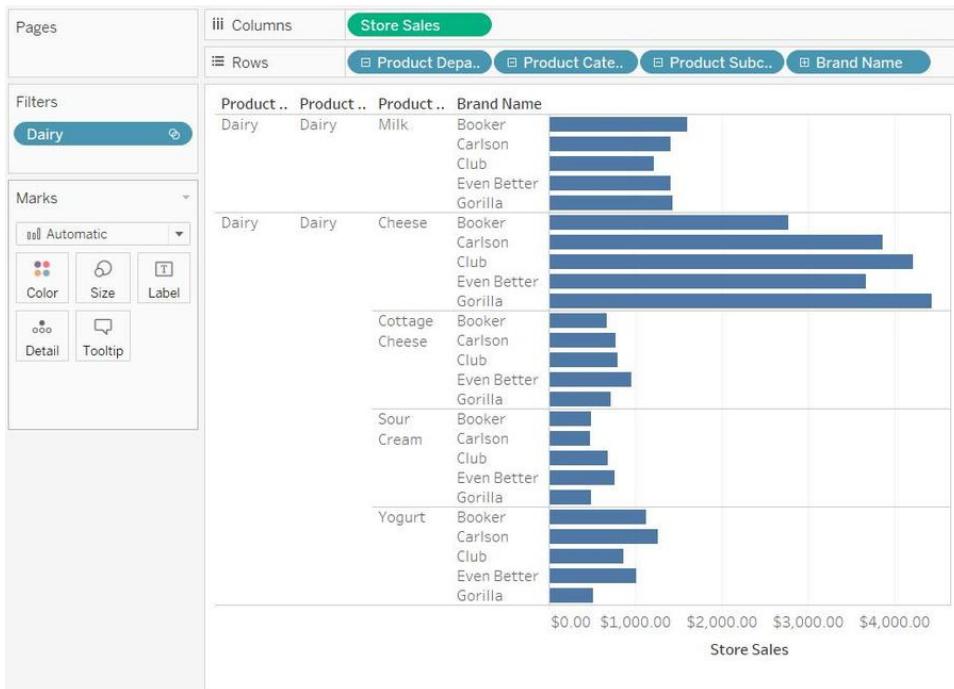
Consider the following view. The **Product Department** dimension is placed on the **Rows** shelf and the **Store Sales** measure is placed on the **Columns** shelf.



If you place the **Dairy** set on the **Filters** shelf, you can see that the view is filtered to include only the Dairy product categories.



As shown below, you can drill down into **Product Department** to reveal the **Product Category**, **Product Subcategory**, and **Brand Name** levels. As these descendants are revealed, row headers are added to the view. This is because a set filter allows you to view the levels of detail contained within the filtered members.



Create Sets for Top N and Others

If you collect large sets of data that you want to visualize, you might find that limiting the amount of information displayed to an important subset of records helps you work with and answer questions about the data more effectively.

This article describes how to create an interactive view that separates your customers into two dynamic groups:

- The top N customers
- All other customers

The view includes a control that your users can adjust to change the number of customers included in the top customers group. When they change the number, the view updates accordingly.

Sets and supported data sources

The method described in this article for creating a view of the top customers uses the In/Out functionality of sets.

Sets were introduced with Tableau Desktop version 8.0.

For live connections, the In/Out functionality requires a relational or multidimensional data source.

If you use a file-based data source, such as a Microsoft Excel workbook or text file, you can take an extract with which you can create sets.

: Create the parameter

1. In Tableau Desktop, open a new workbook and connect to the **Sample-Superstore** data source.
2. Open a new worksheet.
3. In the Data pane, click the drop-down to the right of Dimensions and select **Create Parameter**.
4. In the Create Parameter dialog box, do the following:
 - In the **Name** text box, type **Top Customers 2**.
 - For **Data type**, select **Integer**.
 - For **Current value**, type **5**.
 - For **Allowable values**, click **Range**.
 - Under Range of values, do the following:
 - Click **Minimum** and type **5**.
 - Click **Maximum** and type **20**.
 - Click **Step size** and type **5**.

This parameter will be used, in combination with the top N set you will create in the next step, to quickly adjust the top N value in the view.

: Create the top N customers set

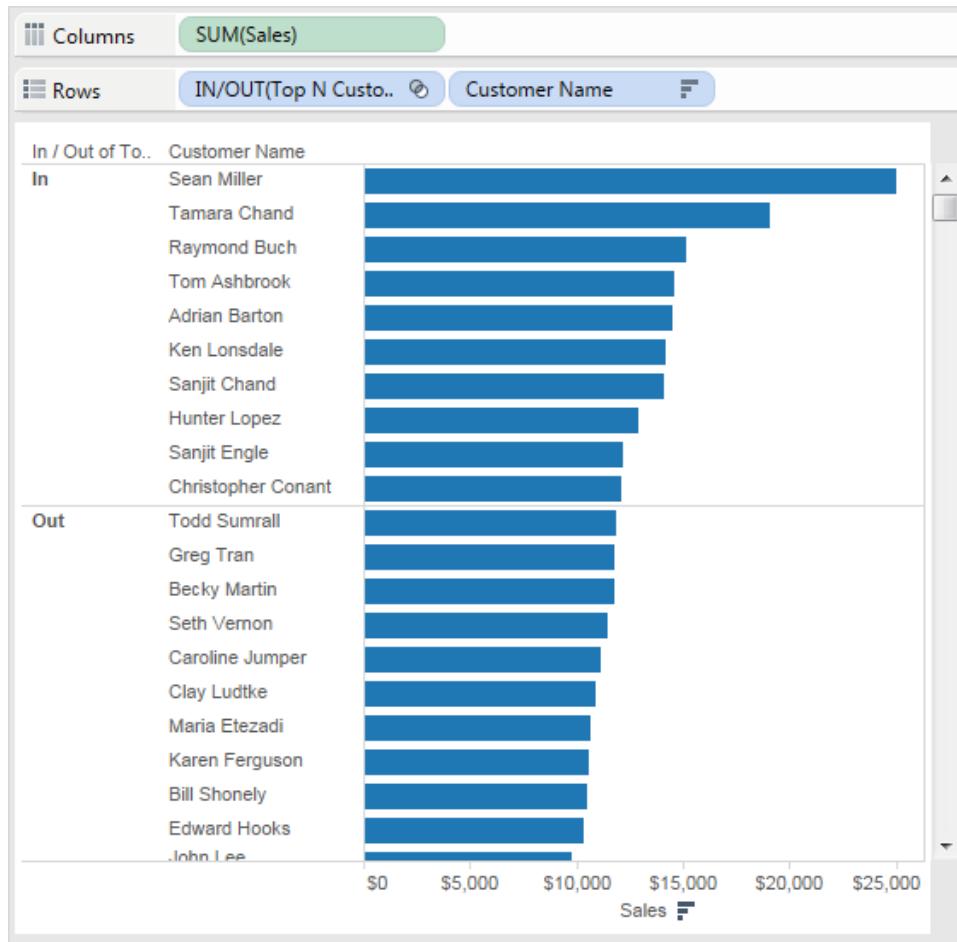
1. In the Data pane, under Dimensions, right-click **Customer Name**, and select **Create > Set**.
2. In the Create Set dialog box that opens, do the following:
 - In the **Name** text box, type **Top N Customers by Sales**.
 - Click the **Top** tab.
 - Select **By Field**.
 - From the field drop-down list (Category), select **Sales**.
 - From the aggregation drop-down list, select **Sum**.
 - When finished, click **OK**.

: Set up the view

1. From **Sets**, drag **Top N Customers by Sales** to the **Rows** shelf.
2. From **Dimensions**, drag **Customer Name** to the **Rows** shelf, positioning it to the right of the set.

3. From **Measures**, drag **Sales** to the **Columns** shelf.

4. On the toolbar, click the Descending Sort button  to make sure that the set is working.



5. In the **Data** pane, under **Sets**, right-click **Top N Customers by Sales**, and then click **Create Calculated Field**.
6. In the Calculated Field dialog box that opens, complete the following steps:
 - In the **Name** text box, type **Subset Labels**.
 - In the **Formula** text box, type the following formula to create dynamic labels for the customers in the set:

```
IF [Top N Customers by Sales]
THEN "Top " + str([Top Customers 2]) + " Customers"
ELSE "Others"
```

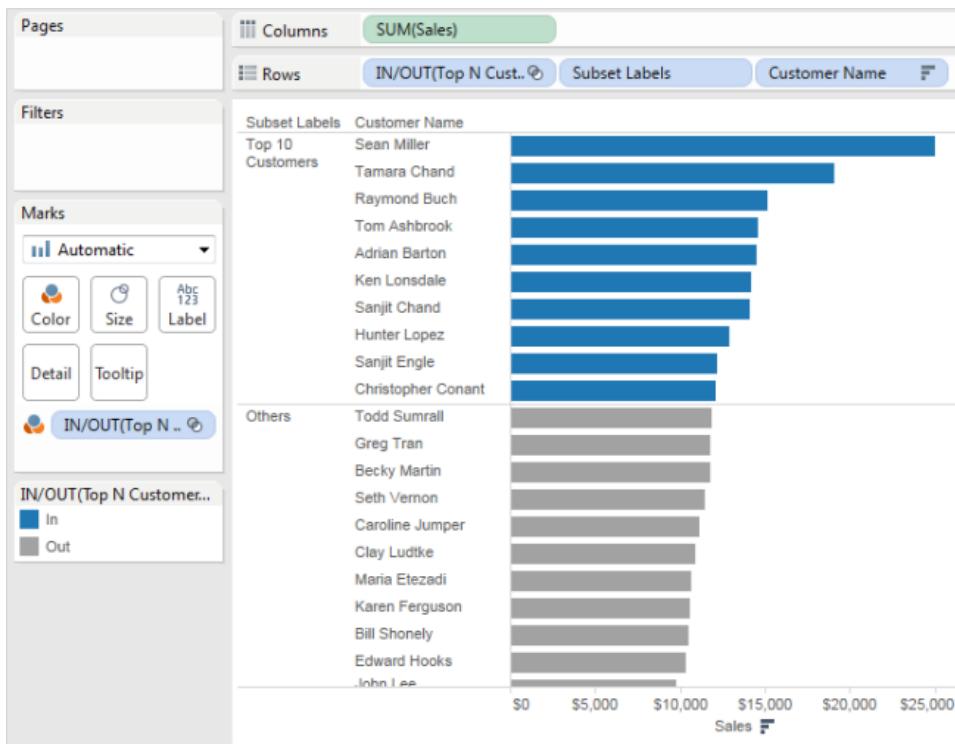
END

- When finished, click **OK**.

7. From **Dimensions**, drag **Subset Labels** to the **Rows** shelf, placing it between the Top N set and the **Customer Name** dimension.
8. On the **Rows** shelf, right-click the **IN/OUT(Top N Customers by Sales)** set, and then clear **Show Header**.

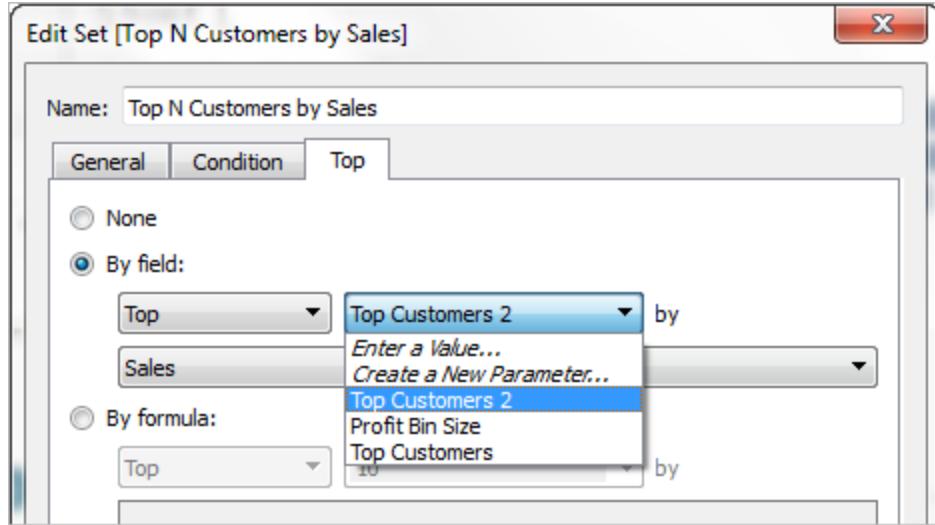
This hides the In/Out labels while retaining the sort order so that your top N subset always appears at the top of the view.

9. From **Sets**, drag **Top N Customers by Sales** to **Color** on the **Marks** card.



: Combine the Top N set with a dynamic parameter

1. In the **Data** pane, right-click **Top N Customers by Sales**, and then select **Edit Set**.
2. In the Edit Set dialog box, do the following:
 - Select the **Top** tab.
 - Click the value drop-down menu, and select the **Top Customers 2** parameter.
 - Click **OK**.

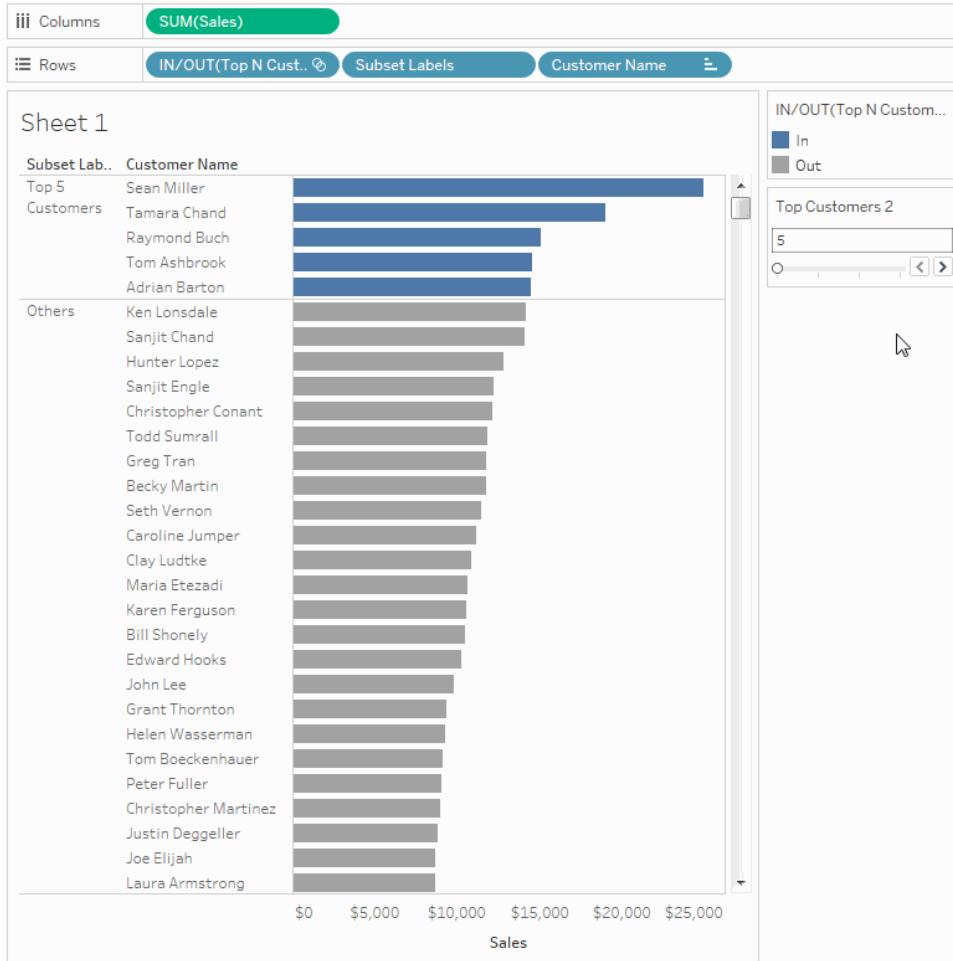


This links the **Top N Customers by Sales** set to the **Top Customers 2** dynamic parameter, instead of to a static list of 10.

This parameter will be used in combination with the Top N Customers by Sales set, to adjust the top N value in the view.

3. In the **Data** pane, under Parameters, right-click the **Top Customers 2** parameter, and select **Show Parameter Control**.

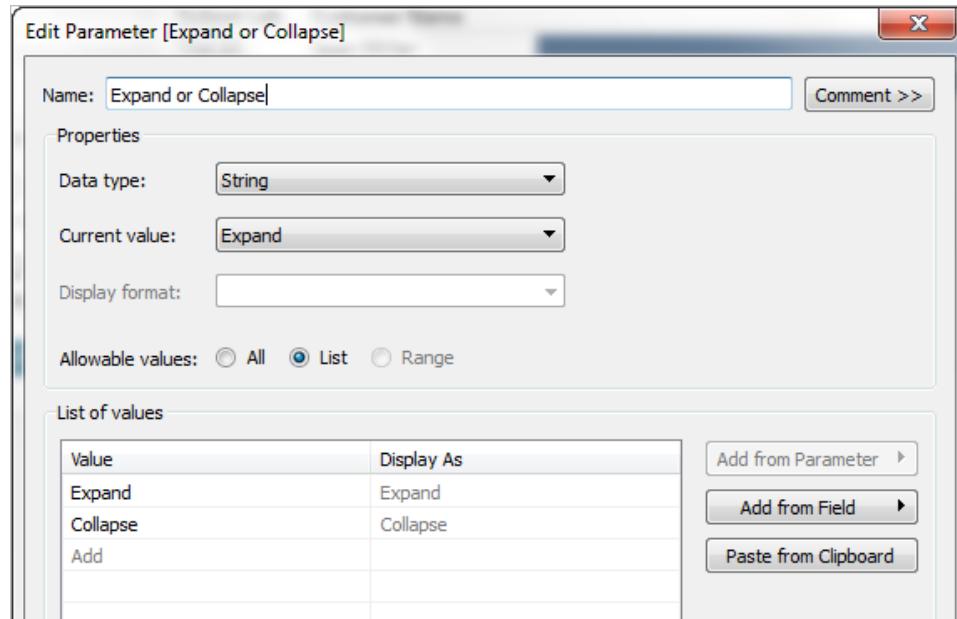
You can control the top N value by using the **Top Customers 2** parameter control that appears in the view.



Additional tips for improving the view's functionality

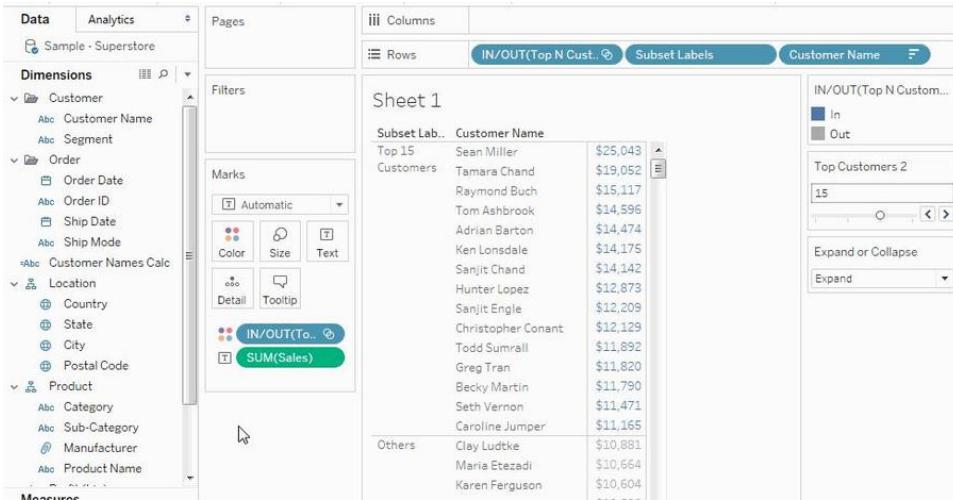
Here are some additional steps you can take to give your viewers more flexibility in displaying the customer subsets.

1. From the **Data** pane drop-down menu, select **Create Parameter**.
2. In the Create Parameter dialog box that opens, do the following:
 - For **Name**, type **Expand or Collapse**.
 - For **Data type**, select **String**.
 - For **Allowable values**, select **List**.
 - In the **List of values**, type the values **Expand** and **Collapse**.

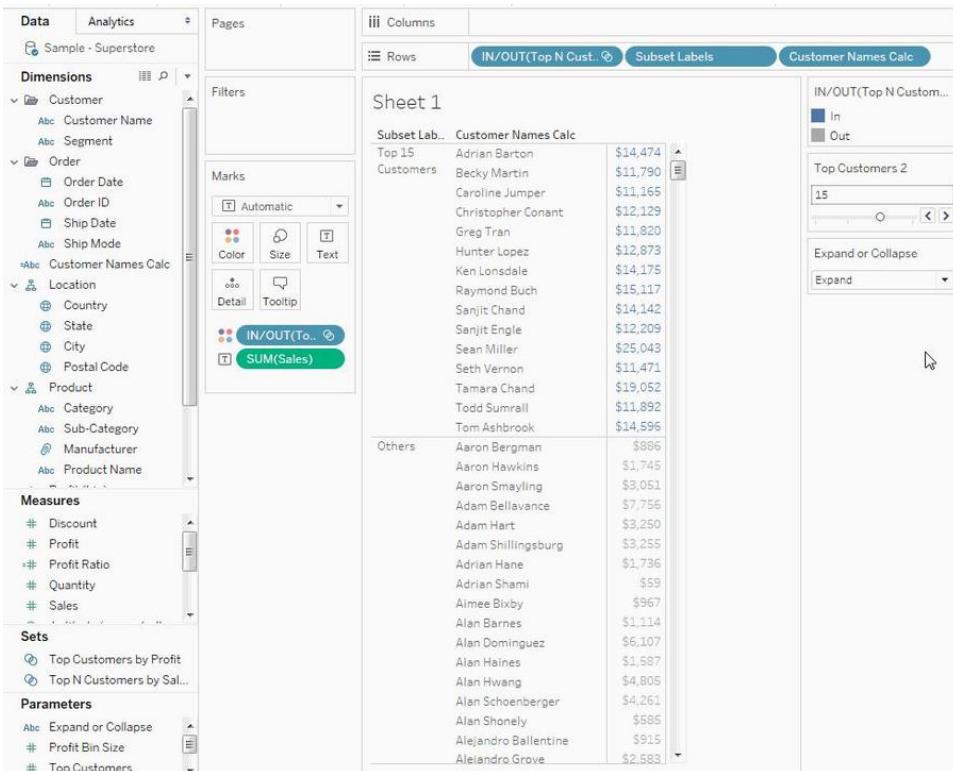


- When finished, click **OK**.
3. Select **Analysis > Create Calculated Field**.
 4. In the Create Calculated Field dialog box that opens, do the following to create a calculation that uses the parameter you just created. This calculation enables viewers to specify how to view customers in the Others subset:
 - For **Name**, enter **Customer Names Calc**.
 - In the formula box, type the following formula, and then click **OK**:

```
IF [Expand or Collapse] = "Collapse" THEN
  IF [Top N Customers by Sales]
    THEN [Customer Name]
  ELSE "Others" END
ELSE [Customer Name] END
```
 5. In the **Data** pane, under Parameters, right-click the **Expand or Collapse** parameter, and select **Show Parameter Control**.
 6. From the **Columns** shelf, drag the **SUM(Sales)** measure to **Label** on the **Marks** card.
 7. From **Dimensions**, drag **Customer Names Calc** directly on top of the **Customer Name** field on the **Rows** shelf, so that it replaces it.



Now you can use the **Expand or Collapse** parameter control to see the list of names in the top N customers and the remaining customers rolled up into a single **Others** entry.



Annotations and Mark Labels

Annotations call attention to specific marks, points, or areas in a view. An annotation, sometimes called a call-out, is most commonly displayed as a text box with a line pointing to a specific point or mark. You can also add an area annotation, which calls out several marks or a region of the view. Additionally, you can use mark labels to call out marks of interest or more commonly to label the view to make it more understandable. You can show mark labels for all the marks in the view, or selectively show and hide individual labels.

Annotations

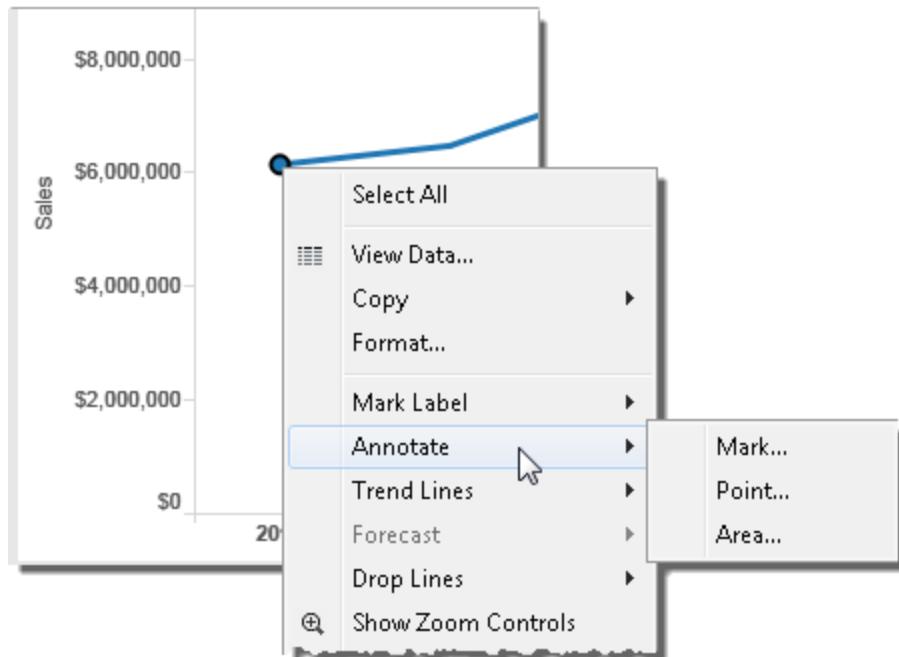
In Tableau there are three kinds of annotations: mark, point, and area. After you add an annotation, you can edit, re-position, format, and remove it.

Adding Annotations

Annotations are an important part of publishing and sharing a view. Use annotations to call out a specific mark, a specific point such as a value on the axis or a reference line, or an area such as a cluster of scatter marks.

To add an annotation:

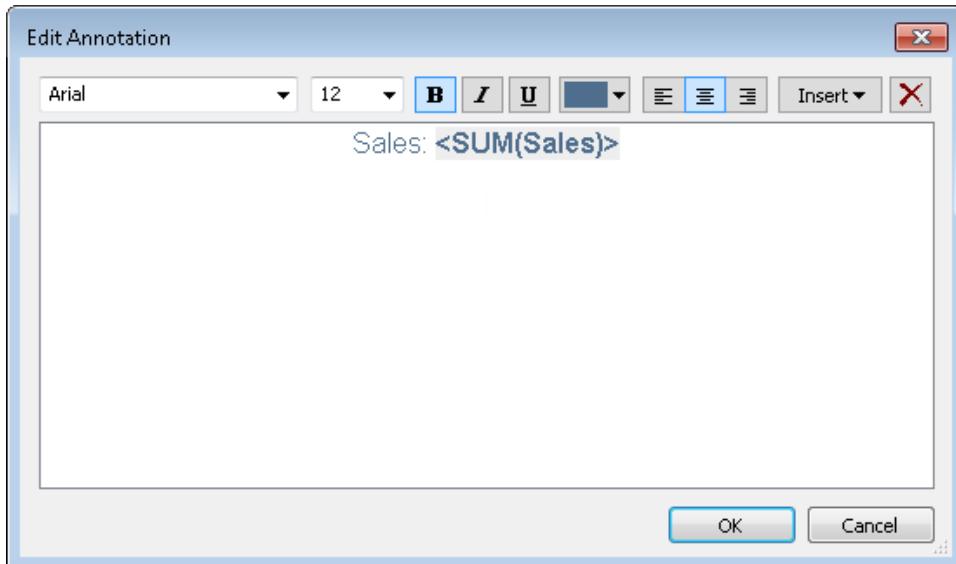
1. Right-click (Control-click on Mac) the view where you want to add an annotation and select **Annotate**.



2. On the sub-menu select one of the following types of annotations:

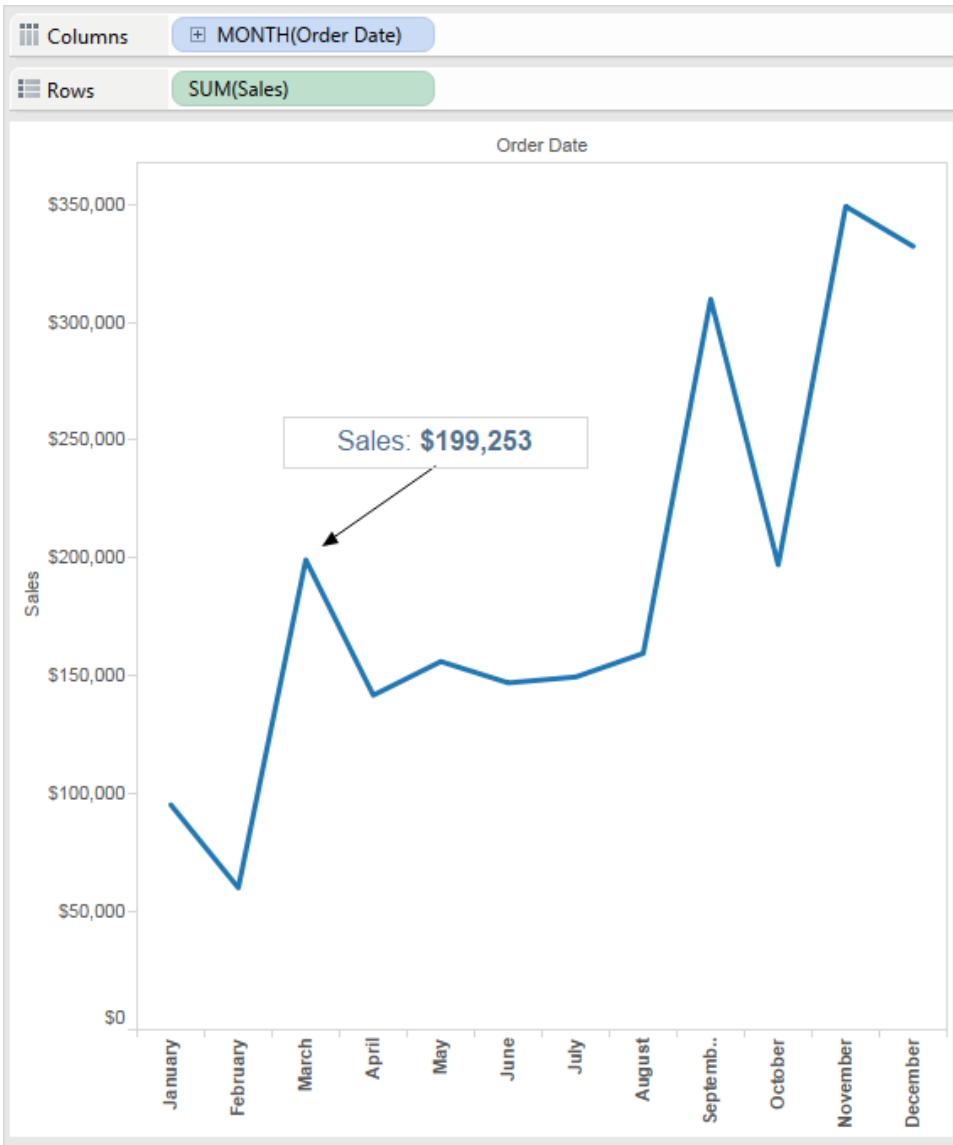
- **Mark** - select this option to add an annotation that is associated with the selected mark. This option is only available if a mark is selected.
 - **Point** - select this option to annotate a specific point in the view.
 - **Area** - select this option to annotate an area in the view such as a cluster of outliers or a targeted region of the view.
3. In the Edit Annotation dialog box, type the text you want to show in the annotation.

Use the **Insert** menu to insert dynamic variables into the annotation text. For example, the annotation can display data values that update as the underlying data changes. The dynamic variables that are available are dependent on whether you are annotating a mark, point, or area. Only relevant variables are available for each of these types of annotations. For example, mark annotations can include dynamic data about that specific data point such as dimension and measure values. Point annotations don't refer to a specific data point but rather a place in the view. For that reason, point annotations can only include dynamic information about that point, such as measure values along the axes. Finally, when adding area annotations, only data relevant to the entire sheet is available, such as title and sheet name.



Insert the **All Values** option using the Insert menu to show the data from the fields in the view. As you add more detail to the view, this text is updated to show the live data.

4. When finished, click **OK**.

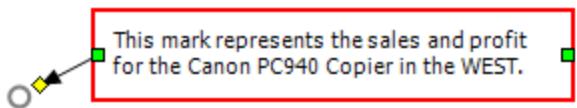


Rearranging Annotations

After you add an annotation, you can move it around, resize it, adjust the line, and move the text. Each type of annotation can be rearranged and modified in different ways. This section discusses how to rearrange, resize, and modify each type of annotation.

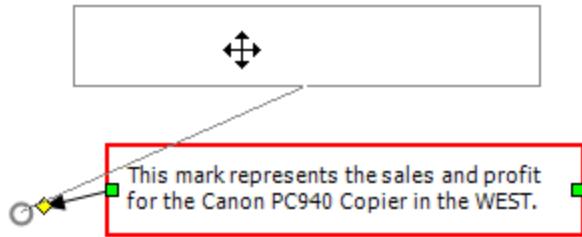
Mark Annotations

When you select a mark annotation the body and line are selected and several resize handles display. Use these handles to resize the body and line.



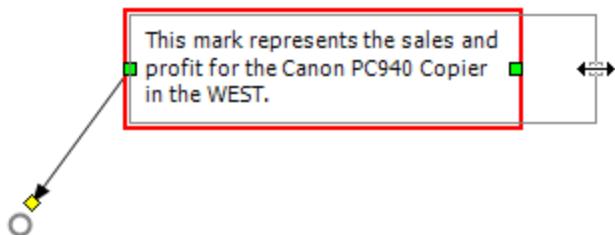
To reposition the body

- Click and drag the body of the selected annotation to a new position.



To resize the body

- Click and drag the body resize handle left and right. The text and height are automatically adjusted to fit the width of the body.



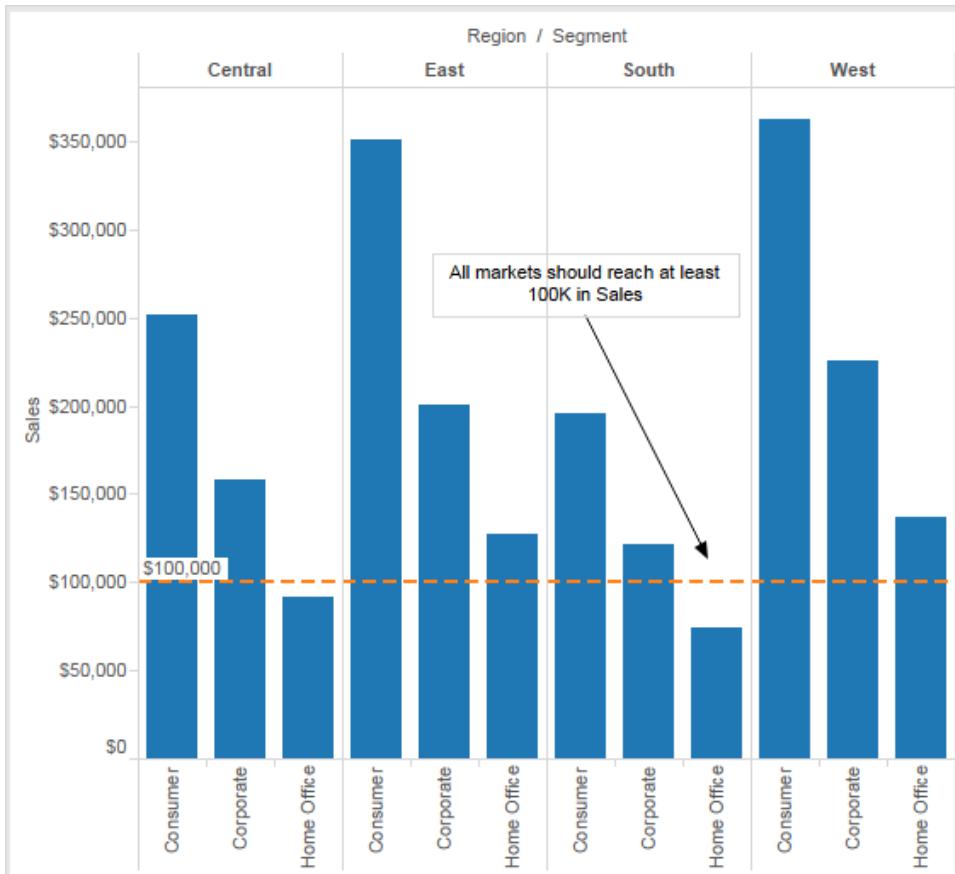
To resize the line

- Click and drag the line resize handle ♦



Point Annotations

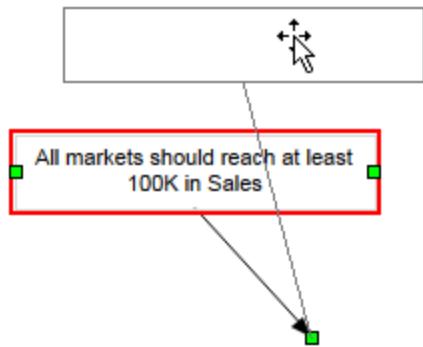
A point annotation marks a specific point in the view such as a reference line or a value on an axis. Point annotations display as text with a line. When you select a point annotation, several resize handles display. Use these handles to reposition and resize the body and line.



To reposition the body:

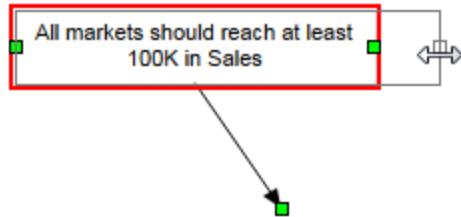
- Click and drag the body of the selected annotation to a new position. As you move the body, the line is automatically resized so that it continues to point at the specific point you

selected.



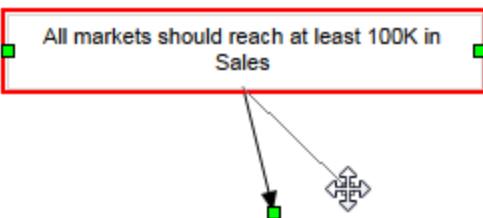
To resize the body:

- Click and drag the side resize handles ■ left and right. The text and height are automatically adjusted to fit the width of the body.



To move the line end point:

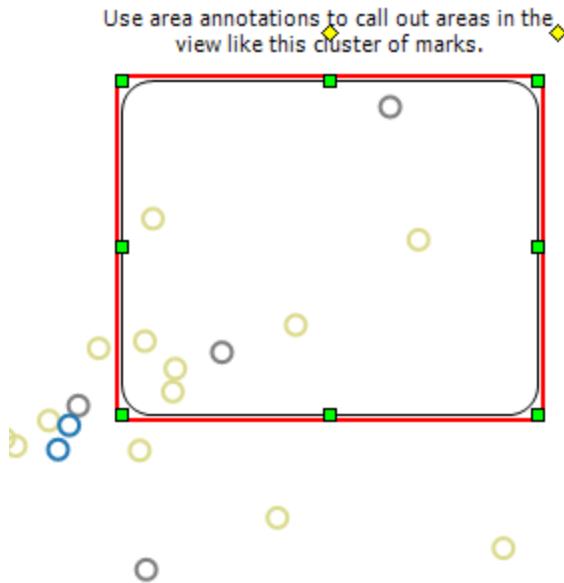
- Click and drag the end point of the line ■ so that it points at a new location.



Area Annotations

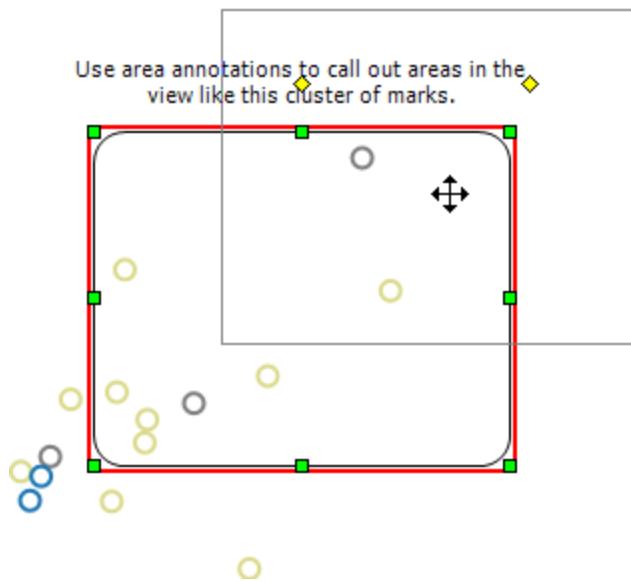
An area annotation is a way to highlight or call out an area in the view. Area annotations are not associated with any particular mark, in fact, these annotations are commonly used to call out

several marks. When you select an area annotation, several resize handles and two text handles display. Use these handles to reposition and resize the box and text.



To reposition the box

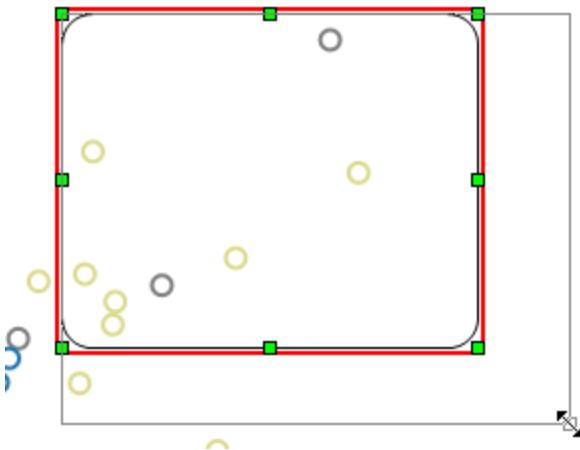
- Click and drag the box of the selected annotation to a new position.



To resize the box

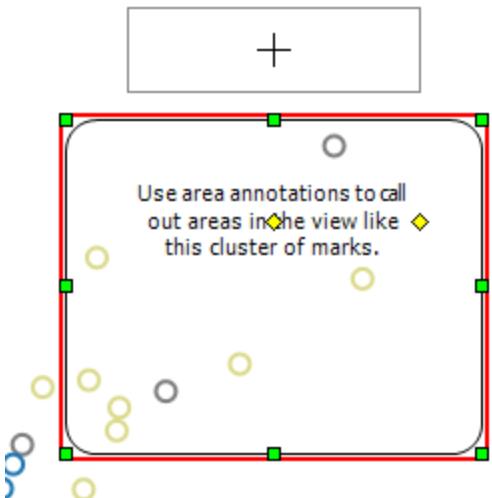
- Click and drag one of the box resize handles 

Use area annotations to call out areas in the view like this cluster of marks.



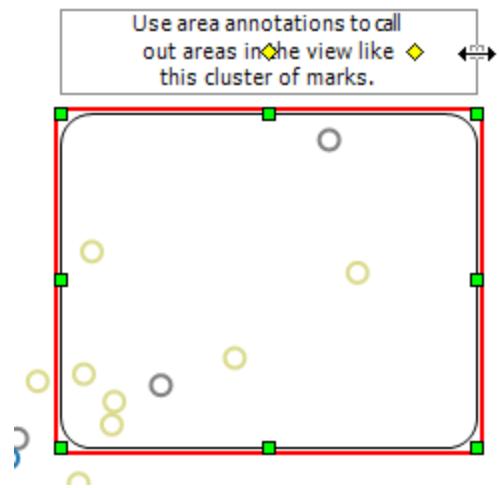
To reposition the text

- Click and drag the center text handle  to a new position.



To resize the text width

- Click and drag the right text handle  left and right. The text height is automatically adjusted to fit the width.

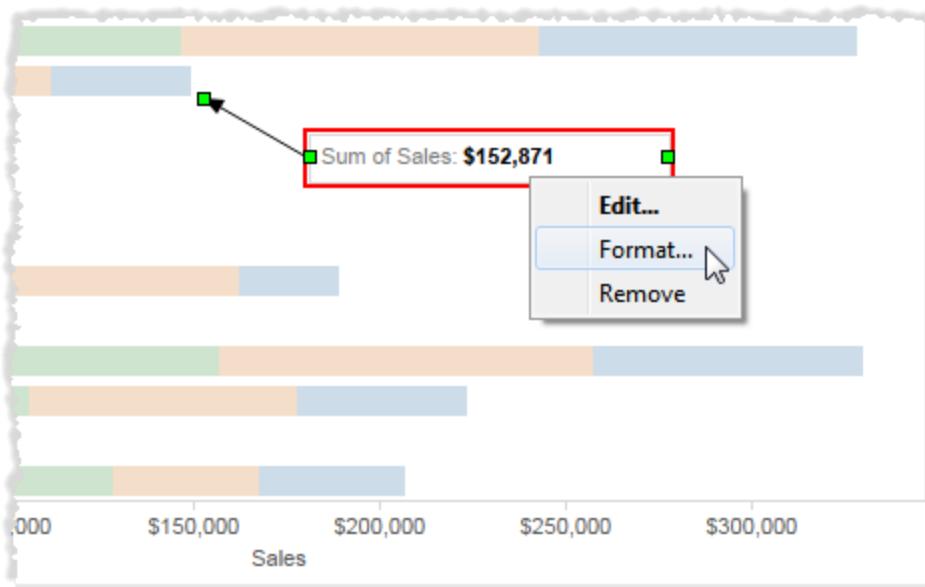


Formatting Annotations

For each annotation you can modify the text, body, and line. For example, you can specify whether the body should be a box, a single edge, or not shown at all. Additionally, you can specify whether the lines on mark and point annotations end with an arrow, dot, or a simple line.

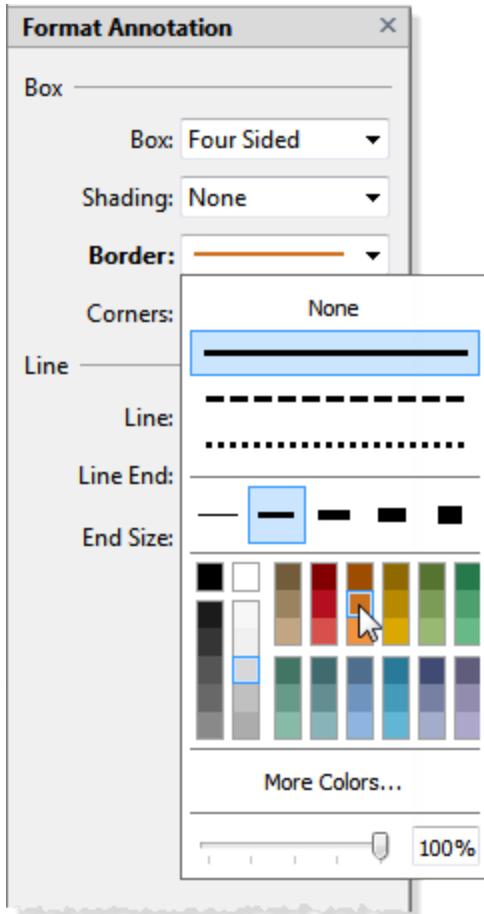
To format annotations:

1. Select one or more annotations, right-click (control-click on Mac) one of the selected annotations, and select **Format**. The **Format** pane opens showing the relevant settings.



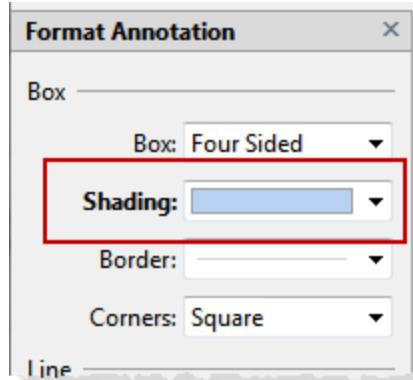
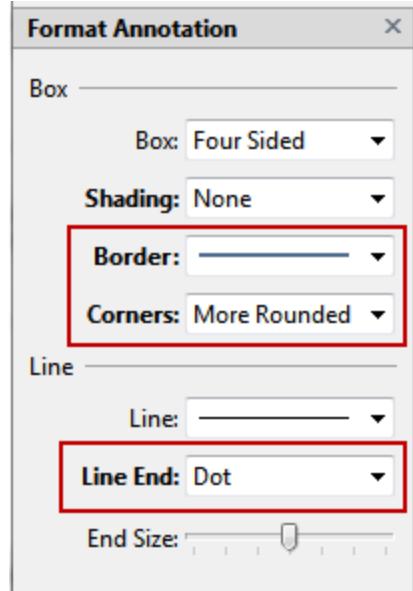
2. In the **Format** pane, use the drop-downs to specify font properties, text alignment, line

style, and shading.

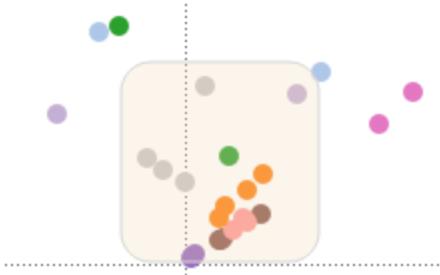
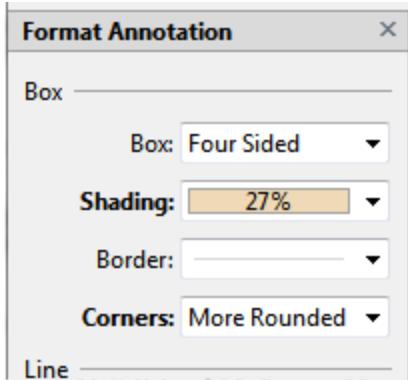
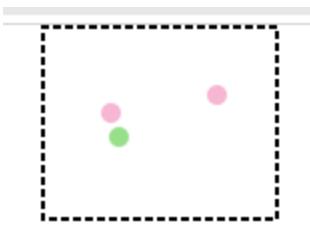
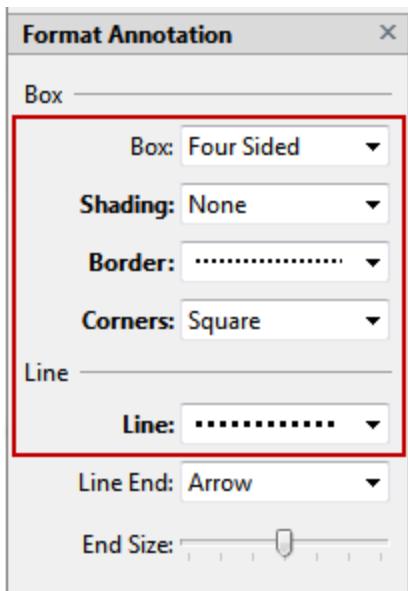


Example Formatting Options

Annotation Style	Format Window Settings
 An annotation box with a yellow border containing text: Profit: \$21,181 and Sales: \$129,770. An arrow points from the text to the border.	Default format settings for point and mark annotations.

Annotation Style	Format Window Settings
	 <p>Format Annotation</p> <p>Box</p> <p>Box: Four Sided</p> <p>Shading: </p> <p>Border:</p> <p>Corners: Square</p> <p>Line</p>
	 <p>Format Annotation</p> <p>Box</p> <p>Box: Four Sided</p> <p>Shading: None</p> <p>Border: </p> <p>Corners: More Rounded</p> <p>Line</p> <p>Line: </p> <p>Line End: Dot</p> <p>End Size: </p>

Annotation Style	Format Window Settings
 <p>Profit: \$21,181 Sales: \$129,770</p>	<p>Format Annotation</p> <p>Box</p> <ul style="list-style-type: none"> Box: Single Edge Shading: None Border: <input type="color" value="#000"/> Solid Corners: More Rounded <p>Line</p> <ul style="list-style-type: none"> Line: <input type="color" value="#000"/> Solid Line End: Dot End Size: <input type="range" value="1"/>
 <p>Profit: \$21,181 Sales: \$129,770</p>	<p>Format Annotation</p> <p>Box</p> <ul style="list-style-type: none"> Box: Four Sided Shading: None Border: <input type="color" value="#FF8C00"/> Dashed Corners: Square <p>Line</p> <ul style="list-style-type: none"> Line: <input type="color" value="#FF8C00"/> Dashed Line End: Open Arrow End Size: <input type="range" value="1"/>

Annotation Style	Format Window Settings
 <p>Annotate areas in the view like this cluster of marks.</p>	
	

Removing Annotations

At any time you can remove one or more annotations.

To remove an annotation:

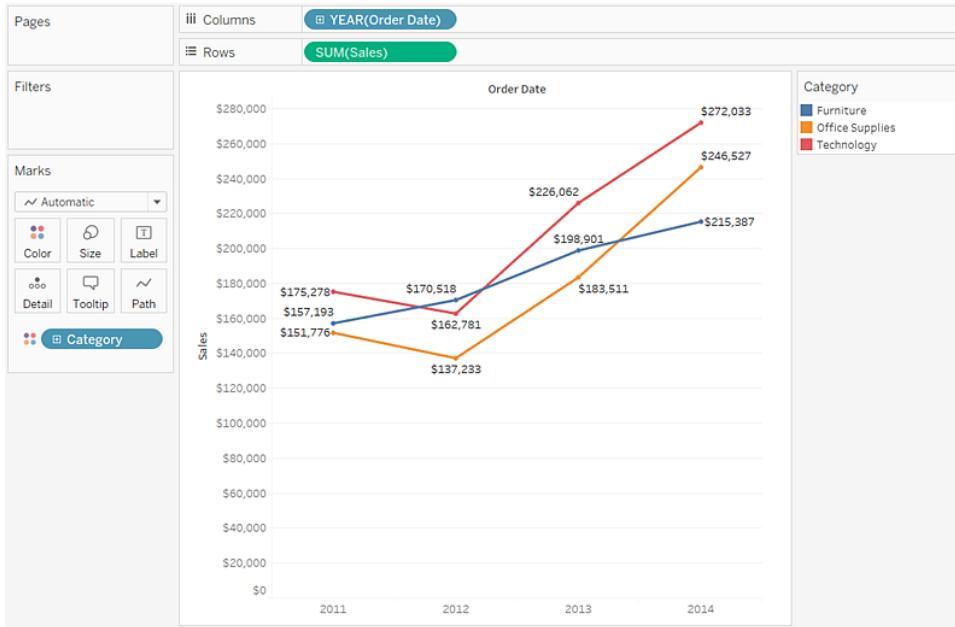
1. Select one or more annotations to remove.
2. Right-click (Control-click on Mac) one of the selected annotations and select **Remove** or press the Delete key on your keyboard.

Mark Labels

Mark labels are values shown next to each data point in a view.

For example, in a view that shows product category sales over time as a line, you can label sales next to each point along the lines.

Or, you might label each product category.



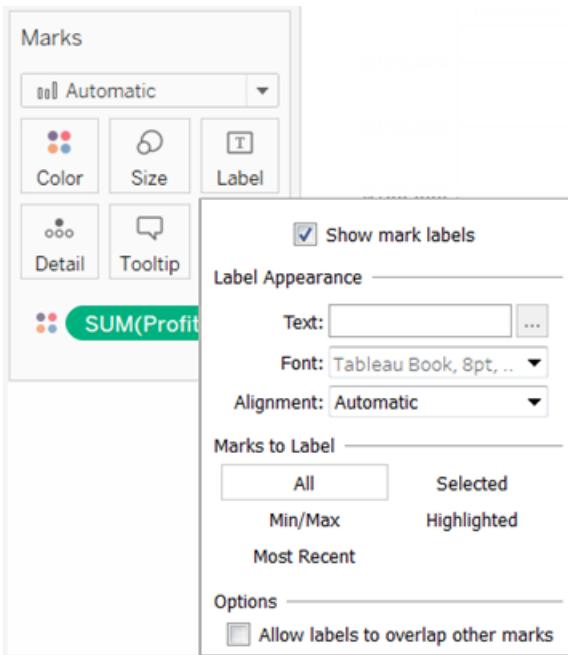
You can turn on mark labels for marks, selected marks, highlighted marks, minimum and maximum values, or only the line ends in a line chart. You can turn on mark labels for an individual marks.

Show and Hide Mark Labels

To add context to your views, you can show mark labels.

To show or hide mark labels:

- Click **Label** on the **Marks** card, and then select **Show mark labels**.



Label only specific marks

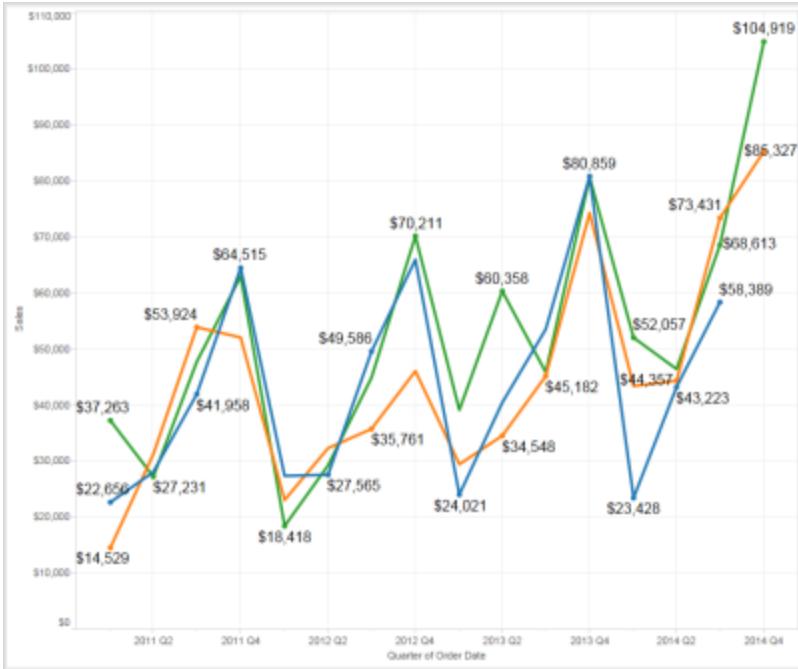
After you show mark labels in a worksheet, you can specify which marks to label.

To specify which marks you want to label:

- Click **Label** on the **Marks** card, and then make a selection from the following options under the **Marks to Label** section:

All

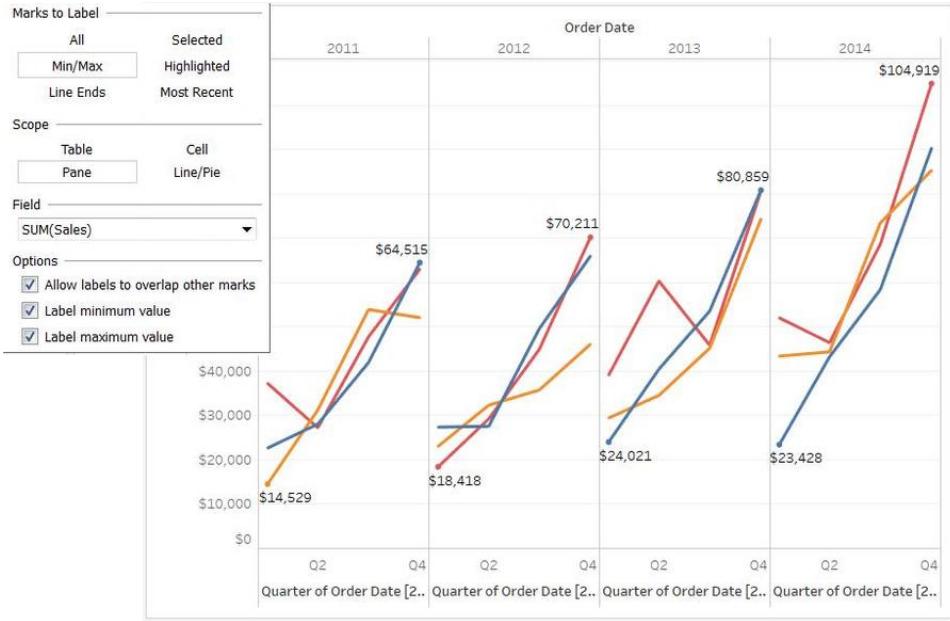
Label all marks in the view.



Min/Max

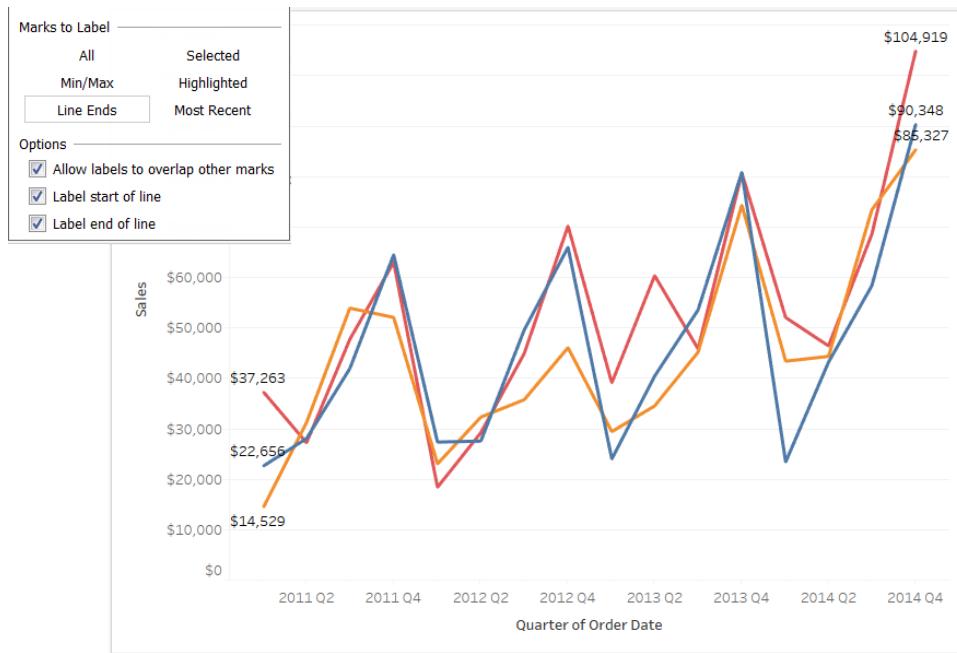
Label only the minimum and maximum values for a field in the view. When you select this option, you must specify a scope and field to label by.

For example, the image below shows the mark labels scoped to pane, and specifies to show the minimum and maximum sum of sales values. This means that the minimum and maximum sales are labeled for each pane in the view. In this case, there are four panes in the view.



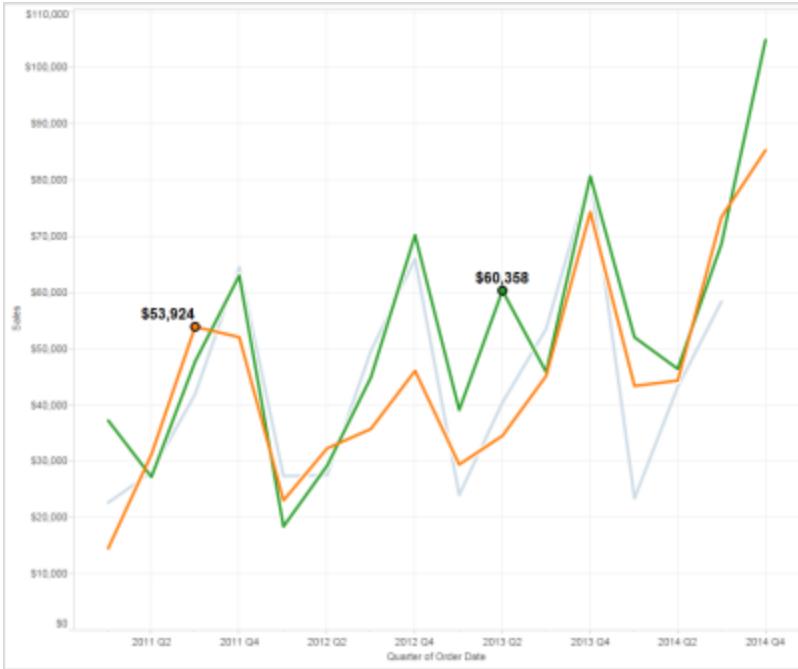
Line Ends

You can label the start or the end of all lines in the view, or both.



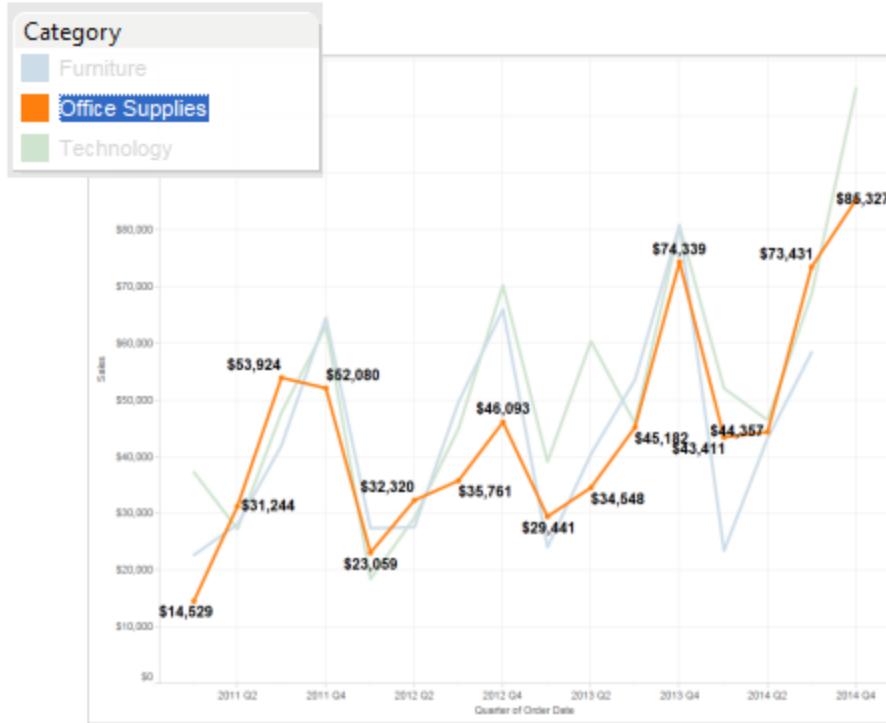
Selected

Label only the selected marks in the view. Labels appear when you select one or more marks in the view.



Highlighted

Label only the highlighted marks in the view. Labels appear when you select a member in a legend (for example, Office Supplies, in the following image), when you select one or more marks in the view, or when you select marks using the Highlighter. For more information about how to use the Highlighter, see [Highlight Data Points in Context on page 1125](#).

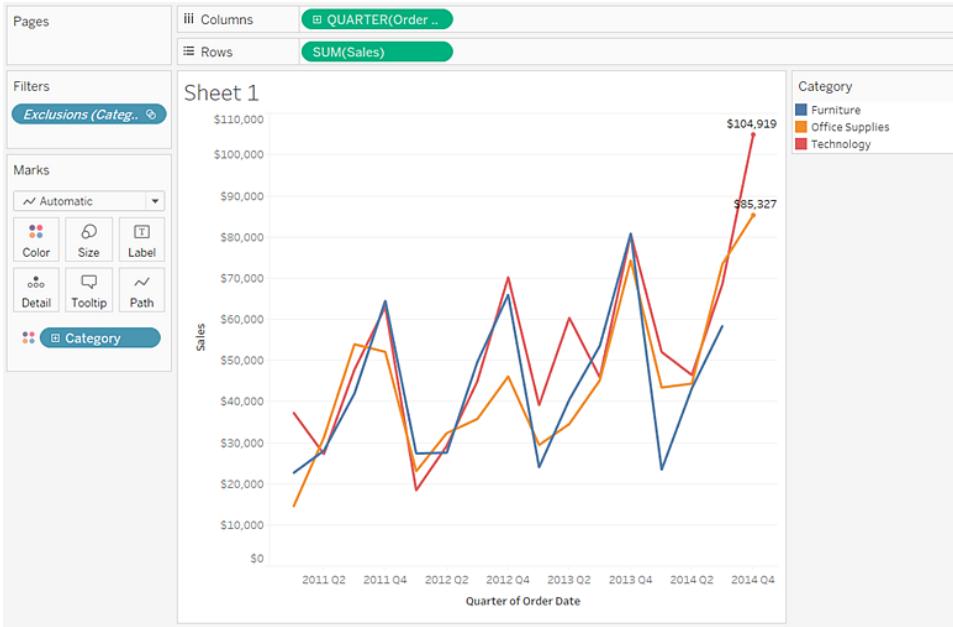


Most Recent

Label the most recent marks in the view. When there is a time or date field in the view, you can label all marks that correspond to the most recent date or time in the view. When you select this option, you can specify a scope for the labels in the view.

Marks labeled are determined by the level of detail of the view.

For example, the following image shows labels for each category, because the **Category** field is on **Color** on the **Marks** card. There is no label for the Furniture category, because the most recent mark in that category is excluded from the view.



Select to overlap other marks

By default, mark labels do not overlap other marks or labels in the view. You can show all labels in the view, even the ones that overlap other marks and labels.

To overlap other marks in the view:

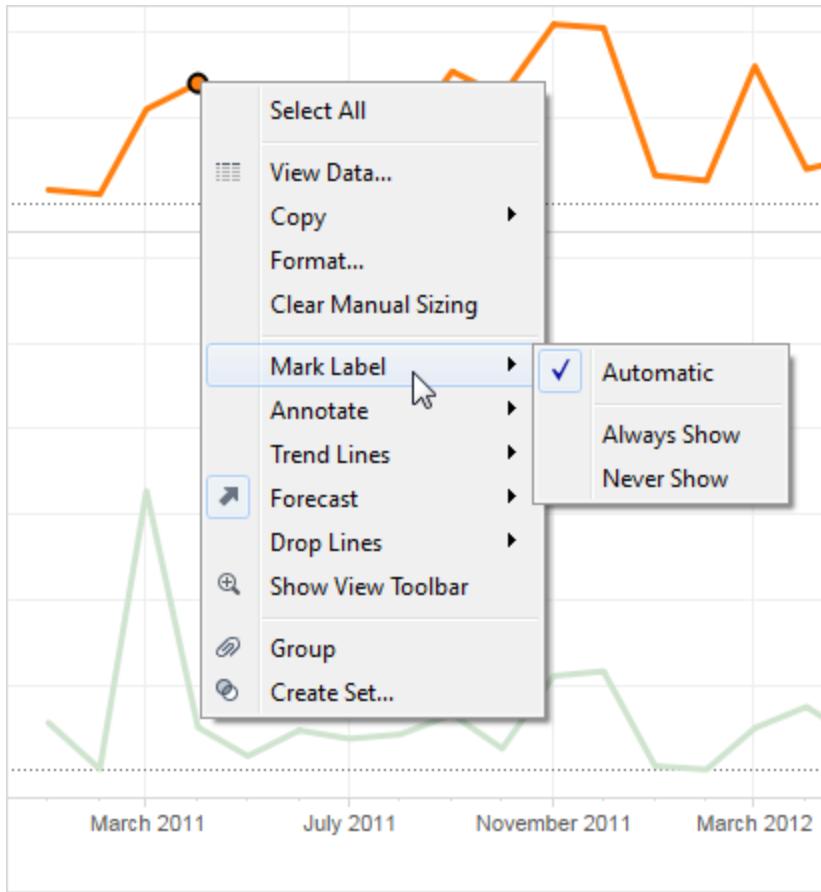
- Click **Label** on the **Marks** card, and then select **Allow labels to overlap other marks** under the **Options** section.

Show and Hide Individual Mark Labels

Rather than showing all mark labels or dynamically showing labels based on the view, you might want to show labels for a selection of individual marks. You can use mark labels to call out the values of specific marks of interest, as well as hide overlapping mark labels.

To show or hide individual mark labels:

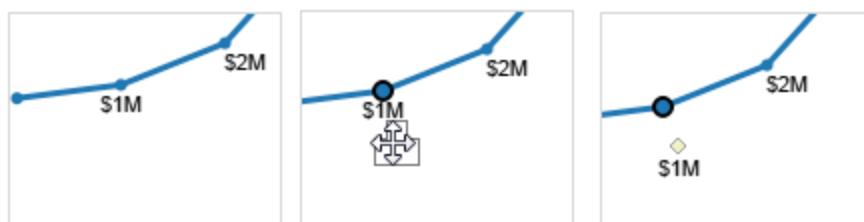
- Right-click (control-click on Mac) the mark you want to show or hide a mark label for, select **Mark Label**, and then select one of the following options:
 - Automatic** - select this option to turn the label on and off depending on the view and the settings in the Label drop-down menu.
 - Always Show** - select this option to show the mark label even when it would otherwise be hidden (based on the settings in the Label drop-down menu).
 - Never Show** - select this option to hide the mark label even when it would otherwise be shown (based on the settings in the Label drop-down menu).



Move Mark Labels

After you show a mark label in a view, you can reposition it to best fit your view and presentation. For example, in a stacked bar chart, the mark labels are automatically placed in the center of each bar. However, you may want to stagger the labels so that the longer ones don't overlap.

1. Select the mark whose mark label you want to move.
2. Click and drag the move handle to a new location.



Edit Mark Labels with Aliases

Another way to modify mark labels is to edit the aliases of a field. An alias is an alternative name assigned to a dimension member, or to a field name. Tableau gives you the ability to display and edit aliases for data sources that support this feature. When you edit the aliases you can change the names of the members in a field, thus modifying the mark labels displayed in a view.

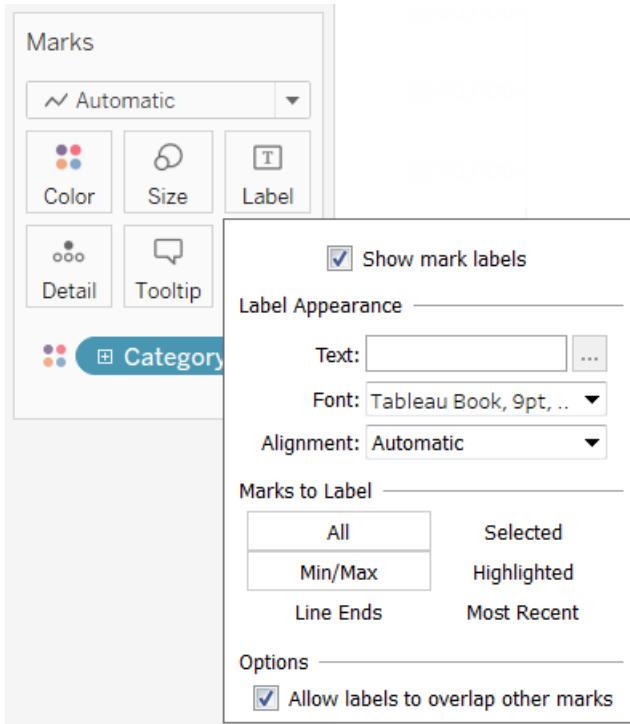
Format Mark Labels

When you select to show mark labels in the view, there are several formatting options to help you adjust the appearance of the labels. These options are available in the **Label** drop-down menu.

You can use the **Label** drop-down menu to adjust the label appearance, as well as specify which labels appear in the view. You can customize the text, adjust the font properties, and set an alignment for all labels.

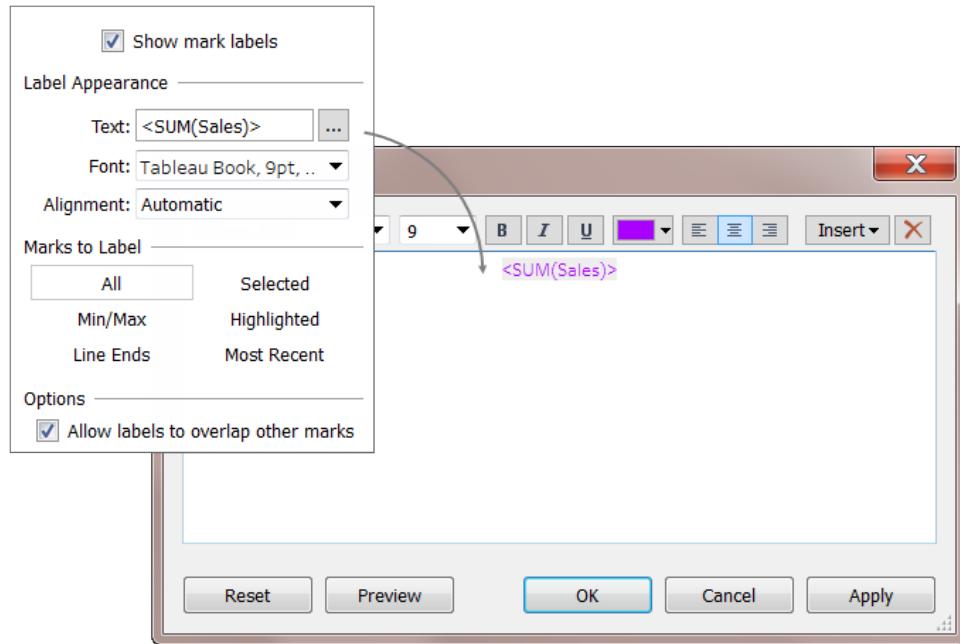
For more information about specifying which labels appear in the view, see [Show and Hide Mark Labels](#) on page 549.

1. On the **Marks** card, add a field to **Label**.
2. On the **Marks** card, click **Label**.



3. Change the text for all labels in the view.

- In the **Label Appearance** section, click the **Edit Text** icon. This opens the Edit Label dialog box where you can customize the label text.

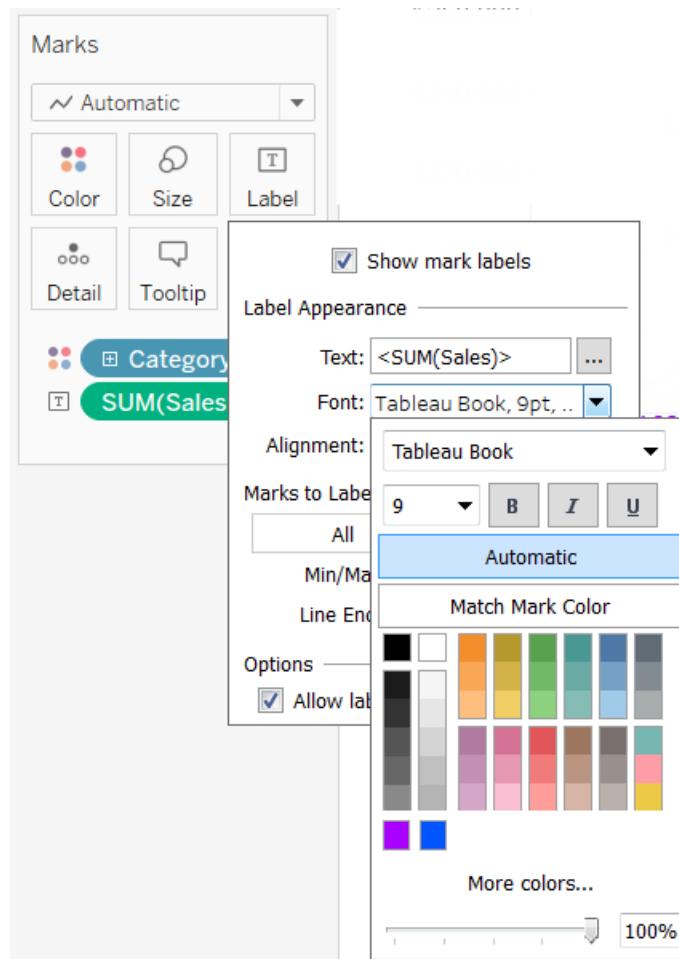


4. Customize the appearance of the font for all labels in the view.

- In the **Label Appearance** section, click the **Font** drop-down arrow. This opens the **Font** drop-down menu.

In the **Font** drop-down menu, you can select a font type and size, select to bold, italicize, or underline the font, adjust the opacity of the labels, and select a color for

the labels. For more information, see [Select Label Color](#).

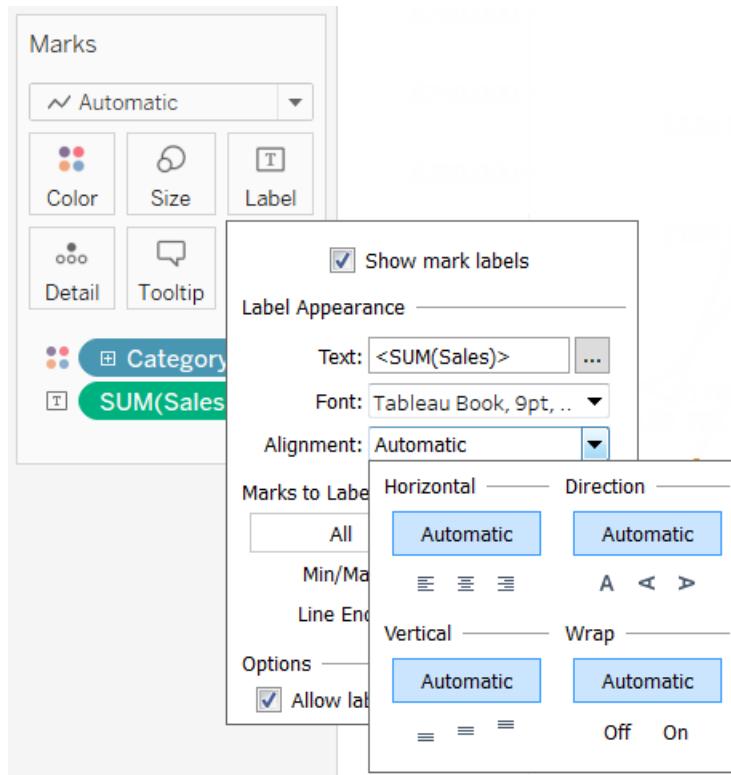


5. Change the alignment, direction, and text wrap for all labels in the view.

- In the **Label Appearance** section, click the **Alignment** drop-down arrow. This opens the Alignment drop-down menu.

In the Alignment drop-down menu, you can select a horizontal and vertical alignment, and a direction for all labels. You can also wrap the text for all labels in

the view.



6. Select label color.

By default, when you choose to show mark labels, the label colors automatically appear black or white in the view. However, you can also customize the color of the labels.

- On the **Label** drop-down menu, click the **Font** drop-down arrow. This opens the **Font** drop-down menu.

In the **Font** drop-down menu, you can select a specific color for all the mark labels in the view, or you can closely match each label to the color of its mark.

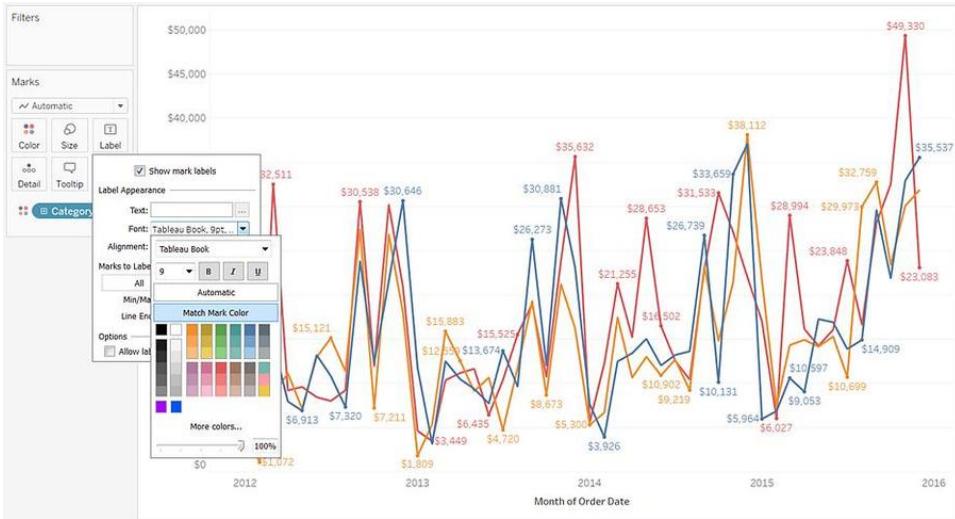
- Select a specific color:**

To select a specific color, click a color in the **Font** drop-down menu.

If you want to create a custom color for the mark labels, click **More Colors**, and then create a color in the **Select Color** dialog box that opens.

- Match label colors to the mark colors:**

To closely match each label to the color of its mark, click **Match Mark Color** in the font drop-down menu.



Inspecting Data

Once you have created a view, Tableau offers a selection of dynamic data inspection tools that help you isolate the data of interest and then continue to explore and analyze. For example, if you have a dense data view, you can focus on a particular region, select a group of outliers, view the underlying data source rows for each mark, and then view a summary of the selected marks including the average, minimum, and maximum values.

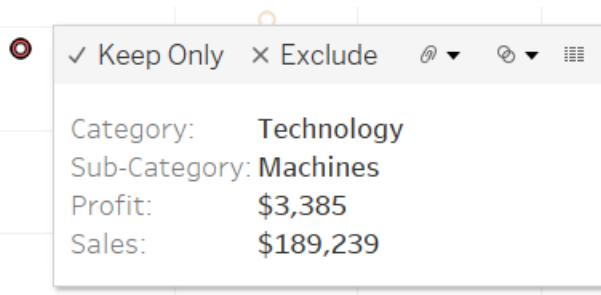
Marks and Data Analysis

Selecting marks in the view can help you gain insight into how a subset of your data compares to the overall data in your view. For example, you can view the underlying data of selected marks using tooltips, or compare the average sales of the selected marks to the overall average sales of the marks in the view using recalculated lines.

Data analysis options in Tooltips

When you select one or multiple marks in a view, several options for inspecting your data are available in the tooltip that appears. Tooltips provide additional information about the marks in your view, and provide the following data analysis options:

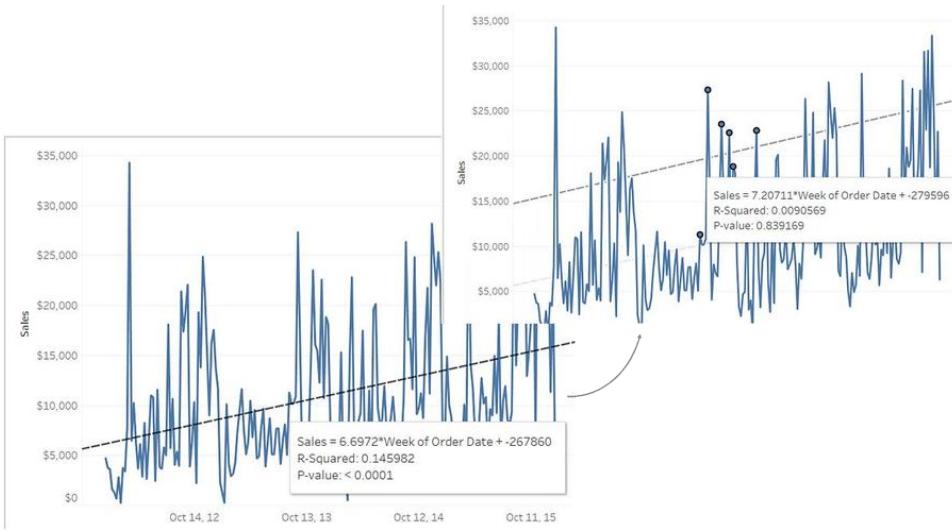
- Keep only the selected marks in the view. [Keep Only](#) For more information, see [Select to keep or exclude data points in your view](#) on page 673.
- Exclude the selected marks from the view. [Exclude](#)
- Create a group based on the selected marks. For more information, see [Groups](#) on page 497.
- Create a set that contains the selected marks. For more information, see [Sets](#) on page 512.
- View the underlying data of the marks selected. For more information, see [View Data](#) on page 573.



Data comparison with recalculated lines

If there is an analytics object in your view, such as an average line, constant line, trend line, reference line or distribution line or band, selecting one or multiple marks in the view lets you instantly compare the analytical data for the selected marks to all data in the view.

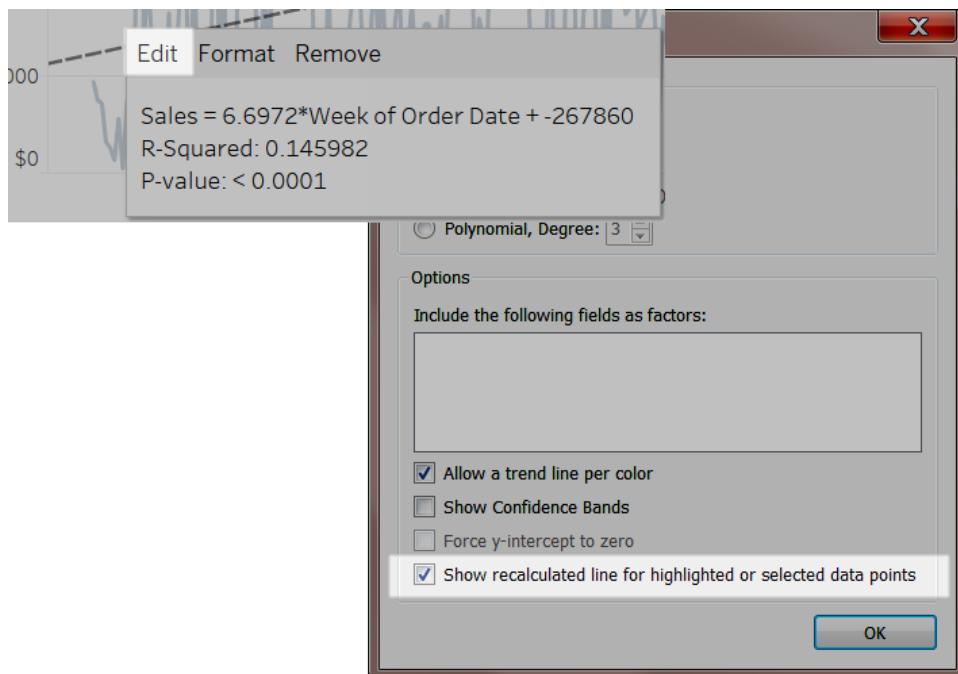
For example, selecting marks in a view that contains a trend line creates a second, recalculated trend line, the value of which is determined by the selected marks only, so you can compare that trend to the overall trend.



Recalculated lines are displayed by default when you select marks in a view that already contains analytics objects. If you don't want recalculated lines to be created you can turn them off.

Turn off recalculated lines:

1. Select an analytics object in the view, such as a trend line, and click **Edit**.
2. In the **Edit** dialog box, clear **Show recalculated line for highlighted or selected data points**.



Alternatively, you can right-click (control-click on Mac) an analytics object in the view and clear **Show Recalculated Line**.

When you return to the view and select or highlight marks, recalculated lines will not appear. For more information about how to highlight marks, see [Selecting Marks to Highlight on page 1122](#).

Recalculated lines and highlight actions

Recalculated lines also work with highlighting actions, both in worksheets and dashboards.

For example, if you create a highlight action in a dashboard, selecting marks in one sheet will highlight marks in the other sheets in the dashboard. If those other sheets have trend lines, reference lines, or other analytics objects, recalculated lines will appear as the views update.



Related Topics:

[Analytics Pane](#) on page 135

[Select Marks](#) on the next page

[Tooltips](#) on page 251

[Highlight Actions](#) on page 1120

View Toolbar

The view toolbar appears in the upper-left corner of the view and lets you select marks, zoom in and out, and pan in the view. By default, the view toolbar appears when you hover over a map view, but you can also show the view toolbar in other types of views, such as scatter plots or box plots.

For more information about showing and hiding the view toolbar in map views, see [Customize How People Interact with your Map](#) on page 781.



Show the view toolbar

- In a worksheet or dashboard, right-click (control-click on Mac) anywhere in the view, and then select **Show View Toolbar**.

Set when the view toolbar appears in the view

- Select **Worksheet > Show View Toolbar** (in a dashboard, select a view, then select **Worksheet > Show View Toolbar**), and then select one of the following options:
 - **Automatic** –The toolbar appears only when you hover over a map view.
 - **Show on hover** – The toolbar appears when you hover over the selected view. You can select this option for any type of view.
 - **Hide** –The toolbar does not appear in the selected view.

Hide the view toolbar

- In a worksheet or dashboard, right-click (control-click on Mac) anywhere in the view, and then select **Hide View Toolbar**.

Note: If the view toolbar is hidden, you can still use keyboard shortcuts to select marks, zoom, and pan. For more information, see [Keyboard Shortcuts on page 1611](#).

Select Marks

You can select marks in the view to inspect your data. When you select a mark or a subset of marks in the view, you can see information about the marks in the tooltip that appears. You can also quickly filter the marks you select from the view, as well as view their underlying data. For more information, see [Tooltips on page 251](#).

Select marks

Click an individual mark to select it. After you select a mark, hold down the Ctrl key (Command key on Mac) to add more marks to a selection.

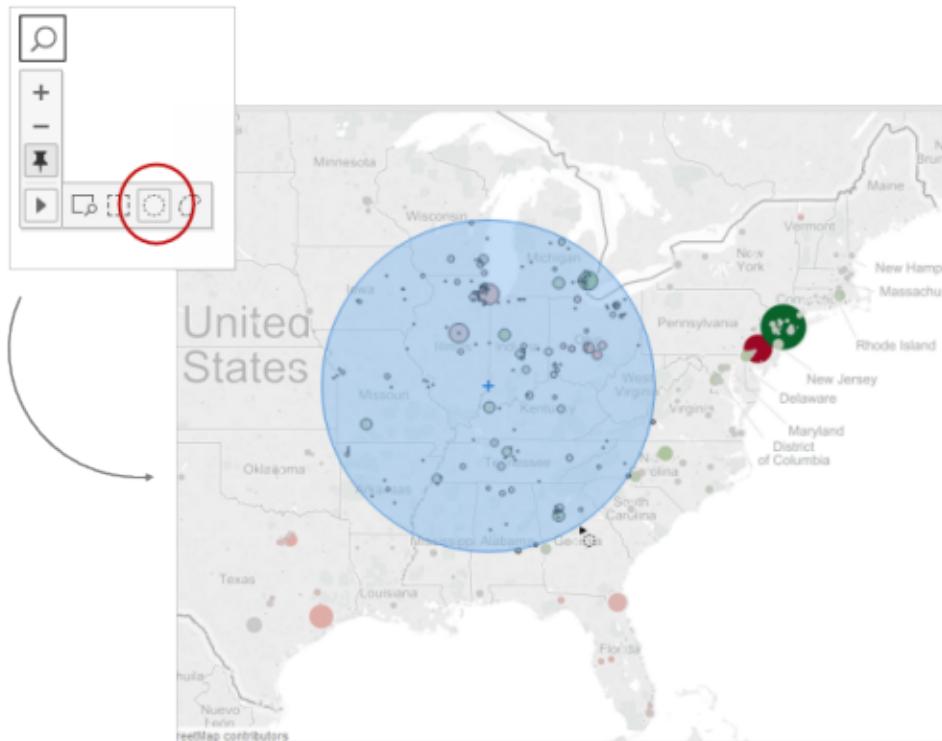
To select multiple marks using the default selection tool, click and drag across the view. In most views, the Rectangular selection tool is the default tool.

You can also use the Radial, Rectangular, and Lasso tools on the view toolbar to select multiple marks. For more information about the view toolbar, see [View Toolbar on the previous page](#).

Radial selection tool

The Radial tool selects marks within a circular area. To use the Radial tool, hover over the

arrow on the view toolbar, click the Radial tool button , and then click and drag across the view.

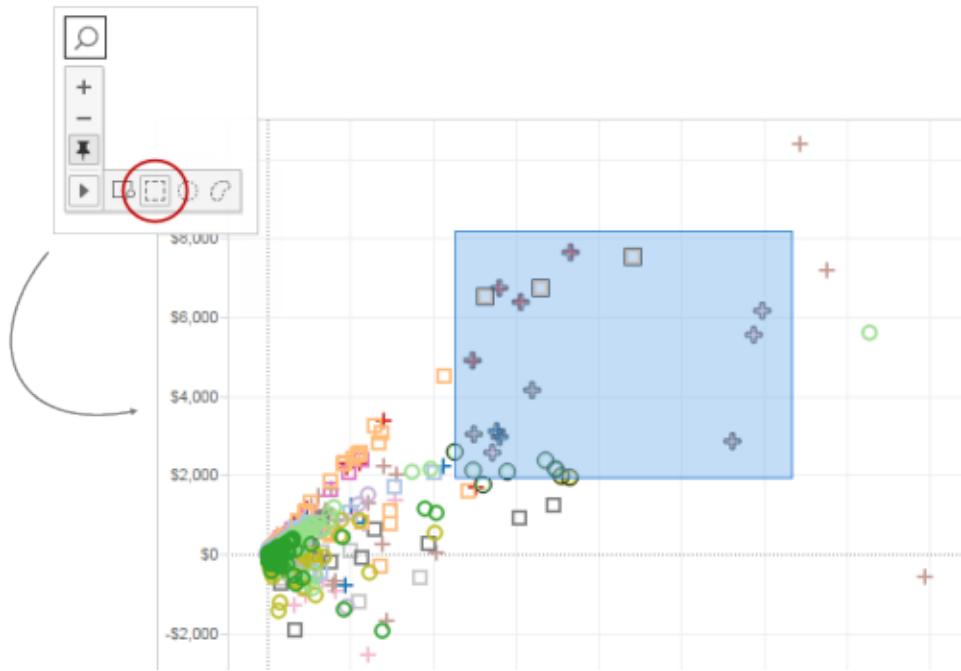


You can also measure distance in a map view with the Radial tool. For more information, see [Measure Distances Between Data Points and Locations in a Map on page 794](#).

Rectangular selection tool

The Rectangular tool selects marks within a rectangular shape. To use the Rectangular tool,

hover over the arrow on the view toolbar, click the Rectangular tool button , and then click and drag across the view.

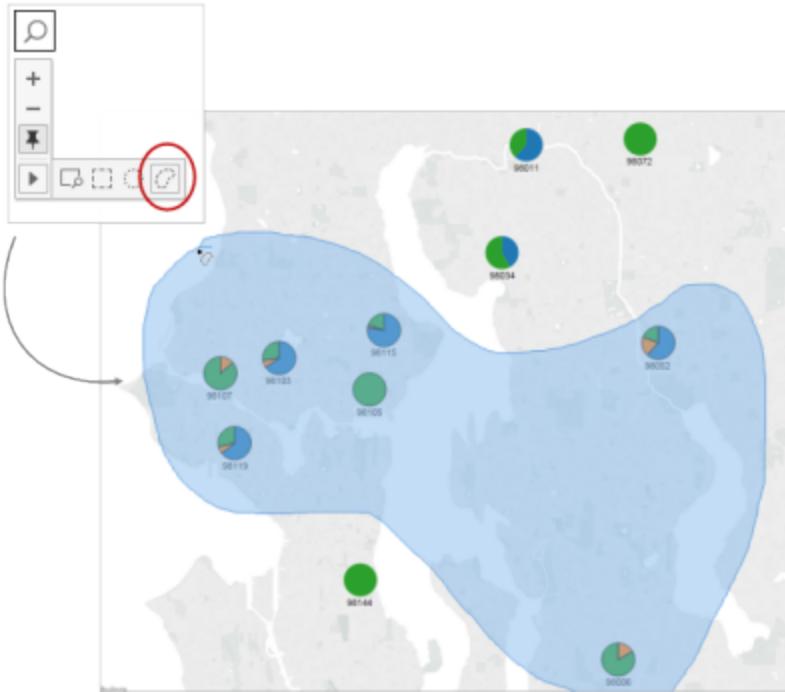


Note: The Rectangular tool is the default tool in most views, and may not appear in the view toolbar. In this case, you can drag across the view to use the rectangular tool.

Lasso selection tool

With the Lasso tool, you can select multiple marks by drawing a freehand shape around them. This tool is useful when you want to include only certain marks, and exclude others around them.

To use the Lasso tool, hover over the arrow on the view toolbar, click the Lasso tool button , and then draw a freehand shape around the marks you want to select.



Pan and Zoom

The pan tool and zoom controls help you interact with the view and inspect your data. They are located in the upper-left corner of the view, on the view toolbar. For more information, see [View Toolbar on page 565](#).

You can use the zoom controls to zoom in and out, zoom to a specific area, and fix or reset the axes in the view. Use the pan tool to move quickly around the view.

Zoom in and out

On the view toolbar, click the Zoom In button to zoom in and the Zoom Out button to zoom out. If the view toolbar is hidden, double-click the view to zoom in; to zoom out, hold down Shift, and then double-click the view.

Zoom to a specific area

To zoom in to a specific area of the view, click the Zoom Area tool button on the view toolbar, and then drag to create the zoom area. If the view toolbar is hidden, hold down Ctrl + Shift (Command-Shift on Mac) to use the Zoom Area tool.

Reset the view

After you zoom in or out, the axes in the view are fixed to a specific range. To quickly reset the axes so they automatically zoom the view to all of your data, click the Reset Axes button



on the view toolbar.

Pan

To pan, do one of the following:

- Hold down Shift, and then drag across the view.
- On the view toolbar, hover over the arrow, select the Pan tool , and then click and drag across the view.

Undo and Redo

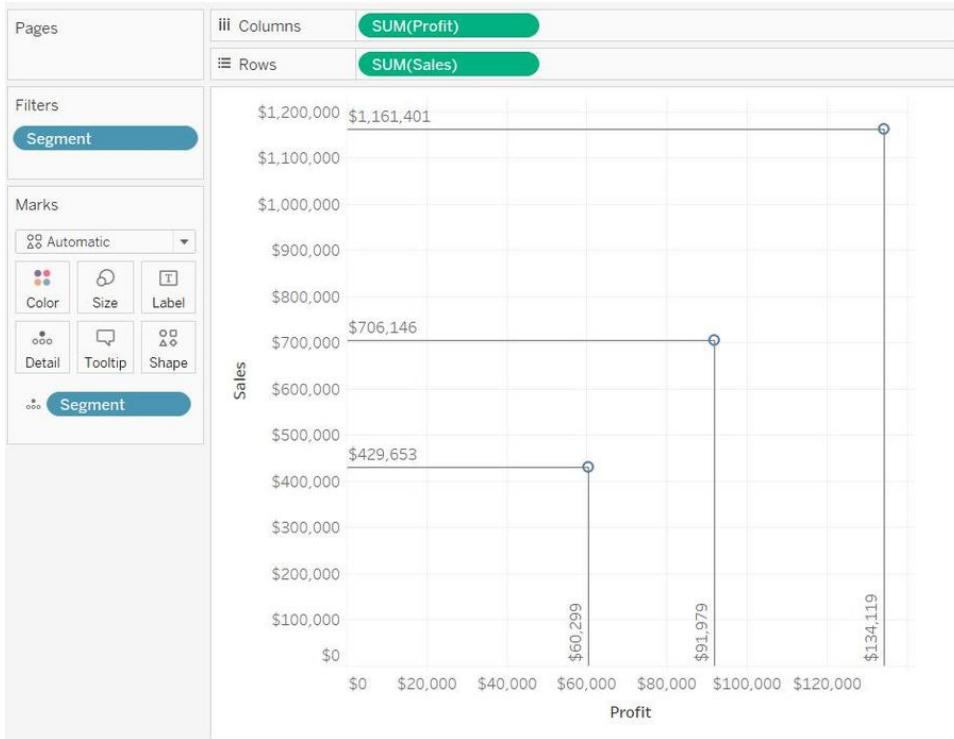
You can perform unlimited undo and redo of your actions. You can undo almost all actions in Tableau by pressing the **Undo** button on the toolbar. Likewise, you can redo almost all actions by pressing the **Redo** button on the toolbar.



In this regard, every workbook behaves like a web browser. You can quickly return to a previous view. Or you can browse all the views of a data source that you have created. Tableau saves the undo/redo history across all worksheets until you exit. The history is not saved between sessions.

Drop Lines

Use drop lines to distinguish marks and call out their position in the view. For example, in a view that is dense with scatter marks, you can turn on drop lines to show the position of a particular data point. When you add drop lines, a line is extended from the marks to one of the axes. You can show drop lines all the time or only when a mark is selected.



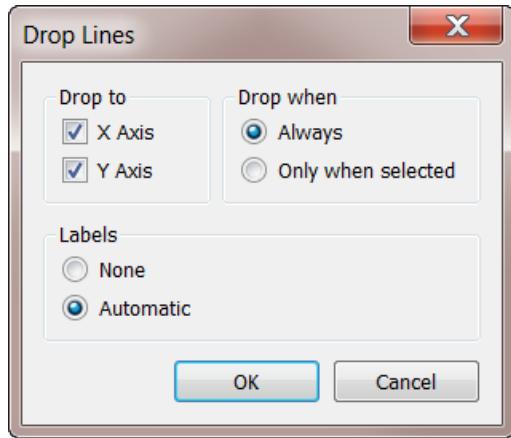
Note: Drop lines do not display when you publish a view to Tableau Server or Tableau Online.

To add drop lines to the view, right-click (control-click on Mac) the pane and select **Drop Lines > Show Drop Lines**.

By default, drop lines are set to only show when the mark is selected. You can change this setting and specify other options in the Drop Lines dialog box.

Right-click (Control-click on Mac) the pane and select **Drop Lines > Edit Drop Lines** to open the Drop Lines dialog box.

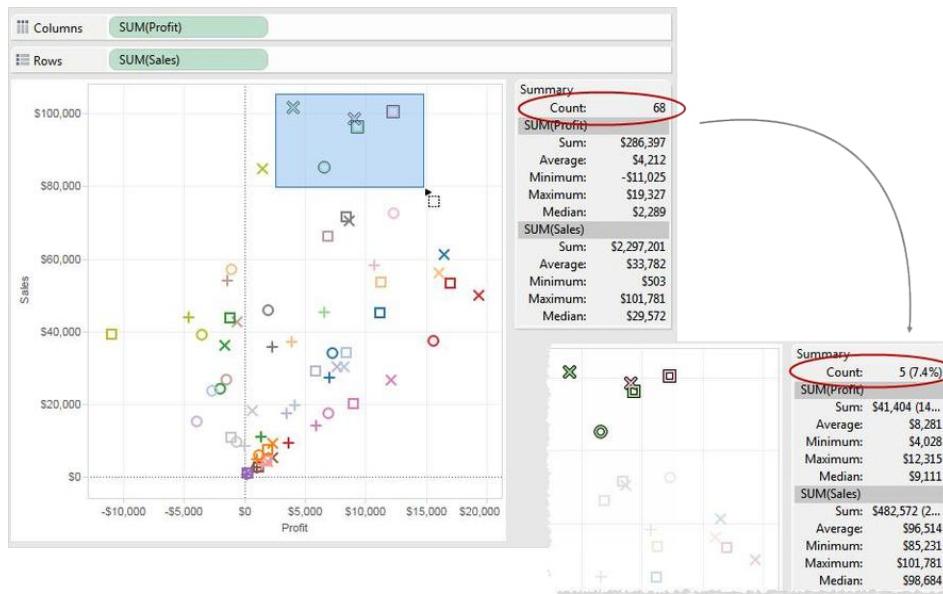
In the Drop Lines dialog box select an axis to draw the line to, whether to always show the drop lines, and whether to show labels.



Summary Card

The **Summary Card**, available on the **Show/Hide Cards** toolbar menu  , provides a quick view of information about a selection or the entire data source.

When you select data in the view, the **Summary Card** updates to show you information only for the data within the selection:



By default, the **Summary Card** shows Sum, Average, Minimum, Maximum, and Median values for the data in the view. The summary card values update to show these values for the current selection of data points. (Average is computed by summing all relevant values and then dividing by the total number of values. Median is computed by sorting values from lowest to

highest and then selecting the middle value.) The Count value at the top of the card indicates the number of marks in the view or selection.

You can use the drop-down menu for the **Summary Card** to show additional statistics:

- **Standard Deviation**

A measure of data spread around its average, measured in the same units as the data itself. The sample standard deviation is an unbiased estimate of the population standard deviation given a slight correction. This standard deviation includes the correction.

- **First Quartile**

A measure of location that is commonly used with other quartiles to provide a robust measure of spread. Robust in this case means not as sensitive to outliers as the standard deviation. The first quartile is the 25th percentile, typically the lower line in a boxplot.

- **Third Quartile**

A measure of location that is commonly used with other quartiles to provide a robust measure of spread. Robust in this case means not as sensitive to outliers as the standard deviation. The third quartile is the 75th percentile, typically the upper line in a boxplot.

- **Skewness**

A measure of the tendency of your data to have extreme values to one side. Positive skewness means the extreme values are to the right, while negative skewness means the extreme values are to the left.

- **Excess Kurtosis**

A measure of the tendency of your data to have more extreme or outlying values than a normal distribution. A normal distribution has a kurtosis of 3 so this value is kurtosis minus three.

View Data

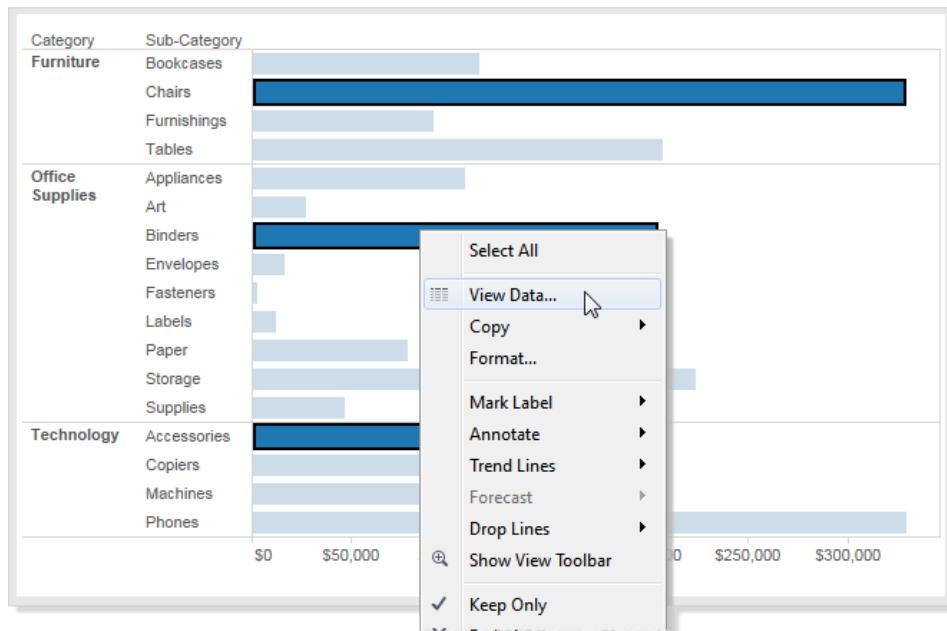
The View Data command lets you display the values for all rows in the data source that underlie a set of marks in the view. It also shows you the summary data based on the aggregations in the view. You can view data to verify the aggregated value associated with a mark, or to isolate and export the individual rows associated with data of interest, such as outliers.

You can view data for a selection of marks, for the fields in the Data pane, and when you're connecting to data.

The View Data command works with all relational and multi-dimensional databases except Oracle Essbase and SAP® Business Information Warehouse databases. While you can view data with the Microsoft Analysis Services and the Teradata OLAP connector multi-dimensional databases, the database must be drill-through enabled; in addition there are some restrictions

to the data you can see. Multi-dimensional data sources are supported only in Tableau Desktop on Windows.

In the view below, sales for two product dimensions (**Department** and **Container**) are displayed as a bar chart. Suppose you wanted to view data for the largest marks in each pane. To do this, you would select those marks, right-click (control-click on Mac) in the table, and select **View Data** on the context menu. Alternatively, you could select the **Analysis > View Data** menu item.



Viewing data may not return any records if you are using a field that contains floating point values as a dimension. This is due to the precision of the data source and mainly occurs when you are connected to Microsoft Excel, Microsoft Access, or text files.

Summary Data

Summarized data is shown on the **Summary** tab. Summarized data is a text table of the aggregated data for the fields shown in the view.

View Data: Sheet 18

Category	Sub-Category	Sales
Furniture	Chairs	\$328,449
Office Supplies	Binders	\$203,413
Technology	Accessories	\$167,380

Summary Underlying 3 rows

Underlying Data

Underlying data for the selected marks are displayed on the **Underlying** tab. In the lower right of the dialog box you can see the number of rows in the underlying data.

View Data: Sheet 18

Category	City	Country	Customer Name	Order Date
Furniture	Henderson	United States	Claire Gute	11/9/2013
Office Supplies	Los Angeles	United States	Brosina Hoffman	6/9/2011
Office Supplies	Seattle	United States	Irene Maddox	12/6/2013
Office Supplies	Fort Worth	United States	Harold Pawlan	11/22/2012
Office Supplies	San Francisco	United States	Zuschuss Donatelli	8/27/2011
Furniture	Philadelphia	United States	Sandra Flanagan	7/17/2014
Office Supplies	Los Angeles	United States	Eric Hoffmann	1/16/2013
Technology	Los Angeles	United States	Eric Hoffmann	1/16/2013
Office Supplies	Philadelphia	United States	Tracy Blumstein	9/17/2012
Office Supplies	Philadelphia	United States	Tracy Blumstein	9/17/2012
Furniture	Houston	United States	Steve Nguyen	12/27/2012

Summary Underlying 2,915 rows

Sort the data by clicking one or more column headers. To restore the original sort order, click the header repeatedly until it is no longer highlighted with a sort arrow.

By default, **Show all fields** is selected. Clear this option to only show the columns used on shelves (or fields referenced by a calculation used on a shelf) in the current worksheet.

If you want to export one or more data source rows, select the data points of interest by selecting the row and then clicking **Copy** to copy the selected data.

[View Data \(Microsoft Analysis Services\)](#)

View Data with a Microsoft Analysis Services database works almost the same way it does with relational data sources. The difference is that a Microsoft Analysis Services cube is generally set up and configured by an administrator who decides whether it is enabled for drill-through and the fields that a user is allowed to see. That means that when you try to view data using a database that is not enabled, you may get an error message alerting you that the cube is not enabled for drill-through.



In addition, Microsoft Analysis Services databases limit viewing data to a single mark at a time. More precisely, viewing the data (which uses MDX drill-through) is not an option when the selection of mark(s) is defined by more than one value of a dimension.

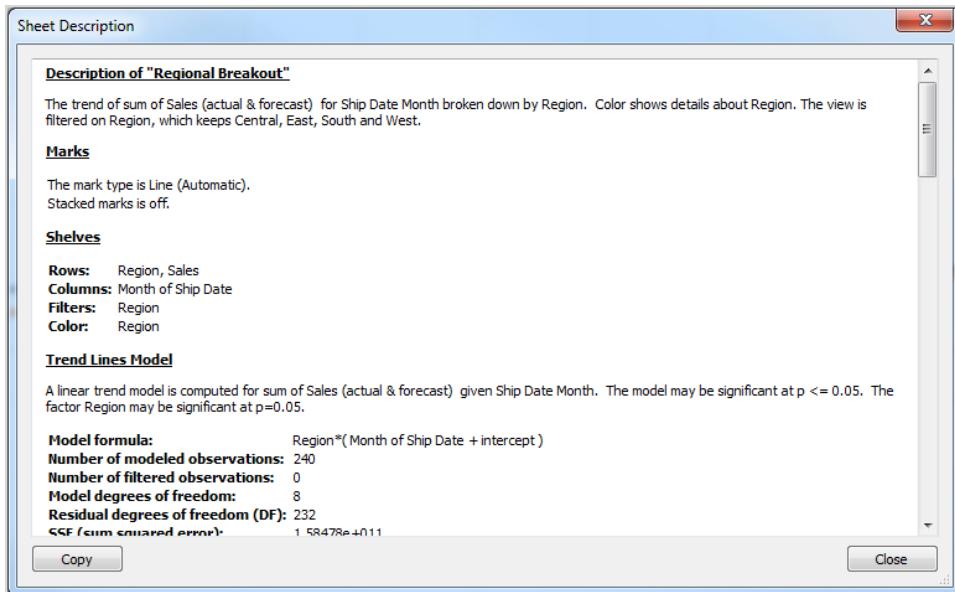
When you are viewing underlying data for a field, the **Show all fields** option is checked and disabled by default. With a Microsoft Analysis Services database, only the fields specified by the administrator are shown, so you cannot choose to include all data source fields in the dialog box.

Describing the View

Occasionally you may want to succinctly summarize an analysis you have completed on a worksheet. You might then want to remind yourself of what it shows (the filters that are applied, etc.), and finally, you may want to share a summary of the analysis with someone else.

When you choose **Worksheet > Describe Sheet**, you can view a description of the workbook, data source, fields and layout of the current worksheet. This summary includes the Caption in the first line, but provides other important summary information. This information can be copied and exported to other applications using the Clipboard.

Note: If you have Trend Lines turned on, the Describe Sheet dialog box includes information about the trend line model. For more information, see [Assessing Trend Line Significance on page 664](#). If you have Forecasting turned on, the Describe Sheet dialog box includes information about estimated data. For more information, see [Forecast Descriptions on page 650](#).



Multidimensional Hierarchies

Multidimensional data sources contain hierarchies. For example, your database might contain a Product dimension that includes members such as product family, product department, and so on organized into a hierarchy, or you might have a Time dimension that includes years, quarters, and months.

Data

Basic (Sample)

Dimensions

- Scenario**
- Market**
- Product**
 - Gen2, Product
 - Gen3, Product
 - Caffeinated
 - Ounces
 - Pkg Type
 - Intro Date
- Year**

Abc *Measure Names*

Data

Staples (TestV1)

Dimensions

- Call Center**
- Customer Segment**
- Employee**
- Market Segment**
- Order Date**
- Product**
 - Prod Type1
 - Prod Type2
 - Prod Type3
 - Prod Type4
 - Product Name
- Supplier**

Abc *Measure Names*

Note: In Tableau, multidimensional data sources are supported only in Windows.

Drilling Down and Up in a Hierarchy

One of the most useful ways to navigate hierarchies is to drill down or drill up. For example, if you are examining the sales totals for various years, you can then drill down and view sales for all of the months within each year. Alternatively, if you are examining sales totals for all months, you can then drill up and view the sales for each year.

You can drill down and drill up in Tableau by clicking on fields placed on shelves, or by selecting a hierarchy header in the table. These two methods are described below.

Using Fields on Shelves

You can drill down or drill up by clicking on a dimension that is placed on any shelf. If the dimension is on the Rows or Columns shelf, drilling down shows more data (more headers) in the table, while drilling up shows less data in the table.

You can click on the plus/minus control that appears on any hierarchical dimension on any shelf. If a dimension member shows the plus sign , then its children are not already showing and you can drill down at least one level. If a dimension member shows the minus sign , then its children are already showing and you can drill up.

The following figure demonstrates drilling down one level in the hierarchy for the Region dimension to expose the states within each region.

The diagram illustrates a drill-down operation. On the left, a table shows data at the Region level (East, West, South, Central) across categories Colas, Root Beer, Cream Soda, Fruit Soda, and Diet Drinks. The 'Region' header is highlighted with a red circle. An arrow points from this table to a second, larger table on the right. This second table shows data at the State level for each region. It includes columns for Region, State, Colas, Root Beer, Cream Soda, and Fruit Soda. The 'Region' header is also highlighted with a red circle in this second table.

Region	Colas	Root Beer	Cream Soda	Fruit Soda	Diet Drinks
East	27,740	23,672	20,241	15,745	7,919
West	28,306	31,234	29,056	20,821	10,200
South	16,280	18,521	15,845	10,234	5,765
Central	33,808	30,672	28,056	20,821	10,200

Region	State	Colas	Root Beer	Cream Soda	Fruit Soda
East	New York	8,940	7,939	9,305	8,514
	Massachusetts	6,518	5,180	1,418	1,541
	Florida	5,867	5,283	4,704	2,487
	Connecticut	3,378	3,090	3,849	2,094
	New Hampshire	3,037	2,180	965	1,109
West	California	12,096	16,794	11,128	7,424
	Oregon	4,250	6,743	2,456	6,543
	Washington	4,937	4,704	4,621	4,774

Using Headers

To drill down and drill up for individual dimension members in a hierarchy, right-click (control-click on Mac) a table header and select **Drill Down** or **Drill Up** from the context menu. This is often referred to as non-uniform drill down because you expose only the members of interest instead of exposing all the members of a given level.

For example, the following figure illustrates drilling down into the Root Beer member of the Gen2,Product dimension. Note that new row headers are displayed in the table and that Gen3,Product, which is the next generation in the hierarchy, is automatically displayed.

The diagram shows a context menu being used on a table header. The menu options are: Keep Only (checked), Exclude, Hide, Drill Down (highlighted with a mouse cursor), Format..., Rotate Label, and Show Header. The 'Drill Down' option is selected. An arrow points from this menu to a second, larger table on the right. This second table shows data at the Gen3,Product level for the Root Beer category. It includes columns for Gen2,Product, Gen3,Product, Qtr1, Qtr2, Qtr3, and Qtr4. The 'Gen2,Product' header is highlighted with a red circle in both tables.

Gen2,Product	Qtr1	Qtr2	Qtr3	Qtr4
Colas	25,048	27,187	28,544	25,355
Root Beer	401	27,942	27,116	
Cream Soda	736	26,650	25,022	
Fruit Soda	355	22,079	20,648	
Diet Drinks	787	27,495	25,665	

Gen2,Product	Gen3,Product	Qtr1	Qtr2	Qtr3	Qtr4
Colas		25,048	27,187	28,544	25,355
Root Beer	Old Fashioned	9,894	10,192	10,939	10,512
	Diet Root Beer	9,486	9,725	9,885	9,144
	Sasparilla	4,440	4,562	4,362	4,195
	Birch Beer	2,807	2,922	2,756	3,265
Cream Soda		23,997	25,736	26,650	25,022
Fruit Soda		20,148	21,355	22,079	20,648
Diet Drinks		25,731	26,787	27,495	25,665

One reason to use non-uniform drill down is if your data source has a ragged hierarchy (asymmetric layout). You also might want to view the children for just the member of interest.

Drilling down and drilling up results in filtering the data.

Building Views with Oracle Essbase

When Tableau is connected to an Oracle Essbase data source, there are three important features that you should know about:

Generations and Levels

In Tableau, you can work with either the generations or the levels of a dimension. The generations of a dimension are all members that are an equal distance from the root of the dimension. The levels are all members that are an equal distance from the leaves of the dimension. For balanced dimensions, you'll typically want to work with generations. However, if your dimension is ragged, then it may make more sense to navigate using levels.

By default, the generations of each dimension are listed in the Data pane. When you drag a dimension to a shelf, all generations that are ancestors of the selected generation (all generations that are above it in the hierarchy) are automatically included in the placement.

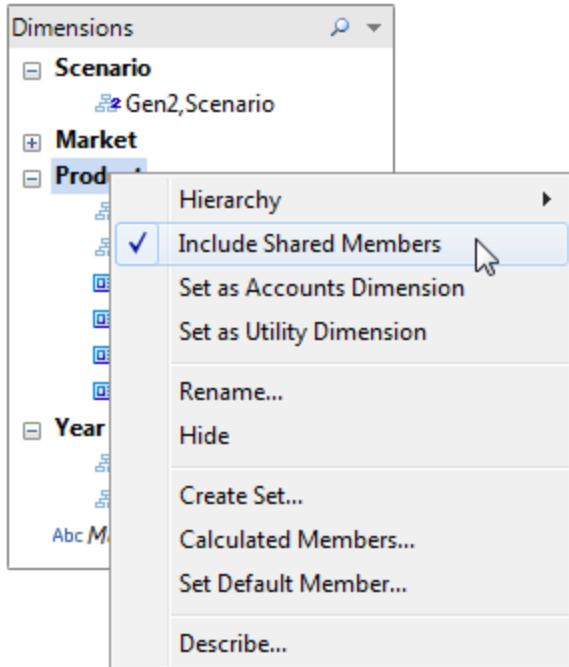
If you would rather navigate using the levels of a dimension, right-click the name of the dimension and then select **Hierarchy > Levels**.

If you are using the same dimension in multiple worksheets, you can use levels in one worksheet and generations in another worksheet simultaneously. Furthermore, you can mix generations and levels from different dimensions in the same worksheet.

Shared Members

Shared members are dimension members that appear in more than one place in a hierarchy. For example, Diet Coke might be part of the product generation. But it might be shared by both the diet colas branch and the colas branch of the product hierarchy above it. In the database, however, the data about Diet Coke is stored just once.

By default, Tableau includes shared members in all generations (or levels) of a dimension. This means that a shared member might appear multiple times in a table. If you choose to exclude shared members, they will appear only once in a table. By default, shared members are included for all dimensions. To exclude shared members for a given dimension hierarchy, right-click the dimension name in the Data pane and select **Include Shared Members** from the menu.



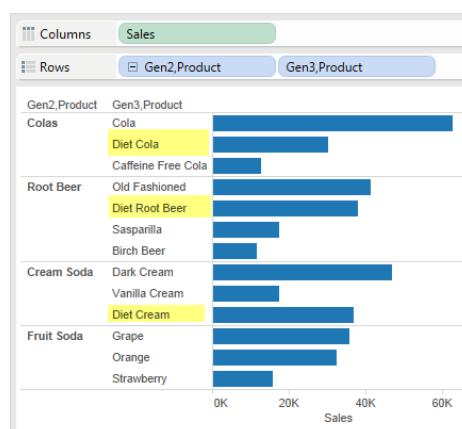
The figure below shows part of a data view where shared members are included (left) and excluded (right). Notice that diet drinks are shared members.

Includes Shared Members

The diet drinks are listed in both the Diet Drinks hierarchy and their respective drink hierarchies.

Does Not Include Shared Members

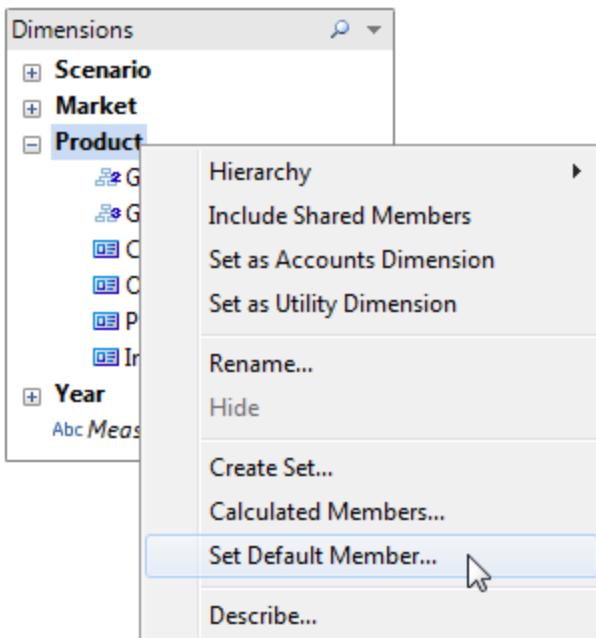
The diet drinks are only listed once, in their respective drink hierarchies.



Setting the Default Member

All multidimensional data sources have default members that are set when the data source is first built. If you find that you are creating filters all the time to look at the same specific data, you may find it useful to change the default member. For example, if you are the regional manager for the Western region in a company and you only want to look at your region's numbers, you can set the default member to the Western region.

To change the default member in Tableau, right-click a dimension hierarchy and select **Set Default Member**.



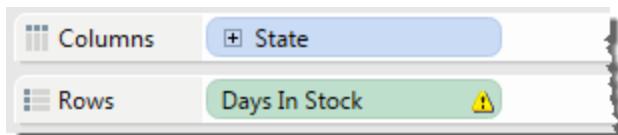
In the subsequent dialog box, select from the following options:

- Default member defined on cube – uses the default member that was defined when the cube was built. This is the default setting in Tableau.
- (All) member for the hierarchy – uses the ALL member for the selected hierarchy as the default member.
- Selected member – uses the member that you select in the bottom half of the dialog box as the default member.

The default member determines how you view the cube and so is much more powerful than applying filters. All fields will be calculated based on the default member you select. In addition, these default member settings are saved with the connection.

Building Views with Microsoft Analysis Services

When you build views in Tableau using a Microsoft Analysis Services Cube it is possible to have measures and dimensions that don't make a lot of sense when placed in the view together. For example, you may have a measure for Sales Quota. It won't make sense to place that measure against a dimension containing products if products don't have sales quotas. Tableau helps you figure out the dimensions and measure that can be used together in meaningful ways by highlighting unrelated dimensions and measures in gray. So in the last example, when we place Sales Quota onto a shelf, the products dimensions are highlighted in gray. Highlighted dimensions are not disabled and can still be added to the view. When you add an incompatible measure to the view, the measure is marked with a caution symbol.

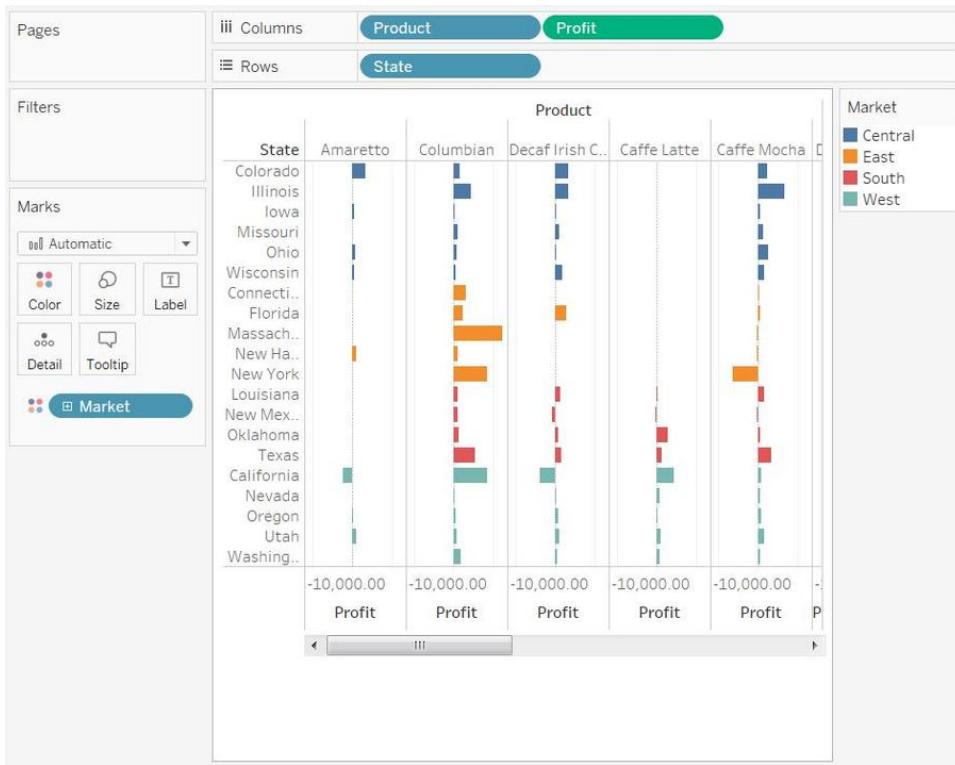


Perfect Pivoting

In Tableau, perfect pivoting refers to working with hierarchies in these ways:

- Using varying levels of detail including skipping levels (for example, Country and City, but not State).
- Using varying levels of detail on different worksheet shelves simultaneously (for example, Product Family on the Columns shelf and Product Department on Color).
- Using varying levels of detail out of order (for example, Quarter before Year).

For example, in the following view the Market hierarchy is broken up to show the State level as Rows and the Region level as Color.

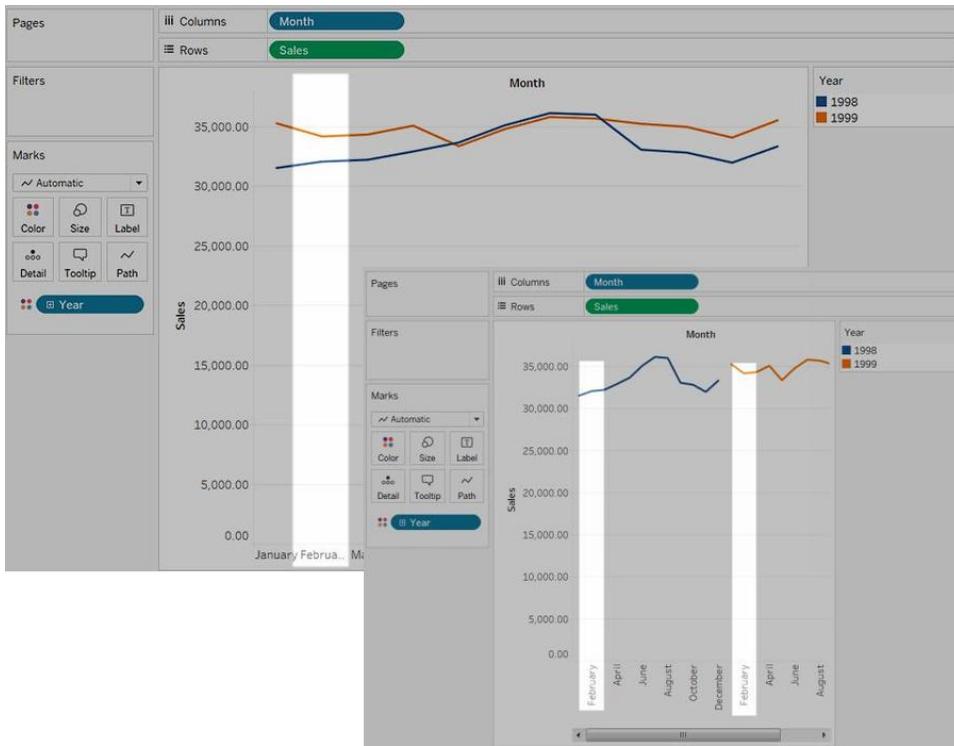


Defining Unique Values

Sometimes, when you are building views in Tableau, a field will have multiple members with the same name. For example, you may have a view showing the average profit by month over several years. The month February appears multiple times (once for each year).



While the name, February, is repeated, each instance of February can either be considered similar or unique. If you consider them similar, they will appear in the same column if you decided to move the Year field to the Color. However, if you consider them unique, they will be treated as two different values.



It is generally okay to consider repeated names within date and time fields (like in the previous example) similar but if there are repeated names in the Customer Name field, you won't want to consider the two customers as the same person.

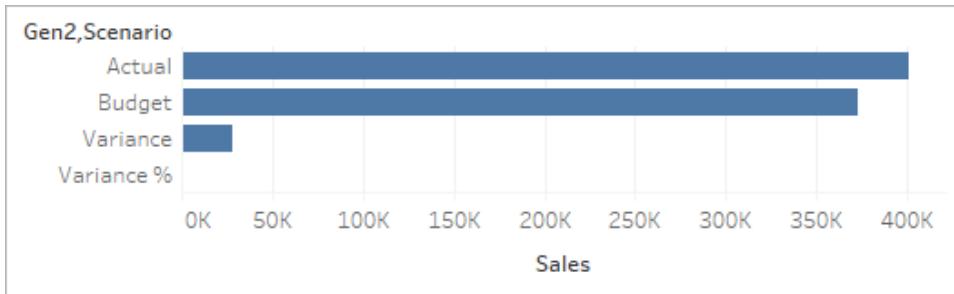
To define how you want Tableau to determine whether repeated values are unique, right-click (control-click on Mac) on the dimension in the **Data** pane and select one of the following on the **Unique Values** context menu:

- By Key: each member is considered unique based on the key given to it by the system administrator when the database is set up. Members with the same name but different keys are treated as unique values.
- By Name: each member is considered unique based on the member name. Members with the same name (regardless of their keys) are treated as if they are the same.

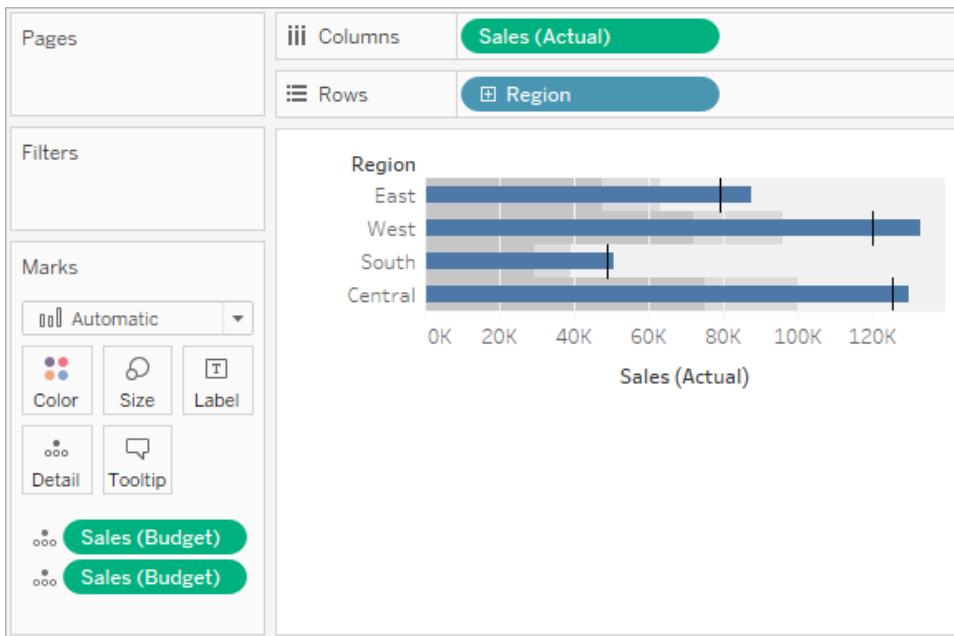
By default, unique date and time values are determined by name and all other values are determined by key.

Utility Dimensions

Oracle Essbase databases sometimes have special dimensions used to model comparative values such as Actual vs. Budgeted or Current Year vs. Previous Years. These dimensions are the utility dimensions and are often set up as Scenario or Years. For example, the members of a Scenario dimension are shown below.



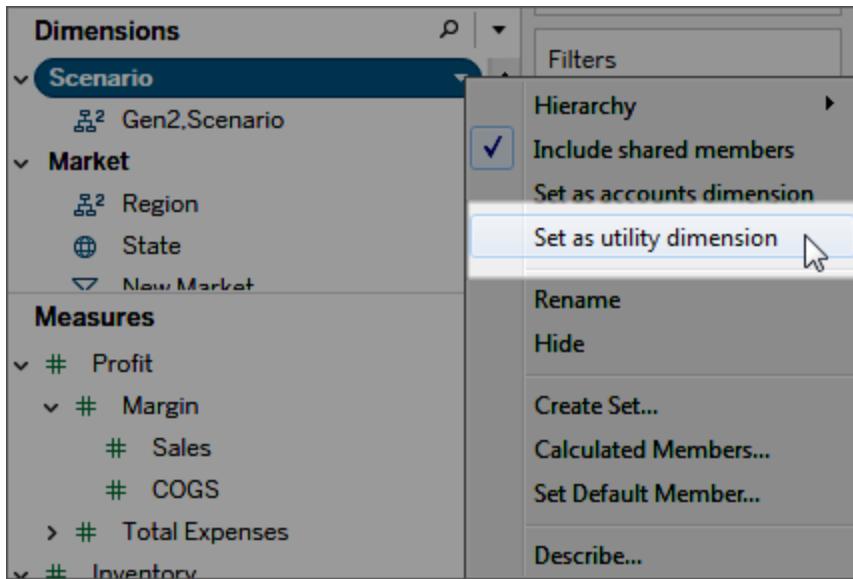
In the above view, you can see Actual Sales, Budgeted Sales, and so on. However, what if you wanted compare Actual Sales to Budgeted Sales in a bullet graph? In that case you need to set the Scenario dimension to be used as the utility dimension. When you set a dimension as the utility dimension you can then specify which member of the utility dimension to use for each measure in the view. For example, below is a bullet graph showing actual sales to budgeted sales by region.



You can see that the Sales measure is used twice in the view: once to show actual and once to show budgeted.

To use a dimension as the utility dimension:

1. Right-click the dimension in the Data pane and select **Set as utility dimension**.



The dimension in the Data pane (in this example, Scenario) can no longer be used as a dimension field in the view. The Measures area of the Data pane indicates that there is a utility dimension (in this example, (by Scenario) indicates the utility dimension).

Data Analytics

Basic (Sample)

Dimensions

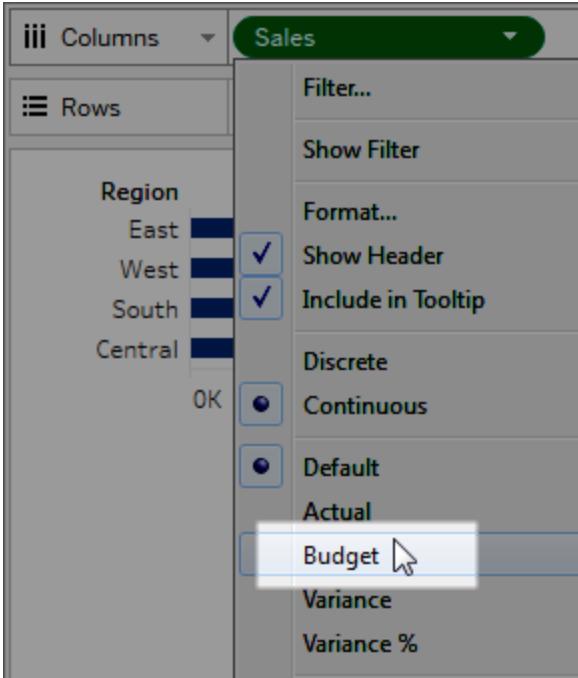
- Scenario
 - Gen2,Scenario !
- Market
 - Region
 - State
 - New Market
 - Major Market
 - Small Market
- Population
- Product

Measures (by Scenario)

- # Profit
- # Margin
 - # Sales
 - # COGS
- # Total Expenses
- # Inventory

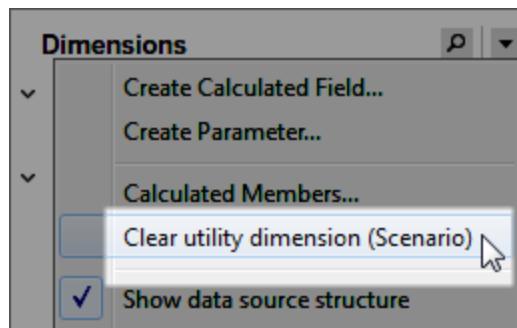
A screenshot of a data visualization interface. At the top, there are tabs for 'Data' and 'Analytics'. Below the tabs, a section labeled 'Basic (Sample)' contains a 'Dimensions' tree view. The tree includes categories like 'Scenario', 'Market' (with sub-options like Region, State, New Market, Major Market, Small Market), 'Population', and 'Product'. Under 'Measures (by Scenario)', there is a list of metrics: Profit, Margin (which further branches into Sales and COGS), Total Expenses, and Inventory. A red exclamation mark is visible next to the 'Gen2,Scenario' item under the Scenario dimension.

2. Drag a measure to the view.
3. Open the field menu for the measure in the view and select the member of the utility dimension you want to anchor the measure to.



To remove a utility dimension:

Open the drop-down menu at the top of the Data pane and select **Clear utility dimension**.



When you remove the utility dimension, measures that reference it in the view are no longer valid.

Pause Updating to Work with Large Views

If you are building a view that involves a large amount of data, you might find it easier to turn off automatic updates, build the view, and then resume updates. Turning off automatic updates queues the queries instead of sending a separate query to the database every time you drag a field.

If you are using filters in your views, you might also want to pause automatic updates for filters. This enables you to pause the query for filter changes until you are finished organizing your data. Otherwise every time you make a change to a filter, Tableau queries the data source, potentially slowing down performance.

For more information, see [Turn off Automatic Updates to Boost Performance](#) on page 1832.

Follow the steps below to pause automatic updates while building the view.

1. Click the **Pause Auto Updates** button  on the toolbar to turn off automatic updates for worksheets.
2. Place all of the selected fields on the shelves.
3. Specify filters to restrict the data to view only the values that you are interested in.
If you also want to pause automatic updates for filters, click the drop-down arrow on the **Pause Auto Updates** button on the toolbar and select **Auto Update Filters** to clear the check mark.

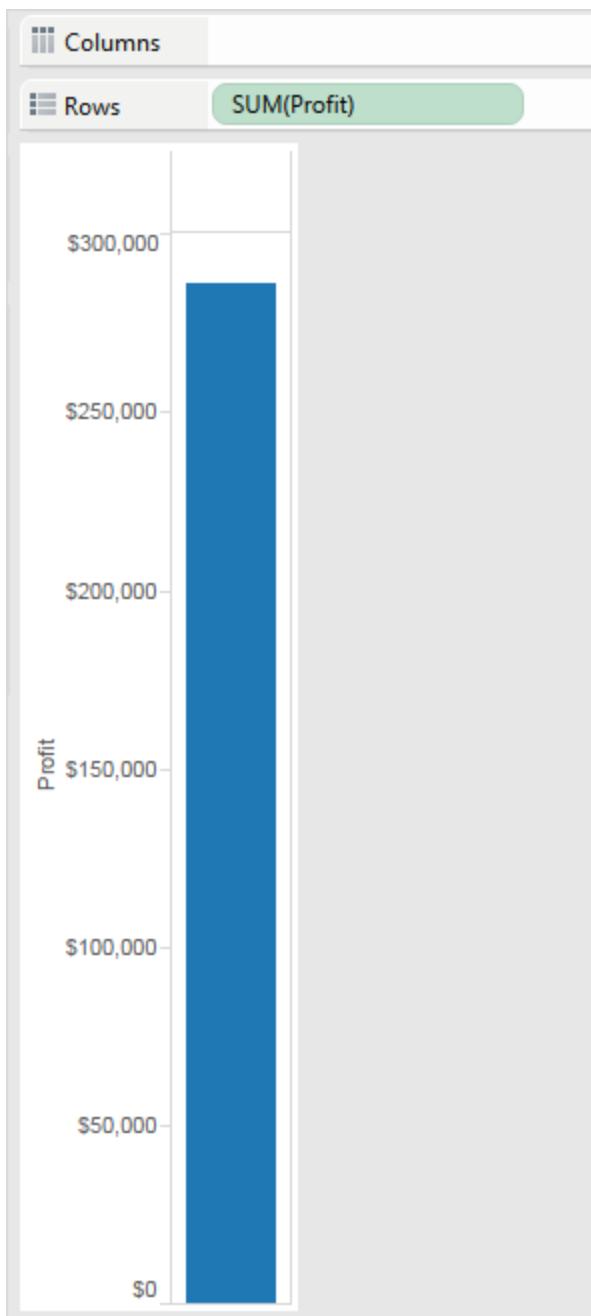
4. Click the **Resume Auto Updates** button  on the toolbar to turn automatic updates back on for worksheets.

To resume automatic updates for filters, click the drop-down arrow on the **Pause Auto Updates** button on the toolbar and select **Auto Update Filters** again.

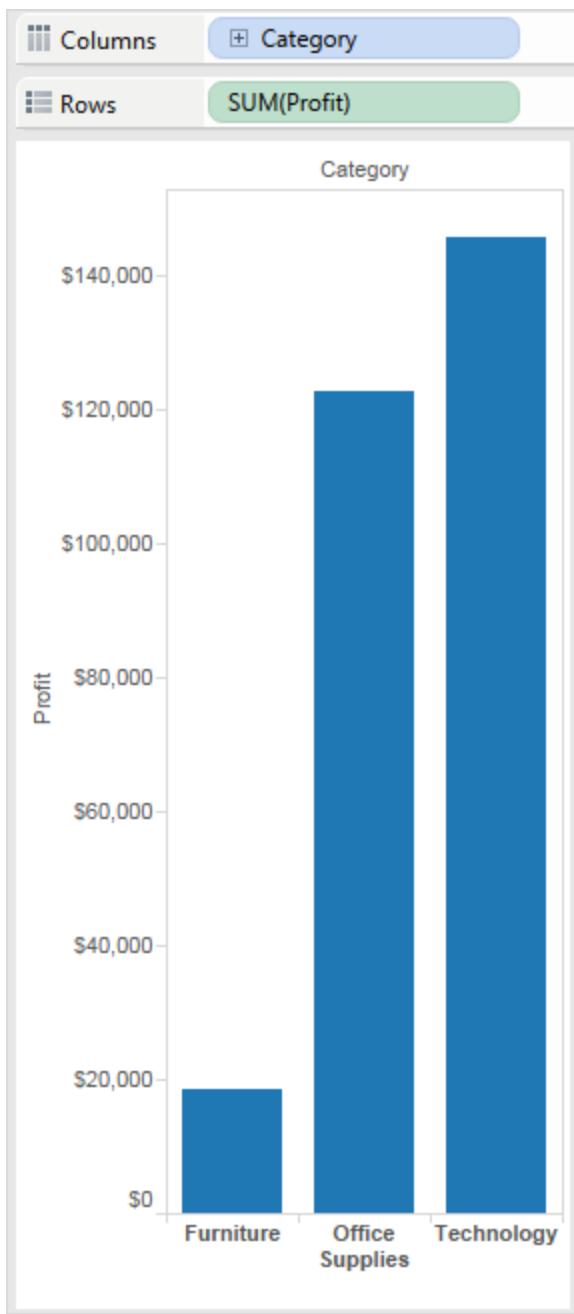
Add to Sheet: Double-Click

Tableau supports an additional method for automatically generating views of data called Automatic Double-Click. To use this method, double-click fields in the Data pane you are interested in. Tableau automatically adds each field to the view. That is, each double-click results in an additional field added to a shelf in an intelligent way. Like Show Me!, this function leverages Tableau's ability to make an intelligent "best guess" of how the data should be displayed. Double-clicking sometimes creates a view that you can use without further modification. More often, it can provide a starting point, which you can then modify to get the ideal result.

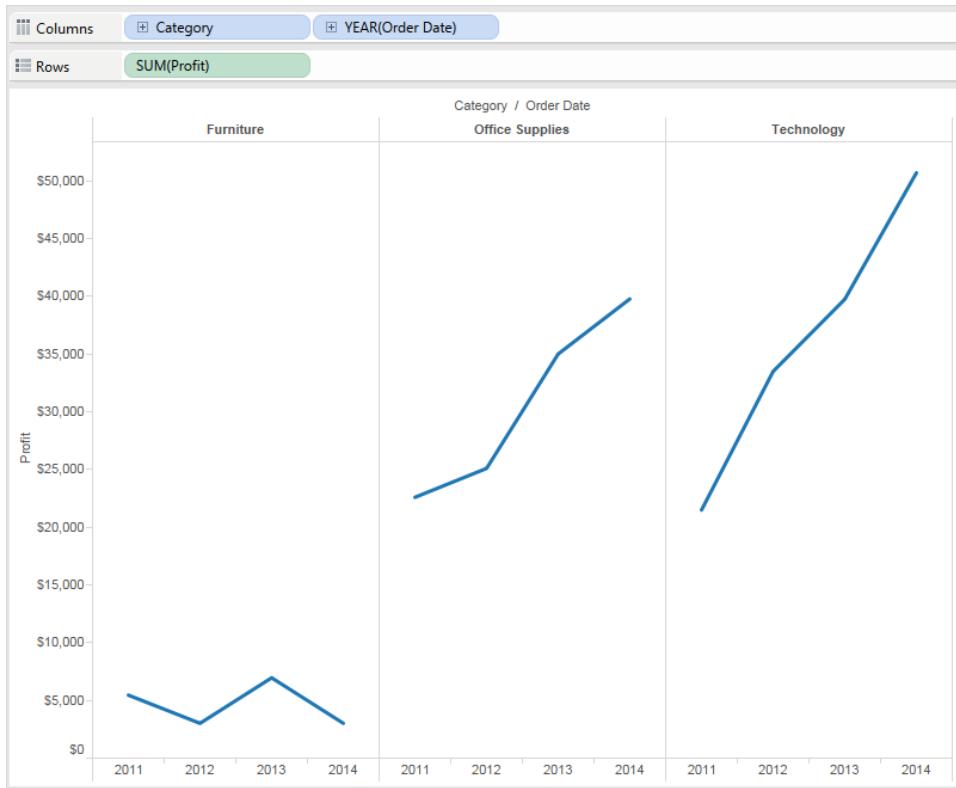
1. Double-clicking the **Profit** measure in the **Data** pane automatically adds that field to the view in an intelligent way.



2. Double-clicking the **Category** dimension in the **Data** pane automatically adds that field to the view based on the fact that Profit is already on the Rows shelf.



3. Double-clicking the **Order Date** dimension in the **Data** pane automatically adds that field to the **Columns** shelf based on the placement of other fields in the view. As you double-click fields they are successively added to the view. The order in which you click fields determines the type of view created.



The following table describes some of the rules used in creating automatic views by double-clicking fields in the **Data** pane.

Text Table	Adding a dimension first produces a text table (or cross-tab). All subsequent clicks on fields result in refinement of the text table.
Bars	Adding a measure first and then a dimension produces a bar view. All subsequent clicks result in refinement of the bar view, unless a date dimension is added, at which time the view is changed to a line.
Line	Adding a measure and then a date dimension produces a line view. All subsequent clicks result in refinement of the line view.
Continuous Line	Adding a continuous dimension and then a measure produces a continuous line view. Subsequent dimensions result in refinement of the continuous line view. Subsequent measures add quantitative axes to the view.
Scatter	Adding a measure and then another measure produces a scatter view. Subsequent dimensions result in refinement to the scatter view. Subsequent measures will create a scatter matrix.
Maps	Adding a geographic field produces a map view with latitude and longitude as axes and the geographic field on the Level of Detail shelf. Subsequent dimen-

sions add rows to the view while subsequent measures further refine the map by adding size and color encoding.

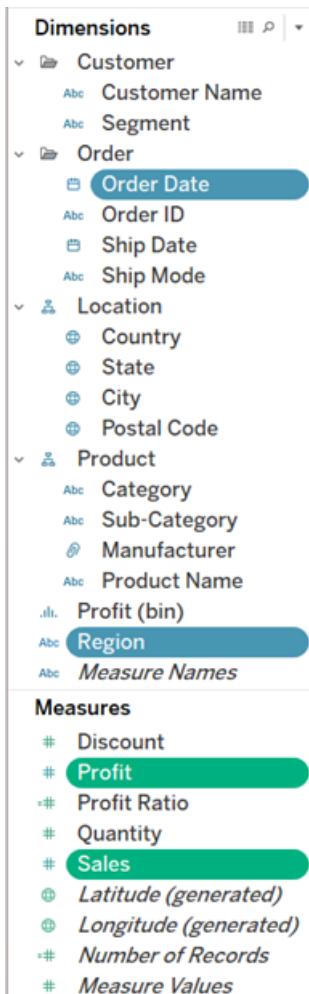
Show Me

Show Me creates a view based on the fields already used in the view and any fields you've

selected in the Data pane. Open Show Me by clicking **Show Me** on the toolbar .

When you use Show Me simply select fields you want to analyze in the Data pane and then select the type of view you want to create. Tableau automatically evaluates the selected fields and gives you the option of several types of views that would be appropriate for those fields. Further, Show Me highlights the visualization type that best matches the data.

1. Select fields in the **Data** pane that you want to analyze. Hold the Ctrl key (Command key on a Mac) to make multiple selections.



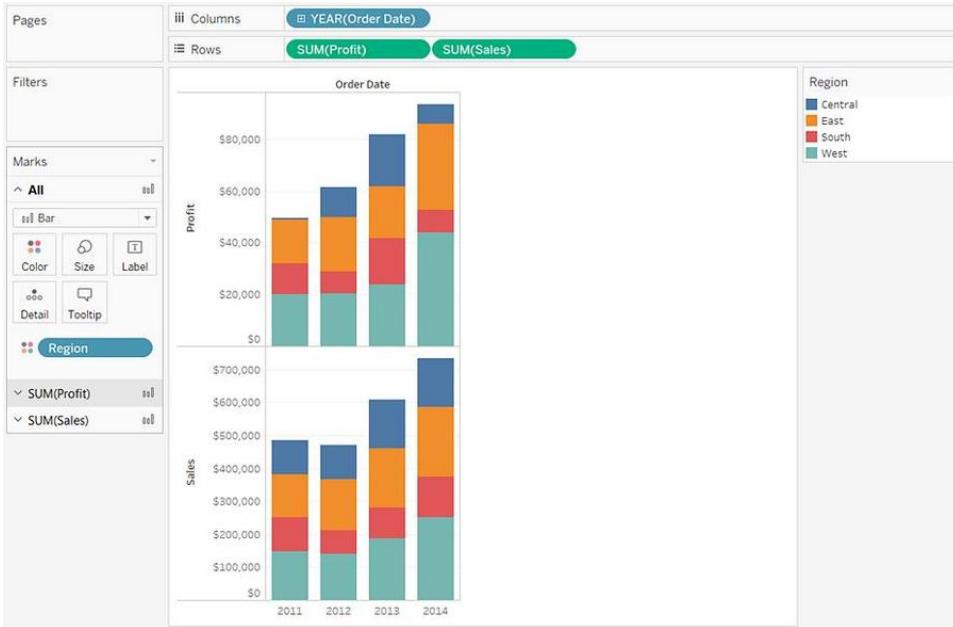
The screenshot shows the Tableau Data pane with two sections: Dimensions and Measures. In the Dimensions section, 'Order Date' and 'Region' are selected, highlighted with blue rounded rectangles. In the Measures section, 'Profit' and 'Sales' are selected, also highlighted with blue rounded rectangles. Other visible fields include Customer Name, Segment, Order ID, Ship Date, Ship Mode, Country, State, City, Postal Code, Category, Sub-Category, Manufacturer, Product Name, Profit (bin), and various generated measures like Latitude (generated) and Longitude (generated).

2. Click **Show Me** on the toolbar and then select the type of view you want to create.



Any view type that is not gray will generate a view of your data. As you hover over each view type, the description at the bottom shows the minimum requirements. In this example, Stacked Bars is selected.

3. View the Result. Tableau automatically creates a view of the data.



Create Bins from a Continuous Measure

Sometimes it's useful to convert a continuous measure (or a numeric dimension) into bins.

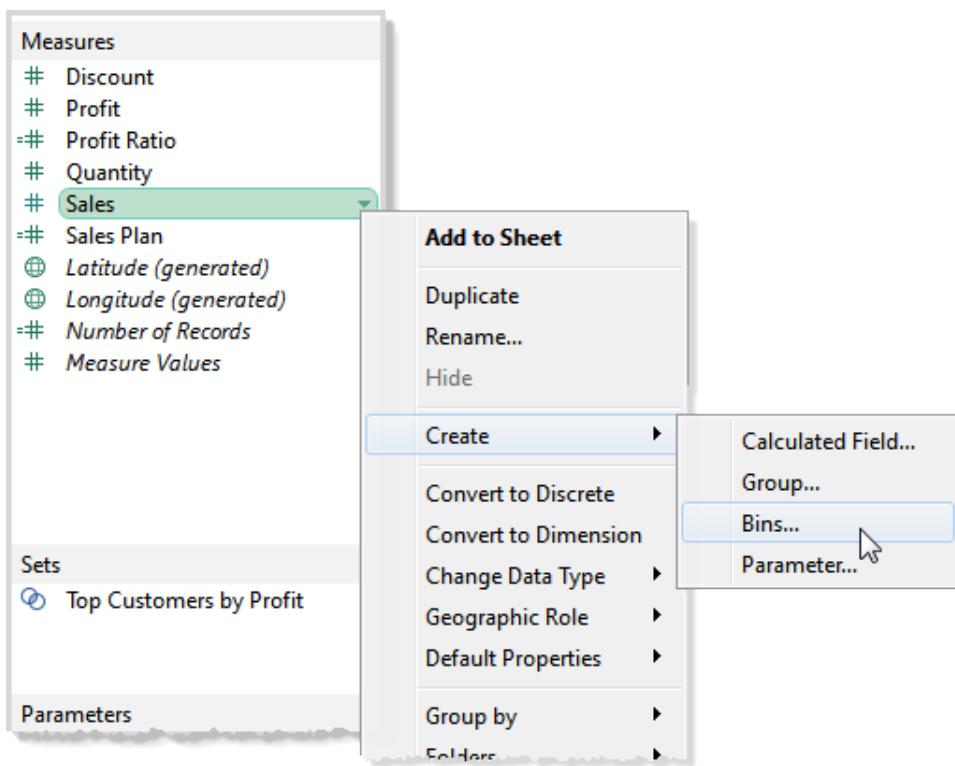
For example, suppose you create a view with **Profit** on **Rows** and **State** on **Columns**. You could consider the **State** field as a set of bins—each profit value is sorted into a bin corresponding to the state from which the value was recorded. But if you want to bin **Profit** values without reference to a dimension, you can create a numeric bin, with each bin corresponding to a range of values so that each profit value is sorted into a bin.

Note: You can bin data only for relational data sources. This feature is not supported for cube (multidimensional) data sources.

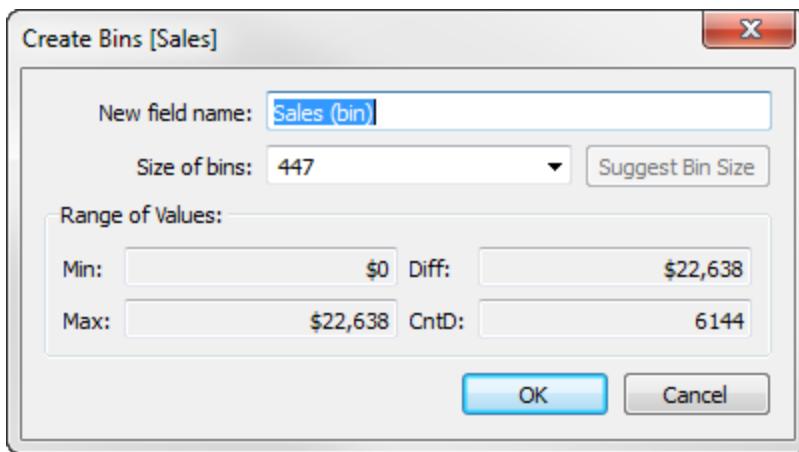
When you bin a measure you create a new dimension. That's because you are creating a field with a limited and discrete set of possible values out of a field with an unlimited, continuous range of values. However, once the dimension is created you can convert it to a continuous dimension. This can be useful if you want to create a histogram. See [Create a Histogram from a Binned Dimension on page 600](#).

Create a Binned Dimension:

1. In the **Data** pane, right-click (control-click on Mac) a measure and select **Create > Bins**.



2. In the Create Bins dialog box, accept the proposed New field name or specify a different name for the new field.



3. Either enter a value in the **Size of bins** field or have Tableau calculate a value for you.

If Tableau can perform the optimizing calculation quickly enough (in less than 1.5 seconds), the value you see initially in **Size of bins** is Tableau's estimate of the optimal bin size.

If Tableau cannot perform the optimizing calculation quickly, the **Size of bins** field defaults to 10. In this case you can click **Suggest Bin Size** to have Tableau perform the optimizing calculation.

The formula that Tableau uses to calculate an optimal bin size is

$$\text{Number of Bins} = 3 + \log_2(n) * \log(n)$$

In the formula, n is the number of distinct rows in the table. The size of each bin is determined by dividing the difference between the smallest and the largest values by the number of bins.

The four read-only fields in the lower part of the Create Bins dialog box show you the data that Tableau uses to suggest a bin size. You can also consider these values if you want to set a bin size manually. The values are:

Min	The field's minimum value.
Max	The field's maximum value.
Diff	The difference between the field's minimum and maximum values.
CntD	The number of distinct values (rows) in the data.

After you click **OK** to dismiss the Create Bins dialog box, a new binned field appears in the **Dimensions** area of the **Data** pane.

The screenshot shows the Tableau Data pane. At the top, there are tabs for 'Data' and 'Analytics'. Below the tabs, it says 'Sample - Superstore'. Under the 'Dimensions' section, there are several categories: 'Customer' (with 'Customer Name' and 'Segment'), 'Order' (with 'Order Date', 'Order ID', 'Ship Date', and 'Ship Mode'), 'Location', 'Product', 'Profit (bin)', 'Region', and 'Sales (bin)'. The 'Sales (bin)' dimension is highlighted with a red box. At the bottom of the list, there is a link 'Measure Names'.

When you add a binned dimension to the view, each bin acts as an equal-sized container that summarizes data for a specific range of values. Column or row headers are created, where each bin label designates the lower limit of the range of numbers that is assigned to the bin. Note that the lower limit is inclusive.

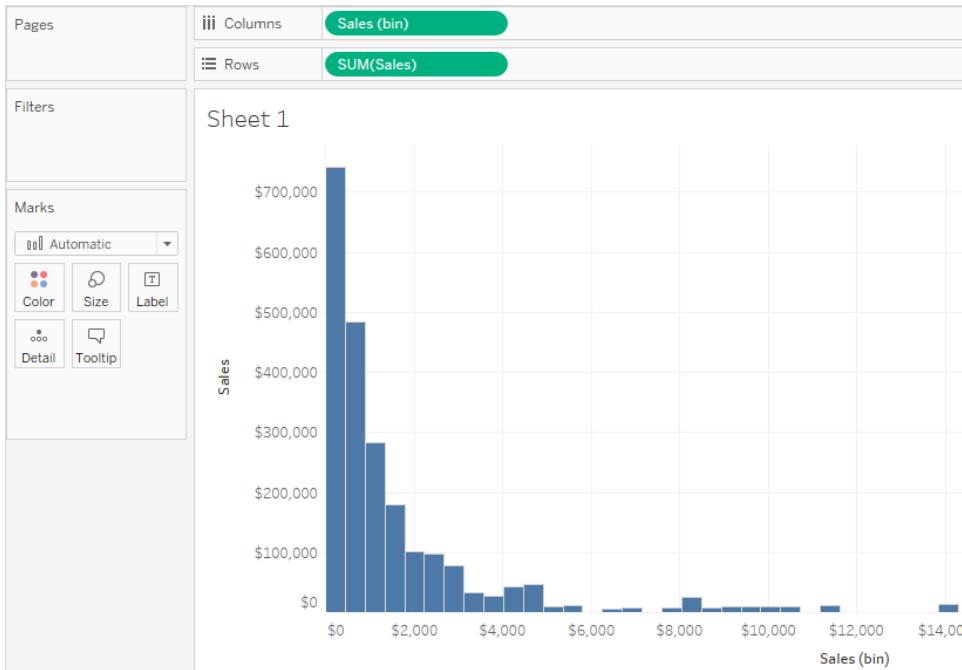
Create a Histogram from a Binned Dimension

If you create a binned dimension, you can use it as the starting point for creating a histogram. Using the **Sales (bin)** dimension created according to the instructions above, use the following steps to create a histogram.

Note: A quicker way to create a histogram is using Show Me. See [Build a Histogram on page 89](#) in the Build-It-Yourself Exercises section for information on creating a histogram using Show Me.

1. Click the **Sales (bin)** dimension in the Data pane and choose **Convert to continuous**.
2. Drag the **Sales (bin)** dimension from the Data pane and drop it on the **Columns** shelf.
3. Drag the original **Sales** field from the Measures area of the Data pane and drop it on the **Rows** shelf.
4. Click **SUM(Sales)** on **Rows** and change the aggregation from Sum to Count.

The result is a histogram:



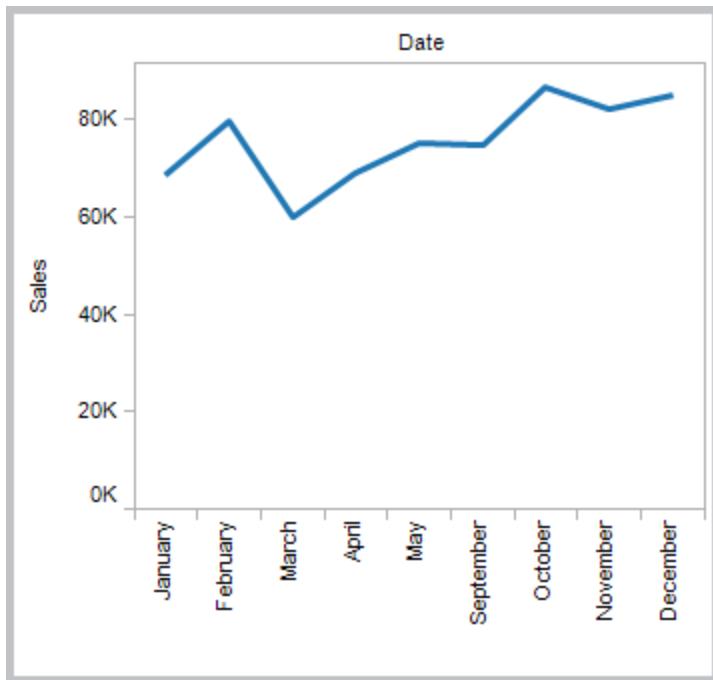
Missing Values

When you're working with dates or numeric bins, Tableau only shows the values that are represented in your data. If your data does not contain the complete range of values, the missing values will not be shown. For example, your data may contain data for January through May and September through December. However, there was no data recorded for June, July, and August. If you create a line chart in Tableau, the missing months will not be shown. You can optionally show the missing months to make it clear that there was no data recorded during that time.

Note: You can replace missing values with a calculated field using the ZN() function. See [Number Functions](#) on page 1552 for details.

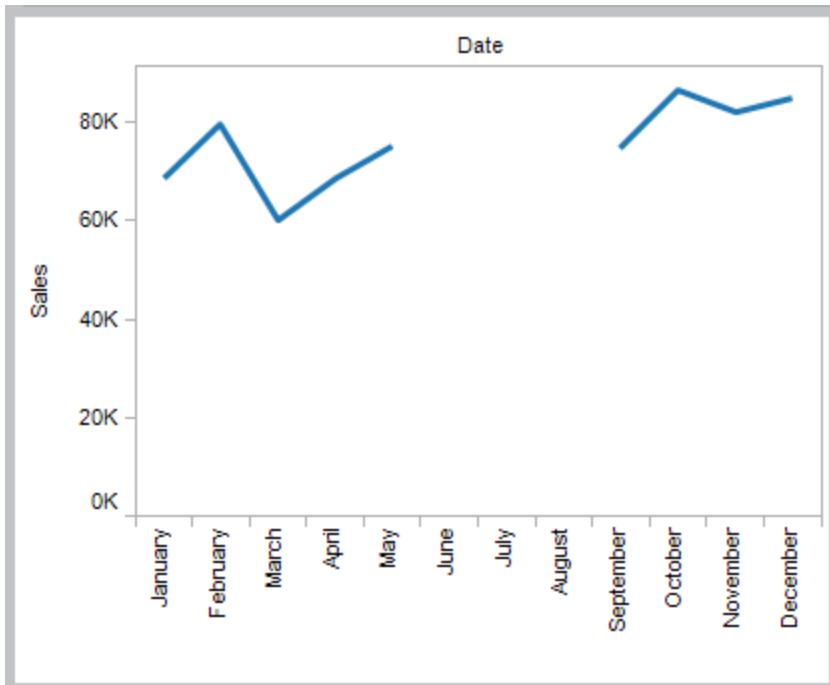
Missing Values Hidden (Default)

By default, missing values in a date range or numeric bins are not shown.

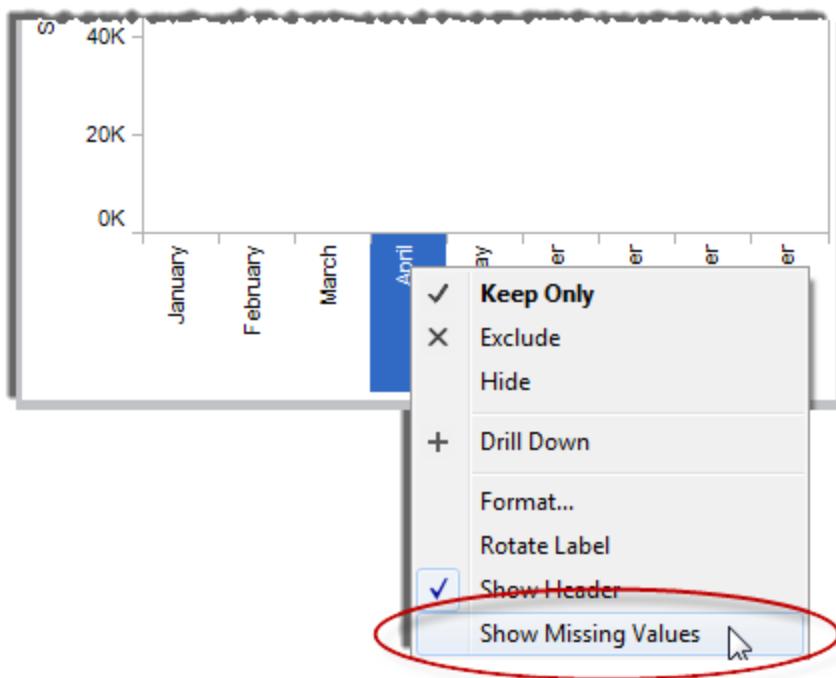


Missing Values Shown

You can show the missing values to indicate incomplete data.



To show missing values in a range, right-click (control-click on Mac) the date or bin headers and select **Show Missing Values**.



Show and Hide Empty Rows and Columns

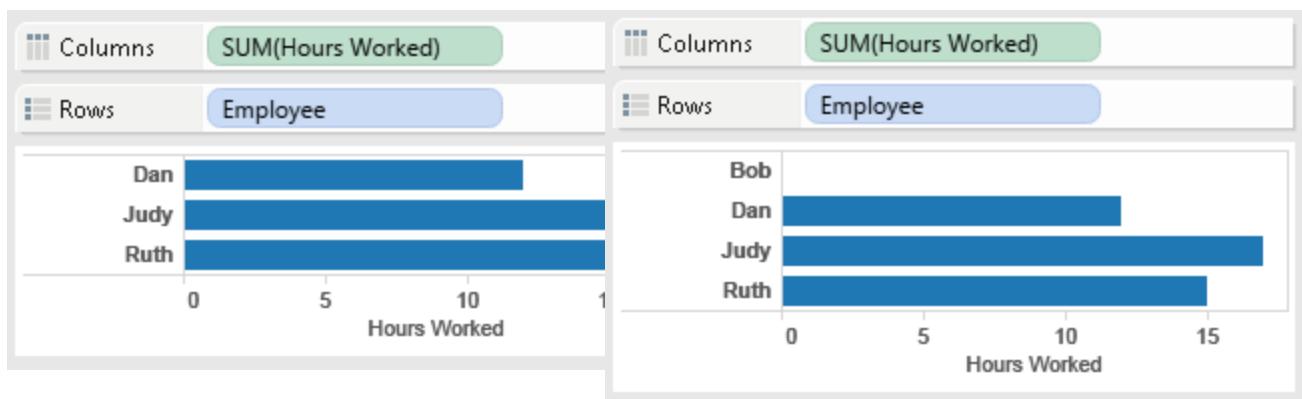
When you are working with fields that are not dates or numeric bins, Tableau hides missing values by default. For example, you may be showing workers and the number of hours worked in a particular month. If a worker didn't work at all that month, there may not be a row in the database for that worker. When you drag the Worker field to the Rows shelf, the workers that didn't work are hidden by default. You can show the empty rows by selecting **Analysis > Table Layout > Show Empty Rows**. Similarly, show the empty columns by selecting **Analysis > Table Layout > Show Empty Columns**.

Empty Rows Hidden (default)

Bob did not work in January so there are no records in the database for him. By default he is not listed.

Empty Rows Shown

Even though Bob did not work in January, he is listed but no marks are drawn.

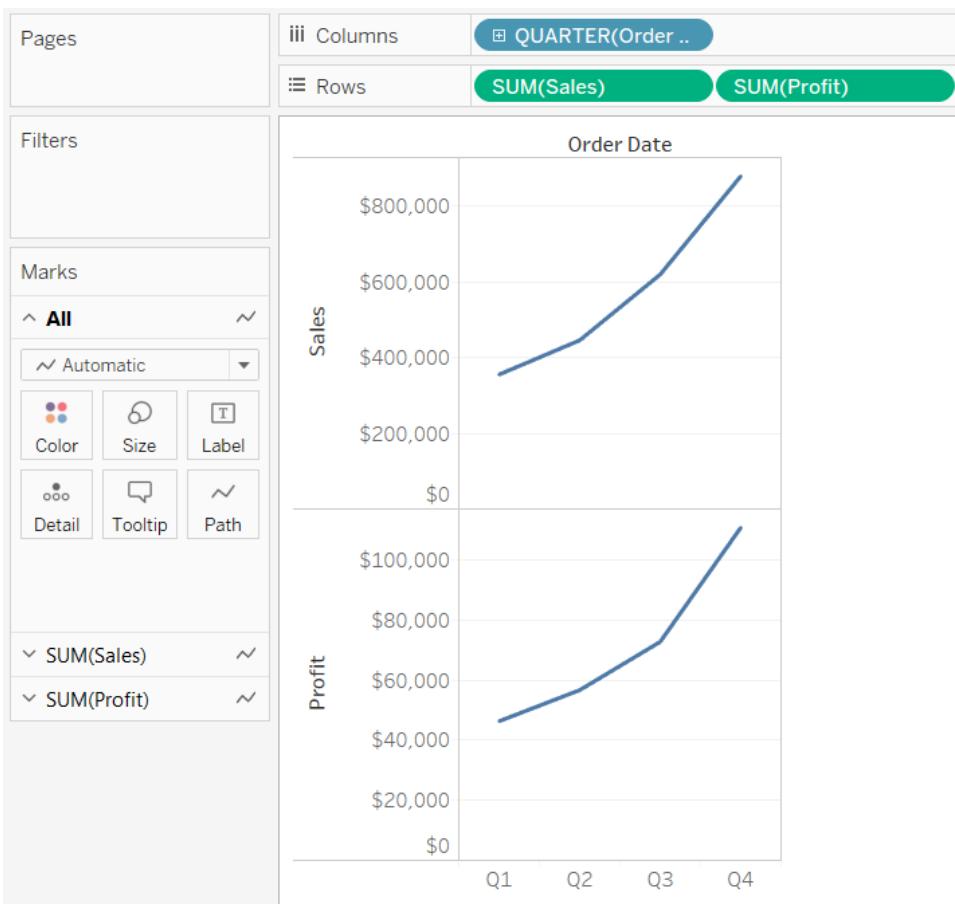


Using Multiple Measures

There are lots of different ways to compare multiple measures in a single view. For example, you can create individual axes for each measure or you can blend the two measures to share an axis and finally, you can add dual axes where there are two independent axes layered in the same pane. In any of these cases you can customize the marks for each axis to use multiple mark types and add different levels of detail. Views that have customized marks are called combination charts.

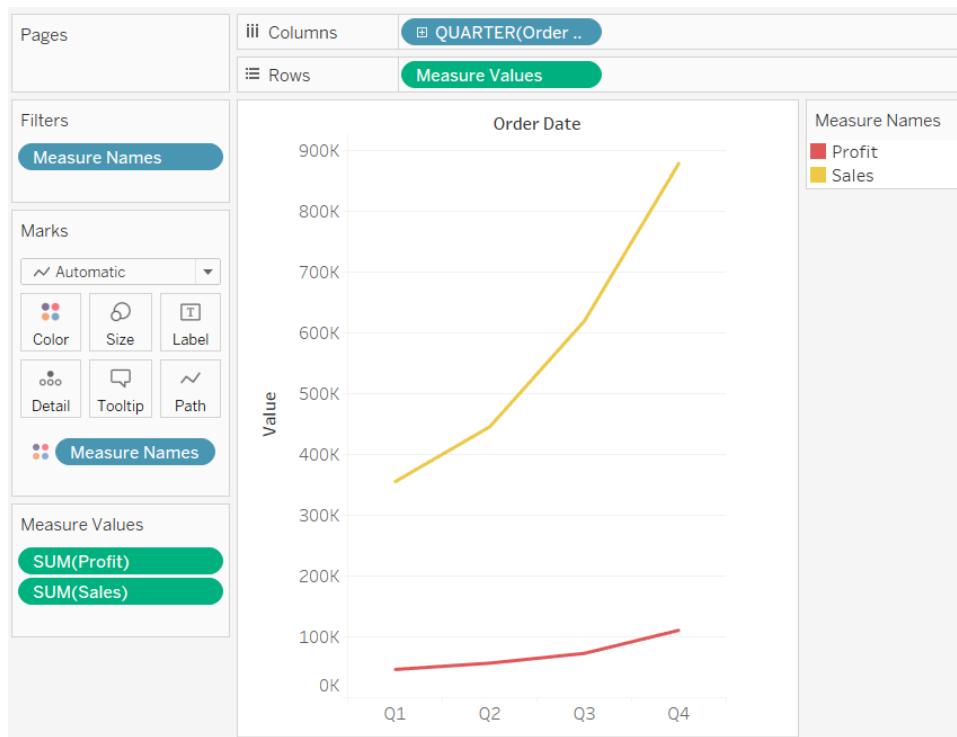
Individual Axes

Add individual axes for each measure by dragging measures to the **Rows** and **Columns** shelves. Each measure on the Rows shelf adds an additional axis to the rows of the table. Each measure on the Columns shelf adds an additional axis to the columns of the table. For example, the view below shows quarterly sales and profit. The Sales and Profit axes are individual rows in the table and have independent scales.



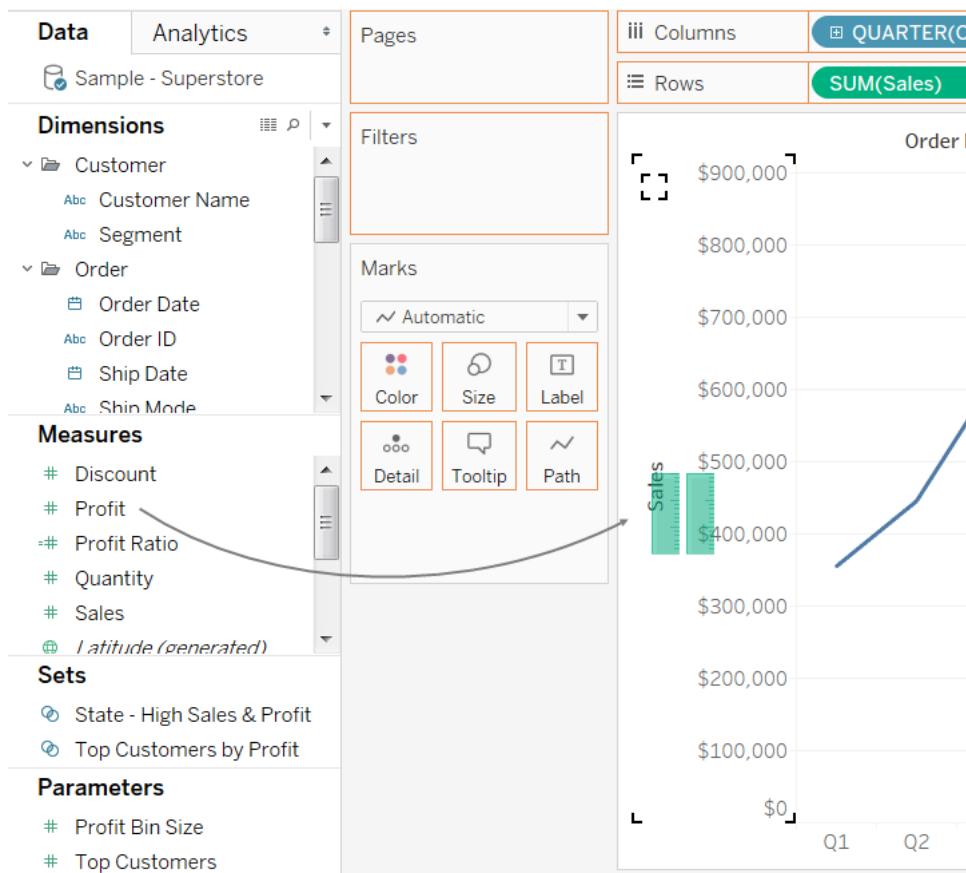
Blended Axes

Measures can share a single axis so that all the marks are shown in a single pane. Instead of adding rows and columns to the view, when you blend measures there is a single row or column and all of the values for each measure is shown along one continuous axis. For example, the view below shows quarterly sales and profit on a shared axis.



To create this view and blend multiple measures, simply drag one measure or axis and drop it onto an existing axis.

Note: If you drag a measure on to the canvas and only see a single ruler indicator instead of the double ruler indicator shown below, Tableau creates dual axes instead of a blended axis. For more information about how to create dual axes, see [Dual Axes on the next page](#).



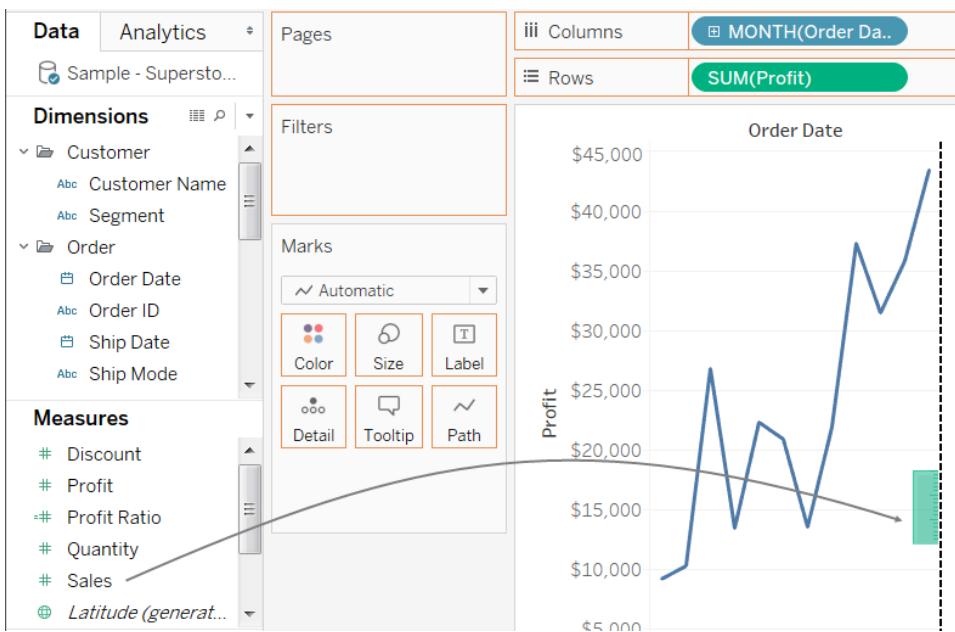
Blending measures uses the **Measure Names** and **Measure Values** fields, which are generated fields that contain all of the measure names in your data source and all of the measure values. The shared axis is created using the **Measure Values** field. The **Measure Names** field is added to **Color** on the Marks card so that a line is drawn for each measure. Finally, the **Measure Names** field is filtered to only include the measures you want to blend.

Note: Blending axes is most appropriate when comparing measures that have a similar scale and units. If the scales of the two measures are drastically different, the trends may be distorted.

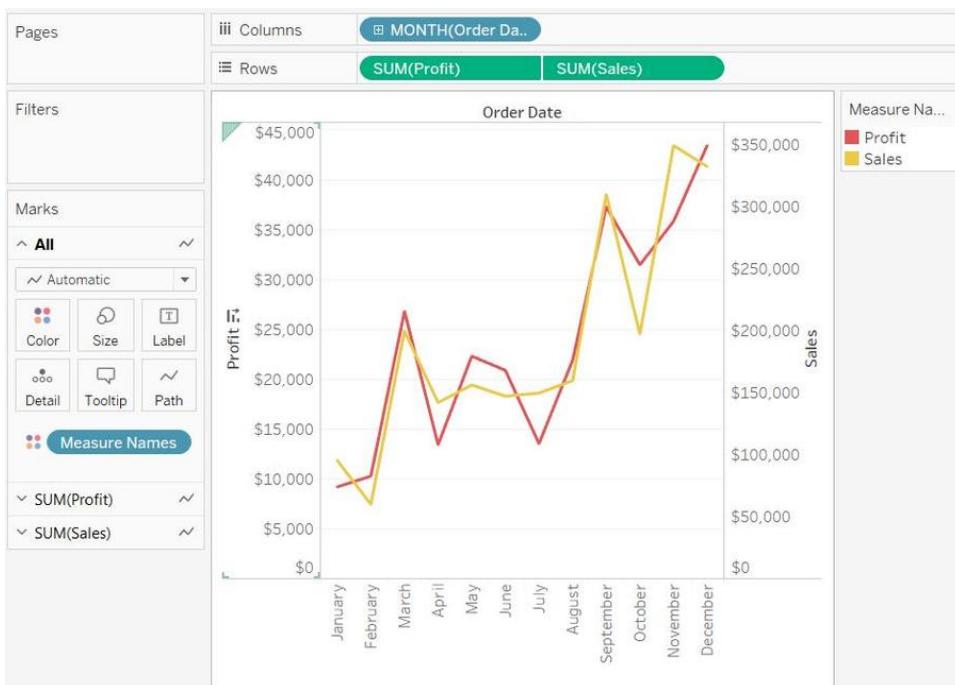
Dual Axes

You can compare multiple measures using dual axes, which are two independent axes that are layered on top of each other.

Dual axes are useful when you have two measures that have different scales. To add a measure as a dual axis, drag the field to the right side of the view and drop it when you see a black dashed line appear. You can also right-click (control-click on Mac) the measure on the Columns or Rows shelf and select **Dual Axis**.



The result is a dual axis view where the Profit axis corresponds to the red line and the Sales axis corresponds to the yellow line.



You can add up to four layered axes: two on the Columns shelf and two on the Rows shelf.

Synchronize axes

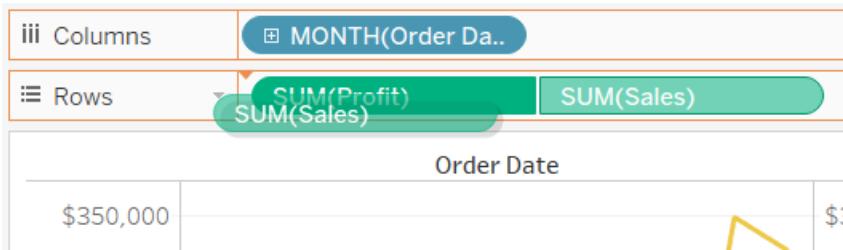
To align the two axes in a dual axis to have the same scale, right-click (control-click on Mac) the secondary axis, and select **Synchronize Axis**. This aligns the scale of the secondary axis to the scale of the primary axis.

In this example, the Sales axis is the secondary axis and the Profit axis is the primary axis.

Note: To synchronize axes, the data types for both measures must be the same. If the data types for your measures are different, see the section below.

If you would like to change which axis is the primary, and which axis is the secondary, select the field on the Columns or Rows shelf that is the secondary, and drag it in front of the primary field on the shelf until you see an orange triangle appear.

In this example, you can select the **SUM(Sales)** field on the **Rows** shelf, and drag it in front of the **SUM(Profit)** field. The Sales axis is now the primary and the Profit axis is the secondary.



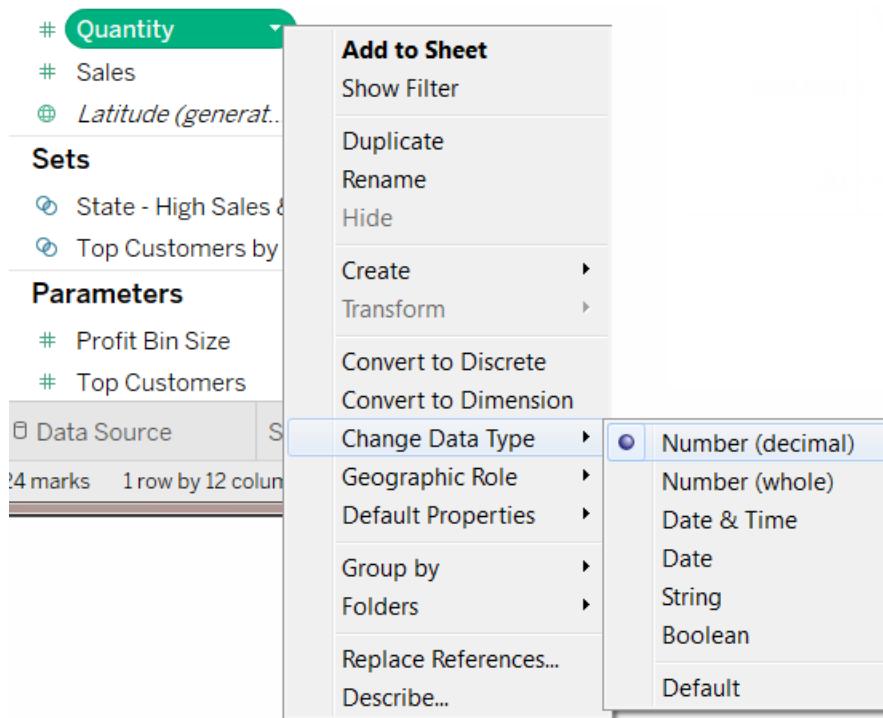
Synchronizing axes with measures that are different data types

The **Synchronize Axis** option ensures that you make a scaled and correct comparison. However, sometimes this option may not be available (grayed out). This is because the data type of one of the axes is different from the other.

To resolve this issue, you must change the data type of one of the axes. Follow the example below to change the data type for an axis.

1. Click the new Worksheet icon to open a new worksheet.
2. Drag **Order Date** to Columns, then click the drop-down arrow on the field on the shelf and select **Month** from the context menu.
3. Drag **Sales** to Rows, then drag **Quantity** to the right side of the view and drop it when you see a black dashed line appear to create a dual axis.
If you right-click on the **Quantity** Axis, you can see that the **Synchronize Axis** option is grayed out
4. In the Data pane, click on the drop-down arrow on the **Quantity** field, and select **Change**

Data Type > Number (decimal) in the context menu. This changes the data type for this field.

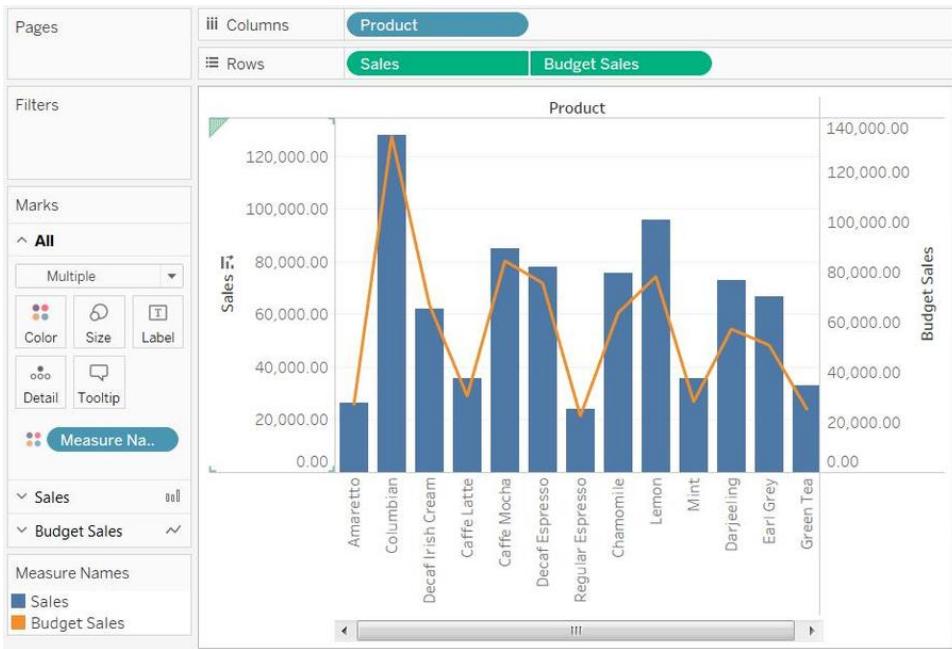


5. Replace the **Quantity** measure on the view.
6. Right-click the secondary axis, and then select **Synchronize Axis**.

Combination Charts

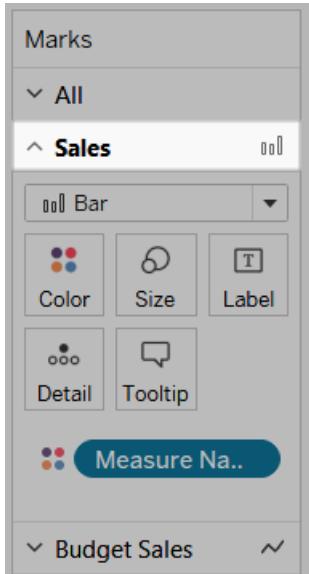
When working with multiple measures in a view, you can customize the mark type for each distinct measure. For example, you can create a view with a line showing a target amount across several months and a bar chart showing the actual attainment for the months. These measures can be displayed as individual axes, blended axes, or dual axes.

Because each measure can have customized marks, you can customize the level of detail, size, shape, and color encoding for each measure too.



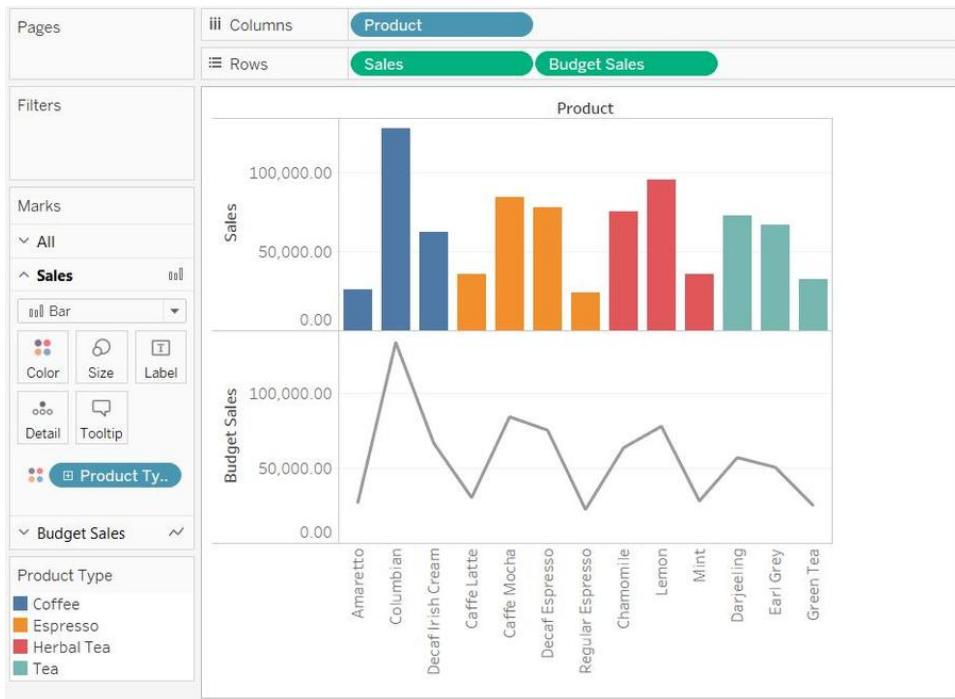
Customize the marks for a measure:

1. Select the Marks card for the measure that you want to customize. There is a Marks card for each measure on the Rows and Columns shelves.



2. Select a new mark type for the measure. Any changes to the mark type, shape, size, color, detail and other mark properties will be applied to the selected measure. For example, in the view below the **Sales** Marks card is active. The Mark Type has been changed to **Bar** and when **Product Type** is placed on **Color** on the **Sales** Marks card,

the encoding and level of detail is only applied to the Sales marks. The Budget Sales mark is not broken down by Product Type.



Select the **All** Marks card to modify properties for all measures at once.

Work with Time

Many visualizations you create with Tableau will involve working with date and time fields. With Tableau, you can identify trends in your data over time, or forecast where your data will go in the future.

Dates and Times

How you work with dates in Tableau depends on whether you are using a relational or cube (multidimensional) data source. This section discusses the differences.

Dates in Cube (Multidimensional) Data Sources

In Tableau, cube (multidimensional) data sources are supported only in Windows.

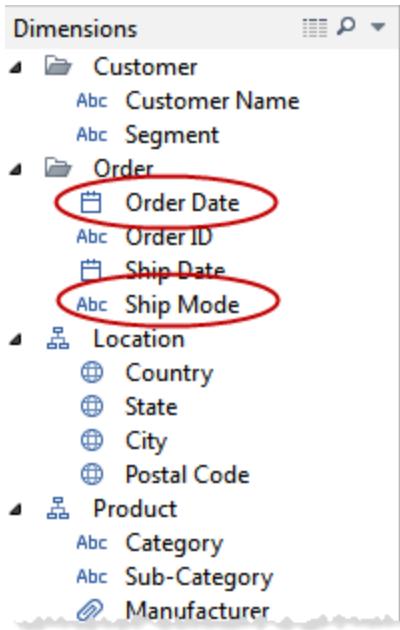
For cube data sources, date dimensions are usually organized into hierarchies that contain levels such as year, quarter, and month. In addition, some multidimensional data sources have time intelligence enabled, which makes it possible to look at data levels different ways, such as Months by Year, Months by Quarter, Weekends, etc. These levels are represented as attributes of the hierarchy. Hierarchies and attributes are defined when the cube is created and you cannot modify them in Tableau. For example, the **Year** dimension from an Oracle Essbase data source is shown below.



When you place a multidimensional date on a shelf, the field is treated like any other dimension. For example, you can drill down, drill up, and so on.

Dates in Relational Data Sources

For relational data sources, dates and times are automatically placed in the Dimensions area of the **Data** pane and are identified by the date or date-time icon. For example, the Order Date and Ship Date dimensions from an Excel data source are shown below.

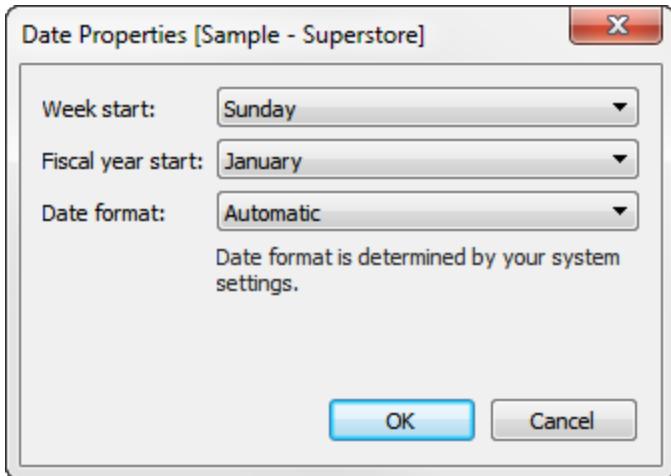


When you place a relational date on a shelf, the field name is automatically modified to reflect the default date level. Tableau defines the default date level to be the level at which there are multiple instances. For example, if the date field includes multiple years, the default level is year. However, if the date field contains data for just one year but includes multiple months, then the default level is month.

If you don't want Tableau to automatically select a date level and would rather have a date dimension be a continuous field, you can right-click (control-click on Mac) the field in the **Data** pane and select **Convert to Continuous**. The dimension then turns green in the **Data** pane; now when you use that dimension in a view, it will be continuous. You can easily revert back by selecting **Convert to Discrete** from the field's context menu in the **Data** pane. You can also convert a field in the view to continuous while it is on a shelf by selecting **Continuous** on its context menu (which you can see when you right-click (control-click on Mac) the field). The field on the shelf turns green but the field in the **Data** pane is still discrete.

Date Properties for a Data Source

You can set date properties for a data source. To do so, right-click (control-click on Mac) a data source in the **Data** pane and choose Date Properties:



The options in the Date Properties dialog box are:

- **Week start** - Specifies which day is considered the first day of the week. You can override the **Week start** value you set at the data source level by including a `start_of_week` value with certain date functions (`DATEDIFF`, `DATENAME`, `DATEPART`, and `DATETRUNC`). See [Date Functions on page 1565](#).

The initial value for this setting is in some cases determined by your data source. If the data source does not initialize the start-of-week day, then Tableau consults the system's locale settings. The day which is considered the first day of the week varies from region to region. For example, Sunday is the first day of the week in the US, while Monday is the first day in the EU.

This setting is disabled for multidimensional data sources because this information is defined when cube designer creates the date/time dimension.

After a data source is created, the **Week start** value does not automatically change if for any reason the default start-of-week setting changes in the host operating system (for example, if the data source is uploaded to Tableau Server in a different country).

Calendar controls reflect the workbook locale rather than the data source's **Week start** setting. The exception is that a week trunc start of week that shows the week number in it will use the data source's **Week start** setting in order to provide a consistent week number value in the calendar.

- **Fiscal year start** - Specifies which month is considered the first month of the fiscal year. To specify whether a date dimension uses the standard calendar (Jan. 1 - Dec. 31) or the fiscal calendar, right-click (control-click on Mac) the field in the Data pane and select [Default Properties > Calendar Type](#) and then select **Standard Calendar** or **Fiscal Calendar**. For any date field in the view, you can then specify whether to use the standard calendar or the fiscal calendar. For details, see [Fiscal Dates on page 623](#).
- **Date format** - Specifies the default format for data dimensions, as displayed, for example, in tooltips. To override the default date format for a date dimension, right-click

(control-click on Mac) the field in the Data pane and select **Default Properties > Date Format** and then select one of the available formats.

[Day of the week sorting](#)

If you are working in a language for which Tableau does not provide a local version, set your workbook locale to assure that Tableau can sort the days of the week in the correct chronological order. Otherwise, Tableau will sort the names of the days alphabetically. For information on setting the locale, see [Language and Locale on page 255](#).

If none of the supported locales is appropriate, you can sort the days of the week manually. See [Manual Sorting on page 495](#).

[Supported date formats](#)

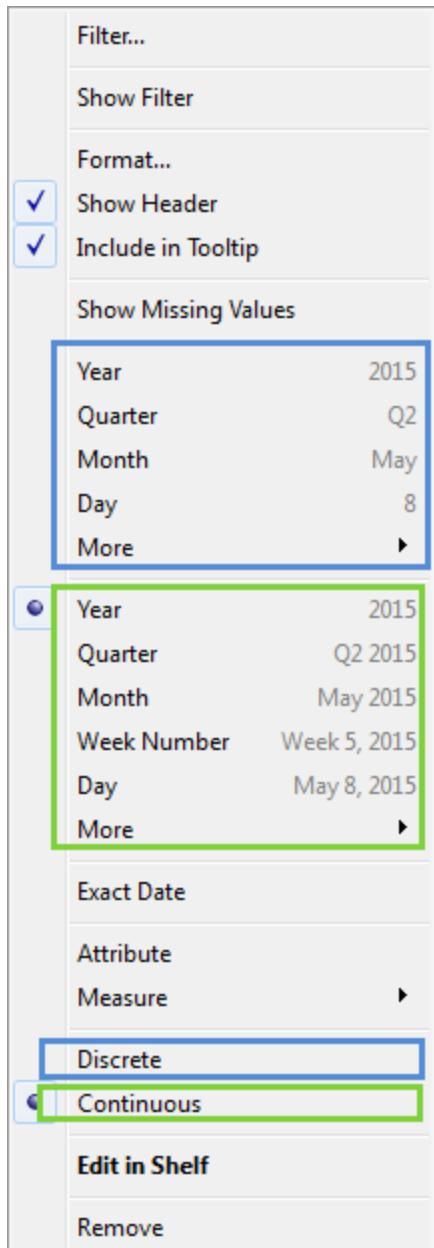
When working with dates, Tableau retrieves date formats automatically from the data source.

For a list of supported date formats, see the table of supported date format symbols in [Custom Date Formats on page 626](#).

Changing Date Levels

For both relational and multidimensional data sources, you can change the date level using the field's context menu after dragging it to a shelf. For cube (multidimensional) dates, the levels available in the context menu are given by the levels defined in the date hierarchy. For relational dates, you can select between the discrete date levels at the top of the menu or the continuous date at the bottom. A preview of each date level is shown. Below is the field menu for a relational date with the discrete Year level selected.

In Tableau, cube (multidimensional) data sources are supported only in Windows.



Click any of the options in the blue areas to configure the field as a discrete date.
Selecting one of these options creates what is known as a "date part."

Click any of the options in the green areas to configure the field as a continuous date.
Selecting one of these options creates what is known as a "truncated date."

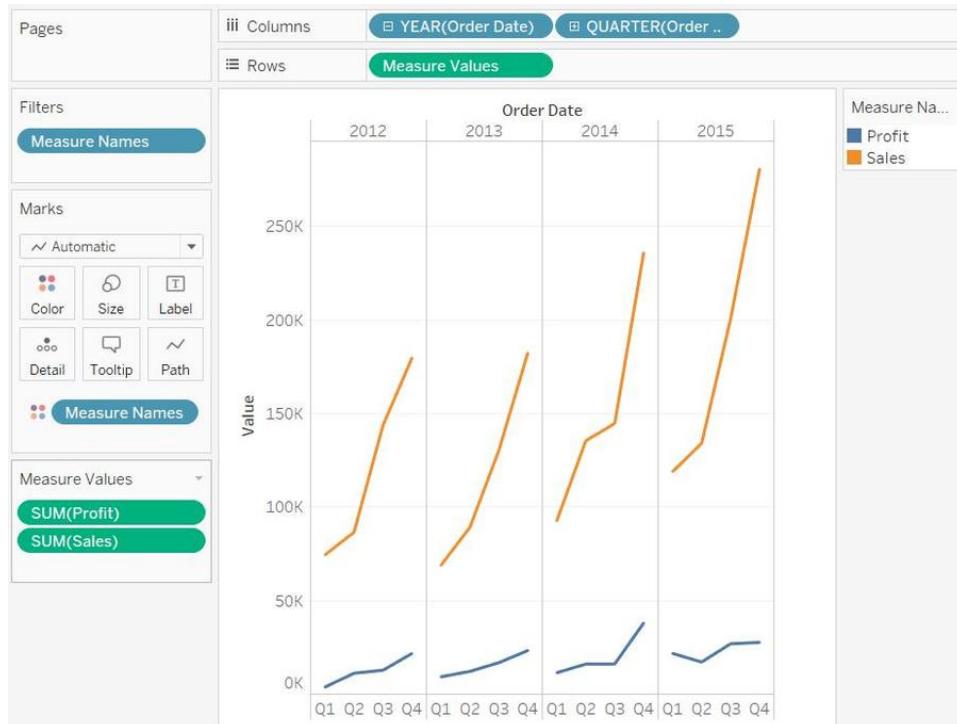
When you select a particular level, Tableau asks the data source to perform a computation on the date field. For example, suppose a particular row in your data source has a date entry of 01/23/16. The year is 2016, the quarter is 1 because January falls in the first quarter, and the week number is 4 because January 23rd falls in the fourth week. How the date level is computed depends on the configured Date Properties for the data source. See [Date Properties for a Data Source](#) on page 614.

Note: Some date levels might not make sense for your relational data source. For example, if the date format does not include time information such as hour, minute, or second, then selecting one of these options will not add any data to your view.

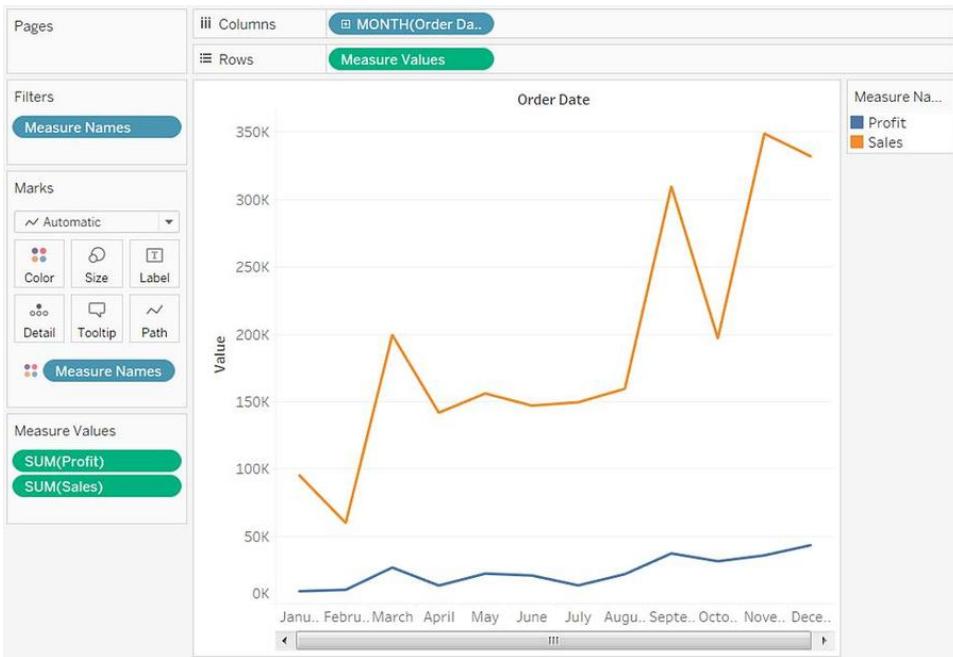
You can work with dates at varying levels of detail simultaneously. You do this by clicking the  control on a date field in the view. This is known as drilling into a date. For example, if you drill into a discrete (blue) date field **YEAR(Order Date)**, Tableau adds a second field to the right of the initial date, **QUARTER(Order Date)**, as in the image below. You can continue drilling down until you reach the finest available granularity for the date field.

You can also drag date fields to the **Rows** or **Columns** shelf multiple times in order to nest them and to drill down into them at varying levels of detail.

For example, the view shown below drills down from the year level to display the quarter level as well.



By removing one of the date fields from the Columns shelf, and then selecting **Month** from the remaining date field's context menu, you can display the data for each month across all years.



To display even finer granularity, select Month from the lower, continuous section of the field's context menu.

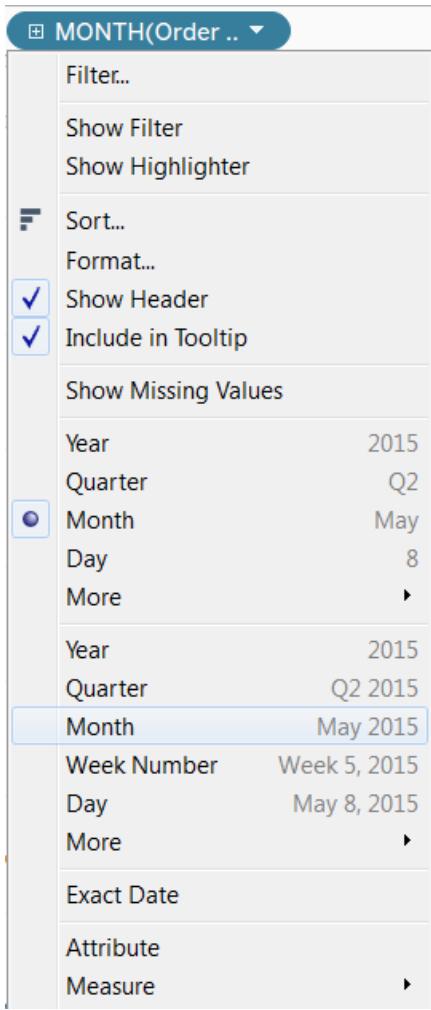
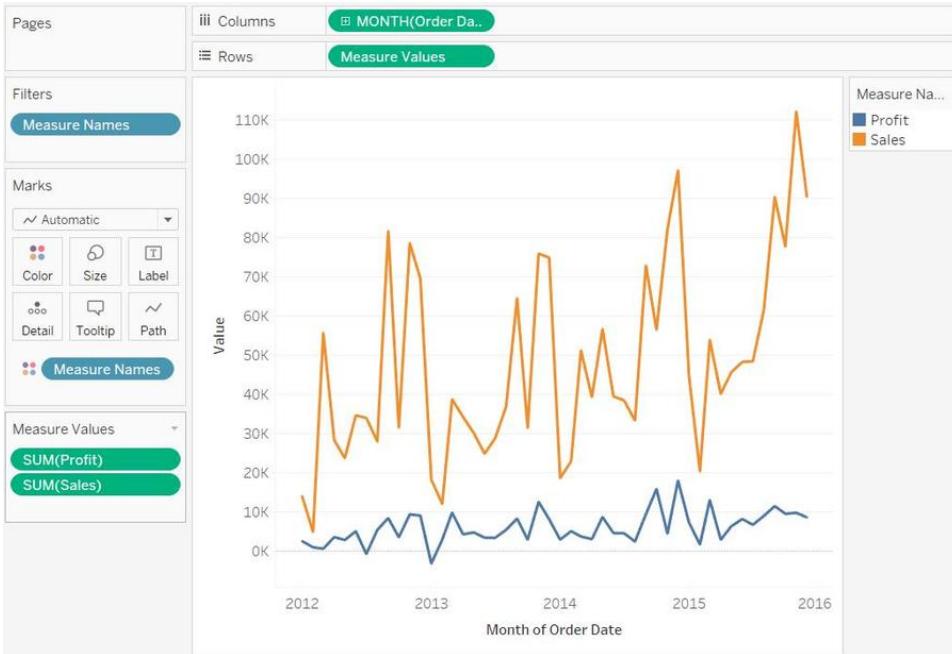


Tableau then displays the dates across the range of available years, at a month-by-month level of granularity.



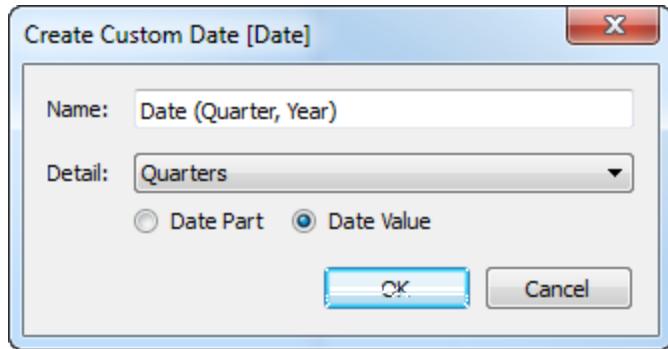
Custom Dates

You might want to create a custom date if you always use a date at a specific level, or if you want to create calculations that rely on a binned or truncated date.

You can create a custom date by working in the **Data** pane or by using the `DATEPART` and `DATETRUNC` functions in a calculated field.

The custom date you create becomes a new field in the **Data** pane.

1. Right-click (Control-click on Mac) a date field in the **Data** pane and select **Create > Create Custom Date**.
2. In the Create Custom Date dialog box, type a name for the custom date, such as Date (Quarter, Year).
3. From the **Detail** list, select the level you want to show the date at.
4. Specify whether you want the custom date to be discrete (select **Date Parts**) or continuous (select **Date Value**).



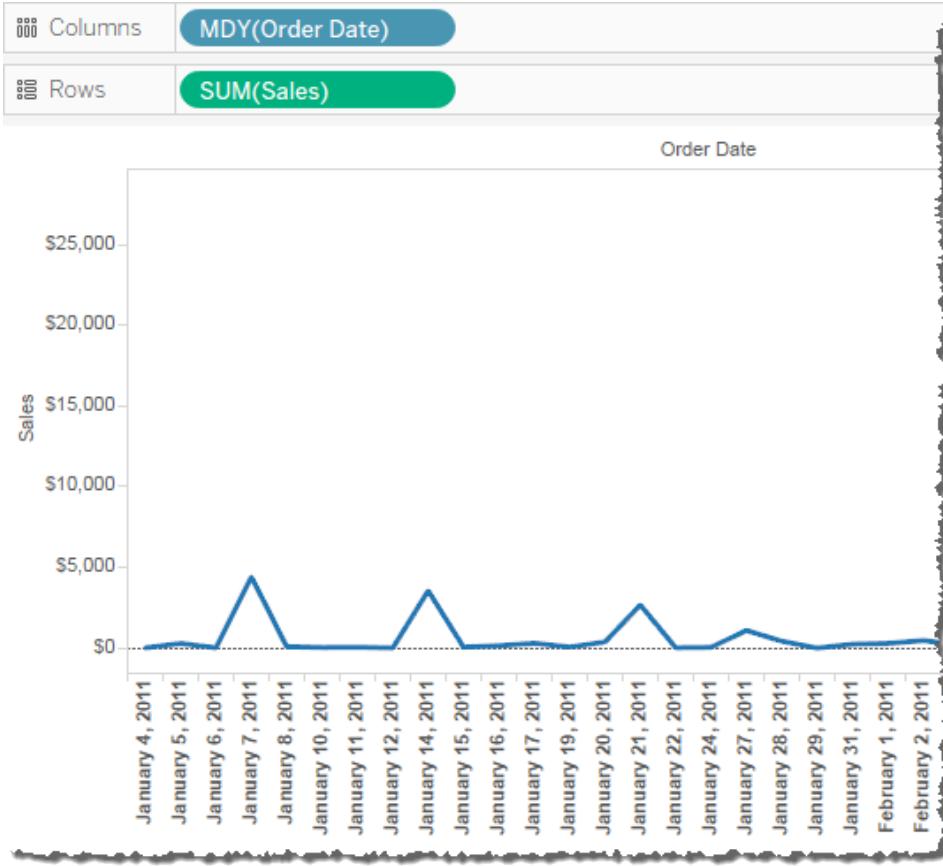
5. When you are finished, click **OK**.

Example - format column headers as Month, Day, Year

To format column headers as Month, Day, Year (for example, January 1, 2011), follow the steps below. This example uses the Sample - Superstore data source provided with Tableau Desktop.

1. Drag **[Order Date]** to **Columns**.
2. Drag **Sales** to **Rows**.
3. Right-click (Control-click on a mac) **[Order Date]** on the **Columns** shelf and choose **More > Custom**.
4. In the Custom Date dialog box, choose **Month / Day /Year** from the **Detail** drop-down list.

The view now has the column headers in the correct format:



Fiscal Dates

In some situations, a date field needs to be expressed in terms of an organization's fiscal year. Calendar years run from January 1st until December 31st, but an organization's fiscal year might start in a different month. For example, a company's fiscal year might run from June 1st through May 31st. In such cases, it's appropriate to display some date values in a view using fiscal equivalents (Fiscal Year, Fiscal Quarter, and Fiscal Week Number) rather than their calendar equivalents.

To set the fiscal year start month for a data source, follow these steps:

1. Right-click (Control-click on Mac) the data source in the **Data** pane to open the Date Properties dialog box.
2. Set the **Fiscal year start** field to the appropriate month.

This change sets the fiscal year for all date dimensions in the data source.

You can set the fiscal year start month for an individual date value to be different from the fiscal year start date for the data source. To do this, right-click (Control-click on Mac) the date dimension in the **Data** pane and select **Default Properties > Fiscal Year Start**.

Note: Date functions do not take account of the configured fiscal year start. See [Date Functions](#) on page 1565.

Whether a given level of a date dimension is affected by the use of the Fiscal Calendar depends on the specific case.

Date Level	When Converted to Fiscal
Year	Reflects the fiscal year. For example, if Fiscal year start is set to April, the year for the date June 1, 2004 would be shown as FY 2005.
Quarter	The Quarter reflects the fiscal quarter. For example, if Fiscal year start is set to April, the quarter for the date June 1, 2004 would be Q1.
Month	No change in behavior. The calendar month is the same as the fiscal month.
Day	No change in behavior. The calendar day is the same as the fiscal day.
Hour	No change in behavior. The calendar hour is the same as the fiscal hour.
Minute	No change in behavior. The calendar minute is the same as the fiscal minute.
Second	No change in behavior. The calendar second is the same as the fiscal second.
Week Number	The Week Number reflects the fiscal week number. For example, if Fiscal year start is set to April, the week number for the date April 1, 2004 would be 1.
Weekday	No change in behavior. The calendar weekday is the same as the fiscal weekday.
MM/YYYY	No change in behavior. This date format always displays calendar dates, even when a fiscal year has been assigned.
M/D/Y	This date format always displays Calendar dates, even when a fiscal year has been assigned.

The only date levels that explicitly indicate that the fiscal calendar is in use are the Year and Quarter level. Specifically, fiscal years and quarters are shown with the FY prefix. This is not true for fiscal quarters or week numbers, however, which are not shown with any special fiscal markings.

Fiscal year designations for any given date dimension are applied to all instances of the field in the Tableau workbook. Fiscal dates can only be applied to dimensions in a relational data source.

Fiscal year formatting is applied to all date formats that include a year, or a year and a quarter. In particular, if you apply a custom date format, and only use the “y” and “q” placeholders, then FY will be prepended to each year.

Date Truncations

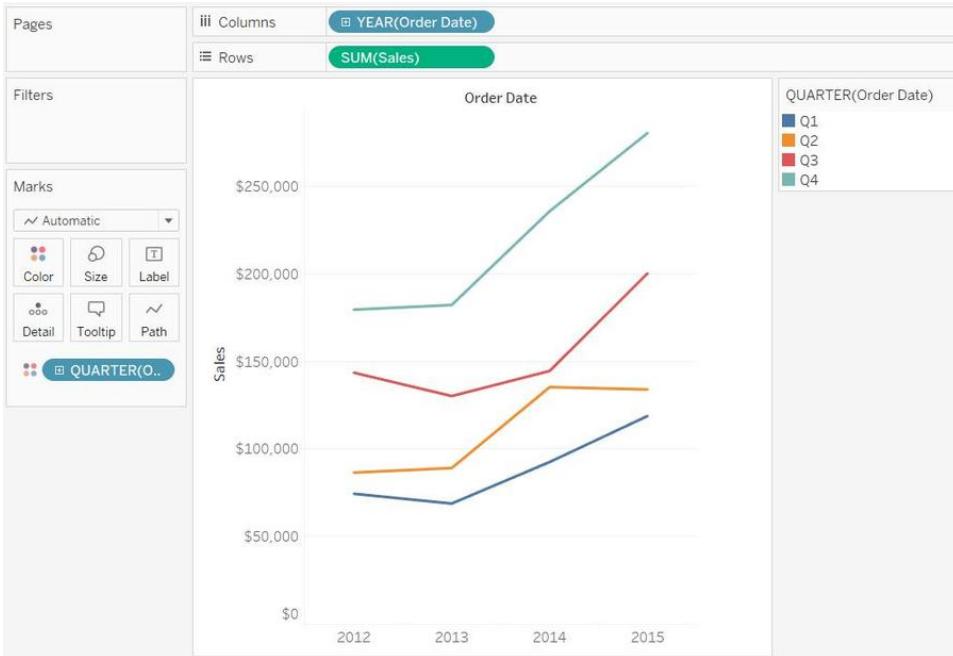
When a date dimension is using a fiscal calendar, only the following date parts and truncations will reflect the fiscal calendar:

Date part or truncation	When using a fiscal calendar
Year date part	Reflects the fiscal year. For example, if Fiscal year start is set to April, the year for the date June 1, 2004 would be shown as FY 2005.
Quarter date part	Reflects the fiscal quarter. For example, if Fiscal year start is set to April, the quarter for the date June 1, 2004 would be Q1.
Week Number date part	Reflects the fiscal week number. For example, if Fiscal year start is set to April, the week number for the date April 1, 2004 would be 1.
Year date truncation	Truncates to the start of the fiscal year. For example, if Fiscal year start is set to May, the date June 1, 2004 would become May 1, 2004.
Quarter date truncation	Truncates to the start of the fiscal quarter. For example, if Fiscal year start is set to July, the date June 1, 2004 would become April 1, 2004.

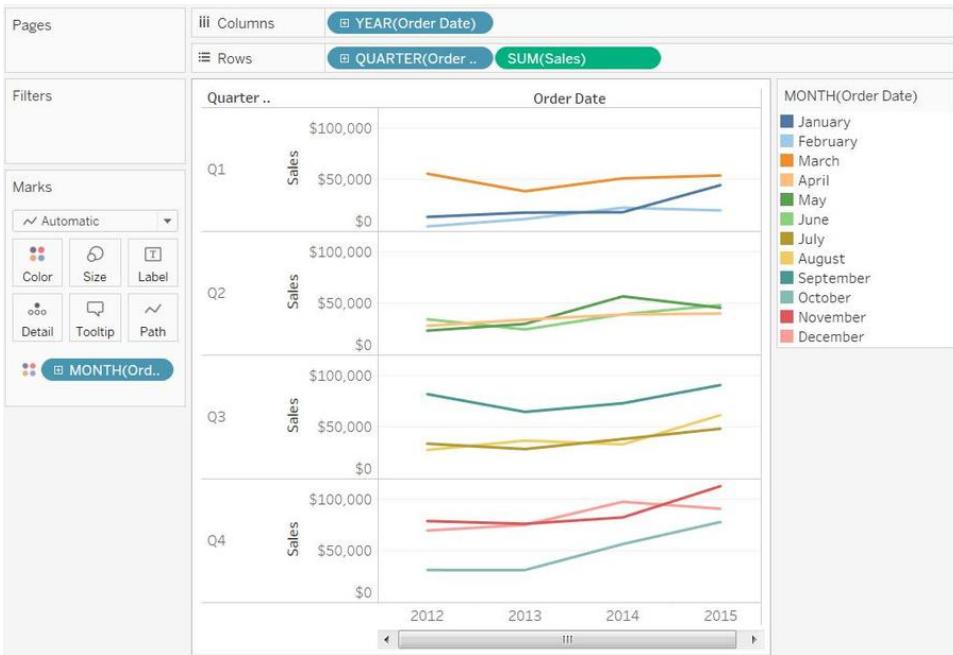
Perfect Pivoting with Dates

You can perfect pivot dates by placing different date levels on different worksheet shelves simultaneously. Place the date field on a variety of shelves and then select the desired date level from the fields' context menus.

For example, the following line chart displays years as column headers and then color-encodes the marks by quarter.



You can separate the marks by month and by quarter as shown below.



Custom Date Formats

There are different ways to format dates in Tableau.

- To format a date field in the view, right-click (Control-click on a Mac) the field and choose **Format**.
- To format a date field in the **Data** pane, right-click the field and choose **Default Properties > Date Format**.
- To format date properties for a data source, right-click a data source in the **Data** pane and choose **Date Properties**. Then in the Date Properties dialog box, choose **Date format**.

When you format dates, Tableau presents a list of available formats. In most cases, the last item in the list is **Custom**. You specify a custom date using a set of format symbols, which are listed in the table below.

The date formats in the table are supported when your workbook is connected to a Tableau data extract or has a live connection to a data source that also supports the date format. (Refer to your data source's documentation to verify that the date format you want is supported.)

Tableau retrieves date formats from the data source. Tableau Server can also retrieve date formats from the Run As user account on the server that is running Tableau Server.

Note: The following date formats might not be the same as those used with the **DATEPARSE()** function. See [Convert Strings to Date Fields](#) on page 418 for more information.

Supported Date Format Symbols

Use the following symbols to construct a custom date format.

Symbol	Description
(:)	Time separator. In some locales, a different character is used to represent the time separator. The time separator separates hours, minutes, and seconds when time values are formatted. The actual character used as the time separator in formatted output is determined by your system settings.
(/)	Date separator. In some locales, a different character is used to represent the date separator. The date separator separates the day, month, and year when date values are formatted. The actual character used as the date separator in formatted output is determined by your system settings.
c	Display the date as dddd and display the time as tttt, in that order. Display only date information if there is no fractional part to the date serial number; display only time information if there is no integer portion.
d	Display the day as a number without a leading zero (1 31).

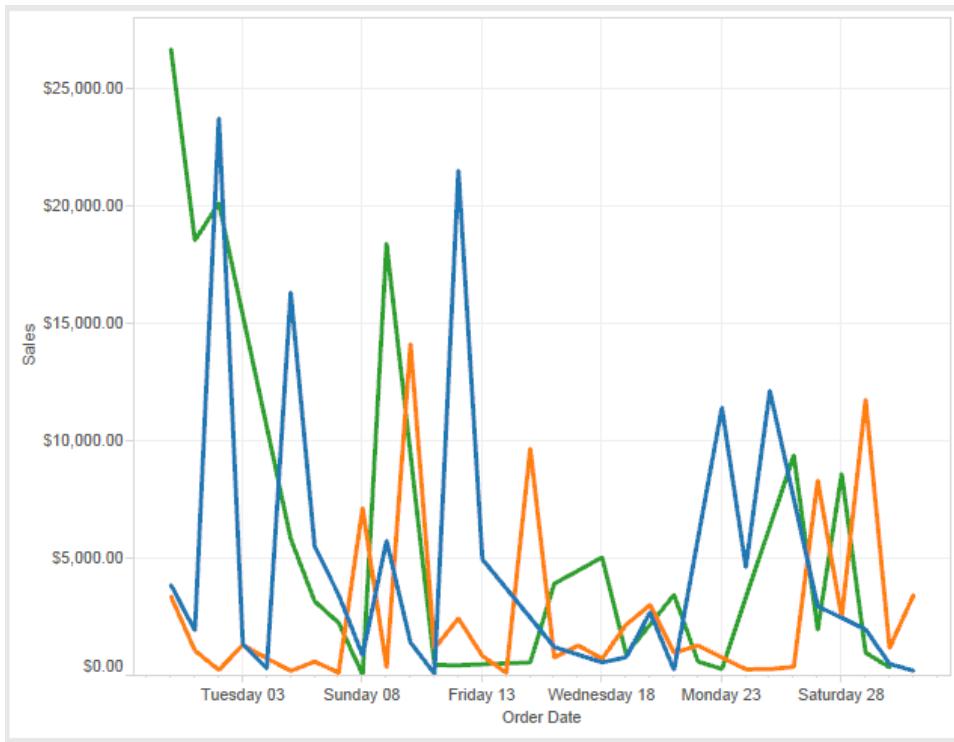
dd	Display the day as a number with a leading zero (01 31).
ddd	Display the day as an abbreviation (Sun Sat).
dddd	Display the day as a full name (Sunday Saturday).
ddddd	Display the date as a complete date (including day, month, and year), formatted according to your system's short date format setting. The default short date format is m/d/yy.
dddddd	Display a date serial number as a complete date (including day, month, and year) formatted according to the long date setting recognized by your system. The default long date format is mmmm dd, yyyy.
aaaa	The same as dddd, only it's the localized version of the string.
w	Display the day of the week as a number (1 for Sunday through 7 for Saturday).
ww	Display the week of the year as a number (1 54).
m	Display the month as a number without a leading zero (1 12). If m immediately follows h or hh, the minute rather than the month is displayed.
mm	Display the month as a number with a leading zero (01 12). If m immediately follows h or hh, the minute rather than the month is displayed.
mmm	Display the month as an abbreviation (Jan Dec).
mmmm	Display the month as a full month name (January December).
oooo	The same as mmmm, but localized.
q	Display the quarter of the year as a number (1 4).
y	Display the day of the year as a number (1 366).
yy	Display the year as a 2-digit number (00 99).
yyyy	Display the year as a 4-digit number (100 9999).
h	Display the hour as a number without leading zeros (0 23).
Hh	Display the hour as a number with leading zeros (00 23).
N	Display the minute as a number without leading zeros (0 59).
Nn	Display the minute as a number with leading zeros (00 59).
S	Display the second as a number without leading zeros (0 59).

Ss	Display the second as a number with leading zeros (00 59).
000	Display milliseconds. Use a period character as a separator before specifying milliseconds.
ttttt	Display a time as a complete time (including hour, minute, and second), formatted using the time separator defined by the time format recognized by your system. A leading zero is displayed if the leading zero option is selected and the time is before 10:00 A.M. or P.M. The default time format is h :mm :ss.
AM/PM	Use the 12-hour clock and display an uppercase AM with any hour before noon; display an uppercase PM with any hour between noon and 11:59 P.M.
am/pm	Use the 12-hour clock and display a lowercase AM with any hour before noon; display a lowercase PM with any hour between noon and 11:59 P.M.
A/P	Use the 12-hour clock and display an uppercase A with any hour before noon; display an uppercase P with any hour between noon and 11:59 P.M.
a/p	Use the 12-hour clock and display a lowercase A with any hour before noon; display a lowercase P with any hour between noon and 11:59 P.M.
AMPM	Use the 12-hour clock and display the AM string literal as defined by your system with any hour before noon; display the PM string literal as defined by your system with any hour between noon and 11:59 P.M. AMPM can be either uppercase or lowercase, but the case of the string displayed matches the string as defined by your system settings. The default format is AM/PM.

Custom Date Format Examples

Specifying a custom format yyyy-MM-dd HH:mm:ss.000 would produce dates in the format 2015-05-10 11:22:16.543. Such a format might be appropriate for scientific data.

Specifying a format DDDD DD would produce dates that show the Weekday and the Day, as shown below.



Support for Japanese era-based date formats

Starting with version 9.3, Tableau supports Japanese emperor-era-based date (Wareki) formats. Here's how to apply an era-based date format to a field in your view:

1. Set your workbook locale to Japanese.
2. Right-click the field in the view for which you want to set a date format.
3. Choose **Format**.
4. In the **Format** pane, from the **Dates** drop-down list, select a format.

If the format that you want isn't listed, you can construct your own date format. To do this, choose **Custom format** in the **Dates** box, then type your format using the Tableau date placeholders. The following era-based year placeholders are available:

Symbol	Description
g	Short era name (such as H for the Heisei era).
gg	Era name (such as 平成).
ggg	Long era name (for Japanese, this is currently the same as the regular era name).
e	Era-based year, such as 1 for the first year of an era.
ee	Era-based year, such as 01 for the first year of an era. If there is only one digit,

then the era-based year will have a zero added to the front.

If your workbook locale is not Japanese, you can create a custom date format, then insert the language code !ja_JP! in front of your format, so that it looks like this:

!ja_JP! gg ee"年"mm"月"dd"日"

The language code forces the date to be treated as if it is a Japanese date.

Era-based dates are not fully supported by the Tableau Server browser view. In particular, if you publish a workbook that contains a quick filter, the e and g placeholders will not be filled in:

Order Date gg ee年01月01日 gg ee年12月31日

To avoid this issue, do not show era-based dates in quick filters if your workbook will be viewed in a browser.

Using literal text in a date format

You may want your date format to include some words or phrases, such as **Fiscal Quarter q of yyyy**. However, if you type that text directly into the Tableau format box, it may treat the letters like date parts:

Quarter of Order Date
Fi01/1/2010al 1quarter 1 of 2010
Fi04/1/2010al 2quarter 2 of 2010
Fi07/1/2010al 3quarter 3 of 2010
Fi10/1/2010al 4quarter 4 of 2010

To prevent Tableau from doing this, put double quotes around the letters and words that should not be treated as date parts: “**Fiscal Quarter**” q “of” yyyy.

If you want a literal quote inside of a quoted section, insert this code: “\””. For example, the format “**Fiscal “\”” Quarter**” would be formatted as **Fiscal “ Quarter**.

Using Week-Based Placeholders in Custom Date Formats

For many European locales, Tableau allows you to format dates using ISO-8601 weeks and years.

Expand to see relevant locales

da-DK, Danish (Denmark)

se-SE, Sami, Northern (Sweden)

rm-CH, Romansh (Switzerland)

sms-FI, Sami, Skolt (Finland)

is-IS, Icelandic (Iceland)
de-DE, German (Germany)
km-KH, Khmer (Cambodia)
nl-NL, Dutch (Netherlands)
smj-SE, Sami, Lule (Sweden)
fi-FI, Finnish (Finland)
nn-NO, Norwegian Nynorsk (Norway)
it-IT, Italian (Italy)
it-CH, Italian (Switzerland)
nb-NO, Norwegian Bokmål (Norway)
pl-PL, Polish (Poland)
sv-SE, Swedish (Sweden)
et-EE, Estonian (Estonia)
lv-LV, Latvian (Latvia)
hsb-DE, Upper Sorbian (Germany)
kl-GL, Greenlandic (Greenland)
mk-MK, Macedonian (Macedonia, FYRO)
fo-FO, Faroese (Faroe Islands)
se-NO, Sami, Northern (Norway)
as-IN, Assamese (India)
fy-NL, Frisian (Netherlands)
se-FI, Sami, Northern (Finland)
ps-AF, Pashto (Afghanistan)
prs-AF, Dari (Afghanistan)
de-CH, German (Switzerland)
sv-FI, Swedish (Finland)
dsb-DE, Lower Sorbian (Germany)
de-LU, German (Luxembourg)
fr-CH, French (Switzerland)
smj-NO, Sami, Lule (Norway)
de-LI, German (Liechtenstein)

sma-NO, Sami, Southern (Norway)

sma-SE, Sami, Southern (Sweden)

smn-FI, Sami, Inari (Finland)

The week and year numbering in an ISO-8601 calendar is different from a standard Gregorian calendar. Here's how January 2nd, 2011 would be represented:

Calendar System	Week Number	Year Number
Standard Gregorian	1	2011
ISO-8601	52	2010

When formatting dates, it's important to ensure that your month, week and year numbers all come from the same calendar system. Otherwise, the date might not make sense. There are two ways to do this:

- Let Tableau guess which calendar system you want to use for each placeholder. For example, if you type `mm yyyy` (a month followed by a year), Tableau uses the standard Gregorian year. In this case, it makes sense to use that year with a month number.
- In your format string, place an annotation after each week (`ww`) or year (`yyyy`) placeholder, to indicate what kind of placeholder it should be:

Annotation	Calendar to Use	Example Format String
[Y]	ISO-8601	<code>ww[Y] yyyy[Y]</code>
[y]	Standard Gregorian	<code>ww[y] yyyy[y]</code>

Important: To get correct ISO-8601 week numbering, you must set your data source start of week to Monday. (For information on how to do this, see [Date Properties for a Data Source on page 614](#).) Otherwise, Tableau will number weeks using your data source start of week setting.

Note: ISO week numbering will not occur if you are using a fiscal year that doesn't start in January, regardless of the locale you choose.

Sample format strings for the date December 31, 2013

Here are sample format strings for the date December 31, 2013, where `m` stands for a month or day-of-month placeholder. Time placeholders, such as hours, minutes, and seconds, don't

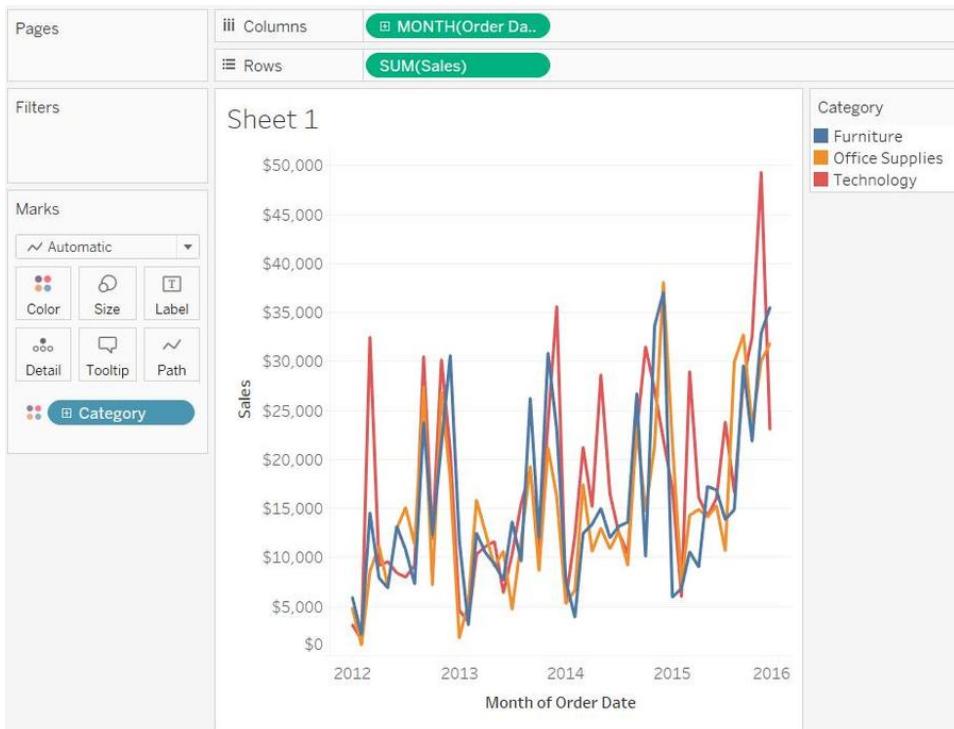
influence whether or not a year is week-based or year-based.

Format string	Formatted output
yyyy	2013
yyyy[Y]	2014
yyyy ww	2014 1
yyyy mm	2013 12
ww yyyy	1 2014
mm yyyy	12 2013
ww yyyy[y]	1 2013
mm yyyy[Y]	12 2014
yyyy mm ww	2013 12 1
mm yyyy ww	12 2013 1
ww mm yyyy	1 12 2013
ww mm yyyy[Y]	1 12 2014
ww yyyy mm	1 2014 12 (format is ambiguous)
ww yyyy[y] mm	1 2013 12
ww yyyy mm yyyy	1 2014 12 2013
ww yyyy yyyy mm	1 2014 2013 12
yyyy mm ww yyyy	2013 12 1 2014

Continuous Dates

You can treat a date as a continuous quantity after placing the field on a shelf. You do this by selecting one of the Continuous date options on the field's context menu (lower list of date levels). Continuous dates draw a quantitative axis for the date values.

For example, the view below displays the sales as a function of a continuous order date and is color-encoded by category. As you can see, the color of the **Order Date** field changes from blue to green after it is converted to a continuous quantity.



Treating dates as a continuous quantity is particularly useful when you use Gantt bars or want to see trends using line charts as shown above.

By default, date dimensions are discrete fields for which Tableau automatically selects a date level when it is placed on a shelf. To make a date dimension continuous by default, right-click (control-click on Mac) the field in the **Data** pane and select **Convert to Continuous**. The field turns green and is automatically converted to a continuous field when you drag it to a shelf. To revert to discrete again, right-click (control-click on Mac) the field in the **Data** pane and select **Convert to Discrete**.

Forecasting

You can forecast quantitative time-series data using exponential smoothing models in Tableau Desktop. With exponential smoothing, recent observations are given relatively more weight than older observations. These models capture the evolving trend or seasonality of your data and extrapolate them into the future. Forecasting is fully automatic, yet configurable. Many forecast results can become fields in your visualizations.

You can add a forecast to a view when there is at least one date dimension and one measure in the view. Choose **Analysis > Forecast > Show Forecast**. When no date dimension is present, you can add a forecast if there is a dimension field in the view that has integer values. See [Forecasting When No Date is in the View](#) on page 645.

When a forecast is showing, future values for the measure are shown next to the actual values.

Forecasting Constraints

Forecasting is not supported for Multidimensional data sources. In Tableau, multidimensional data sources are supported only in Windows.

In addition, the view cannot contain any of the following:

- Table calculations
- Disaggregated measures
- Percent calculations
- Grand Totals or Subtotals
- Date values with aggregation set to Exact Date

How Forecasting Works in Tableau

Forecasting in Tableau uses a technique known as exponential smoothing. Forecast algorithms try to find a regular pattern in measures that can be continued into the future.

You typically add a forecast to a view that contains a date field and at least one measure. However, in the absence of a date, Tableau can create a forecast for a view that contains a dimension with integer values in addition to at least one measure. For details on forecasting using an integer dimension, see [Forecasting When No Date is in the View on page 645](#).

All forecast algorithms are simple models of a real-world data generating process (DGP). For a high quality forecast, a simple pattern in the DGP must match the pattern described by the model reasonably well. Quality metrics measure how well the model matches the DGP. If the quality is low, the precision measured by the confidence bands is not important because it measures the precision of an inaccurate estimate.

Tableau automatically selects the best of up to eight models, the best being the one that generates the highest quality forecast. The smoothing parameters of each model are optimized before Tableau assesses forecast quality. The optimization method is global. Therefore, choosing locally optimal smoothing parameters that are not also globally optimal is not impossible. However, initial value parameters are selected according to best practices but are not further optimized. So it is possible for initial value parameters to be less than optimal. The eight models available in Tableau are among those described at the following location on the OTexts web site: [A taxonomy of exponential smoothing methods](#).

When there is not enough data in the visualization, Tableau automatically tries to forecast at a finer temporal granularity, and then aggregates the forecast back to the granularity of the visualization. Tableau provides prediction bands which may be simulated or calculated from a closed form equation. All models with a multiplicative component or with aggregated forecasts have simulated bands, while all other models use the closed form equations.

Exponential Smoothing and Trend

Exponential smoothing models iteratively forecast future values of a regular time series of values from weighted averages of past values of the series. The simplest model, *Simple Exponential Smoothing*, computes the next level or smoothed value from a weighted average of the last actual value and the last level value. The method is exponential because the value of each level is influenced by every preceding actual value to an exponentially decreasing degree—more recent values are given greater weight.

Exponential smoothing models with trend or seasonal components are effective when the measure to be forecast exhibits trend or seasonality over the period of time on which the forecast is based. *Trend* is a tendency in the data to increase or decrease over time.

Seasonality is a repeating, predictable variation in value, such as an annual fluctuation in temperature relative to the season.

In general, the more data points you have in your time series, the better the resulting forecast will be. Having enough data is particularly important if you want to model seasonality, because the model is more complicated and requires more proof in the form of data to achieve a reasonable level of precision. On the other hand, if you forecast using data generated by two or more different DGPs, you will get a lower quality forecast because a model can only match one.

Seasonality

Tableau tests for a seasonal cycle with the length most typical for the time aggregation of the time series for which the forecast is estimated. So if you aggregate by months, Tableau will look for a 12-month cycle; if you aggregate by quarters, Tableau will search for a four-quarter cycle; and if you aggregate by days, Tableau will search for weekly seasonality. Therefore, if there is a six-month cycle in your monthly time series, Tableau will probably find a 12-month pattern that contains two similar sub-patterns. However, if there is a seven-month cycle in your monthly time series, Tableau will probably find no cycle at all. Luckily, seven-month cycles are uncommon.

Tableau can use either of two methods for deriving season length. The original temporal method uses the natural season length of the temporal granularity (TG) of the view. Temporal granularity means the finest unit of time expressed by the view. For example, if the view contains either a continuous green date truncated to month or discrete blue year and month date parts, the temporal granularity of the view is month. The new non-temporal method, introduced with Tableau 9.3, uses periodic regression to check season lengths from 2 to 60 for candidate lengths.

Tableau automatically selects the most appropriate method for a given view. When Tableau is using a date to order the measures in a view, if the temporal granularity is quarterly, monthly, weekly, daily or hourly, the season lengths are almost certainly 4, 12, 13, 7 or 24, respectively. So only the length natural to the TG is used to construct the five seasonal exponential smoothing models supported by Tableau. The AIC of the five seasonal models and the three non-seasonal models are compared and the lowest returned. (For an explanation of the AIC metric, see Forecast Descriptions.)

When Tableau is using an integer dimension for forecasting, the second method is used. In this case there is no temporal granularity (TG), so potential season lengths must be derived from the data.

The second method is also used if the temporal granularity is yearly. Yearly series rarely have seasonality, but, if they do, it must also be derived from the data.

The second method is also used for views with temporal granularity of minute or second. If such series have seasonality, the season lengths are likely 60. However, when measuring a regular real world process, the process may have a regular repetition which does not correspond to the clock. So, for minutes and seconds, Tableau also checks for a length different from 60 in the data. This does not mean that Tableau can model two different season lengths at the same time. Rather, ten seasonal models are estimated, five with a season length of 60 and another five with the season length derived from the data. Whichever of the ten seasonal models or three non-seasonal models has the lowest AIC, that model is used to compute the forecast.

For series ordered by year, minute, or second, a single season length from the data is tested if the pattern is fairly clear. For integer ordered series, up to nine somewhat less clear potential season lengths are estimated for all five seasonal models, and the model with the lowest AIC is returned. If there are no likely season length candidates, only the non-seasonal models are estimated.

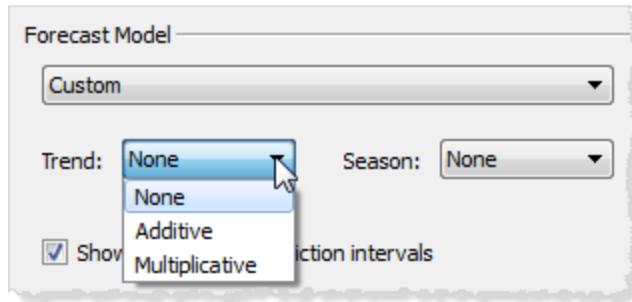
Since all selection is automatic when Tableau is deriving potential season lengths from the data, the default Model Type of “Automatic” in the Forecast Options Dialog Model Type menu does not change. Selecting “Automatic without seasonality” improves performance by eliminating all season length searching and estimation of seasonal models.

The heuristic that Tableau uses to decide when to use season lengths derived from the data depends on the distribution of errors for the periodic regression of each candidate season length. Since the assembly of season length candidates by periodic regression usually produces one or two clear winning lengths if seasonality actually exists in the data, the return of a single candidate indicates likely seasonality. In this case, Tableau estimates seasonal models with this candidate for year, minute and second granularity. The return of less than the maximum of ten candidates indicates possible seasonality. In this case, Tableau estimates seasonal models with all returned candidates for integer ordered views. The return of the maximum number of candidates indicates that errors for most length are similar. Therefore, the existence of any seasonality is unlikely. In this case, Tableau estimates only non-seasonal models for an integer-ordered or yearly ordered series, and only the seasonal models with a natural season length for other temporally ordered views.

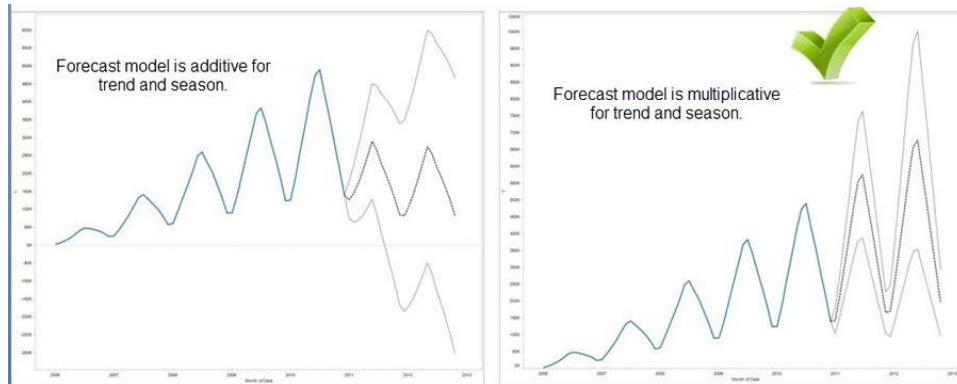
For Model Type “Automatic” in integer-, year-, minute- and second-ordered views, candidate season lengths are always derived from the data whether or not they are used. Since model estimation is much more time consuming than periodic regression, the performance impact should be moderate.

Model Types

In the Forecast Options dialog box, you can choose the model type Tableau users for forecasting. The **Automatic** setting is typically optimal for most views. If you choose **Custom**, then you can specify the trend and season characteristics independently, choosing either **None**, **Additive**, or **Multiplicative**:



An additive model is one in which the contributions of the model components are summed, whereas a multiplicative model is one in which at least some component contributions are multiplied. Multiplicative models can significantly improve forecast quality for data where the trend or seasonality is affected by the level (magnitude) of the data:



Keep in mind that you do not need to create a custom model to generate a forecast that is multiplicative: the **Automatic** setting can determine if a multiplicative forecast is appropriate for your data. However, a multiplicative model cannot be computed when the measure to be forecast has one or more values that are less than or equal to zero.

Forecasting with Time

When you are forecasting with a date, there can be only one base date in the view. Part dates are supported, but all parts must refer to the same underlying field. Dates can be on **Rows**, **Columns**, or **Marks** (with the exception of the Tooltip target).

Tableau supports three types of dates, two of which can be used for forecasting:

- Truncated dates and reference a particular point in history with specific temporal granularity, such as February 2017. They are usually continuous, with a green background in the view. Truncated dates are valid for forecasting.
- Date parts refer to a particular member of a temporal measure such as February. Each date part is represented by a different, usually discrete field (with a blue background). Forecasting requires at least a Year date part. Specifically, it can use any of the following sets of date parts for forecasting:
 - Year
 - Year + quarter
 - Year + month
 - Year + quarter + month
 - Year + week
 - Custom: Month/Year, Month/Day/Year

Other date parts, such as Quarter or Quarter + month, are not valid for forecasting. See [Convert Fields between Discrete and Continuous on page 284](#) for more details about different date types.

- Exact dates refer to a particular point in history with maximum temporal granularity such as February 1, 2012 at 14:23:45.0. Exact dates are invalid for forecasting.

It is also possible to forecast without a date. See [Forecasting When No Date is in the View on page 645](#).

Granularity and Trimming

When you create a forecast, you select a date dimension that specifies a unit of time at which date values are to be measured. Tableau dates support a range of such time units, including Year, Quarter, Month, and Day. The unit you choose for the date value is known as the *granularity* of the date.

The data in your measure typically does not align precisely with your unit of granularity. You might set your date value to quarters, but your actual data may terminate in the middle of a quarter—for example, at the end of November. This can cause a problem because the value for this fractional quarter is treated by the forecasting model as a full quarter, which will typically have a lower value than a full quarter would. If the forecasting model is allowed to consider this data, the resulting forecast will be inaccurate. The solution is to trim the data, such that the trailing periods that could mislead the forecast are ignored. Use the Ignore Last option in the Forecast Options dialog box to remove—or *trim*—such partial periods. The default is to trim one period.

Getting More Data

Tableau requires at least five data points in the time series to estimate a trend, and enough data points for at least two seasons or one season plus five periods to estimate seasonality. For example, at least nine data points are required to estimate a model with a four quarter seasonal cycle ($4 + 5$), and at least 24 to estimate a model with a twelve month seasonal cycle ($2 * 12$).

If you turn on forecasting for a view that does not have enough data points to support a good forecast, Tableau can sometimes retrieve enough data points to produce a valid forecast by querying the datasource for a finer level of granularity:

- If your view contains fewer than nine years of data, by default, Tableau will query the data source for quarterly data, estimate a quarterly forecast, and aggregate to a yearly forecast to display in your view. If there are still not enough data points, Tableau will estimate a monthly forecast and return the aggregated yearly forecast to your view.
- If your view contains fewer than nine quarters of data, by default Tableau will estimate a monthly forecast and return the aggregated quarterly forecast results to your view.
- If your view contains fewer than nine weeks of data, by default, Tableau will estimate a daily forecast and return the aggregated weekly forecast results to your view.
- If your view contains fewer than nine days of data, by default, Tableau will estimate an hourly forecast and return the aggregated daily forecast results to your view.
- If your view contains fewer than nine hours of data, by default, Tableau will estimate an minutely forecast and return the aggregated hourly forecast results to your view.
- If your view contains fewer than nine minutes of data, by default, Tableau will estimate an secondly forecast and return the aggregated minutely forecast results to your view.

These adjustments happen behind the scene and require no configuration. Tableau does not change the appearance of your visualization, and does not actually change your date value. However, the summary of the forecast time period in the Forecast Describe and Forecast Options dialog will reflect the actual granularity used.

Tableau can only get more data when the aggregation for the measure you are forecasting is SUM or COUNT. See [Aggregations on page 329](#) for information on available aggregation types and information on how to change the aggregation type.

To Create a Forecast

Forecasting requires a view that uses at least one date dimension and one measure. For example:

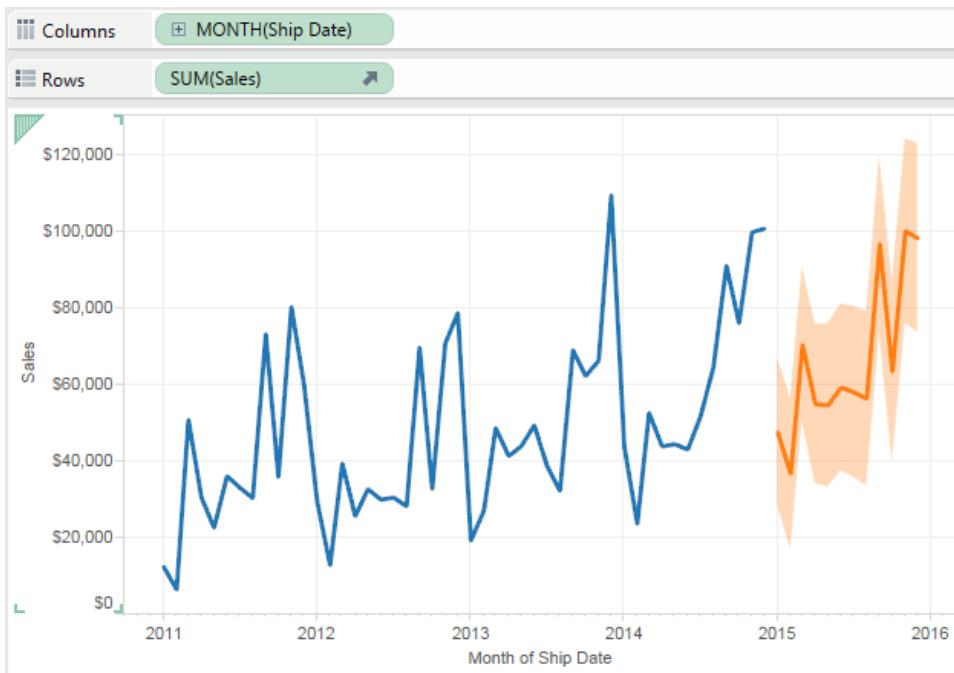
- The field you want to forecast is on the **Rows** shelf and a continuous date field is on the **Columns** shelf.
- The field you want to forecast is on the **Columns** shelf and a continuous date field is on the **Rows** shelf.

- The field you want to forecast is on either the **Rows** or **Columns** shelf, and discrete dates are on either the **Rows** or **Columns** shelf. At least one of the included date levels must be Year.
- The field you want to forecast is on the Marks card, and a continuous date or discrete date set is on **Rows**, **Columns** or **Marks**.

Note: You can also create a forecast when no date dimension is present if there is a dimension in the view that has integer values. See [Forecasting When No Date is in the View on page 645](#).

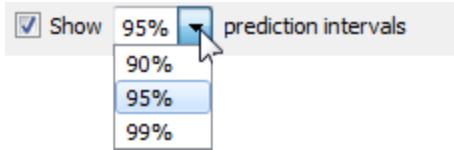
To turn forecasting on, either right-click (control-click on Mac) on the visualization and choose **Forecast > Show Forecast**, or choose **Analysis > Forecast > Show Forecast**.

With forecasting on, Tableau visualizes estimated future values of the measure, in addition to actual historical values. The estimated values are shown by default in a lighter shade of the color used for the historical data:



Prediction Intervals

The shaded area in the image above shows the 95% prediction interval for the forecast. That is, the model has determined that there is a 95% likelihood that the value of sales will be within the shaded area for the forecast period. You can configure the confidence level percentile for the prediction bands, and whether prediction bands are included in the forecast, using the **Show prediction intervals** setting in the Forecast Options dialog box:

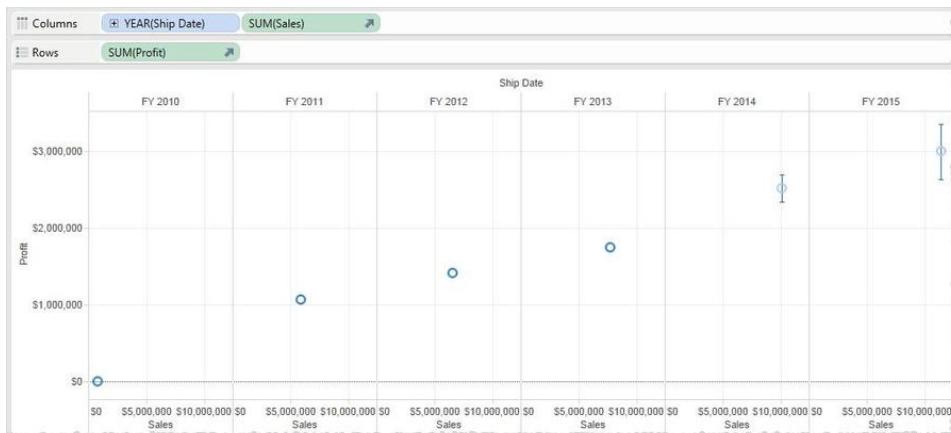


Clear the check box if you do not want to display prediction bands in forecasts. To set the prediction interval, select one of the values or enter a custom value. The lower the percentile you set for the confidence level, the narrower the prediction bands will be.

How your prediction intervals are displayed depends on the mark type of your forecasted marks:

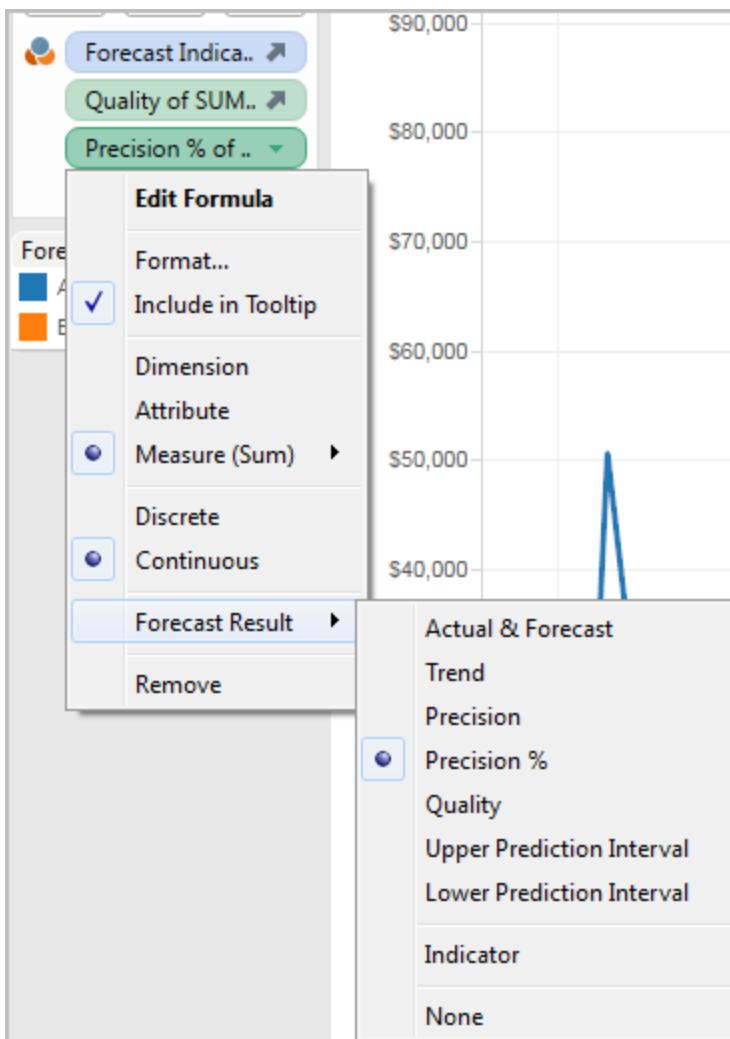
Forecast mark type	Prediction intervals displayed using
Line	Bands
Shape, square, circle, bar, or pie	Whiskers

In the following example, forecast data is indicated by lighter shaded circles, and the prediction intervals are indicated by lines ending in whiskers:



Enhancing Forecasts

For each forecast value, consider verifying the quality or precision of your forecast by dragging another instance of the forecast measure from the **Data** pane to the Detail shelf on the Marks card and then after right-clicking the field to open the content menu, choosing one of the available options:



For descriptions of these options, see [Forecast Field Results](#) on the next page.

You can repeat the process to add additional result types for each forecast value. See "Changing the Forecast Result" type in [Forecast Field Results](#) on the next page for information on changing the result type.

By adding such result types to the Details shelf, you add information about the forecast to tooltips for all marks that are based on forecasted data.

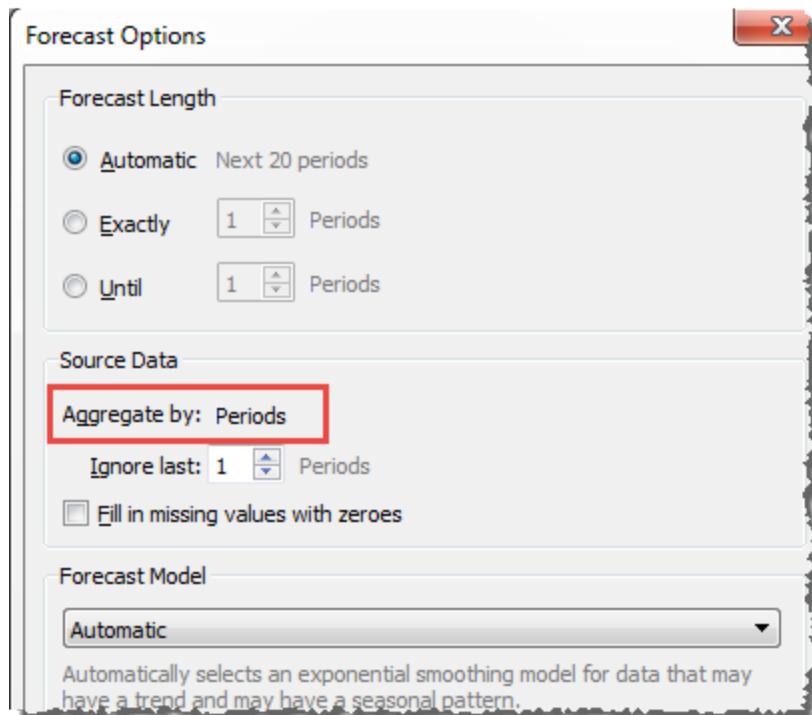
Forecast Indicator:	Estimate
Month of Ship Date:	April 2015
Precision % of Sales:	±38.00%
Quality of Sales:	64
Sales:	\$54,753

Forecasting When No Date is in the View

If a valid date is not in the view, Tableau will look for a dimension in the view that has integer values. If it finds such a dimension it will use that to forecast additional values for measures in the view. As with a date, when an integer dimension is selected to order the measures to be forecast, it is no longer used to partition the data. If there is more than one such integer dimension, Tableau will go in this order:

- An integer dimension on the Columns shelf. If there is more than one such dimension, it will use the first one (farthest to the left on the shelf).
- An integer dimension on the Rows shelf.
- An integer dimension on the Pages shelf.
- An integer dimension on the Marks card.

When Tableau is using an integer dimension to forecast, the Forecast Option and Forecast Description dialog boxes will automatically specify that forecasting is aggregating by periods:



Forecast Field Results

Tableau provides several types of forecast results. To view these result types in the view, right-click (control-click on Mac) on the measure field, choose **Forecast Result**, and then choose one of the options.

The options are:

- **Actual & Forecast**—Show the actual data extended by forecasted data.
- **Trend**—Show the forecast value with the seasonal component removed.
- **Precision**—Show the prediction interval distance from the forecast value for the configured confidence level.
- **Precision %**—Show precision as a percentage of the forecast value.
- **Quality**—Show the quality of the forecast, on a scale of 0 (worst) to 100 (best). This metric is scaled MASE, based on the MASE (Mean Absolute Scaled Error) of the forecast, which is the ratio of forecast error to the errors of a naïve forecast which assumes that the value of the current period will be the same as the value of the next period. The actual equation used for quality is:

$$100 * \max(1 - MASE, 0)$$

The Quality for a naïve forecast would be 0. The advantage of the MASE metric over the more common MAPE is that MASE is defined for time series which contain zero, whereas MAPE is not. In addition, MASE weights errors equally while MAPE weights positive and/or extreme errors more heavily.

- **Upper Prediction Interval**—Shows the value above which the true future value will lie confidence level percent of the time assuming a high quality model. The confidence level percentage is controlled by the Prediction Interval setting in the Forecast Options dialog box. See [Configure Forecast Options](#) on the next page.
- **Lower Prediction Interval**—Shows 90, 95, or 99 confidence level below the forecast value. The actual interval is controlled by the **Prediction Interval** setting in the Forecast Options dialog box.
- **Indicator**—Show the string **Actual** for rows that were already on the worksheet when forecasting was inactive and **Estimate** for rows that were added when forecasting was activated.
- **None**—Do not show forecast data for this measure.

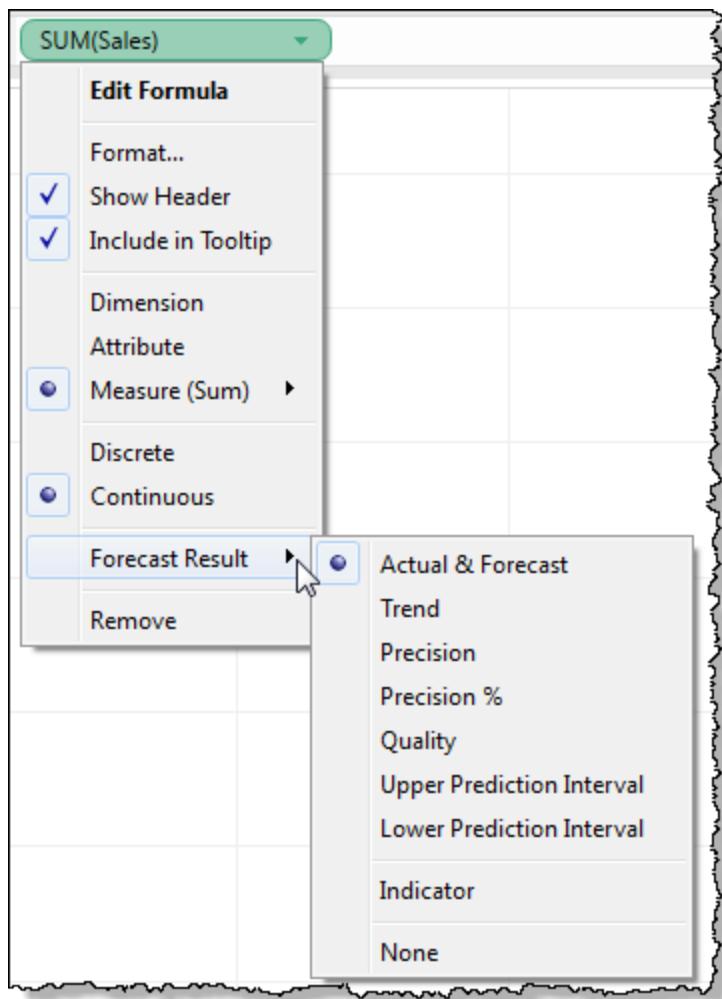
Forecast description information is also included in the worksheet description. See [Describing the View](#) on page 576.

Forecasting a New Measure

When you add a new measure to a visualization that already has forecasting enabled, Tableau attempts to forecast future values.

Changing the Forecast Result Type

To change the forecast result type for a measure, right-click (control-click on Mac) on the measure field, select **Forecast Result**, and then choose a result type.



Configure Forecast Options

Use the Forecast Options dialog box to configure forecast options, including:

- The length of the forecast
- The range and temporal aggregation of source data from which to generate the forecast
- The forecast model
- Prediction intervals

When forecasting is enabled, you can open the Forecast Options dialog box by choosing **Analysis > Forecast > Forecast Options**.

Forecast Length

The **Forecast Length** section determines how far into the future the forecast extends. Select one of the following:

- **Automatic:** Tableau determines the forecast length based on the data.
- **Exactly:** Extends the forecast for the specified number of units.
- **Until:** Extends the forecast to the specified point in the future.

Source Data

Use the **Source Data** section to specify.

- **Aggregate by:** Specifies the temporal granularity of the time series. With the default value (**Automatic**), Tableau chooses the best granularity for estimation. This will typically match the temporal granularity of the visualization (that is, the date dimension that the forecast is based on). However, it is sometimes possible and desirable to estimate the forecast model at a finer granularity than the visualization when the time series in the visualization is too short to allow estimation.

Note: When you are using an integer dimension instead of a date dimension for forecasting, the Aggregate by value is always Periods. See [Forecasting When No Date is in the View on page 645](#).

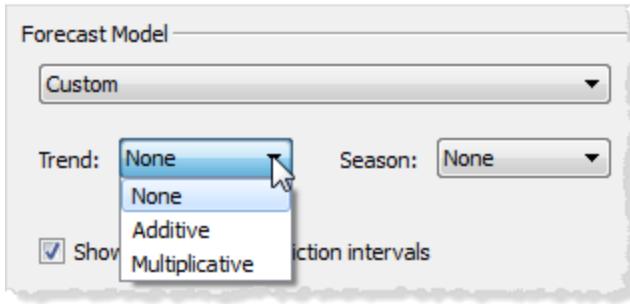
- **Ignore last:** Specifies the number of periods at the end of the actual data that should be ignored in estimating the forecast model. Forecast data is used instead of actual data for these time periods. Use this feature to trim off unreliable or partial trailing periods which could mislead the forecast. When the estimation granularity specified in the **Source Data** section is finer than in the visualization, the trimmed periods are estimation periods. As a result, the trailing actual visualization period may become a forecast period, which is an aggregate of both actual and forecast periods of estimation granularity. In contrast, null values are not filled with zeros and must be filtered to allow forecast.
- **Fill in missing values with zeros:** If there are missing values in the measure you are attempting to forecast, you can specify that Tableau fill in these missing values with zero.

Forecast Model

The **Forecast Model** section specifies how the forecast model is to be produced.

Use the drop down to specify whether Tableau selects what it determines to be the best of all models (**Automatic**), the best of those with no seasonal component (**Automatic without seasonality**), or the model you specify (**Custom**).

When you choose the **Custom** option, two new fields appear in the Forecast Options dialog box, which you use to specify the trend and season characteristics for your model:



The choices are the same for both fields:

- **None**: When you select **None** for Trend, the model does not assess the data for trend. When you select **None** for Season, the model does not assess the data for seasonality.
- **Additive**: An additive model is one in which the combined effect of several independent factors is the sum of the isolated effects of each factor. You can assess the data in your view for additive trend, additive seasonality, or both.
- **Multiplicative** : A multiplicative model is one in which the combined effect of several independent factors is the product of the isolated effects of each factor. You can assess the data in your view for multiplicative trend, multiplicative seasonality, or both.

If there is more than one time series in your visualization, the Custom option forces them all to be forecast using the same custom model. Constraining the models in this way usually results in lower quality models than would be produced by automatic model selection.

Constraints on Multiplicative Models

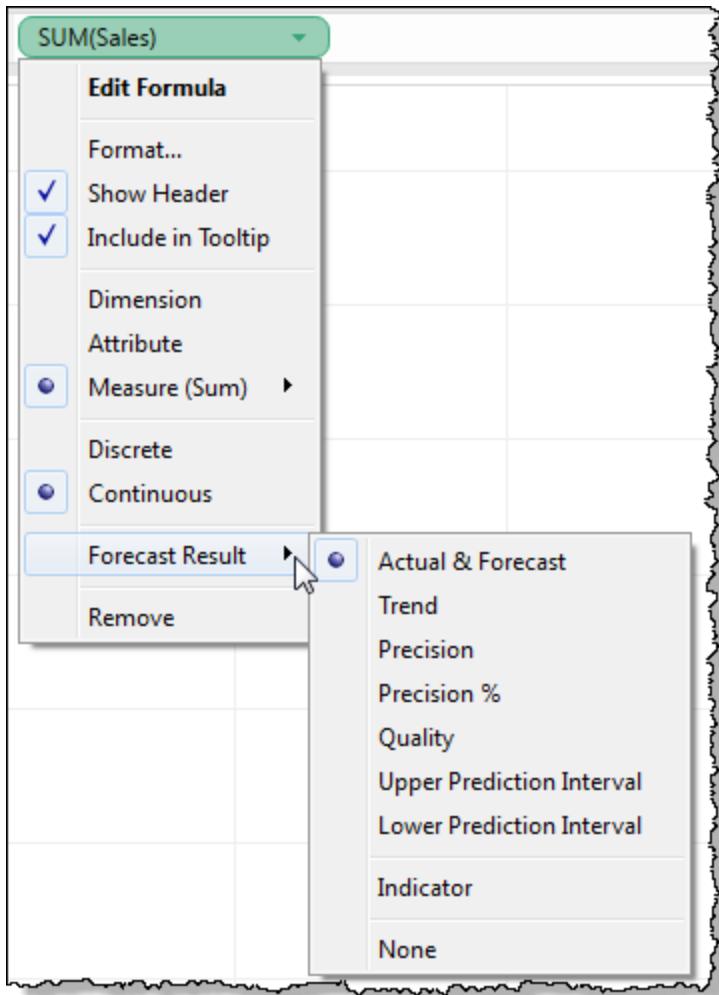
- You cannot use a multiplicative model when the measure to be forecast has one or more values that are less than or equal to zero, or even when some of the data points are too close to zero, relative to other data points.
- You cannot specify a model with multiplicative trend and additive season because the result may be numerically unstable.

Prediction Interval

You can set the prediction interval to 90, 95, or 99 percent, or enter a custom value. This value is used in two locations:

- In the prediction bands displayed with a forecast.
- For the prediction interval options (**Upper Prediction Interval** and **Lower Prediction**

Interval) that are available as forecast result types for a measure in the view:



Forecast Summary

The text box at the bottom of the Forecast Options dialog box provides a description of the current forecast. The forecast summary updates whenever you change any of the forecast options above. If there is a problem with the forecast, the text box provides an error message that may help you resolve the issue. See [Resolving Forecasting Errors](#) on page 655.

Forecast Descriptions

The **Describe Forecast** dialog box describes the forecast models that Tableau computed for your visualization. When forecasting is enabled, you can open this dialog by selecting **Analysis > Forecast > Describe Forecast**. The information in the Describe Forecast dialog box is read-only, though you can click **Copy to Clipboard** and then paste the screen contents into a writeable document.

The **Describe Forecast** dialog box has two tabs: a **Summary** tab and a **Models** tab.

Describe Forecast – Summary Tab

The **Summary** tab describes the forecast models Tableau has created, as well as the general patterns Tableau discovered in the data.

Options Used To Create Forecasts

This section summarizes the options Tableau used to create forecasts. These options were either picked automatically by Tableau or specified in the Forecast Options dialog box.

- **Time series**—The continuous date field used to define the time series. In some cases this value might not actually be a date. See [Forecasting When No Date is in the View on page 645](#).
- **Measures**—The measures for which values are estimated.
- **Forecast forward**—The length and date range of the forecast.
- **Forecast based on**—The date range of the actual data used to create the forecast.
- **Ignore last**—The number of periods at the end of the actual data that are disregarded--forecast data is displayed for these periods. This value is determined by the **Ignore Last** option in the Forecast Options dialog box.
- **Seasonal pattern**—The length of the seasonal cycle that Tableau found in the data, or None if no seasonal cycle was found in any forecast.

Forecast Summary Tables

For each measure that is forecasted, a summary table is displayed describing the forecast. If the view is broken into multiple panes using dimensions, a column is inserted into each table that identifies the dimensions. The fields in summary forecast tables are:

- **Initial**—The value and prediction interval of the first forecast period.
- **Change From Initial**—The difference between the first and the last forecast estimate points. The interval for those two points is shown in the column header. When values are shown as percentages, this field shows the percentage change from the first forecast period.
- **Seasonal Effect**—These fields are displayed for models identified as having seasonality—that is, a repeating pattern of variation over time. They show the high and low value of the seasonal component of the last full seasonal cycle in the combined time series of actual and forecast values. The seasonal component expresses the deviation from the trend and so varies around zero and sums to zero over the course of a season.
- **Contribution**—The extent to which Trend and Seasonality contribute to the forecast. These values are always expressed as percentages and add up to 100%.
- **Quality**—Indicates how well the forecast fits the actual data. Possible values are GOOD,

OK, and POOR. A naïve forecast is defined as a forecast that estimates that the value of the next period will be identical to the value of the current period. Quality is expressed relative to a naïve forecast, such that OK means the forecast is likely to have less error than a naïve forecast, GOOD means that the forecast has less than half as much error, and POOR means that the forecast has more error.

Describe Forecast – Models Tab

The **Models** tab provides more exhaustive statistics and smoothing coefficient values for the Holt-Winters exponential smoothing models underlying the forecasts. For each measure that is forecasted, a table is displayed describing the forecast models Tableau created for the measure. If the view is broken into multiple panes using dimensions, a column is inserted into each table that identifies the dimensions. The table is organized into the following sections:

Model

Specifies whether the components **Level**, **Trend**, or **Season** are part of the model used to generate the forecast. The value for each component is one of the following:

- **None**—The component is not present in the model.
- **Additive**—The component is present and is added to the other components to create the overall forecast value.
- **Multiplicative**—The component is present and is multiplied by the other components to create the overall forecast value.

Quality Metrics

This set of values provides statistical information about the quality of the model.

Value	Definition
RMSE: Root mean squared error	$\sqrt{\left(\frac{1}{n}\right) \sum e(t)^2}$
MAE: Mean absolute error	$\frac{1}{n} \sum e(t) $
MASE: Mean absolute scaled error. MASE measures the magnitude of the error compared to the magnitude of the error of a naive one-step ahead forecast as a ratio. A naive forecast assumes that whatever the value is today will be same value tomorrow. So, a MASE of 0.5 means that your forecast is likely to have half as much error as a naive forecast, which is better than a MASE of 1.0, which is no better than a naive forecast. Since this is a normalized statistic that is defined for all values and weighs errors evenly, it is an excellent	$\frac{\frac{1}{n} \sum e(t) }{\frac{1}{(n-1)} \sum_{t=2}^n Y(t) - Y(t-1) }$

<p>metric for comparing the quality of different forecast methods. The advantage of MASE over the more common MAPE metric is that MASE is defined for time series that contain zero, whereas MAPE is not. Also, MASE weights errors equally, whereas MAPE weights positive and/or extreme errors more heavily.</p>	
<p>MAPE: Mean absolute percentage error. MAPE measures the magnitude of the error compared to the magnitude of your data, as a percentage. So, a MAPE of 20% is better than a MAPE of 60%. Errors are the differences between the response values, which the model estimates, and the actual response values for each explanatory value in your data. Since this is a normalized statistic, it can be used to compare the quality of different models computed in Tableau. However, it is unreliable for some comparisons because it weights some kinds of error more heavily than others. Also, it is undefined for data with values of zero.</p>	$100 \frac{1}{n} \sum \left \frac{e(t)}{A(t)} \right $
<p>AIC: Akaike information criterion. AIC is a model quality measure, developed by Hirotugu Akaike, that penalizes complex models to prevent overfitting. In this definition, k is the number of estimated parameters, including initial states, and SSE is the sum of the squared errors.</p>	$n * \log(SSE / n) + 2 * (k + 1)$

In the preceding definitions, the variables are as follow:

Variable	Meaning
t	Index of a period in a time series.
n	Time series length.
m	Number of periods in a season/cycle.
A(t)	Actual value of the time series at period t.
F(t)	Fitted or forecast value at period t.

Residuals are: $e(t) = F(t) - A(t)$

Smoothing Coefficients

Depending on the rate of evolution in the level, trend, or seasonal components of the data, smoothing coefficients are optimized to weight more recent data values over older ones, such that within-sample one-step-ahead forecast errors are minimized. Alpha is the level smoothing

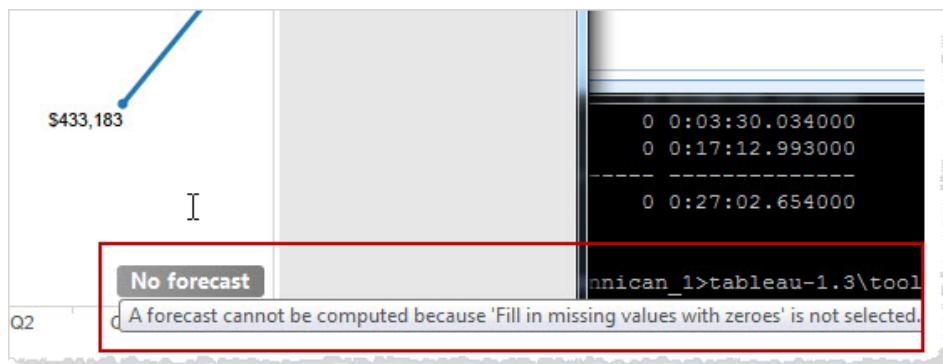
coefficient, beta the trend smoothing coefficient, and gamma the seasonal smoothing coefficient. The closer a smoothing coefficient is to 1.00, the less smoothing is performed, allowing for rapid component changes and heavy reliance on recent data. The closer a smoothing coefficient is to 0.00, the more smoothing is performed, allowing for gradual component changes and less reliance on recent data.

Troubleshooting Forecasting

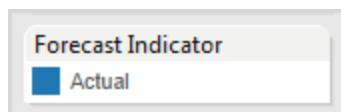
This topic discusses some issues that may arise as you use forecasting in Tableau.

Null Forecasts

A null forecast results when you are using forecasting and modify the view in a manner that is incompatible with forecasting. The most obvious indication that you have a null forecast is that no forecast data is displayed in your view and the text No Forecast is displayed. Hover the cursor over this message to see why Tableau is unable to create a forecast:



Another indication is that the **Forecast Indicator** field on the Marks card shows Actual values, but no Estimate values:



To diagnose a null forecast, open the Describe Forecast dialog box from the Analysis menu (**Analysis > Forecast > Describe Forecast**) to see the error message. Then see [Resolving Forecasting Errors on the next page](#) for a suggested resolution.

Tableau also displays the error message in the Forecast Options dialog box (**Analysis > Forecast > Forecast Options**).

Because a forecast cannot be computed for a time series with null date values in the middle, actions that filter data, either explicitly or implicitly can trigger a null forecast. The **Keep Only** and **Exclude** commands on tooltips are examples of actions that can filter data implicitly—in

some cases, these commands are removed when a forecast is shown. For example, if you have a time series of sales for each quarter from 2008 until 2012 and you exclude the Sales value for the second quarter of 2010, you will get a Null forecast because the time series is irregular. If, instead, you exclude the first quarter of 2008, you have shortened the time series but it remains regular. So, a valid forecast is still possible.

Resolving Forecasting Errors

If Tableau is unable to provide a forecast for your view, the problem can often be resolved by changing the Date value in the view (see [Changing Date Levels on page 616](#)).

Forecasting errors can result when the aggregation level of the time series (months, weeks, etc.) is either too fine or too coarse for the data to be forecast. This can lead to the "too much data" or "too little data" errors described below. Date aggregation can trigger a "too many Nulls" scenario when forecasting attempts to extract more data from the measure than is possible. For example, if the underlying granularity of the sales data is months but you aggregate by weeks, the result may be a significant number of Null values.

Other problems arise when the view's aggregation and the aggregation specified for the forecast (using the **Aggregate by** field in the Forecast Options dialog box) are not compatible. Tableau can create a forecast when the forecast aggregation is a finer level of detail than the view's aggregation, but not when it is at a coarser level of detail; even when it is finer, the two values are only compatible if there is a strict hierarchy that Tableau can use (for example, quarters can be evenly divided into three months, but months can't be evenly divided into weeks). Avoid these scenarios by setting **Aggregate by** to Automatic.

The following list shows errors that can be result from invalid forecasts in Tableau, and provides advice on how to resolve them.

Error message	Suggestion for Resolution
A continuous date cannot be derived from the date fields in the view.	Forecasting requires a date field that can be interpreted continuously. If the date field is not explicitly continuous, then one of the included date levels must be Year. This error is returned if there are no dates in the view, or if the dates in the view don't constitute a full hierarchy (for example, the date includes Year and Day, but not Month), or if they constitute a hierarchy that is not supported (for example, Year, Week, Day).
The time series is too short to forecast.	Expand the time series in your view to include more date values. This error is returned if there are fewer than four data points after trimming off unreliable or partial trailing periods which could mislead the forecast.

A forecast cannot be computed for a time series with Null date values.	Eliminate any Null values from the date field or fields in the view, either by filtering the date field or by using a less detailed date granularity (for example, by switching from months to quarters).
A forecast cannot be computed when the view contains multiple distinct date fields.	This error is returned if there are multiple date fields in the view. For example, if both Order Date and Ship Date are in the same view, forecasting is not supported.
The selected 'Aggregate by' value in Forecast Options is not compatible with the visualization.	<p>The date in the view must be compatible with the value of Aggregate by in the Forecast Options dialog box. For example, if Aggregate by is set to Weeks and the date in the view is set to Months, this error occurs.</p> <p>Change one of the dates so that the two are compatible, or set Aggregate by to Automatic.</p>
A forecast cannot be computed because there are too many missing values.	<p>This error is returned if more than 40% of the data is missing.</p> <p>Selecting Fill in missing values with zeros in the Forecast Options dialog box will not resolve this error. You must modify the source data or use data from a different source.</p>
There is no measure to forecast.	This error is returned if no measure that can be forecast is present in the view. Forecast measures must be on the Rows or Columns shelf, or on the Marks card.
The measure to forecast must be a number.	Some measures cannot be interpreted numerically and therefore cannot be forecast.
A forecast cannot be computed for a dimension.	The value to be forecast must be a measure, and not a dimension.
There is too much data to compute a forecast.	Forecasting is not possible when the result set from the query is too large. The limit is about 10,000 rows. To fix the forecast, aggregate the time series value at a higher level (for example, Month instead of Week) or filter the data.
A forecast cannot be computed because the data is divided into too many rows, columns, or colors.	Simplify the view to resolve the error by filtering or removing some of the dimensions.
A forecast cannot be computed because the view con-	Create a version of the view that does not contain table calculations.

tains table calculations.	
A forecast cannot be computed because there is a measure on the Filters shelf.	Remove the measure from the Filters shelf.
A forecast cannot be computed because Aggregate Measures is not selected.	Aggregate Measures is an option on the Analysis menu. See Aggregating Data on page 334 .
A forecast cannot be computed because the view contains percent calculations.	Percentage of is an option on the Analysis menu. See Percentages on page 941 .
A forecast cannot be computed because Grand Totals or Subtotals is enabled.	These options are controlled from the Totals command in the Analysis menu. See Totals on page 976 .
A multiplicative model cannot be computed because the measure to be forecast has one or more values that are less than or equal to zero.	You have created a custom model with Trend or Seasonality set to Multiplicative . Change this value, or set the Forecast Model to Automatic .
A model with multiplicative trend and additive season is not allowed because it is numerically unstable.	You have created a custom model configured as described in the error message. Change the settings for the custom model, or set the Forecast Model to Automatic .
A seasonal model cannot be computed because the time series is too short.	Expand the time series in your view to include more date values.
The selected multiplicative model cannot be computed because some of the data is too close to zero relative to the rest of the data.	You have created a custom model configured as described in the error message. Change the settings for the custom model, or set the Forecast Model to Automatic .

Trend Lines

Use the Tableau trend line feature to incrementally construct interactive models of behavior. With trend lines you can answer such questions as whether profit is predicted by sales, or

whether average delays at an airport are significantly correlated with the month of the year.

Adding Trend Lines

When you add trend lines to the view, you can specify how you want them to look and behave.

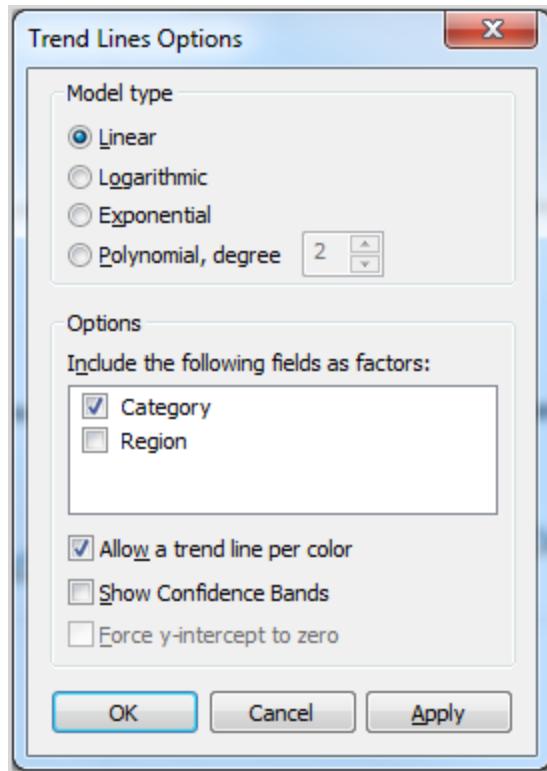
Add Trend Lines to the View

1. Select **Analysis > Trend Lines > Show Trend Lines**, or right-click (Control-click on a Mac) on the pane and choose **Trend Lines > Show Trend Lines**.

This command adds a linear trend line for each page, pane, and color on the worksheet. You can continue with the steps below to configure trend lines.

You can also drag a trend line in from the Analytics pane. See [Analytics Pane on page 135](#).

2. Select **Analysis > Trend Lines > Edit Trend Lines**, or right-click on the pane and choose **Trend Lines > Edit Trend Lines** to open the Trend Lines Options dialog box.
 - a. Select either a Linear, Logarithmic, Exponential, or Polynomial model type.
 - b. For trend models that are considering multiple fields, you can eliminate specific fields you want to exclude as factors in the trend line model. For example, on a view of sales across categories and regions, you may want to see the overall sales trend across all categories, rather than a different line for each region. In this case, you would exclude **Region** as a factor:



For more information on excluding factors, see [Removing Factors from the Model](#) on page 662.

- c. Decide whether to exclude color, using the **Allow a trend line per color** option. When you have color encodings in your view, you can use this option to add a single trend line that models all of the data, ignoring the color encoding.
- d. Decide whether to **Show Confidence Bands**. Tableau confidence bands show upper and lower 95% confidence lines by default when you add trend lines. Confidence lines are not supported for Exponential models.
- e. Select whether to **Force the y-intercept to zero**. This option is useful when you know that you want your trend line to begin at zero. This option is available only when both the **Rows** shelf and the **Columns** shelf contain a continuous field, as with a scatterplot.

3. When finished, click **OK**.

When Can't You Add Trend Lines?

To add trend lines to a view, both axes must contain a field that can be interpreted as a number. For example, you cannot add a trend line to a view that has the Product **Category** dimension, which contains strings, on the **Columns** shelf and the **Profit** measure on the **Rows** shelf. However, you can add a trend line to a view of sales over time because both sales and time can be interpreted as numeric values.

For multidimensional data sources, the date hierarchies actually contain strings rather than numbers. Therefore, trend lines are not allowed. Additionally, the 'm/d/yy' and 'mmmm yyyy' date formats on all data sources do not allow trend lines.

If you have trend lines turned on and you modify the view in a way where trend lines are not allowed, the trend lines do not show. When you change the view back to a state that allows trend lines, they reappear.

Tableau automatically stacks bar marks in many cases. However, trend lines cannot be turned on for stacked bars. You can turn off stacked marks by clearing the **Analysis > Stack Marks** option.

Removing Trend Lines

The easiest way to remove a trend line from a view is just to drag it off. You can also click a trend line and select **Remove**.

To remove all trend lines from the view, choose **Analysis > Trend Lines > Show Trend Lines** to remove the check mark, or right-click (Control-click on a Mac) the pane and choose **Trend Lines > Show Trend Lines** to remove the check mark.

Trend line options are retained for the next time you turn on trend lines. However, if you close the workbook with trend lines turned off, trend line options revert to defaults.

Trend Line Model Types

When you add a trend line to a view, you are building a statistical model. You are answering a question: Do the factors in your view predict a specific value (measure)? More precisely, do the categorical factors in your view split your data into meaningful samples and does the data in those samples allow you to predict values of the response measure given values of the explanatory measure or date? For example, you might be asking whether profit is predicted by time in a view that shows the profit of a company over four years.

Each trend line in the view visually represents a linear regression statistical model. Each model is estimated using data in the same pane and of the same color as the corresponding trend line. Although trend lines may be of type linear, logarithmic, exponential, or polynomial, this does *not* indicate that any of these models is not a linear regression. These types just identify transformations of either the explanatory or response variable in the linear regression formula. The word linear means that the formula is always of the form

$$y = b_0 + b_1 * x_1 + \dots + b_n * x_n$$

In other words, the formula is always linear in its coefficients: $b_0 \dots b_n$.

Model Types

Choose from the following model types. In the following formulas, X represents the explanatory variable, Y to response variable, and (ϵ) (epsilon) represents random error. Random errors are

uncorrelated with each other and with the explanatory variable, and have equal variance.

Linear

With the linear model type, no transformations are performed on either the explanatory or response variable. So the formula is

$$Y = b_0 + b_1 * X + \epsilon$$

Logarithmic

With the Logarithmic model type, the explanatory variable is transformed by the natural log before estimation of the model. So the formula is

$$Y = b_0 + b_1 * \ln(X) + \epsilon$$

Because a logarithm is not defined for number less than zero, any marks for which the explanatory variable is negative are filtered before estimation of the model. Avoid using a model that discards some data unless you know that the data being filtered out is invalid. The trend line description reports how many marks were filtered before model estimation.

Exponential

With the exponential model type, the response variable is transformed by the natural log before estimation of the model. So the formula is

$$\ln(Y) = b_0 + b_1 * X + \epsilon$$

With an exponential model, your response axis does *not* become logarithmic. Instead, the marks plotted in your view are found by plugging in various explanatory values to find values of $\ln(Y)$. These values are then exponentiated to plot the trend line. What you see is the exponential model:

$$Y = e^{(b_0 + b_1 * X + \epsilon)}$$

Because a logarithm is not defined for numbers less than zero, any marks for which the response variable is negative are filtered before model estimation.

Polynomial

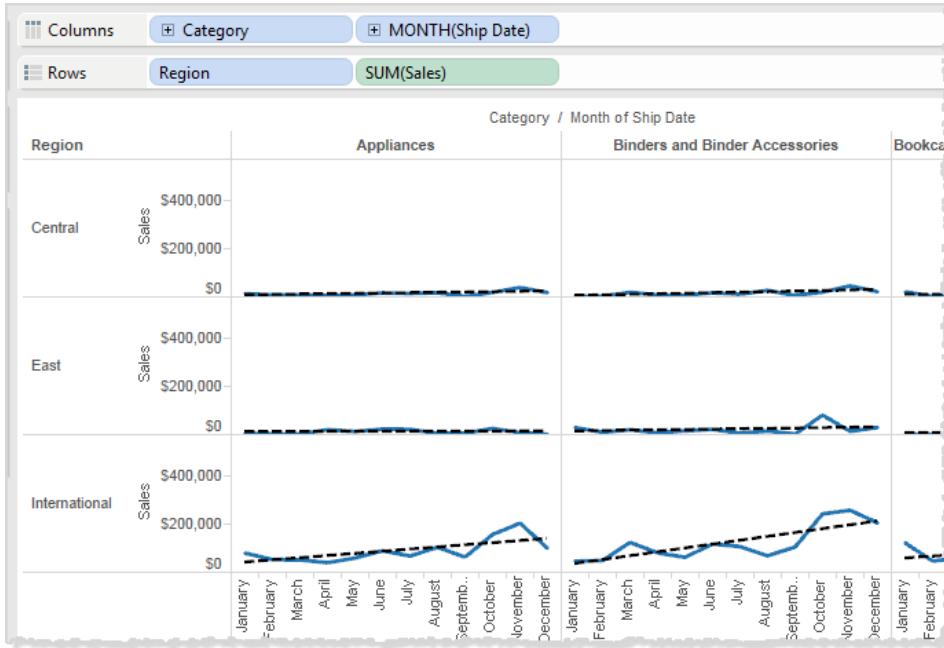
With the polynomial model type, the response variable is transformed into a polynomial series of the specified degree. So the formula is

$$Y = b_0 + b_1 * X + b_2 * X^2 + \dots + \epsilon$$

With a polynomial model type, you also select a **Degree** between 2 and 8. The higher polynomial degrees exaggerate the differences between the values of your data. So if your data increases very rapidly, the lower order terms may have almost no variation compared to the higher order terms, rendering the model impossible to estimate accurately. Also, more complicated higher order polynomial models require more data to estimate. So check the model description of the individual trends line for a red warning message indicating that an accurate model of this type is not possible.

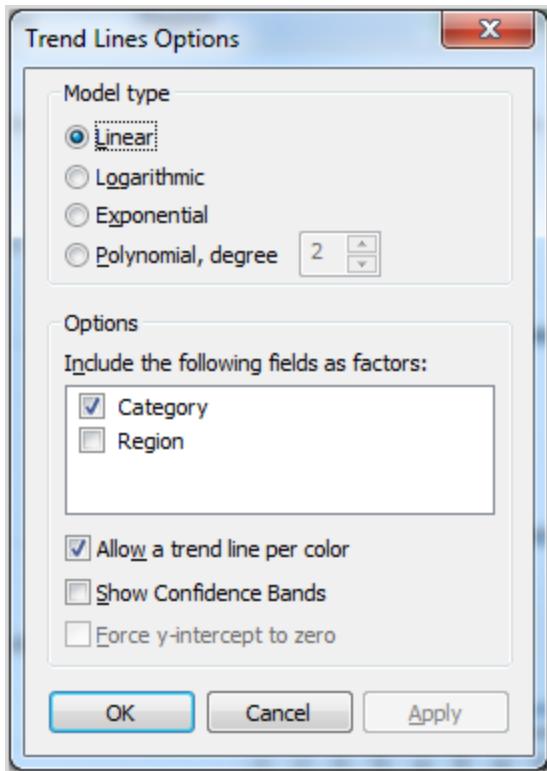
Removing Factors from the Model

You can remove factors from a trend line model in the Trend Lines Options dialog box. Often you will want to remove factors because you want the trend line model to be based on the entire row in the table rather than broken up by the members or values of a field. Consider the following example. The view below shows the monthly sales for various products categories, broken out by region.

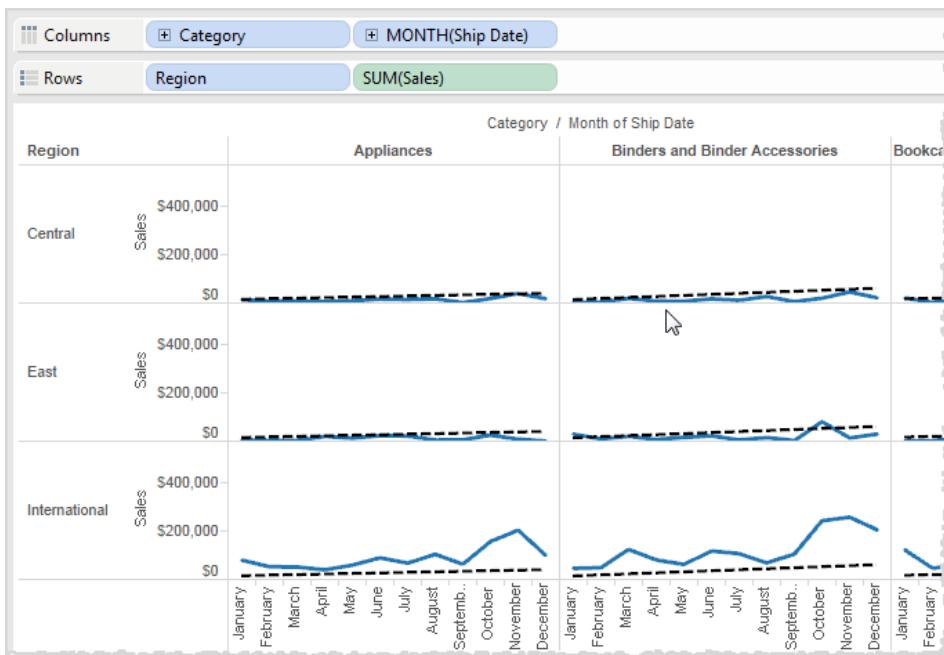


You can see that a separate model is created for each region.

Now remove **Region** as a factor in the model by deselecting it in the Trend Lines Options dialog box.

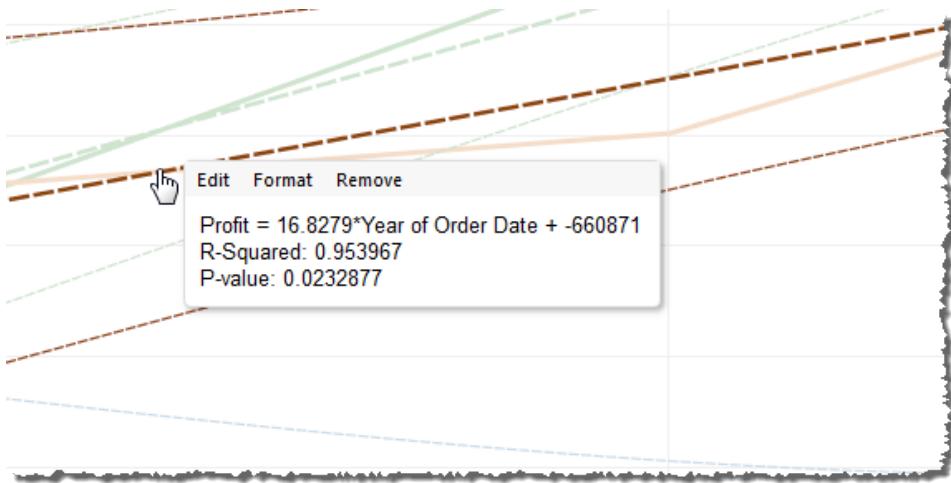


You can see that the trend line model within a category is now the same across all regions. This allows you to compare actual sales against a trend line that is the same for all regions.



Assessing Trend Line Significance

To see relevant information for any trend line in the view, hover the cursor over it:



The first line in the tooltip shows the equation used to compute a value of **Profit** from a value of **Year of Order Date**.

The second line, the R-Squared value, shows the ratio of variance in the data, as explained by the model, to the total variance in the data. For more details, see [Trend Line Model Terms on page 669](#).

The third line, the P-value, reports the probability that the equation in the first line was a result of random chance. The smaller the p-value, the more significant the model is. A p-value of 0.05 or less is often considered sufficient.

Entire Model Significance

Once you've added a trend line to a view, you typically want to know the goodness of fit of the model, which is a measure of the quality of the model's predictions. In addition, you may be interested in the significance of each factor contributing to the model. To view these numbers, open the Describe Trend Model dialog box, right-click (Control-click on a Mac) in the view and select **Trend Lines > Describe Trend Model**.

When you are testing significance, you are concerned with the p-values. The smaller the p-value, the more significant the model or factor is. It is possible to have a model that has statistical significance but which contains an individual trend line or a term of an individual trend line that does not contribute to the overall significance.

Under Trend Lines Model, look for the line that shows the p-value (significance) of the model: The smaller the p-value, the less likely it is that the difference in the unexplained variance between models with and without the relevant measure or measures was a result of random chance.

Describe Trend Model

Trend Lines Model

A linear trend model is computed for sum of Sales given Ship Date Month. The model may be significant at $p \leq 0.05$. The factor Category may be significant at $p=0.05$.

Model formula: Category*(Month of Ship Date + intercept)

Number of modeled observations: 204

Number of filtered observations: 0

Model degrees of freedom: 34

Residual degrees of freedom (DF): 170

SSE (sum squared error): 6.32246e+011

MSE (mean squared error): 3.71909e+009

R-Squared: 0.850309

Standard error: 60984.4

p-value (significance): < 0.0001

Analysis of Variance:

Field	DF	SSE	MSE	F	p-value
Category	32	3.08378e+012	9.6368e+010	25.9117	< 0.0001

Individual trend lines:

Panes	Line	Coefficients						
Row	Column	p-value	DF	Term	Value	StdErr	t-value	p-value
Sales	Appliances	0.0221712	10	Month of Ship Date	11355.9	4200.11	2.70372	0.0221712
				Intercept	52856.9	30912	1.70992	0.118073
Sales	Binders and Binder Accessories	0.0048827	10	Month of Ship Date	21028.2	5848.16	3.5957	0.0048827
				Intercept	38108.4	43041.3	0.885391	0.396736
Sales	Bookcases	0.0582459	10	Month of Ship Date	12823	5998.1	2.13785	0.0582459

This p-value for a model compares the fit of the entire model to the fit of a model composed solely of the grand mean (the average of data in the data view). That is, it assesses the explanatory power of the quantitative term $f(x)$ in the model formula, which can be linear, polynomial, exponential, or logarithmic with the factors fixed. It is common to assess significance using the "95% confidence" rule. Thus, as noted above, a p-value of 0.05 or less is considered good.

Significance of Categorical Factors

In the Analysis of Variance table, sometimes referred to as the ANOVA table, each field that is used as a factor in the model is listed. For each field, among other values, you can see the p-value. In this case, the p-value indicates how much that field adds to the significance of the entire model. The smaller the p-value the less likely it is that the difference in the unexplained variance between models with and without the field was a result of random chance. The values displayed for each field are derived by comparing the entire model to a model that does not include the field in question.

The following image shows the Analysis of Variance table for a view of quarterly sales for the past two years of three different product categories.

Analysis of Variance:

Field	DF	SSE	MSE	F	p-value
Category	160	1.65811e+012	1.03632e+010	20.9732	< 0.0001
Region	136	2.64345e+012	1.94371e+010	39.3372	< 0.0001

As you can see, the p-values for **Category** and **Region** are both quite small. Both of these factors are statistically significant in this model.

For information on specific trend line terms, see [Trend Line Model Terms](#) on page 669.

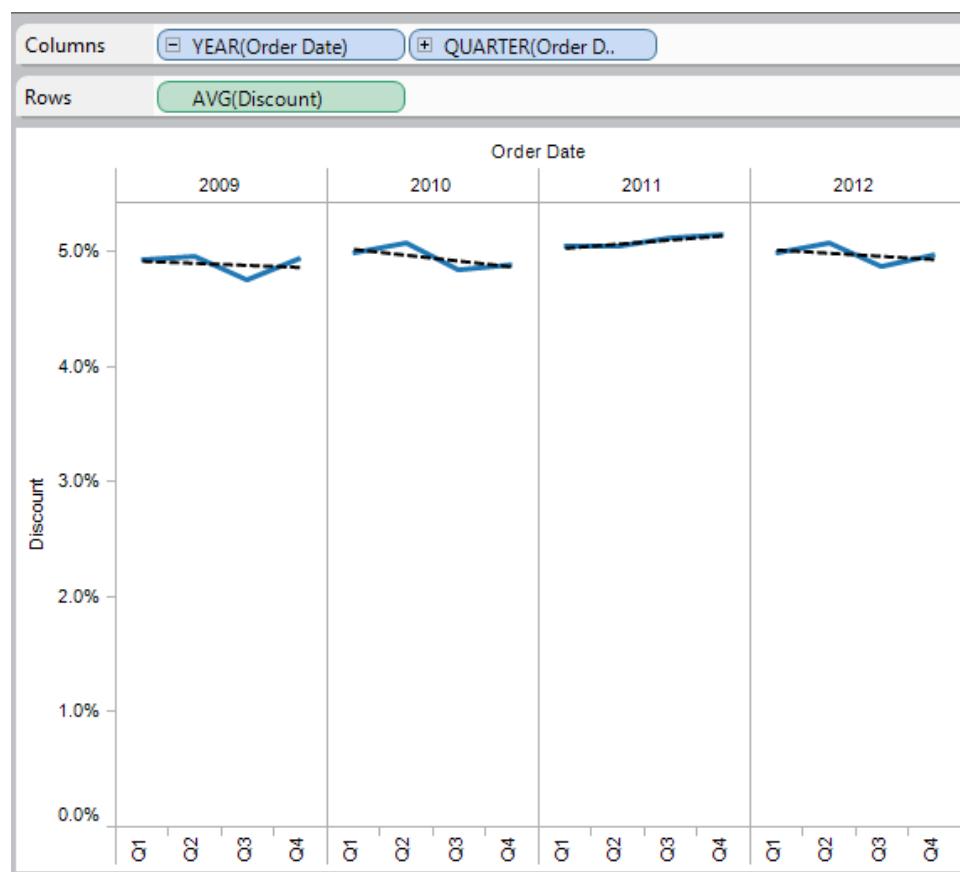
For ANOVA models, trend lines are defined by the mathematical formula:

$$Y = \text{factor 1} * \text{factor 2} * \dots * \text{factorN} * f(x) + \epsilon$$

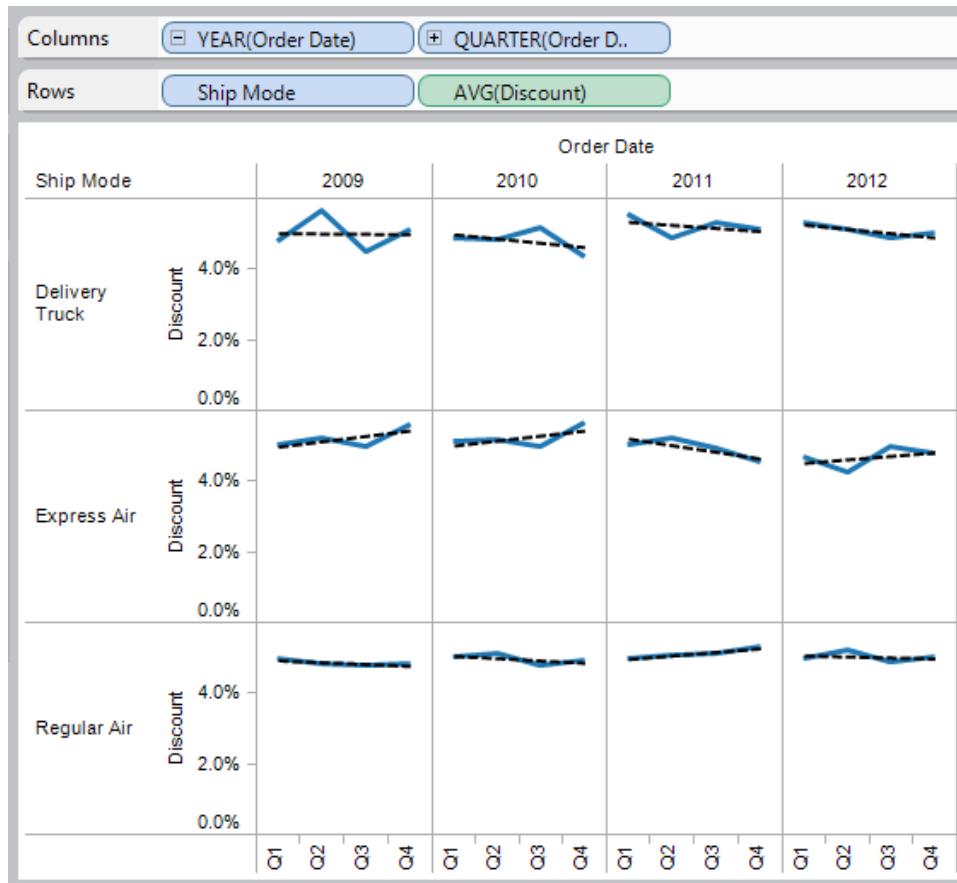
The term Y is called the *response variable* and corresponds to the value you are trying to predict. The term x is the *explanatory variable*, and ϵ (epsilon) is random error. The factors in the expression correspond to the categorical fields in the view. In addition, each factor is represented as a matrix. The $*$ is a particular kind of matrix multiplication operator that takes two matrices with the same number of rows and returns a new matrix with the same number of rows. That means that in the expression $\text{factor 1} * \text{factor 2}$, all combinations of the members of factor 1 and factor 2 are introduced. For example, if factor 1 and factor 2 both have three members, then a total of nine variables are introduced into the model formula by this operator.

Answering Questions with Trend Lines

We start with a question: what is causing high discount rates at a superstore? Using trend lines, we can find out which variables correlate with high discounts. The view below shows the average monthly discount rates for all stores over a four-year range.



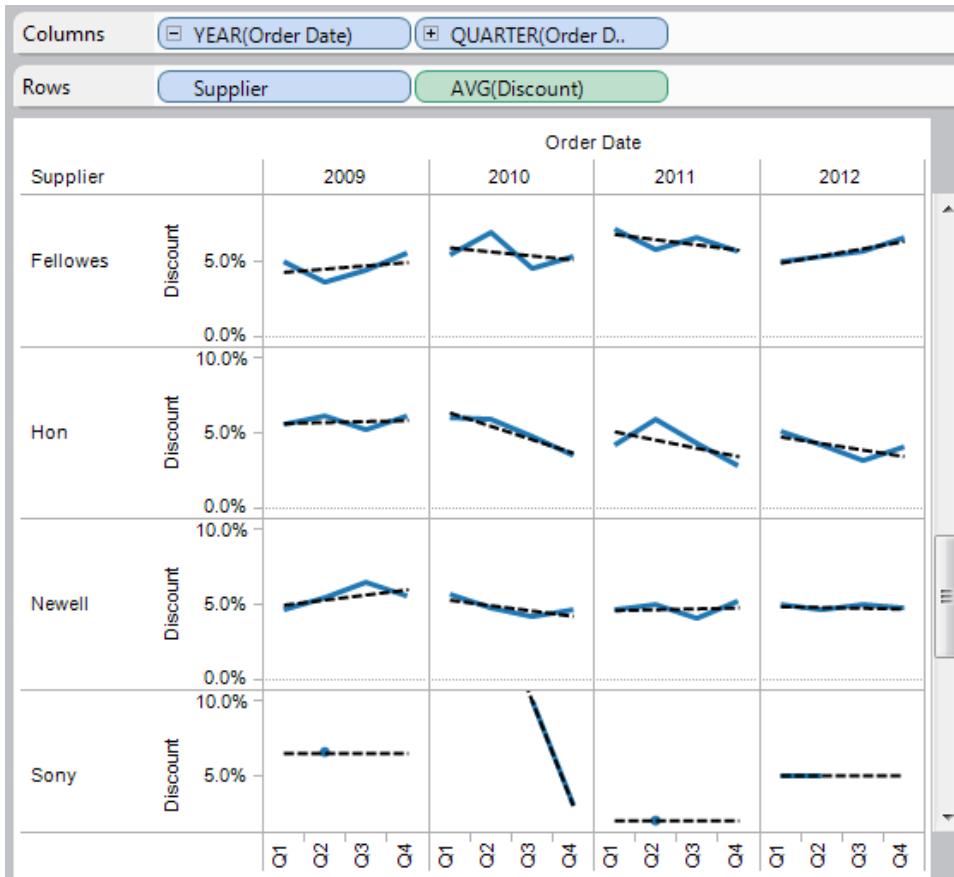
Our first thought is that discount rates may be high for specific ship modes. When we put the **Ship Mode** field on the **Rows** shelf, we see that there was a spike in discount rates for items shipped by Delivery Truck in 2009.



However, in the description of the trend line mode, the Analysis of Variance list shows that the p-value for Ship Mode is 0.22. (To see the description, right-click (Control-click on a Mac) the trend line and choose **Describe Trend Model**). This number is too high to be significant, indicating that we cannot predict discount rates based on ship mode. (It is common to test significance by applying the “95% confidence” rule. This equates to a p-value of 0.05 or less.)

Analysis of Variance:					
Field	DF	SSE	MSE	F	p-value
Ship Mode	16	0.000181235	1.133e-005	1.39986	0.222194
Year of Order Date	18	0.00018312	1.017e-005	1.25726	0.295853

The next variable we test is supplier. Suppliers sometimes offer substantial discounts. So we remove **Ship Mode** and drag **Supplier** to the **Rows** shelf.



Now when we open the Describe Trend Model dialog box, we see that the p-value for Supplier is 0.0004, while the p-value for the entire model is less than 0.003. With that, we can be confident in concluding that changes in average discount rates over time are in fact quite different for different suppliers. In other words we can be statistically confident that the different trend lines slopes for each supplier isn't just due to randomness, but rather to a real correlation between a supplier and the slope of this line.

Analysis of Variance:

<u>Field</u>	<u>DF</u>	<u>SSE</u>	<u>MSE</u>	<u>F</u>	<u>p-value</u>
Supplier	126	0.0349464	0.0002774	1.60454	0.0045179
Year of Order Date	100	0.0318394	0.0003184	1.84197	0.000667

In addition to **Supplier**, we notice that the **Year of Order Date** field offers a statistically significant improvement to the model. See [Assessing Trend Line Significance](#) for more information.

Assumptions

The p-values reported in Tableau trend lines depend on some assumptions about the data.

The first assumption is that, whenever a test is performed, the model for the mean is (at least approximately) correct.

The second assumption is that the "random errors" referred to in the model formula (see [Trend Line Model Types on page 660](#)) are independent across different observations, and that they all have the same distribution. This constraint would be violated if the response variable had much more variability around the true trend line in one category than in another.

Assumptions Required to Compute Trend Lines

The Assumptions required to compute (using Ordinary Least Squares) each individual trend line are:

- Your model is an accurate functional simplification of the true data generating process (for example, no linear model for a log linear relationship).
- Your errors average to zero and are uncorrelated with your independent variable (for example, no error measuring the independent variable).
- Your errors have constant variance and are not correlated with each other (for example, no increase in error spread as your independent variable increases).
- Explanatory variables are not exact linear functions of each other (perfect multicollinearity).

Trend Line Model Terms

When you view the description for a trend line model, there are several values listed. This section discusses what each of these values means.

Model formula

This is the formula for the full trend line model. The formula reflects whether you have specified to exclude factors from the model.

Number of modeled observations

The number of rows used in the view.

Number of filtered observations

The number of observations excluded from the model.

Model degrees of freedom

The number of parameters needed to completely specify the model. Linear, logarithmic, and exponential trends have model degrees of freedom of 2. Polynomial trends have model degrees of freedom of 1 plus the degree of the polynomial. For example a cubic trend has

model degrees of freedom of 4, since we need parameters for the cubed, squared, linear and constant terms.

Residual degrees of freedom (DF)

For a fixed model, this value is defined as the number of observations minus the number of parameters estimated in the model.

SSE (sum squared error)

The errors are the difference between the observed value and the value predicted by the model. In the Analysis of Variance table, this column is actually the difference between the SSE of the simpler model in that particular row and the full model, which uses all the factors. This SSE also corresponds to the sum of the differences squared of the predicted values from the smaller model and the full model.

MSE (mean squared error)

The term MSE refers to "mean squared error" which is the SSE quantity divided by its corresponding degrees of freedom.

R-Squared

R-squared is a measure of how well the data fits the linear model. It is the ratio of the variance of the model's error, or unexplained variance, to the total variance of the data.

When the y-intercept is determined by the model, R-squared is derived using the following equation:

$$1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

When the y-intercept is forced to 0, R-squared is derived using this equation instead:

$$1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n y_i^2}$$

In the latter case, the equation will not necessarily match Excel. This is because R-squared is not well defined in this case, and Tableau's behavior matches that of R instead of that of Excel.

Standard error

The square root of the MSE of the full model. An estimate of the standard deviation (variability) of the "random errors" in the model formula.

p-value (significance)

The probability that an F random variable with the above degrees of freedom exceeds the observed F in this row of the Analysis of Variance table.

Analysis of Variance

This table, also known as the ANOVA table, lists information for each factor in the trend line model. The values are a comparison of the model without the factor in question to the entire model, which includes all factors.

Individual trend lines

This table provides information about each trend line in the view. Looking at the list you can see which, if any, are the most statistically significant. This table also lists coefficient statistics for each trend line. A row describes each coefficient in each trend line model. For example, a linear model with an intercept requires two rows for each trend line. In the Line column, the p-value and the DF for each line span all the coefficient rows. The DF column under the shows the residual degrees of freedom available during the estimation of each line.

Terms

The name of the independent term.

Value

The estimated value of the coefficient for the independent term.

StdErr

A measure of the spread of the sampling distribution of the coefficient estimate. This error shrinks as the quality and quantity of the information used in the estimate grows.

t-value

The statistic used to test the null hypothesis that the true value of the coefficient is zero.

p-value

The probability of observing a t-value that large or larger in magnitude if the true value of the coefficient is zero. So, a p-value of .05 gives us 95% confidence that the true value is not zero.

Commonly Asked Questions

This section describes some commonly asked questions regarding trend lines in Tableau.

How do I change the confidence level used in the model?

Tableau does not enforce a confidence level. It simply reports the significance of the whole model, or of a specific field, by showing the p-value. The p-value will measure the probability of obtaining the same trend result without taking the dimensions into account. For example, a trend of sales per time p-value of 0.05 means that there is 5% chance that the same value could be obtained without taking the time into consideration.

[What does it mean if the p-value for the model is significant but the p-value for the specific field in the Analysis of Variance table is not significant?](#)

The p-value in the Analysis of Variance table indicates whether the field adds or detracts from the significance of the entire model. The smaller the p-value the less likely it is that the difference in the unexplained variance between models with and without the field was a result of random chance. The values displayed for each field are derived by comparing the entire model to a model that does not include the field in question. So, for the situation where the p-value for the model is significant but the p-value for the specific field is not, you know that the model is statistically significant, but you cannot be confident that the specific field in question adds anything to it. Consider whether you might not be better off removing the factor from the model.

[What does it mean if the p-value for the specific field in the Analysis of Variance table is significant but the p-value for the model is not significant?](#)

This could happen in a case when there is no "trend" within each pane. For example, the lines are flat, but the mean varies across a given factor.

Filter Data from Your Views

In this topic:

- [Filtering Order of Operations](#) below
 - [Select to keep or exclude data points in your view](#) on the next page
 - [Drag dimensions, measures, and date fields to the Filters shelf](#) on page 675
 - [Display interactive filters in the view](#) on page 680
 - [Set options for filter card interaction and appearance](#) on page 681
 - [See Also:](#) on page 686 additional filtering resources
-

There are many different ways to filter certain values or a range of values from your view in Tableau. You can:

- Drag a field to the Filters shelf.
- Select to exclude or keep data points (marks) or headers in the view.
- Show an interactive filter in the view.

Other activities, including creating a level of detail expression or adding a table calculation can filter your data, even though these activities are not specifically about filtering.

Filtering Order of Operations

Before you begin filtering data in Tableau, it's important to understand the order in which Tableau executes filters in your workbook.

Tableau performs actions on your view in a very specific order; this is called the Order of Operations. When it comes to filtering, different types of filters are executed in the following order:

1. Extract filters
2. Data source filters
3. Context filters
4. Filters on dimensions (whether on the Filters shelf or in filter cards in the view)
5. Filters on measures (whether on the Filters shelf or in filter cards in the view)

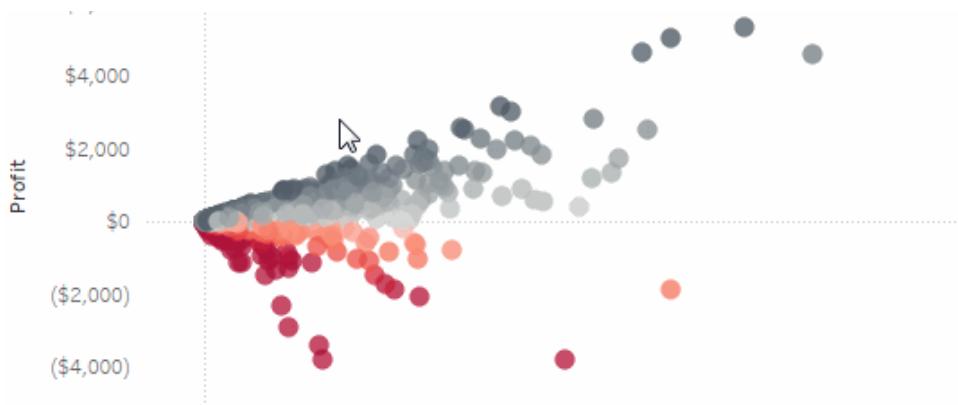
To learn more about how you can change the order of operations, see [Tableau's Order of Operations](#) on page 285.

Select to keep or exclude data points in your view

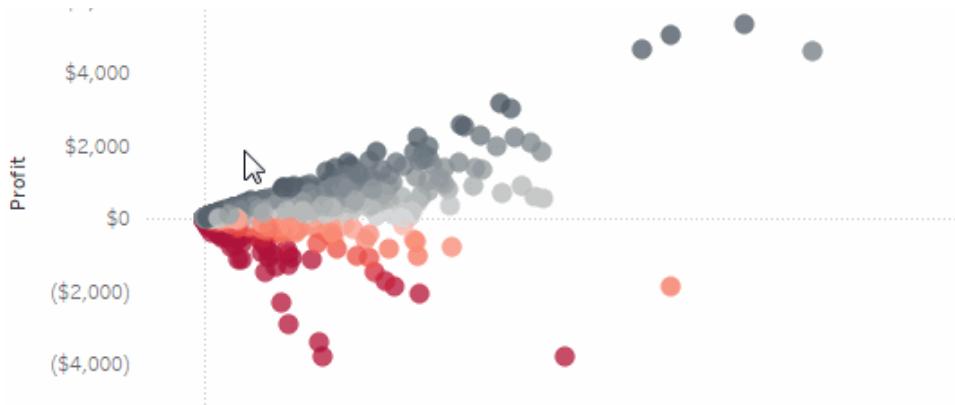
You can filter individual data points (marks), or a selection of data points from your view. For example, if you have a scatter plot with outliers, you can exclude them from the view so you can better focus on the rest of the data.

To filter marks from the view, select a single mark (data point) or click and drag in the view to select several marks. On the tooltip that appears, you can:

- Select **Keep Only** to keep only the selected marks in the view.



- Select **Exclude** to remove the selected marks from the view.



Note: These filtering options are not available if a Wildcard Match filter is already specified for the same field. See [Drag dimensions, measures, and date fields to the Filters shelf on the next page](#) to learn more about Wildcard Match filters.

Select headers to filter data

You can also select headers to filter them from your view.

To filter entire rows or columns of data from your view, select the header in the view. On the tooltip that appears, select to **Exclude** or **Keep Only** the selected data.

When you select a table header that is part of a hierarchy, all of the next level headers are also selected. For example, the view shown below consists of two unrelated dimensions placed on the **Columns** shelf, and two levels of the same hierarchy placed on the **Rows** shelf.

The selected row headers include the Furniture member of the **Category** dimension, and the Binders and Labels members of the **Sub-category** dimension. When Furniture is selected, all members from the next (inner) level in the hierarchy are automatically selected. In this case, that means the Bookcases, Chairs, Furnishings, and Tables members.

III Columns Region State

Rows Category Sub-Category

Sheet 21

		Region / State									
		Central									
Category	Sub-Catego..	Illinois	Indiana	Iowa	Kansas	Michigan	Minnes..	Missouri	Nebras..	Dakota	North
Furniture	Bookcases	\$4,283				\$810		\$213			\$342
	Chairs	\$14,563	\$6,463	\$1,408		\$13,878	\$6,079	\$61	\$564		\$3,963
	Furnishings	\$2,878	\$1,514	\$49	\$111	\$1,916	\$232	\$941	\$1,381		\$1,445
	Tables	\$6,551	\$3,519	\$1,185		\$5,717	\$1,300	\$1,722			\$2,534
Office Supplies	Appliances	\$975	\$4,160			\$82	\$4,324	\$2,844	\$3,670	\$501	\$1,491
	Art	\$930	\$389	\$147	\$163	\$1,005	\$103	\$240	\$19	\$182	\$59
	Binders	\$4,539	\$4,012	\$248	\$612	\$22,822	\$12,470	\$1,876	\$128	\$26	\$445
	Envelopes	\$384	\$890	\$13		\$310	\$31	\$71			\$407
	Fasteners	\$141	\$8	\$46	\$24	\$110	\$43		\$58	\$7	
	Labels	\$225	\$276		\$19	\$881	\$161	\$15	\$14		\$64
	Paper	\$3,456	\$1,880	\$316	\$303	\$2,011	\$320	\$302	\$333		\$197
	Storage	\$9,080	\$4,120	\$13	\$394	\$6,187	\$3,398	\$1,792	\$1,165	\$705	\$2,345
	Supplies	\$178			\$358	\$74	\$37	\$4,217	\$17		\$22
Technology	Accessories	\$5,536	\$2,279		\$92	\$4,933	\$1,520	\$1,022	\$240		\$1,817
	Copiers	\$5,920	\$18,500			\$1,150	\$550	\$5,500			
	Machines	\$3,756	\$84			\$3,411					
	Phones	\$16,772	\$5,460	\$1,154	\$757	\$6,731	\$775	\$565	\$3,046		\$4,551

([Back to top](#))

Drag dimensions, measures, and date fields to the Filters shelf

Another way to create a filter is to drag a field directly from the Data pane to the Filters shelf.

The screenshot shows the Tableau Data pane with the following structure:

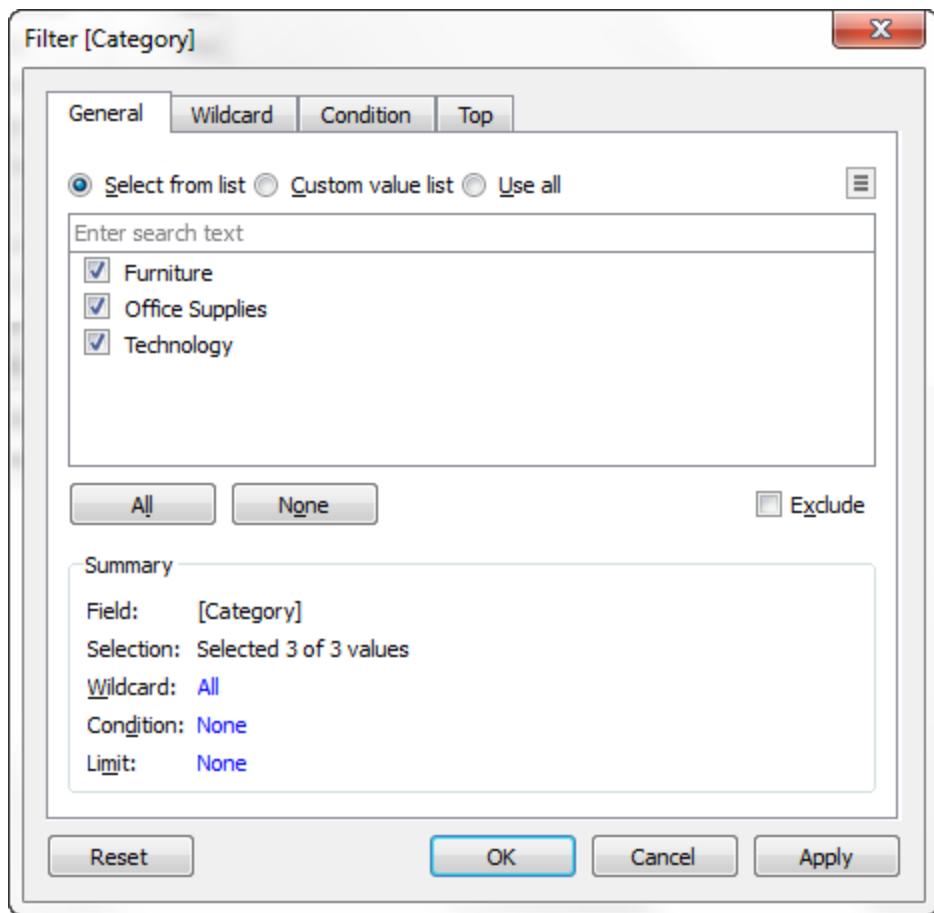
- Data** tab selected.
- Dimensions** section:
 - Customer:
 - Customer Name
 - Segment
 - Order:
 - Order Date
 - Order ID
 - Ship Date
 - Ship Mode
 - Location
- Measures** section:
 - Discount
 - Profit
 - Profit Ratio
 - Quantity
 - Sales (highlighted with a green background)
 - Latitude (generated)
 - Longitude (generated)
 - Number of Records
 - Measure Values

When you add a field to the Filters shelf, the Filter dialog box opens so you can define the filter. The Filter dialog box differs depending on whether you are filtering categorical data (dimensions), quantitative data (measures), or date fields.

Filter categorical data (dimensions)

Dimensions contain discrete categorical data, so filtering this type of field generally involves selecting the values to include or exclude. You can also define conditions and limits to create a more complex filter definition.

When you drag a dimension from the Data pane to the Filters shelf, the following Filter dialog box appears:



There are four tabs in the dialog box:

- **General:** Use the General tab to select the values you want to include or exclude.
- **Wildcard:** Use the Wildcard tab to define a pattern to filter on. For example, when filtering on email addresses you might want to only include emails from a specific domain. You can define a wildcard filter that ends with "@gmail.com" to only include Google email addresses.
- **Condition:** Use the Condition tab in the Filter dialog box to define rules to filter by. For example, in a view showing the average Unit Price for a collection of products, you may want to only show the Products that have an average unit price that is greater than or equal to \$25. You can use the built-in controls to write a condition or you can write a custom formula.
- **Top:** Use the Top tab in the Filter dialog box to define a formula that computes the data that will be included in the view. For example, in a view that shows the average Time to Ship for a collection of products, you can decide to only show the top 15 products by Sales. Rather than having to define a specific range for Sales (e.g., greater than

\$100,000), you can define a limit (top 15) that is relative to the other members in the field (products).

Important Note: Each tab adds additional definitions to your filter. For example, you can select to exclude values under the General tab, and also add limits under the Top tab. Selections and configurations from both tabs are applied to your filter.

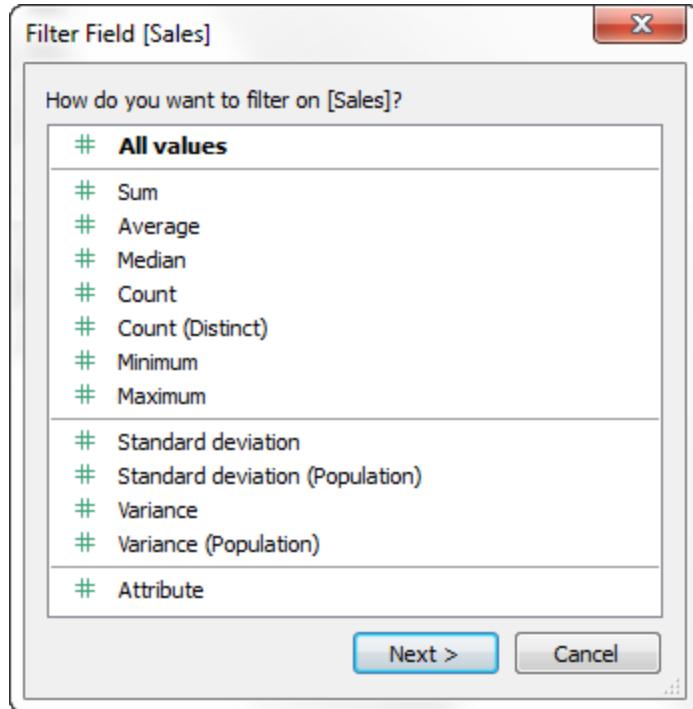
At any time, you can see the definitions of your filter under Summary on the General tab.

Filter quantitative data (measures)

Measures contain quantitative data, so filtering this type of field generally involves selecting a range of values that you want to include. You can create four types of quantitative filters: Range of Values, At Least, At Most, and Special.

Note: If you have a large data source, filtering measures can lead to a significant degradation in performance. It is sometimes much more efficient to filter by creating a set containing the measure and then apply a filter to the set.

When you drag a measure from the Data pane to the Filters shelf, the following dialog box appears:



Select how you want to aggregate the field, and then click **Next**.

In the subsequent dialog box, you're given the option to create the following types of quantitative filters:

Range of Values: Select the Range of Values option to specify the minimum and maximum values of the range to include in the view. The values you specify are included in the range.

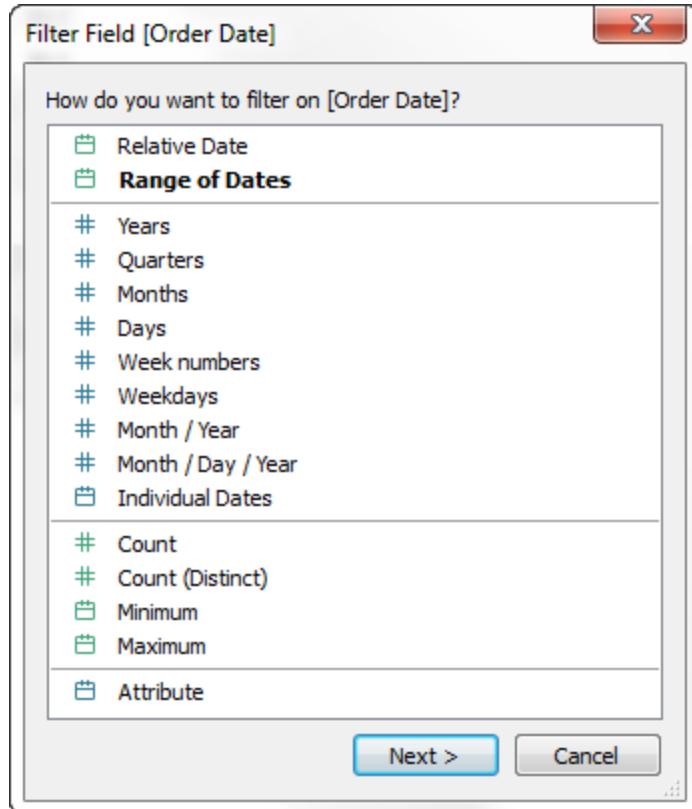
At Least: Select the At Least option to include all values that are greater than or equal to a specified minimum value. This type of filter is useful when the data changes often so specifying an upper limit may not be possible.

At Most: Select the At Most option to include all values that are less than or equal to a specified maximum value. This type of filter is useful when the data changes often so specifying a lower limit may not be possible.

Special: Select the Special option to filter on Null values. Include only Null values, Non-null values, or All Values.

Filter dates

When you drag a date field from the Data pane to the Filters shelf, the following Filter Field dialog box appears:



You can select whether you want to filter on a relative date; filter between a range of dates; or select discrete dates or individual dates to filter from the view.

- **Filter relative dates:** Click Relative dates to define a range of dates that updates based on the date and time you open the view. For example, you may want to see Year to Date sales, all records from the past 30 days, or bugs closed last week. Relative date filters can also be relative to a specific anchor date rather than today.
- **Filter a range of dates:** Select Range of dates to define a fixed range of dates to filter. For example, you may want to see all orders placed between March 1, 2009 and June 12, 2009.
- **Filter discrete dates:** Select a discrete date value in the dialog box if you want to include entire date levels. For example, if you select Quarters, you can choose to filter specific quarters (e.g. Q1, Q2, Q3, Q4) from your view, regardless of the year.
- **Filter individual dates:** Select Individual dates to filter specific dates from your view.
- **Additional date filter options:** When you select Relative dates or Range of dates, the Filter dialog box opens. In that dialog box, you can define a Starting date or Ending date. You can also select Special to include null dates, non-null dates, or all dates.

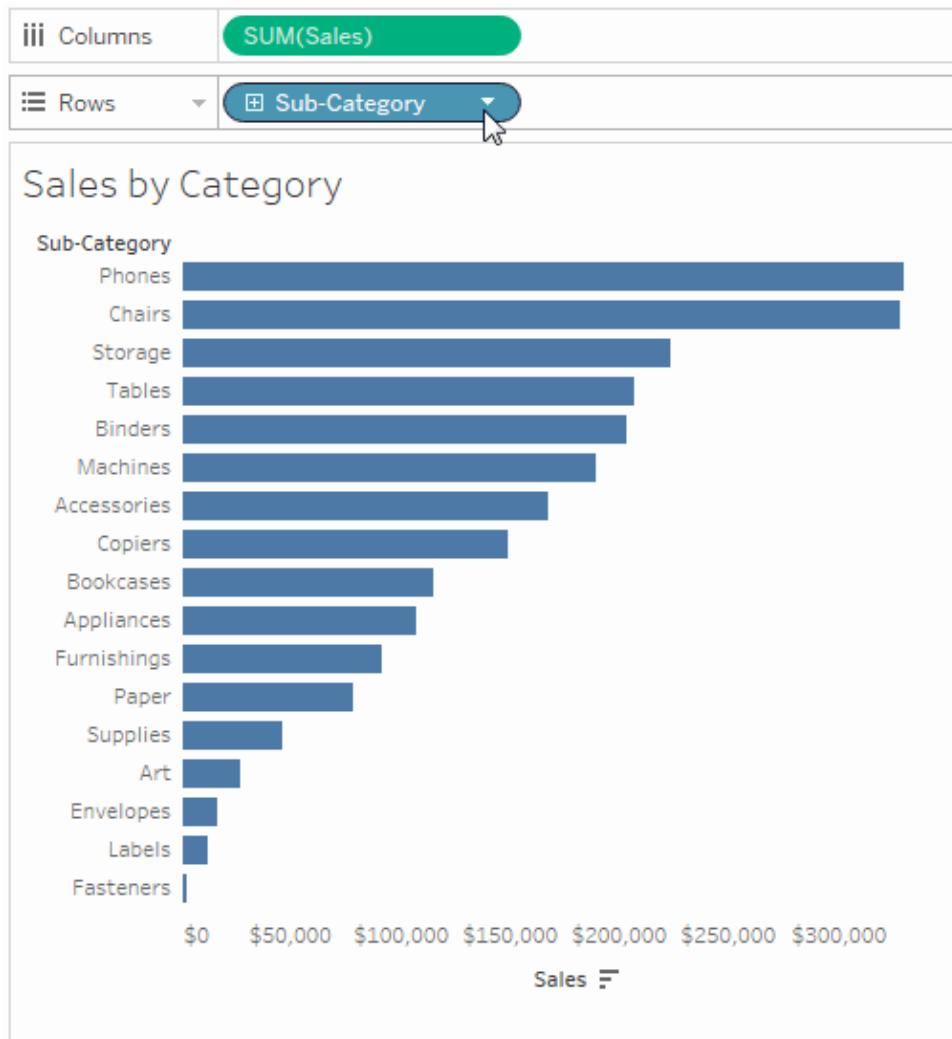
([Back to top](#))

Display interactive filters in the view

You can quickly add and modify filters by showing them in the view. When you show a filter, an interactive filter card appears in the view. Use the card to quickly include or exclude data in the view.

To show a filter in the view:

1. In the view, click the field drop-down menu and select **Show Filter**.



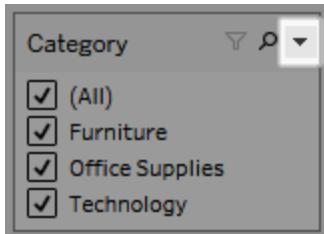
The field is automatically added to the Filters shelf (if it is not already being filtered), and a filter card appears in the view. Interact with the card to filter your data.

Note: You can add a quick filter for a field that is not currently used in the view. To do so, In the Data pane, click the field drop-down menu, and then select **Show Filter**. The field is automatically added to the **Filters** shelf, and a filter card appears in the view.

Set options for filter card interaction and appearance

After you show a filter, there are many different options that let you control how the filter works and appears. You can access these options by clicking the drop-down menu in the upper right

corner of the filter card in the view.



Some options are available for all types of filters, and others depend on whether you're filtering a categorical field (dimension) or a quantitative field (measure).

Finally, you can customize how filters appear in the view, in dashboards, or when published to Tableau Server.

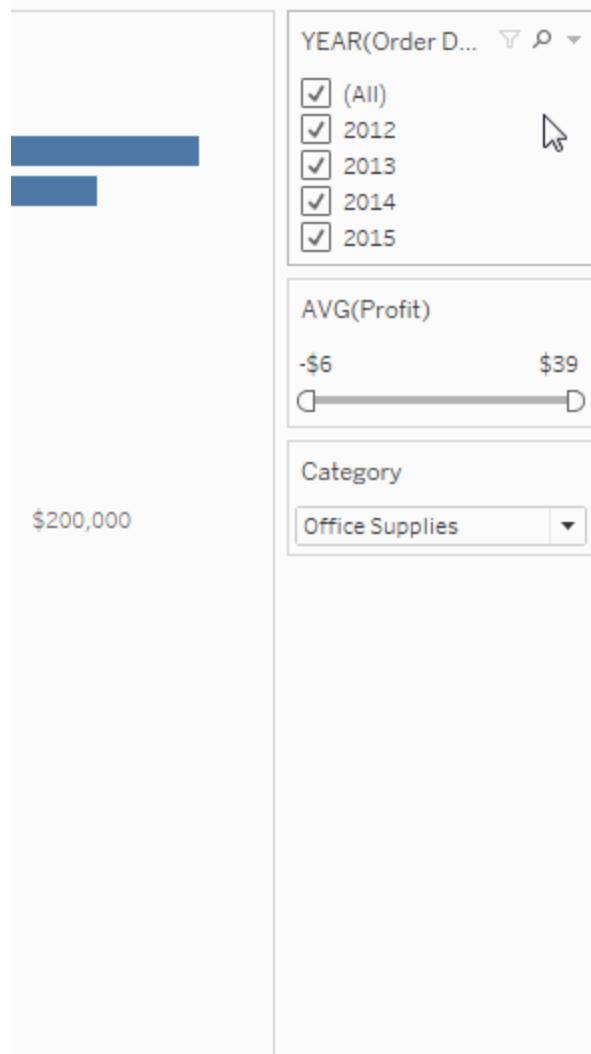
Here are some of the general filter card options:

- **Edit Filter** - This option opens the main Filter dialog box so you can further refine the filter by adding conditions and limits.
- **Remove Filter** - Removes the filter from the Filters shelf and removes the filter card from the view.
- **Apply to worksheets** - Allows you to specify whether the filter should apply to only the current worksheet or be shared across multiple worksheets. For more information, see [Apply Filters to Multiple Worksheets on page 686](#).
- **Format Filters** - Customize the font and colors of all your filter cards in the view.
- **Only relevant values** - Specifies which values to show in the filter. When you select this option other filters are considered and only values that pass these filters are shown. For example, a filter on State will only show the Eastern states when a filter on Region is set. You can use the toggle at the top of the filter card to switch between this option and the All Values in Database option.
- **All values in database** - Specifies which values to show in the filter. When you select this option all values in the database are shown regardless of the other filters on the view.
- **All values in context** - When one of the filters in the view is a context filter, select this option on a different filter to only display values that pass through the context filter. For more information, see [Improve View Performance with Context Filters on page 694](#).
- **Include values** - When this option is selected, the selections in the filter card are included in the view.
- **Exclude values** - When this option is selected, the selections in the filter card are excluded from the view.
- **Hide Card** - Hides the filter card but does not remove the filter from the Filters shelf.

Filter card modes

You can control the appearance and interaction of your filter card in the view by selecting a filter card mode.

To select a filter card mode, in the view, click the drop-down menu on the filter card and then select a mode from the list.



The types of filter card modes you see in the list of options depend on whether your filter is on a dimension or a measure. Below you can find brief descriptions of the types of filter card modes available for dimensions and measures.

Give them all a try!

For dimensions, you can choose from the following filter modes:

- **Single Value (List)**: Displays the values of the filter as a list of radio buttons where only a single value can be selected at a time.
- **Single Value (Dropdown)**: Displays the values of the filter in a drop-down list where only a single value can be selected at a time.
- **Single Value (Slider)**: Displays the values of the filter along the range of a slider. Only a single value can be selected at a time. This option is useful for dimensions that have an implicit order such as dates.
- **Multiple Values (List)**: Displays the values in the filter as a list of check boxes where multiple values can be selected.
- **Multiple Values(Dropdown)**: Displays the values of the filter in a drop-down list where multiple values can be selected.
- **Multiple Values (Custom List)**: Displays a text box where you can type a few characters and search for the value. Alternatively, you can type or paste a list of values into the text box to create a custom list of values to include.
- **Wildcard Match**: Displays a text box where you can type a few characters. All values that match those characters are automatically selected. You can use the asterisk character as a wildcard character. For example, you can type “tab*” to select all values that begin with the letters “tab”. Pattern Match is not case sensitive. If you are using a multidimensional data source, this option is only available when filtering single level hierarchies and attributes.

For measures, you can choose from the following filter modes:

- **Range of Values/Dates**: shows the filtered values as a pair of sliders that you can adjust to include or exclude more values. Click on the upper and lower limit readouts to enter the values manually.

The darker area inside the slider range is called the data bar. It indicates the range in which data points actually lie in the view. Use this indicator to determine a filter that makes sense for the data in your data source. For example, you may filter the Sales field to only include values between \$200,000 and \$500,000 but your view only contains values between \$250,000 and \$320,000. The range of data you can see in the view is indicated by the data bar while the sliders show you the range of the filter.

Note: Data bars only show in filters where the filtered field is also used in the view (e.g., on Columns, Rows, or on the Marks card, and so on) and are at the same aggregation level as the field on the Filters shelf. For example, a filter on SUM (Sales) will only display data bars if the SUM(Sales) field is used in the view. It won't show if AVG(Sales) is used in the view. Even though in both scenarios, the

filtered field, Sales is used in the view; in the latter case the aggregation is different than the aggregation of the filter.

- **At Least/Starting Date:** shows a single slider with a fixed minimum value. Use this option to create a filter using an open ended range.
- **At Most/Ending Date:** shows a slider with a fixed maximum value. Use this option to create a filter using an open ended range.
- **Relative to Now:** this option shows a control where you can define a dynamic date range that updates based on when you open the view. The option is only available for filters on continuous date fields.
- **Browse periods:** shows common date ranges such as past day, week, month, three months, one year, and five years. This option is only available for filters on continuous date fields.

Note: When you expose a filter for Measure Values or Measure Names as a single value list, selecting All will automatically convert the filter to a multiple values list. For information on Measure Values and Measure Names, see [Measure Values and Measure Names](#) on page 302.

Customize filter cards

In addition to the general filter options and the filter modes, you control how your filter appears in the worksheet, on dashboards, or when published to the web even further.

To customize filters, click the filter card drop-down menu and select **Customize**.

You can select from the following options:

- **Show “All” Value** - toggles whether to show the “All” option that displays by default in multiple values and single value lists.
- **Show Search Button** - toggles whether to show the search button at the top of the filter.
- **Show Include/Exclude** - toggles whether to show the Include Values and Exclude Values commands on the filter card menu. When shown, users can switch the filter between include and exclude modes.
- **Show Filter Types** - toggles whether to let users change the type of quick filter shown. For example, when shown, a user can change a multiple values list to a compact list.
- **Show More/Fewer Button** - toggles whether to show the More/Fewer button  at the top of the filter.

- **Show All Values Button** - toggles whether to display the Show All Values button  on the filter card.
- **Show Apply Button** - toggles whether to show the Apply button at the bottom of the filter. When shown, changes to the filter are only applied after you click the button. Pending changes are indicated with a green color. This option is only available in multiple values lists and dropdowns.
- **Show Readouts** - controls whether the minimum and maximum values are displayed as text above a range of values. The readouts can be used to manually type a new value instead of using the sliders.
- **Show Slider** - controls whether the slider displays. When this option is cleared, the filter only displays the readouts.
- **Show Null Controls** - shows a drop-down list that lets you control how the filter handles null values. You can select from values in a range; values in a range and null values; null values only, non-null values, or all values.

Note: Not all of the above options are available for views published to Tableau Server.

See Also:

- [Apply Filters to Multiple Worksheets](#) below
- [Filter Data Across Multiple Data Sources](#) on page 689
- [Improve View Performance with Context Filters](#) on page 694
- [Filter Data from Data Sources](#) on page 434
- [Restrict Access at the Data Row Level](#) on page 1252

[\(Back to top\)](#)

Apply Filters to Multiple Worksheets

When you add a filter to a worksheet, by default it applies to the current worksheet. Sometimes, however, you might want to apply the filter to other worksheets in the workbook.

You can select specific worksheets to apply the filter to or apply it globally to all worksheets that use the same data source or related data sources. For example, you might have a filter that only includes a specific region or product of interest. Rather than adding this filter every time you create a new worksheet, you can simply create the filter once and then apply it to multiple worksheets.

Apply filters to all worksheets that use a related primary data source

This option applies the filter to all worksheets that use related data sources as their primary data source. For more information, see [Filter Data Across Multiple Data Sources](#) on page 689.

To apply a filter to all worksheets using a related primary data source:

- On the Filters shelf, right-click the field and select **Apply to Worksheets > All Using Related Data Sources**.

Filters that use this option are global across the workbook.

Filters that apply to all related data sources are marked with an icon. The filter is automatically created on any existing worksheets, and on any new worksheets you create that use a related data source.

Any changes you make to the filter affects all of those worksheets.

Apply filters to all worksheets that use the current primary data source

This option applies the filter to all worksheets that use the current worksheet's primary data source as their primary data source.

To apply a filter to all worksheets using the current primary data source:

- On the Filters shelf, right-click the field and select **Apply to Worksheets > All Using This Data Source**.

Filters that use this option are global across the workbook.

Filters that apply to all worksheets are marked with a data source icon . The filter is automatically created on any new worksheets you create after you drag a field to the view.

Any changes you make to the filter affects all of those worksheets.

Note: If blending multiple data sources in a view, the **All Using This Data Source** option applies to the filter if it is the primary data source.

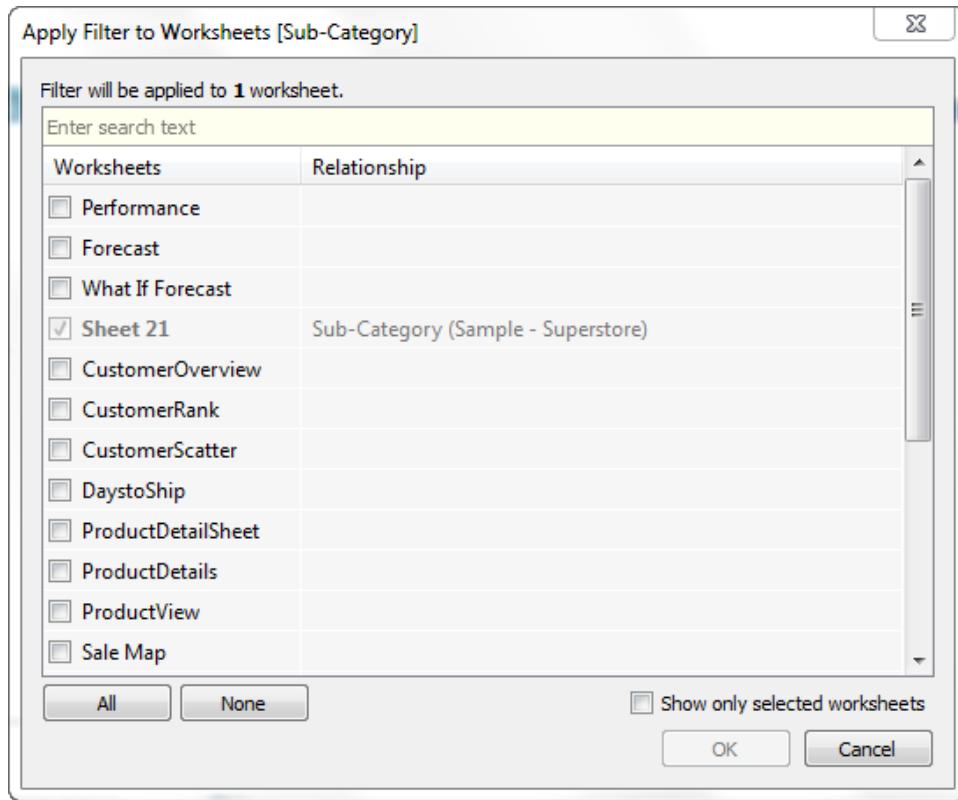
Apply filters to select worksheets

This option opens a dialog box where you can select from a list of worksheets that use the same data source or related data sources.

To apply a filter to select worksheets:

- On the Filters shelf, right-click the field and select **Apply to Worksheets > Selected Worksheets**.
- In the Apply Filter to Worksheets dialog box, select the worksheets that you want to apply the filter. If any of the sheets already contain a filter on the same field, the dialog box will provide details about the filter.

If you select the sheet, the current filter will override any existing filter selections.



Filters that apply to a selection of worksheets are marked with the worksheet icon . Any changes you make to the filter affect all of the selected worksheets.

Apply filters to the current worksheet only

This option only applies to the current worksheet. This option is selected by default when you create new filters. Filters that are local to the current worksheet are shown without any additional icons.

To apply a filter to the current worksheet only:

- On the Filters shelf, right-click the field and select **Apply to Worksheets > Only This Worksheet**.

If you apply a filter to all worksheets or selected worksheets and then change the setting to apply the filter to just the current worksheet, the filters are not removed from all other worksheets. Rather, the filters are disconnected and they are all made local to their respective worksheets. You can go to each worksheet and remove the filter or modify selections.

Filter all worksheets on a dashboard

This option applies the filter to all worksheets in the dashboard that use the same data source or related data sources as their primary data source.

To filter all worksheets on a dashboard:

- In a dashboard, click the drop-down menu on a filter card and select **Apply to Worksheets > Selected worksheets**.
- In the Apply Filter to Worksheets dialog box, click **All on dashboard**, and then click **OK**.

Note: In previous versions of Tableau Desktop, the **All Using This Data Source** option was called Make Global and the **Only This Worksheet** option was Make Local.

Filter Data Across Multiple Data Sources

While working with multiple data sources in a workbook, you might want to compare the data between them using a field they have in common. To do so, you can apply a filter across multiple primary data sources.

For example, the following dashboard shows the order quantity, average sales, and average profit for customers. It has three views. Each of the views uses a different data source as its primary data source, and those data sources all have a field in common: Customer Name.

There is also a filter card in the view for Customer Name.



This is an interesting dashboard with a lot of great information, but you might want to update all of the views in the dashboard at the same time by the customer you're analyzing. For example, maybe you want to see the average sales, profit, and number of orders you've received from one of your customers, Aaron Riggs.

To do so, you can filter all three data sources on the Customer Name field.

Follow the steps below to learn how to filter data across multiple data sources.

[Before you start, you might want to know the following:](#)

In 10.0, you can only filter data across multiple primary data sources. You cannot filter data across secondary data sources. Filtering data across a worksheet's secondary data source is not currently supported in Tableau Desktop.

If you want to filter data across secondary data sources, consider the following alternatives:

- Join tables that are in the same data source, instead of blending them. For more information, see [Join Your Data on page 354](#).
- Create a cross-database join if your tables are in different data sources. For more information, see [Join Your Data on page 354](#).
- Make the primary data source the secondary data source, and the secondary data source the primary. **Note:** This might not always be feasible given the level of detail you want in the final view.

For more FAQs about cross data-source filters, see the [Cross data-source filtering FAQs](#) forum post in the Tableau Community.

[: Define relationships between your data sources](#)

Before you can create relationships between data sources, you must ensure that there is a common field between the data sources you're mapping. The fields do not need to be named the same in each data source, but they should have some data in common.

After you've identified the common fields, you must create relationships between them, or *map the fields* to one another.

If the fields happen to have the same name, Tableau might recognize them as related automatically. You can modify an existing relationship that was created automatically by Tableau, or create a new relationship between two fields in different data sources, by following the procedure below.

To define a relationship between your two data sources:

1. Select **Data > Edit Relationships**.
2. In the Edit Relationships dialog box, do the following:
 - Select a data source for **Primary data source**, and select a data source for **Secondary data source**.

- Select **Custom**.
 - To define a relationship between fields that have different names, click **Add**.
To edit an existing relationship, select the fields on the right, and then click **Edit**.
3. In the Add/Edit Field Mapping dialog box, do the following, and then click **OK**:
- Under **Primary data source field**, select a field.
 - Under **Secondary data source field**, select a field with similar data to the field you selected for the primary data source.
- For more information about editing relationships, see [Step 4: \(Optional\) Define or edit relationships on page 374](#).
- Note:** After you define your relationships, you do not need to enable blends (that is, you do not have to click the link icon in the **Data** pane) to filter across your data sources.
- For more information about the difference between primary and secondary data sources, see [Primary and secondary data sources on page 371](#).

: Add a filter to the view

After you have defined relationships between your data sources, go to one of your worksheets and drag a dimension to the **Filters** shelf. Then select to include or exclude data from the view. For more information about filtering your data, see [Filter Data from Your Views on page 672](#).

Optional Step: Display a filter card in the view. Right-click the field on the **Filters** shelf and select **Show filter**. A filter card for that field appears in the view.

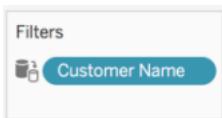
For more information about filter cards (previously known as quick filters), see [Display interactive filters in the view on page 680](#).

: Apply the filter to your worksheets

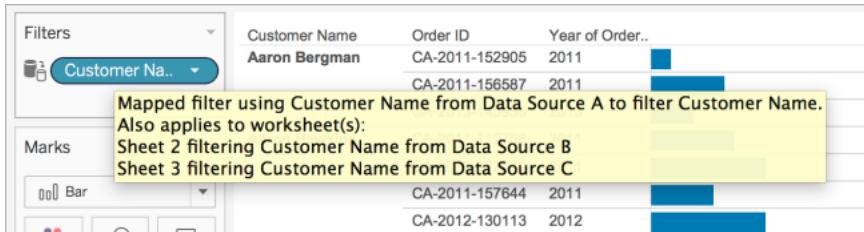
After you set up your filter, you can apply that filter to all the worksheets in your workbook that use related data sources, or you can apply it to selected worksheets.

- To apply the filter to all worksheets that use related data sources, right-click the field on the **Filters** shelf and select **Apply to worksheets > All using related data sources**.

The Customer Name field is added to the **Filters** shelf on every worksheet that uses a related data source. An icon is added next to the field on the **Filters** shelf, indicating that the filter is being applied to multiple data sources.



If you hover over the field on the Filters shelf in any worksheet, you can find details about the type of filter, the source field for the filter, and the other sheets the filter applies to.

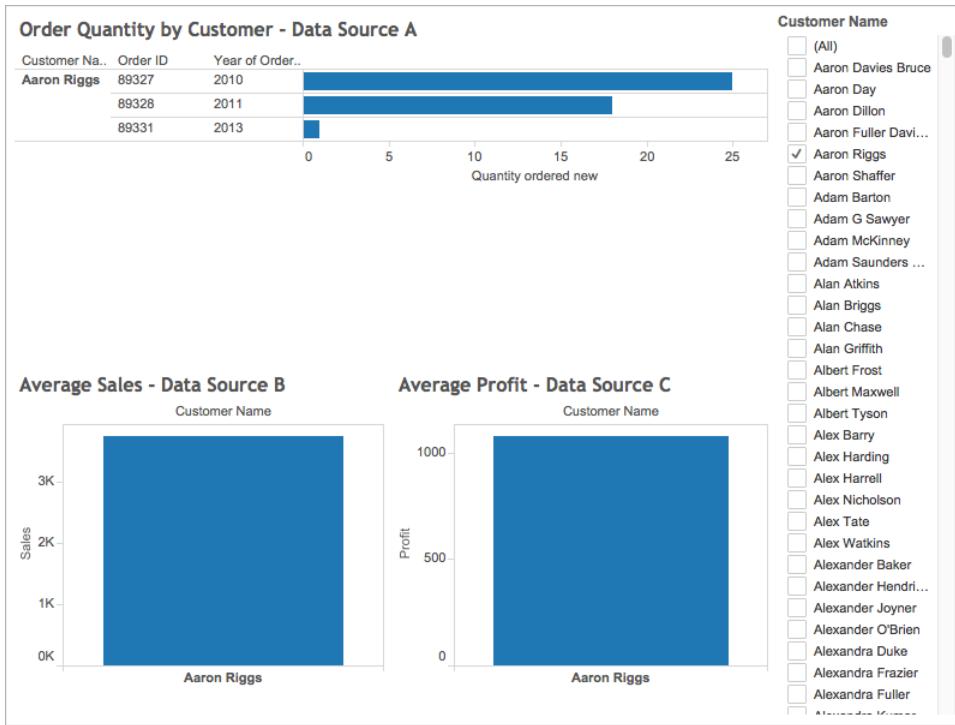


- To apply the filter to selected worksheets, right-click the field on the **Filters** shelf and select **Apply to worksheets > Selected worksheets**.

The Customer Name field is added to the **Filters** shelf on the worksheets you selected. An icon is added next to the field on the **Filters** shelf to indicate that the filter is being applied to select worksheets.



Now, on the dashboard, when you filter the view down to Aaron Riggs, all three views update and you can see that Aaron made orders in 2010, 2011, and 2013, and spent an average of 3,700 USD. The average profit for these orders was approximately 1,000 USD.



Source fields and target fields

When you apply a filter to multiple data sources, you create a source field and one or more target fields. Both source fields and target fields appear on the **Filter** shelf in their respective worksheets.

The source field is the field you're filtering with.

The target field on any given worksheet is a field from another data source that is related to the source field. This is the field that is actually being filtered on that worksheet.

The source field determines the data that is included or excluded from the target fields. For example, let's say you have three worksheets that use three separate data sources (A, B, and C) as their primary data source. Each of the data sources has a field in common (Fruit), and the data is as follows:

Fruit from data source A	Fruit from data source B	Fruit from data source C
Apples	Apples	Apples
Oranges	Grapefruit	Grapefruit
Bananas	Oranges	Oranges

	Pears	Tomatoes
		Bananas

If the Fruit field from data source A is the source field for the cross data source filter, then the data that appears for the target fields is as follows:

Fruit from data source A - Source field	Fruit from data source B - Target field	Fruit from data source C - Target field
Apples	Apples	Apples
Oranges	Oranges	Oranges
Bananas		Bananas

Any data that does not match the data in the source field is excluded from the target fields, and will not appear in your worksheets or in your filter cards.

On the Filters shelf, source fields are indicated with either a  icon when the source field applies to all worksheets with a related data source, or a  icon when it applies to select worksheets. Target fields are indicated with a  icon on the field on the Filters shelf. They are also indicated with a  or  icon next to the field on the Filters shelf.

Improve View Performance with Context Filters

By default, all filters that you set in Tableau are computed independently. That is, each filter accesses all rows in your data source without regard to other filters. However, you can set one or more categorical filters as context filters for the view. You can think of a context filter as being an independent filter. Any other filters that you set are defined as dependent filters because they process only the data that passes through the context filter.

You may create a context filter to:

- Improve performance – If you set a lot of filters or have a large data source, the queries can be slow. You can set one or more context filters to improve performance.

- Create a dependent numerical or top N filter – You can set a context filter to include only the data of interest, and then set a numerical or a top N filter.

For example, suppose you're in charge of breakfast products for a large grocery chain. Your task is to find the top 10 breakfast products by profitability for all stores. If the data source is very large, you can set a context filter to include only breakfast products. Then you can create a top 10 filter by profit as a dependent filter, which would process only the data that passes through the context filter.

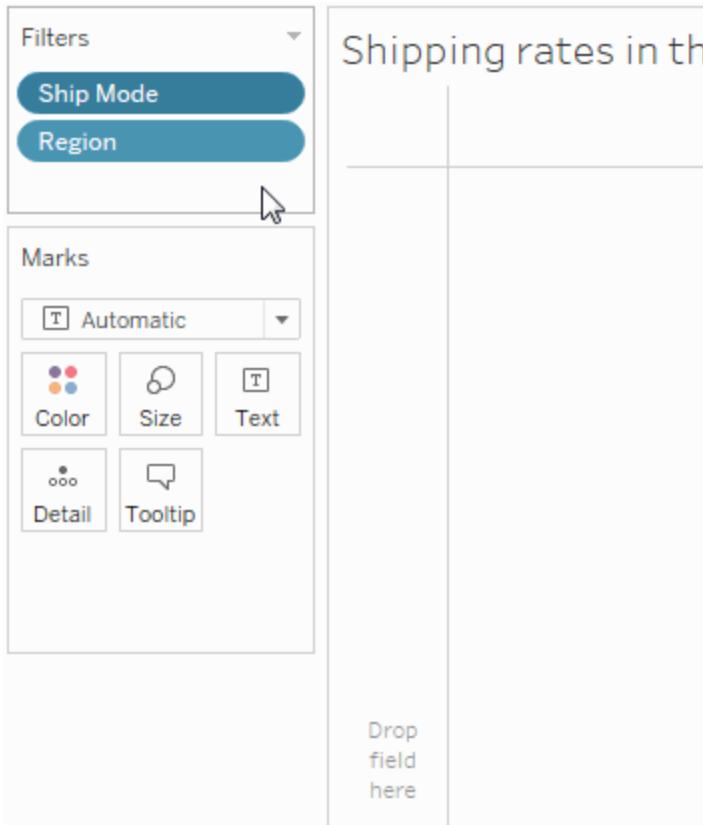
Note: As of Tableau 9.0, context filters no longer create temporary tables, except for generic ODBC data sources and customized data sources.

Create Context Filters

To create a context filter, select **Add to Context** from the context menu of an existing categorical filter. The context is computed once to generate the view. All other filters are then computed relative to the context. Context filters:

- Appear at the top of the Filters shelf.
- Are identified by a gray color on the Filters shelf.
- Cannot be rearranged on the shelf.

As shown below, the **Ship Mode** dimension is set to be the context for a view. The **Region** filter is computed using only the data that passes through **Ship Mode**.



You can modify a context filter by:

- Removing the field from the Filters shelf – If other context filters remain on the shelf, a new context is computed.
- Editing the filter – A new context is computed each time you edit a context filter.
- Selecting **Remove from Context** – The filter remains on the shelf as a standard filter. If other context filters remain on the shelf, a new context is computed.

Speed up Context Filters

To improve performance of context filters, especially on large data sources, follow these general rules.

- Using a single context filter that significantly reduces the size of the data set is much better than applying many context filters. In fact, if a filter does not reduce the size of the data set by one-tenth or more, it is actually worse to add it to the context because of the performance cost of computing the context.
- Complete all of your data modeling before creating a context. Changes in the data model, such as converting dimensions to measures, require recomputing the context.
- Set the necessary filters for the context and create the context before adding fields to

other shelves. Doing this work first makes the queries that are run when you drop fields on other shelves much faster.

- If you want to set a context filter on a date you can use a continuous date. However, using date bins like YEAR(date) or context filters on discrete dates are very effective.

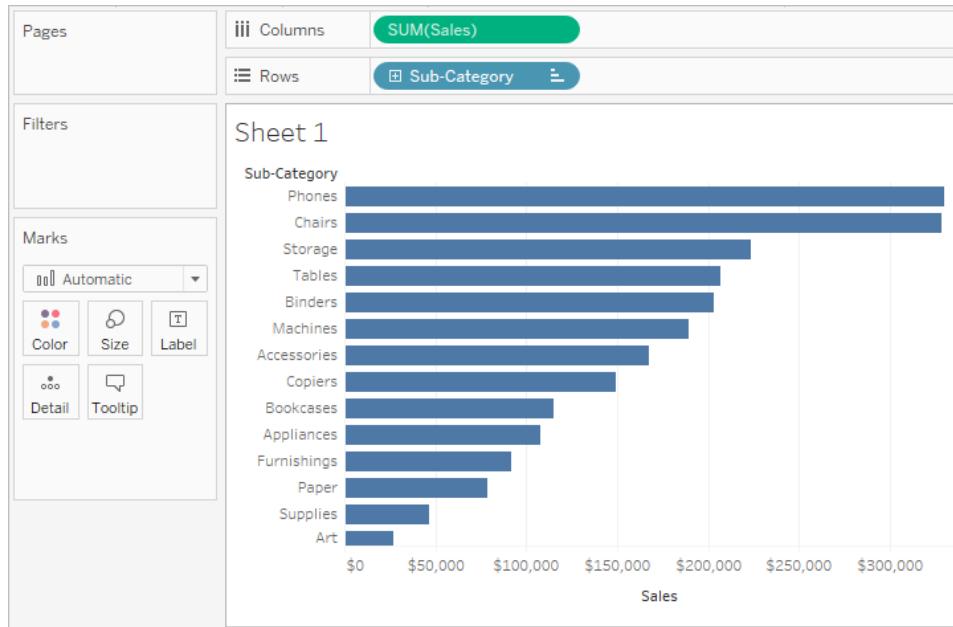
If your data set is heavily indexed, context filters may not provide performance improvement and may actually cause slower query performance.

Context filters can adversely affect any query performance improvements when using the **Include joined tables only when referenced** option in the Tables dialog box. See the note at the bottom of [Join Your Data on page 354](#).

Example – Create Context Filters

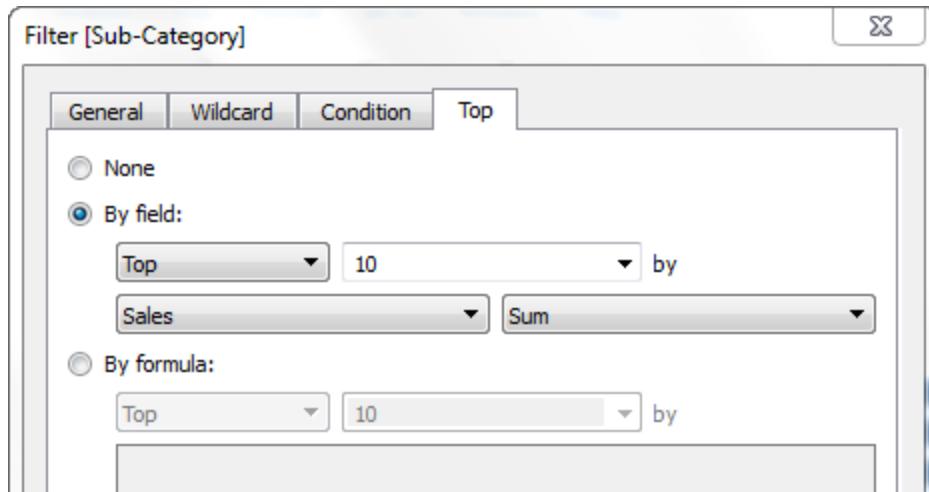
This example walks you through how to create a context filter. First you'll filter a view to show the top 10 products by sales. Then you'll create a context filter on product category so you can see the top 10 furniture products.

1. Use the **Sample - Superstore** data source to create the initial view shown below. The view shows the sales for all sub-categories, sorted with the highest sale at the top.

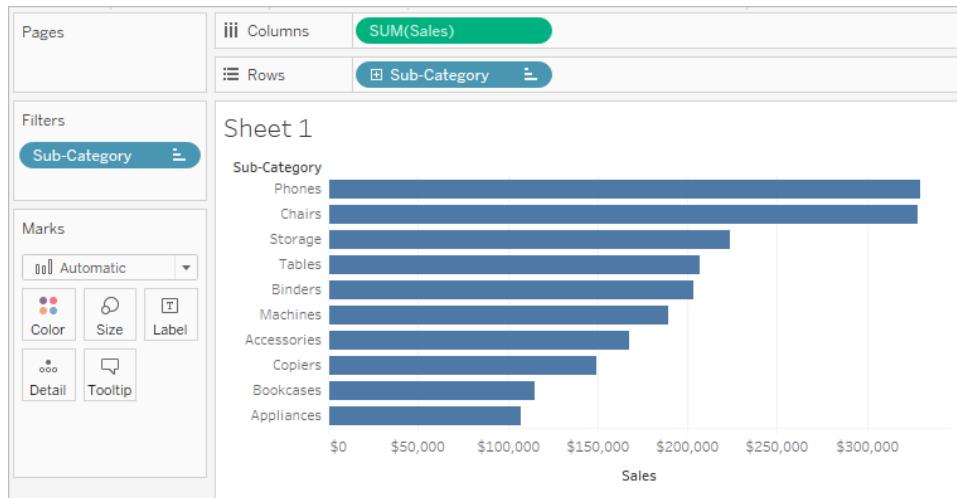


2. Now create a Top 10 filter to just show the top selling products. You can create this filter by dragging the **Sub-Category** field to the Filters shelf. In the Filter dialog box, switch to the **Top** tab and define a filter that is Top 10 by Sum of Sales. See [Filter Data from](#)

Your Views on page 672 to learn more about defining a Top N filter.

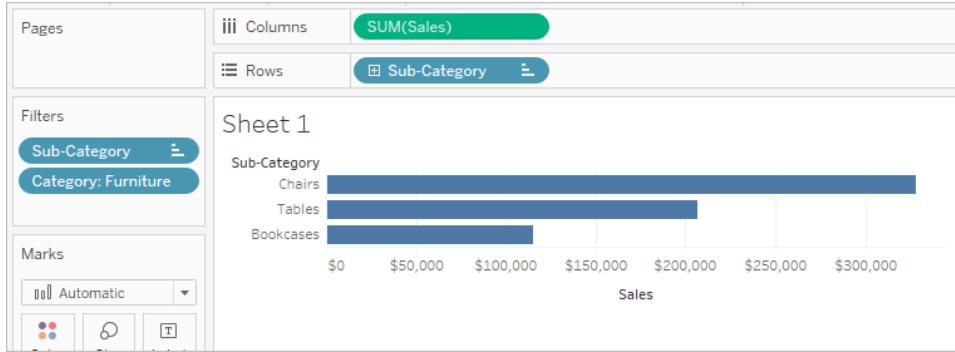


- When you click **OK**, you'll see that the view is filtered to show the top 10 product sub-categories in terms of sales.

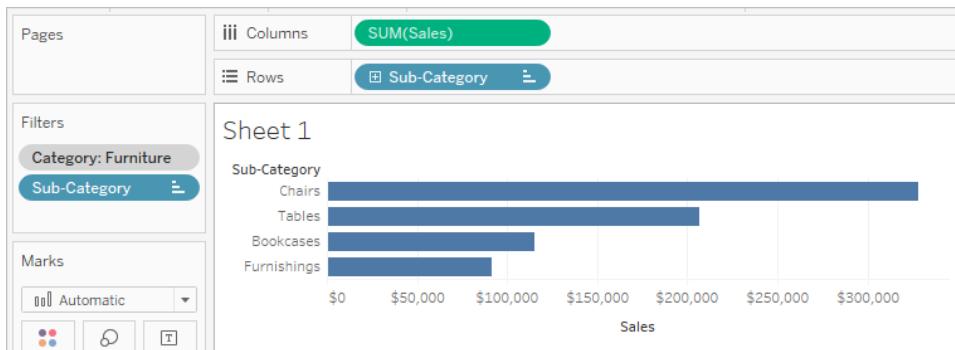


- Now, let's add another filter to show only furniture products. Drag the **Category** field to the Filters shelf and select only **Furniture**. When finished, click **OK**.

The view is filtered but instead of 10 products, it now shows 3. This is because by default all filters are evaluated separately and the view shows the intersection of the results. So this view shows that three of the top 10 overall products are furniture products.



5. To find out what the top 10 furniture products are we need to make the Category filter a context filter. Right-click the field on the Filters shelf and select **Add to Context**.
6. The filter is marked as a context filter and the view updates to show the top four furniture products. Why not 10? Because only four of the sub-categories contain furniture. But we now know that the Top 10 filter is being evaluated on the results of that context.



Create Slicing Filters

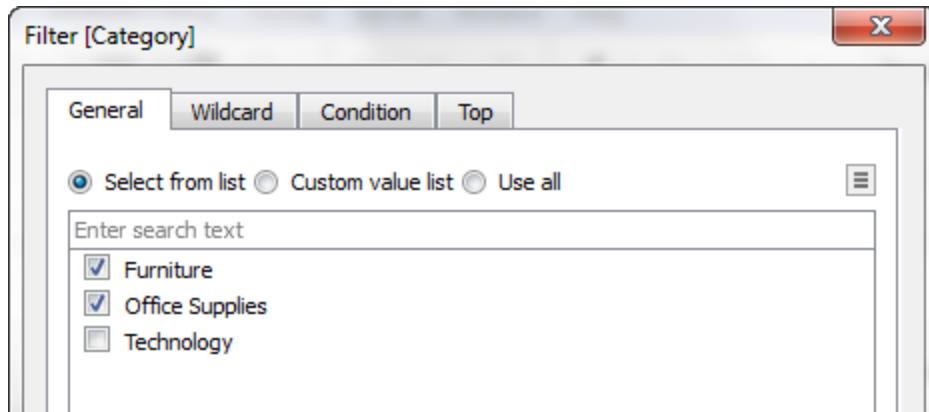
Slicing filters, also known as slicers or calculation filters, are dimension filters that behave differently for multidimensional data sources than they do for relational data sources.

In Tableau, multidimensional data sources are supported only in Windows.

A slicing filter exists when:

- The dimension that you place on the Filters shelf is not also on any other shelf in the view.

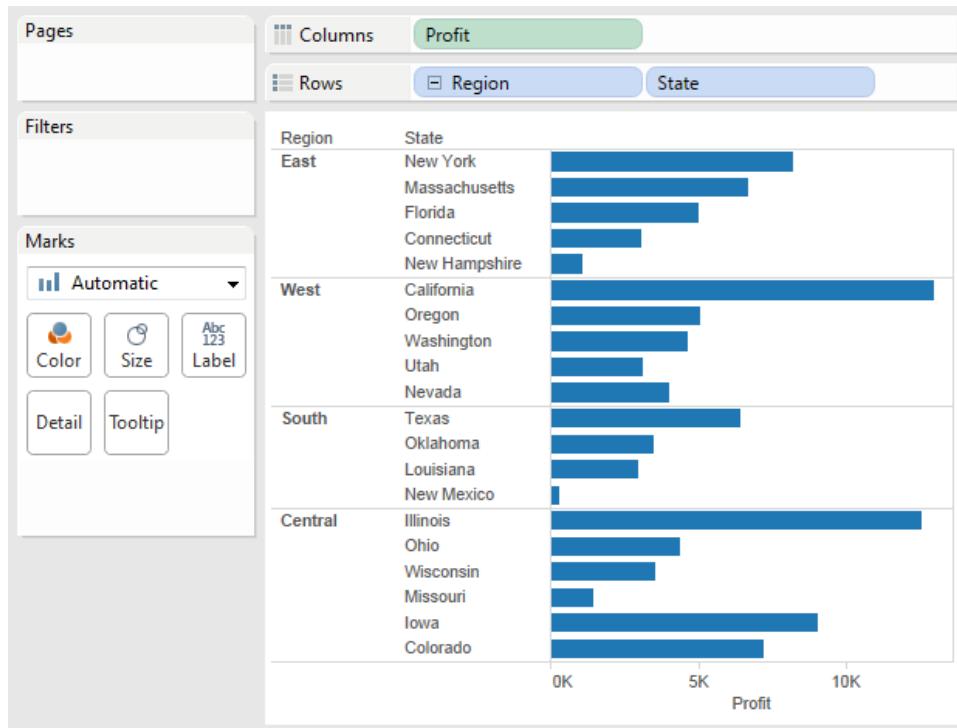
- The filter is defined to include multiple values. For example:



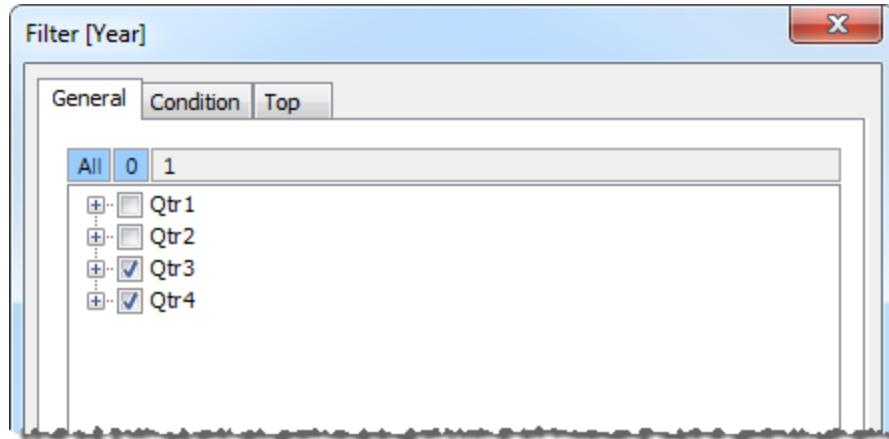
For relational data sources, there is nothing remarkable about a slicing filter. Tableau recalculates values for each measure in the view using the configured aggregations.

But because multidimensional data sources contain data that is aggregated in the data source, no aggregation can be defined for the “slice” that the dimension filter cuts through the values of individual measures. Tableau therefore automatically performs a summation. And because multiple values are defined, the result is the sum of a group of sums.

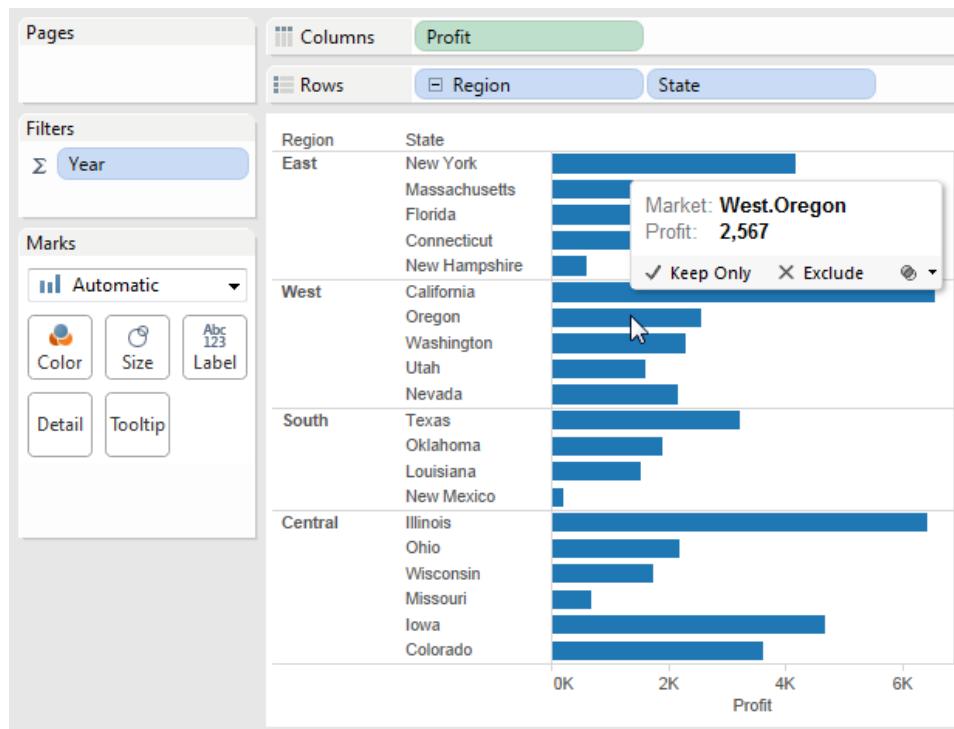
For example, consider the view shown below, which shows profit by region and state.



A slicing filter that includes the Q3 and Q4 members of the Quarter dimension is then applied to the data.



In the resulting view, the profit for Oregon in the West region is 2,567. This number was calculated by summing data values for Qtr3 and Qtr4.



The summation symbol (Σ) you see on the Filters shelf in the view above indicates that this is a slicing filter. If the original aggregation for a field is non-additive (for example, Count Distinct) the values you see after applying a slicing filter may be unexpected.

Build and Use Maps

Authoring maps in Tableau is as easy as dragging and dropping a field onto the view. Whether you're plotting disease outbreaks, tracking bird migrations, or analyzing public transportation usage, you can put your data on a map in Tableau.

All you need is some location data.

Here are some resources to help you bring that data into Tableau and plot it on a map view:

Bring location data into Tableau

- Convert shapefile data for use in Tableau
- Custom geocode your data
- Blend geographic data

Build a map view

- Format geographic fields in Tableau
- Build a map view

Customize your map view

- Create custom territories
- Customize the appearance of your map
- Customize how people interact with your map
- Use WMS (Web Map Service) Servers
- Use Mapbox Maps

Interact with Maps

- Explore data in maps
- Search for locations in your map
- Measure distances in your map

For a complete list of all topics about maps, see below:

Create Tableau Maps from Shapefiles

If you work with shapefile data, you can use a Geographic Information System (GIS) such as QGIS or ArcGIS to convert the data into a text file that you can import into Tableau Desktop. You can then connect to that text file and build a map view in Tableau.

In this topic:

- [Create Tableau polygons from shapefiles using QGIS](#) below
- [Add point data from shapefiles using QGIS](#) on page 713
- [Create Tableau polygons from shapefiles using ArcGIS](#) on page 715

Create Tableau polygons from shapefiles using QGIS

If you have shapefile data, you can use QGIS Desktop to convert it to a CSV file for use in Tableau Desktop.

For example, if you have shapefile data for U.S. counties, you can use QGIS Desktop to convert the shapefile to CSV files that you can then connect to in Tableau Desktop. You can create Tableau polygons for all the counties in the United States using the data in the CSV files.

Follow the steps below to learn how to use QGIS Desktop to convert shapefile data to text files and then create Tableau polygons.

Before you start

Before you follow the steps in this section, the coordinates in your shapefile should already be in the form of decimal latitude and longitude units, specifically using EPSG: 4326, WGS 84.

Additionally, this procedure requires you to use QGIS Desktop, an open source GIS that supports many different shapefile formats. To download this software, go to the [QGIS website](#).

Note: If you are using Windows, install QGIS Standalone Installer Version 2.14 (64-bit).

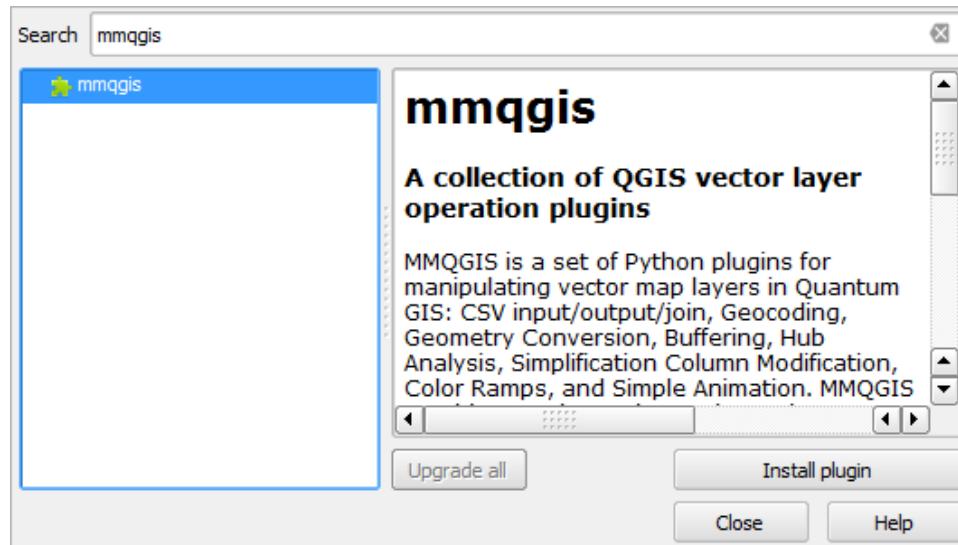
Disclaimer: Clicking this link will take you away from the [Tableau](#) website. Although we make every effort to ensure these links to external websites are accurate, up to date, and relevant, Tableau cannot ensure the accuracy or freshness of pages maintained by external providers. Contact the external site for answers to questions regarding its content.

: Download shapefile data

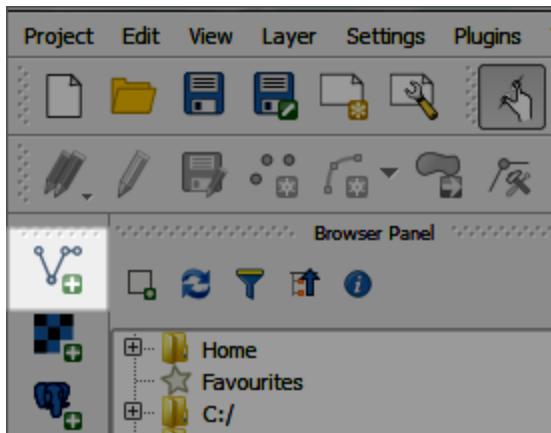
1. Go to the [United States Census Bureau website](#) and click `cb_2015_us_county_500k.zip` to download the file.
2. Extract the shapefiles to a folder on your computer. In this example, the folder is named US Counties and is saved on the Desktop.

: Use QGIS Desktop to convert shapefiles to CSV file format

1. Open QGIS Desktop.
2. Select **Plugins > Manage and Install Plugins**.
3. In the Plugins dialog box, do the following:
 - In the search bar at the top, search for **mmqgis**.
 - Select the **mmqgis** plugin that appears, and click **Install plugin**.

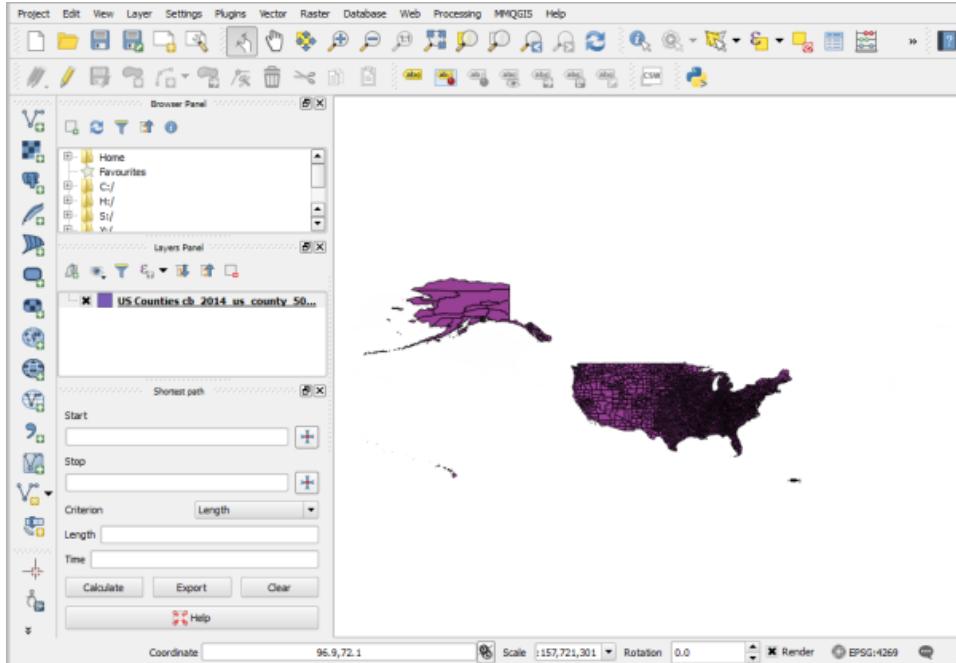


- Click **Close**.
4. On the left side of the workspace, click the **Add Vector Layer** icon.



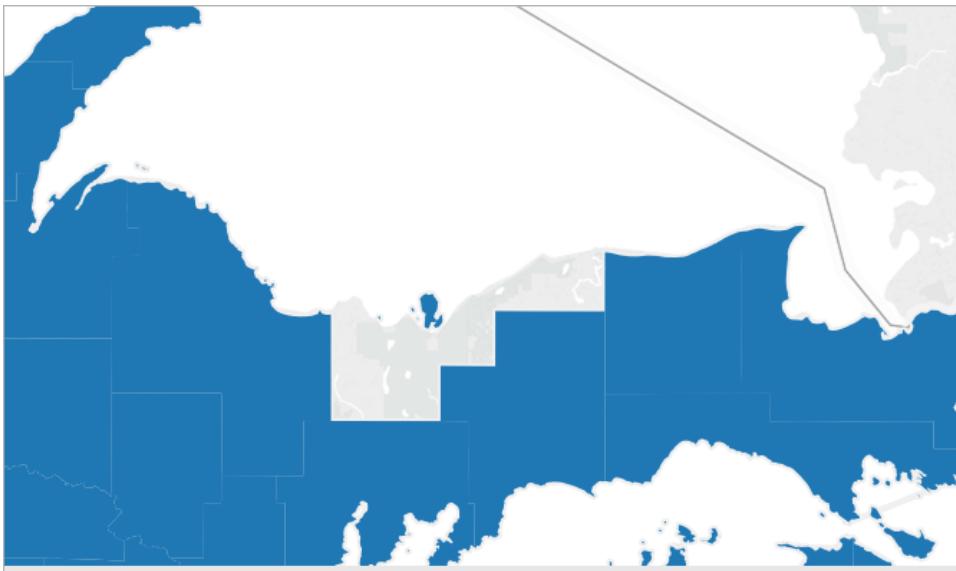
5. In the Add vector layer dialog box, do the following:

- Under **Source type**, click **Directory**.
 - Under **Source**, click **Browse**, and then navigate to and select the **US Counties** folder.
 - Click **Open**.
6. If the Select vector layers to add dialog box appears, select **cb_2015_us_county_500k**, and then click **OK**.
- The polygons in the shapefile are plotted on the canvas and a layer is added to the Layers Panel.



7. Select **Vector > Geometry Tools > Multipart to Singleparts**.

This breaks any multiple polygons in the shapefile into single polygons so that Tableau can read and plot them properly on a map view. If you skip this step, certain locations in your map view might not plot correctly because Tableau does not recognize that they might be separate polygons. For example, in the image below, some of the Wisconsin counties along Lake Superior are not plotted properly.



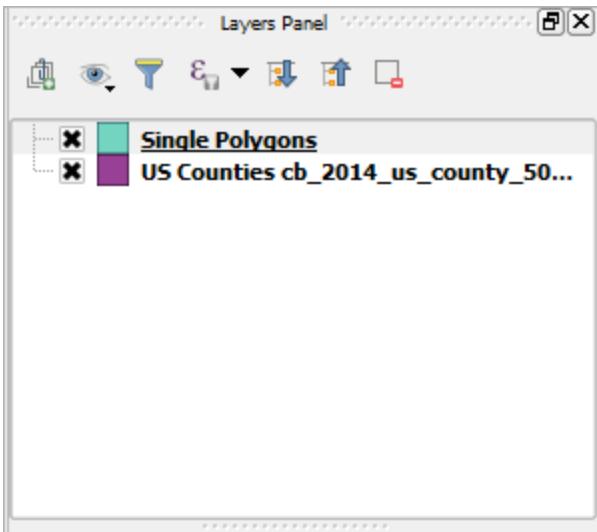
8. In the Multipart to singleparts dialog box, do the following:

- For **Input line or polygon vector layer**, make sure **US Counties cb_2015_us_county_500k any** is selected.
- For **Output shapefile**, click **Browse**.

In the new dialog box that opens, name the file **Single Polygons** and save it on your computer.

- Click **OK**, and then click **Close** to return to the QGIS workspace.

The Single Polygons layer is added to the Layers Panel.



9. Select **MMGIS > Import/Export > Geometry Export to CSV File**.

10. In the Export Geometry to CSV dialog box, do the following:

- For **Source Layer**, make sure that **Single Polygons** is selected.
- For **Output Nodes CSV File**, click **Browse**.

In the subsequent dialog box that opens, name the file **nodes.csv** and save it to a location on your computer.

- For **Output Attributes CSV File**, click **Browse**.

In the subsequent dialog box that opens, name the file **attributes.csv** and save it in the same location that you saved the **nodes.csv** file.

- For **Line Terminator**, select **LF**. This ensures that any spaces are removed from cells in the CSV file.
- Click **OK**.

The two CSV files, **nodes.csv** and **attributes.csv**, are saved to the location you chose.

: Prepare the Nodes CSV file for Tableau

1. Open the **nodes.csv** file you saved.

Note: The CSV file used in this example contains approximately 1,037,335 rows of data. Due to worksheet size limits, you might not be able to open this file in older versions of Microsoft Excel.

2. Add a new column called **Path**, which lists a sequence of numbers from **1** to however many records are in the file.

For this example, there are 1048575 records. For shortcuts on adding a sequence of numbers in Microsoft Excel, see the [Microsoft Office Support website](#).

	A	B	C	D	E
1	shapeid	x	y	Path	
2	0	-86.9212	32.65754	1	
3	0	-86.9204	32.65856	2	
4	0	-86.9204	32.66008	3	
5	0	-86.9176	32.66417	4	
6	0	-86.9146	32.66436	5	
7	0	-86.875	32.66253	6	
8	0	-86.8739	32.6626	7	
9	0	-86.8166	32.66012	8	
10	0	-86.7715	32.66072	9	
11	0	-86.7266	32.66158	10	
12	0	-86.7134	32.66173	11	
13	0	-86.7135	32.66622	12	
14	0	-86.7142	32.70569	13	
15	0	-86.6537	32.70619	14	
16	0	-86.6262	32.70638	15	
17	0	-86.625	32.70639	16	
18	0	-86.6206	32.70626	17	

3. Save and close the file.

: Connect to CSV files in Tableau and prepare your fields for creating map views

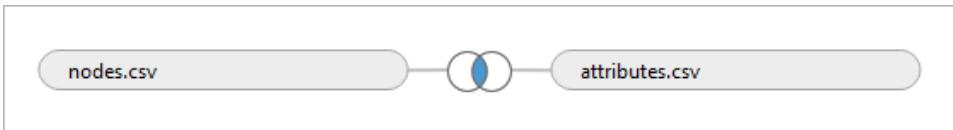
1. Open Tableau Desktop.
2. On the **Connect** section of the start page, click **Text File**.
3. Select the **nodes.csv** file that you created with QGIS.

The **nodes.csv** file is added to the top of the canvas on the data source page. If you saved the **nodes.csv** and **attributes.csv** files in the same place, they both appear in the left pane.

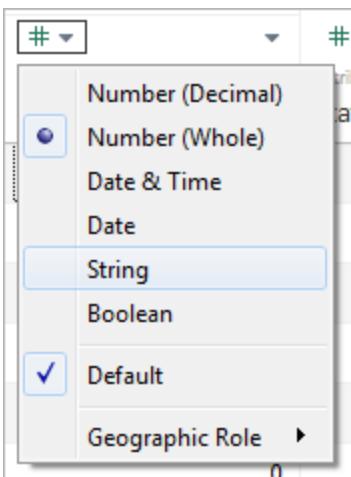
The screenshot shows the Tableau Data Source pane for a connection named "nodes". The status bar at the top says "Connected to Text File". The "Directory" section shows "C:\Users\Desktop". The "Files" section contains an input field "Enter file name" and two listed files: "attributes.csv" and "nodes.csv".

- From the left pane, drag `attributes.csv` to the top of the canvas, to the right of the `nodes.csv` file.

Tableau creates an inner join between the two data sources because they have a field in common: **Shapeid**.



- In the grid on the bottom half of the workspace, click **Update Now** to view your data.
- In the grid, for the **Shapeid (Attributes)** and **Shapeid** columns, click the data type icon and select **String**.



Important: If you do not change both **Shapeid** columns to be string data types, the join between the two data sources breaks.

7. In the grid, for the **X** column, click the data type icon and select **Geographic role > Longitude**.
8. For the **Y** column, click the data type icon and select **Geographic role > Latitude**.
9. Click **Sheet 1**.

: Build the map view

1. From the **Measures** pane, under nodes#csv, drag **Shapeid(Count)** and **Path** up to the **Dimensions** part of the **Data** pane.

This converts the two fields from measures to dimensions and ensures that Tableau treats these fields as discrete locations.

Data

Analytics

Nodes+

Dimensions

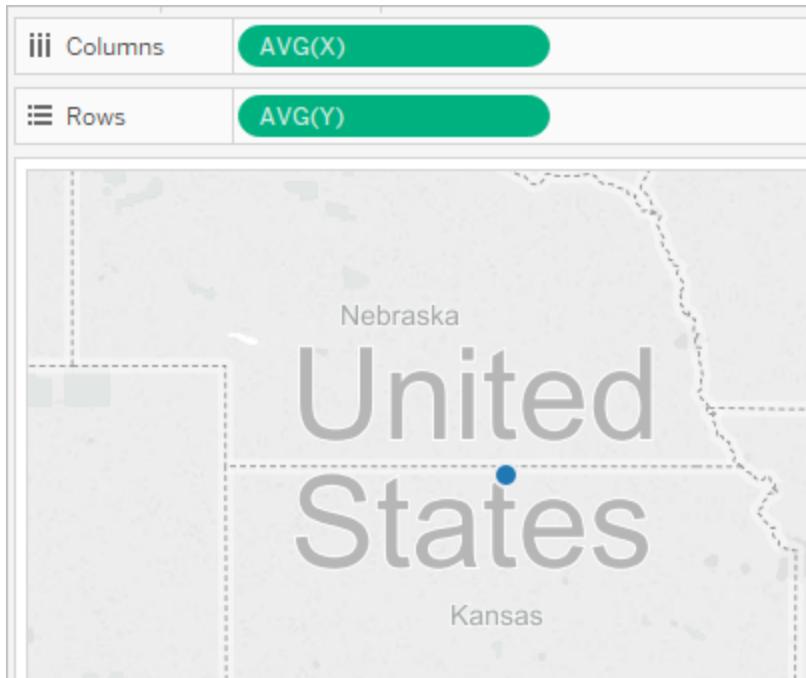
Affgeoid
Name
Measure Names

Measures

- (attributes#csv)
 - Aland
 - Awater
 - Countyfp
 - Countyns
 - Geoid
 - Lsad
 - Shapeid (Attributes....)
 - Statefp
- (nodes#csv)
 - Path
 - Shapeid (Count) (Disti...)
 - X
 - Y
 - Number of Records
 - Measure Values

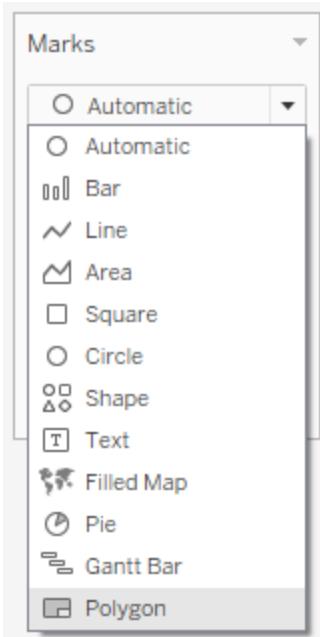
- From **Measures**, under nodes#csv, drag the **X** field to the **Columns** shelf, and the **Y** field to the **Rows** shelf.

Tableau adds an aggregation to each field on the **Columns** and **Rows** shelves and creates a map view with one mark.



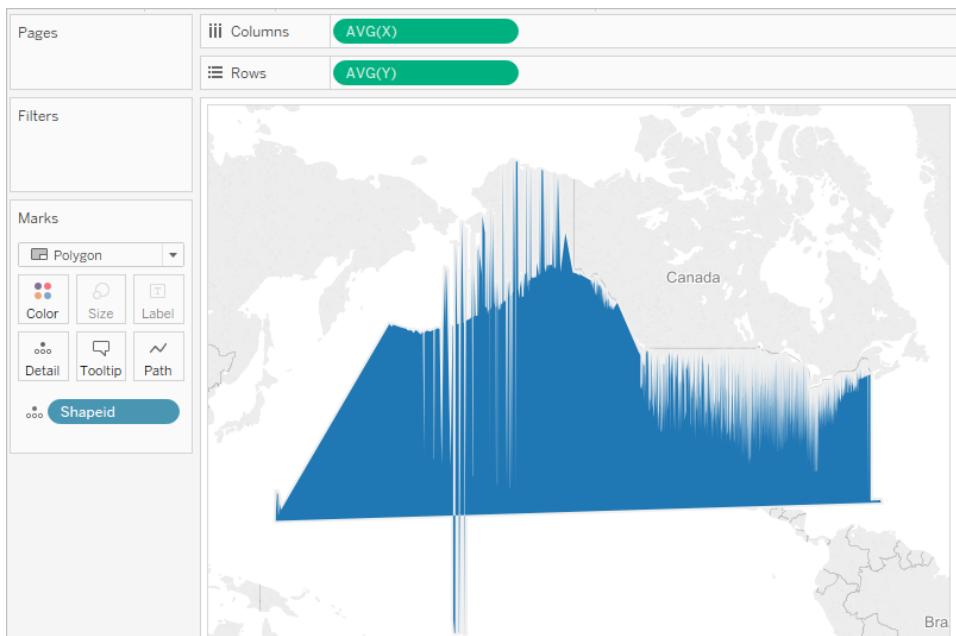
- On the **Marks** card, click the **Mark Type** drop-down arrow and select **Polygon**.

This ensures that you can create a polygon map, instead of a point data map.



4. From **Dimensions**, drag **Shapeid** to **Detail** on the **Marks** card.

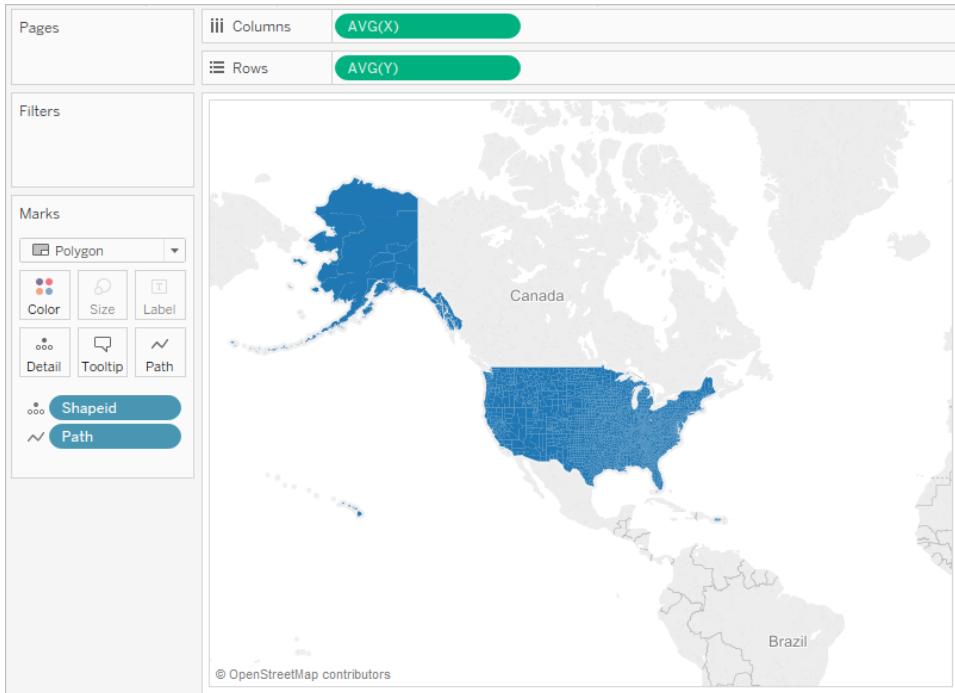
The view temporarily updates to look like the following. Next, you tell Tableau where and in which order to plot each of the polygons.



5. From **Dimensions**, drag the **Path** field to **Path** on the **Marks** card.

Note: If you do not see the Path option on the Marks card, check that the mark-type is set to **Polygon**, not Automatic or Filled Map. See [Step 3](#).

The view updates to a legible map of the United States, with a separate polygon for every county. By adding the field **Path**, which is the column you manually created in the `nodes.csv` file, you tell Tableau which order to draw the polygons on the map.



You can zoom in and out of the map view, adjust the color of the marks, and even customize the background map. You can also blend additional data, such as sales data, to one of the CSV files to perform additional analyses on your location data. For more information about blending geographic data, see [Data Blending vs. Custom Geocoding on page 725](#).

Add point data from shapefiles using QGIS

You can also use QGIS Desktop to convert point data from shapefile to CSV file format, and then add that point data on a map in Tableau.

Follow the steps below to learn how to use QGIS Desktop to convert shapefile data to text files and then use that data to create a point data map in Tableau. For information on how to download QGIS Desktop, see the [Before you start on page 703](#) section.

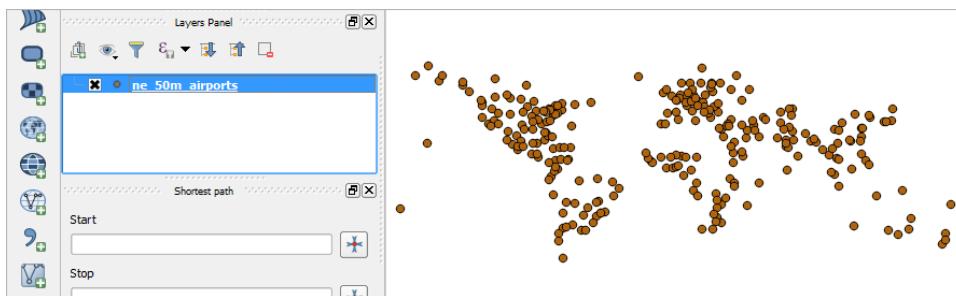
: Download shapefile data

1. Go to the [Natural Earth website](#) and download the Airports data.
2. Extract the shapefiles to a folder on your computer. In this example, the folder is named Airports and is saved on the Desktop.

: Use QGIS Desktop to convert shapefiles to CSV file format

1. Open QGIS Desktop.
2. On the left side of the workspace, click the **Add Vector Layer** icon.
3. In the Add vector layer dialog box, do the following:
 - Under **Source type**, click **Directory**.
 - Under **Source**, click **Browse**, and then navigate to and select the airports folder.
 - Click **Open**.

The points in the shapefile are plotted on the canvas and a layer is added to the Layers Panel (highlighted in blue).



4. In the Layers Panel, right-click the airports layer, then click **Save As**.
5. In the Save vector layer as dialog box, do the following:
 - For **Format**: Select **Comma Separated Value**.
 - Click **Browse** to select the location where you want to save the file, and then click **OK**.

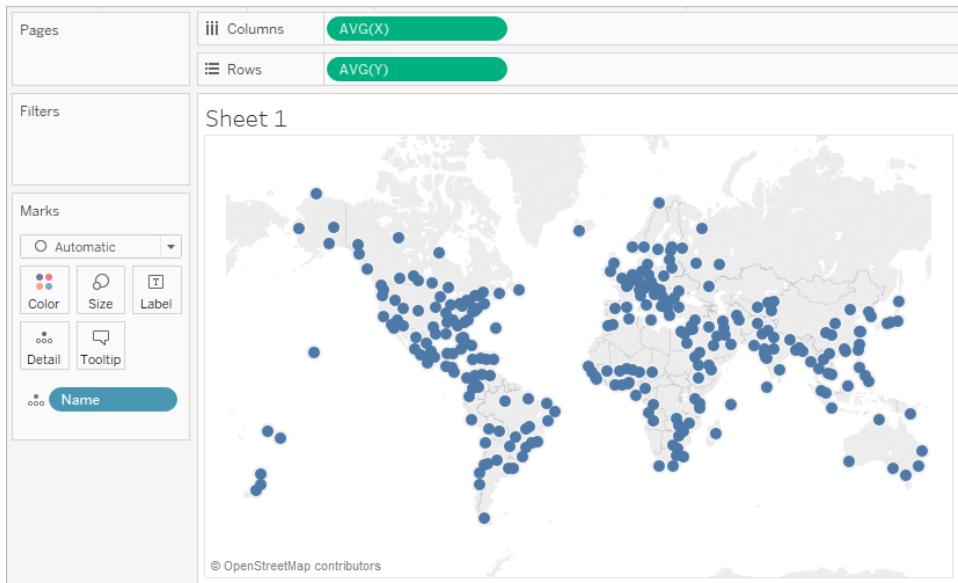
: Connect to the CSV file in Tableau and build the map view

1. Open Tableau Desktop.
2. On the **Connect** section of the start page, click **Text File**.
3. Select the airports CSV file that you created with QGIS.
The `airports.csv` file is added to the top of the canvas on the Data Source page.
4. In the grid, for the **X** column, click the data type icon and select **Geographic role >**

Longitude.

X	Y	Scalerank	Featurecla	Type	Name
113.935	22.315	2	Airport	major	Hong Kong Int'l
121.231	25.077	2	Airport	major	Taoyuan
4.764	52.309	2	Airport	major	Schiphol

5. For the **Y** column, click the data type icon and select **Geographic role > Latitude**.
 6. Click **Sheet 1**.
 7. On **Measures**, double-click the **X** (Longitude) and **Y** (Latitude) fields to add them to the **Columns** and **Rows** shelves.
- A map with a single point appears.
8. From **Dimensions**, drag **Name** to **Detail** on the **Marks** card.
- Every airport listed in the **Name** field is displayed as a point on the map because each airport corresponds to a latitudinal and longitudinal value.



Create Tableau polygons from shapefiles using ArcGIS

You can also use ArcGIS to convert shapefiles to a text file format that can be imported into Tableau Desktop as a polygon layer.

Note: The ArcGIS method requires a licensed version of ArcGIS (ArcView, ArcEditor, or ArcInfo) and the **ET GeoWizards tool for ArcGIS**. The software installs as a toolbar in ArcMap and the functionality required for preparing the shapefile is covered under their free features.

: Prepare the shapefile

1. In an ArcMap view that contains the polygon shapefile layer of interest, verify that the polygon is in a geographic coordinate system using decimal latitude and longitude units, such as GCS NAD 1983. If it is in a projected coordinate system, use ArcToolbox to project the shapefile to a geographic coordinate system.
2. Launch the ET GeoWizards tool in ArcMap and follow the step-by-step wizard to output a **Polygon to Point** feature. Be sure that the output points are the **Vertices** only.

: Prepare the Tableau data source

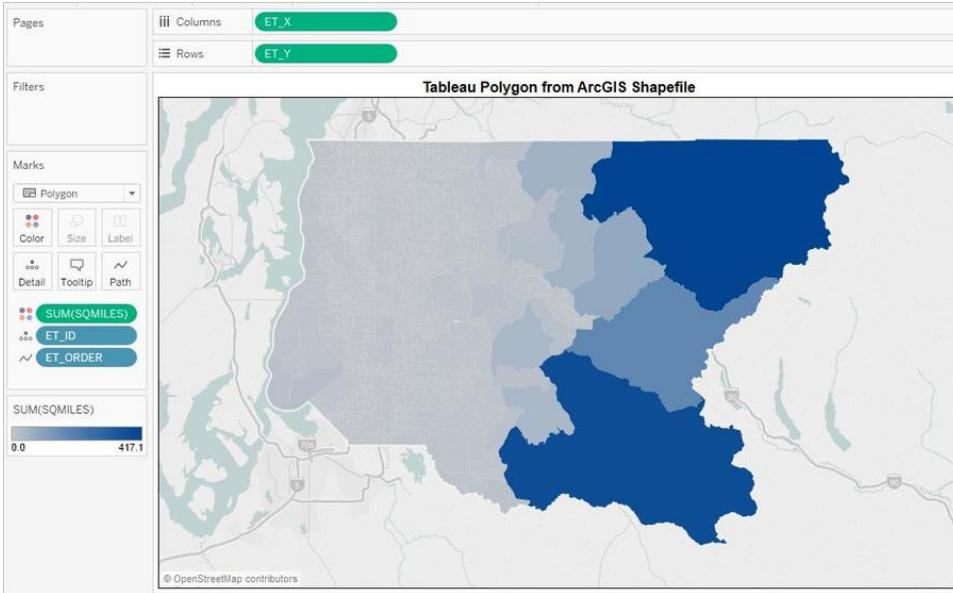
1. Open the .DBF of the output shapefile into a program such as Excel. At a minimum, the file must contain the following fields for correct import into Tableau:
 - **[ID] or [ET_ID]** Either of these two fields can be used for the **Level of Detail** shelf.
 - **[ET_X]** This field corresponds to the longitude coordinate of the record.
 - **[ET_Y]** This field corresponds to the latitude coordinate of the record.
 - **[ET_ORDER]** This field contains the draw order of the points and is used in the **Path** shelf for the polygon.

You can change the header names here to be more informative to the Tableau user.

Save the updated file to a data source supported in Tableau, such as in Excel or Access format.

ET_ID	ET_X	ET_Y	ET_ORDER
2	-122.75632	48.99688	0.9938
2	-122.75604	48.99951	0.9965
3	-123.09055	49.00198	0.0000
3	-123.03539	49.00215	0.3163
3	-123.03018	48.99286	0.3774
3	-123.02784	48.98869	0.4049
3	-123.02236	48.97890	0.4692
3	-123.02158	48.97752	0.4783
3	-123.02146	48.97730	0.4797
3	-123.02152	48.97727	0.4801
3	-123.02809	48.97394	0.5224
3	-123.04097	48.97731	0.5987
3	-123.05199	48.97623	0.6622
3	-123.06072	48.97539	0.7125
3	-123.06295	48.97546	0.7253
3	-123.06400	48.97549	0.7313
3	-123.08383	48.97614	0.8451
3	-123.08395	48.97790	0.8553
3	-123.08450	48.98654	0.9049
3	-123.08814	48.99583	0.9521
4	-121.52410	48.99896	
4	-121.39554	48.99985	

2. Refer to the remaining steps in the [Creating Polygon-Shaded Maps](#) article to complete the view you want in Tableau.



Disclaimer: This information refers to a third-party product as an example. This example is not an endorsement of this product over any other competing products.

Please note that while we make every effort to keep references to third-party content accurate and up to date, the information we provide here might change without notice as the third-party product changes.

Custom Geocode Your Data

Tableau recognizes country names, state/province names, city names, and area codes for many countries world wide. However, there are some locations that Tableau does not recognize immediately.

If you have location data that Tableau doesn't recognize, you can custom geocode your data so that you can use it to create map views in Tableau with ease. Custom geocoding your data allows you to create custom geographic roles that you can use as you create map views in Tableau.

For example, if your data contains country, state/province, and street address data, Tableau Desktop will geocode your data to the country and state/province level, but will not recognize the street address data. In this case, you can create a custom geographic role for the street address data so that you can plot it on a map view in Tableau.

Follow the steps below to learn how to custom geocode your data.

Here's what you need to do:

1. Create one or more CSV files with your location data
2. Create a schema.ini file (optional)
3. Import the files into Tableau
4. Assign geographic roles to your fields

Before you start: Custom geocoding your data can be tricky. We recommend that you check out the [Prepare Your Geographic Fields](#) on page 735 topic to learn more about geographic roles and the type of data Tableau recognizes before you get started.

: Create a CSV file with your location data

The first step to custom geocoding your data is creating a CSV file to import into Tableau. The contents of this file differ depending on whether you are [extending an existing geographic hierarchy](#), [adding a new geographic role](#), or [adding a new hierarchy](#) to Tableau.

However, the following applies to all custom geocoding files:

- The .csv file must contain Latitude and Longitude columns.
- The latitude and longitude values you add must be real numbers. Make sure to include at least one decimal place when specifying these values.

Extend an Existing Role

The built-in geographic roles in Tableau contain hierarchies that can be extended to include locations relevant to your data. For example, the existing hierarchy of Country > State/Province may not contain all of the states or provinces in your data. You can extend this level to include missing states or provinces.

The import file for this type of geocoding should contain every level of the hierarchy above the level you are extending. For example, if you are extending (adding data to) the State/Province geographic role, the existing State/Province hierarchy has a column for Country and State/Province, along with Latitude and Longitude.

	A	B	C	D
1	Country (Name)	State/Province	Latitude	Longitude
2	England	United Kingdom	51.5000	0.1167
3	Scotland	United Kingdom	55.9500	3.1833
4	Wales	United Kingdom	51.4833	3.1833
5	Northern Ireland	United Kingdom	54.6000	5.9167

In the import file, the names of the columns define the geographic roles. When extending an existing role, the column names must match the existing geographic roles in the hierarchy that

you are extending. This will ensure that the new locations are added to the proper roles and hierarchies.

See the table below for information on how to organize hierarchies in your import file.

Built-In Hierarchy	Columns to include in the .csv file (in order; left to right)
Country (Name)	Country (Name) Latitude Longitude
Country (Name), State/Province	Country (Name) State/Province Latitude Longitude
Country (Name), State/Province, City	Country (Name) State/Province City Latitude Longitude
Country (Name), State/Province, County	Country (Name) State/Province County Latitude Longitude
Country (Name), ZIP Code/Post-code	Country (Name) ZIP Code/Postcode Latitude Longitude
Country (Name), Area Code	Country (Name) Area Code Latitude Longitude

Country (Name), CBSA	Country (Name) CBSA Latitude Longitude
----------------------	---

In addition to the Country (Name) column, you can optionally include the following columns: Country 2 char (ISO 3166-1), Country 3 char (ISO 3166-1), and Country (FIPS 10). If you include these columns, they should be just to the right of the Country (Name) column in any order.

Note: The highest level in the hierarchy is Country and cannot be extended to include higher levels such as Continent, etc.

Add New Roles

To add new geographic roles to the existing geographic hierarchy in Tableau, format your import file to include the new roles along with their parent roles.

When adding new roles to an existing hierarchy, the import file for those roles needs to contain the columns for each level in the existing hierarchy. For more information on the columns to include in each geographic hierarchy, see the [built in hierarchy table](#) in the Extend An Existing Role section.

Below is an example of an import file containing airport codes. Importing the file below would add the geographic roles Airport (ICAO), Airport (IATA), and Airport (City) to the existing Country (Name) hierarchy. Notice that the column name for country matches the existing Country (Name) geographic role.

	A	B	C	D	E	F
1	Airport (ICAO)	Airport (IATA)	Airport (City)	Country (Name)	Latitude	Longitude
2	AYGA	GKA	GOROKA	PAPUA NEW GUINEA	-6.08167	145.39167
3	AYLA	LAE	LAE	PAPUA NEW GUINEA	0	0
4	AYMD	MAG	MADANG	PAPUA NEW GUINEA	-5.20694	145.78861
5	AYMH	HGU	MOUNT HAGEN	PAPUA NEW GUINEA	-5.82611	144.29611
6	AYNZ	LAE	NADZAB	PAPUA NEW GUINEA	-6.56972	146.72611
7	AYPY	POM	PORT MORESBY	PAPUA NEW GUINEA	-9.44333	147.22
8	AYRB	RAB	RABAUL	PAPUA NEW GUINEA	0	0
9	AYWK	WWK	WEWAK	PAPUA NEW GUINEA	-3.58361	143.66917
10	BGAM	N/A	ANGMAGSSALIK	GREENLAND	0	0
11	BGAS	N/A	ANGISSOQ	GREENLAND	0	0
12	BGAT	N/A	ANIJITIEQ	GREENLAND	0	0

Once you have created your import file, save the file as a Comma Delimited (.csv) file (Windows Comma Separated if on a Mac) in a folder on your computer.

Add New Hierarchies

To add a new hierarchy in Tableau, you must create multiple import files, each representing a level in the new hierarchy. For example, you may have custom sales that are defined by Theater, Country, Region, and City. To create a new hierarchy from these geographic roles, you need to create multiple import files, each representing a level in the new hierarchy.

For this example, the top level in the hierarchy is theater so the import file would include Theater, Latitude, and Longitude.

	A	B	C
1	Theater	Latitude	Longitude
2	Shakespeare's Globe theater	51.508342	-0.096984
3	The Düsseldorfer Schauspielhaus	51.227420	6.783620
4	Balboa Theatre	32.714329	-117.161364

The next level is Country, so its import file should contain columns for Theater, Country, Latitude and Longitude.

	A	B	C	D
1	Theater	Country	Latitude	Longitude
2	Shakespeare's Globe theater	England	51.508342	-0.096984
3	The Düsseldorfer Schauspielhaus	Germany	51.227420	6.783620
4	Balboa Theatre	United States	32.714329	-117.161364

Make sure that each file in the hierarchy contains columns for the current level and all levels above it. For example, the City import file, the last geographic role in the hierarchy, would contain columns for Theater, Country, Region and City.

	A	B	C	D	E	F
1	Theater	Country	Region	City	Latitude	Longitude
2	Shakespeare's Globe theater	England	London	London	51.508342	-0.096984
3	The Düsseldorfer Schauspielhaus	Germany	North Rhine-Westphalia	Düsseldorf	51.227420	6.783620
4	Balboa Theatre	United States	Anza Borrego	San Diego	32.714329	-117.161364

Save each import file into a single folder and name each file with the level it represents. For this example there are four import files named theater.csv, country.csv, region.csv, and city.csv.

[\(Back to top\)](#)

(Optional): Create a schema.ini file

Sometimes when you attempt to import custom geographic information in Tableau using a .csv file, you may see a ".csv could not be used because it does not contain a unique column" error message.

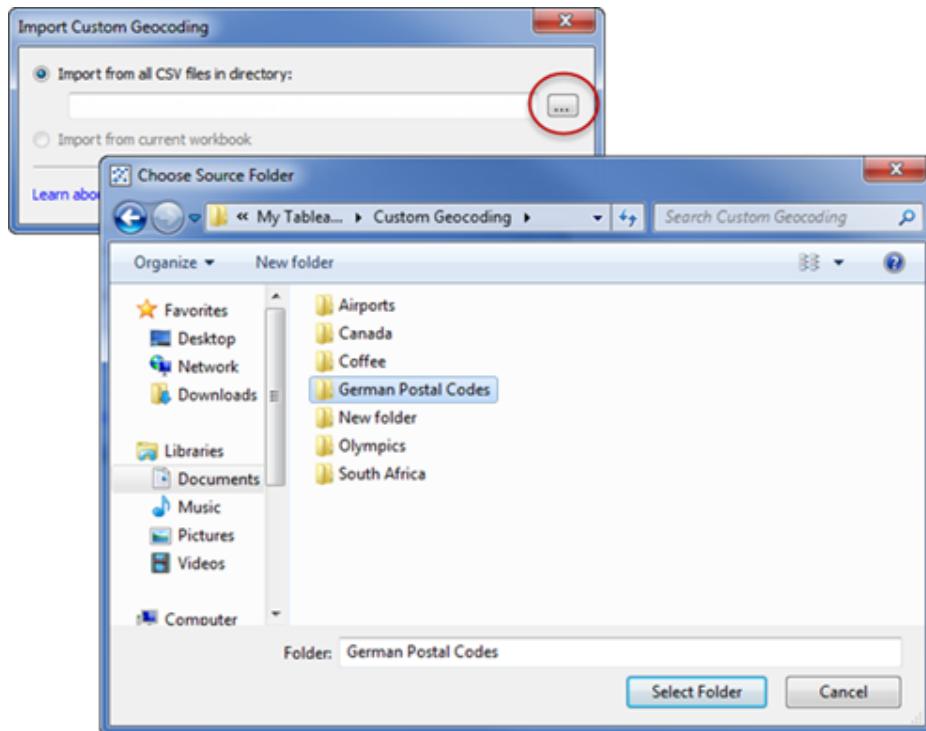
This is because your .csv file might contain numerical data, such as numeric postcodes. Tableau will only accept text fields for new geographic roles. However, you can create a schema.ini file to tell Tableau that the numeric field you want to import should be treated as a text field.

To learn how to create a schema.ini file, see [Create a schema.ini File](#) on page 733.

: Import custom geocoding file(s) into Tableau Desktop

Once you have created a .csv file with custom geocoding you can import that file into Tableau.

1. Open Tableau Desktop and navigate to a new or existing worksheet.
2. Select **Map > Geocoding > Import Custom Geocoding**.
3. In the Import Custom Geocoding dialog box, click the button to the right of the text field to browse to the folder your import file is saved in. All files in the folder will be imported into Tableau.



4. Click **Import**.

The custom geocoding data is imported into the workbook and the new geographic roles become available.

Where does Tableau store my custom geocoding data?

When you import custom geocoding, the data is stored in the Local Data folder in your Tableau Repository. To import custom geocoding, your Tableau Repository must be on a local hard drive. The custom geocoding is then available for all workbooks.

Save custom geocoding files in a separate folder on your local computer. If you have several different sets of custom geocoding files, save each of them under their own folder names.

You can remove the custom geocoding stored in your Tableau Repository by clicking **Map > Geocoding > Remove Custom Geocoding**. This will not remove the geocoding from a packaged workbook, but it will remove it from the Local Data folder in your Tableau Repository.

What happens to my custom geocoding data when I save my workbook?

When you save your workbook as a packaged workbook, the custom geocoding data is packaged with the workbook. When you open a packaged workbook you can import the custom geocoding from that workbook into your own Tableau Repository.

Note: Importing a new custom geocoding file will replace any custom geographic roles previously imported into Tableau.

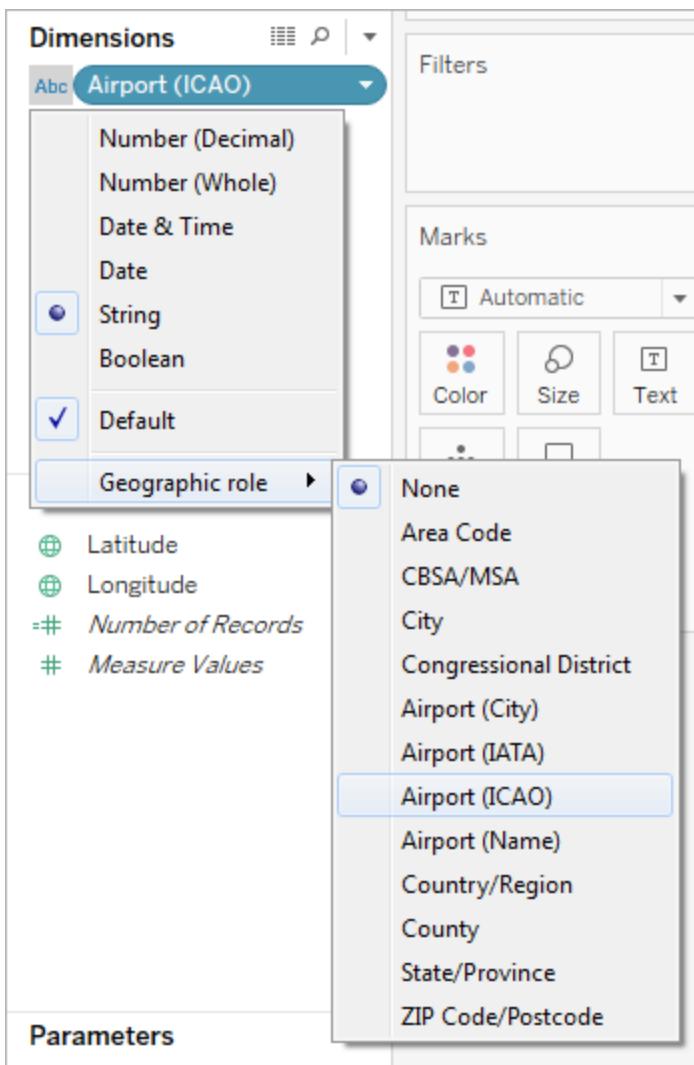
[\(Back to top\)](#)

: Assign geographic roles to your fields

Once you import custom geocoding into your workbook, the custom geographic roles become available. The next step is to assign those geographic roles to fields in your data source.

For example, if you decided to add a new geographic role that lists the latitude and longitude coordinates of airports around the world, you can assign that geographic role to a field in your data source that lists the names of airports so when you create a map view with that field, the airports are plotted in the correct locations on the map view.

To assign a custom geographic role to a field, in the Data pane, click the data type icon next to the field and select **Geographic Role**. Next, select a custom geographic role from the list.



You're now ready to start building a map view with your custom geographic roles. For more information on how to build a map view, see [Build a Map View on page 744](#).

See also:

[Data Blending vs. Custom Geocoding](#) below

[Blend Geographic Data](#) on page 727

[\(Back to top\)](#)

Data Blending vs. Custom Geocoding

If you have geographic locations in your data that are not automatically geocoded in Tableau, there are two ways to plot them on a map view—data blending and custom geocoding. Both data blending and custom geocoding allow you to plot your own locations on a map. Data blending is

easier to set up and you can work with data from any data source. Custom geocoding lets you add to existing roles, and create hierarchies. Custom geocoding can be easier to use once it is set up and imported.

This topic explains the difference between blending geographic information with another data source and importing custom geocoding data into Tableau.

The following is a quick comparison of data blending and custom geocoding.

Capability	Data Blending	Custom Geocoding
Plot your own locations on a map view	Yes	Yes
Use any data	Yes	No, text files only
Add new geographic roles	No	Yes
Add to an existing geographic role	No	Yes
Create new geographic hierarchies	No	Yes
Can be reused for other workbooks	No	Yes

Data Blending

Data blending is the easiest way to plot your data on a map view. Data blending works great if you are adding a single level of geographic information with a latitude and longitude. You can use any data source, unlike Custom geocoding where you can only use text files. You can share the geographic data source with other workbooks on Tableau Server. You can also speed up your map performance by creating an extract that contains only your geocoding data.

The following is a high-level procedure for blending geographic data. For a more detailed example, see [Blend Geographic Data](#) on the next page.

To blend geographic data

1. Create a data source that defines your geographic data.
2. In Tableau Desktop, connect to the original data you want to map, and then connect to the data source that defines your geographic data.
3. Blend the two data sources together. For more information, see [Blend Your Data on page 369](#).
4. Plot the data on a map view.

Custom Geocoding

Custom geocoding is a more flexible way to plot your data on a map. Custom geocoding is available for all workbooks on a computer once the custom geocoding data is imported. The

custom geocoding data will be copied into any packaged workbook (.twbx) or published workbook that uses a custom geographic role. This will make the workbooks about 40 MB larger.

If you upgrade Tableau Desktop, you may need to refresh your custom geocoding to take advantage of any fixes made to the geocoding data in Tableau.

In custom geocoding, you can use additional columns to define larger geographic locations. For instance, if you are creating a set of US census tracts, you may need to define which US county they fall in. You can include additional columns to define larger geographic locations in the import file.

You can also use multiple files for multiple geographic roles that have a matching relationship, meaning they share larger geographic data, such as country or state/province. Once the custom geocoding data is imported, you will see additional geographic roles that can be assigned to your geographic data.

Finally, custom geocoding lets you add additional places to an existing geographic role, such as adding new cities to the city role. It also allows you to define a hierarchy of geographic roles that extends the built-in hierarchies (e.g. census tracts in counties) or defines a new hierarchy (e.g. sub-territories in territories).

For more information about custom geocoding, see [Custom Geocode Your Data on page 718](#). Additionally, you can learn how to custom geocode data by watching the [Custom Geocoding training video](#) on the Tableau website.

See also:

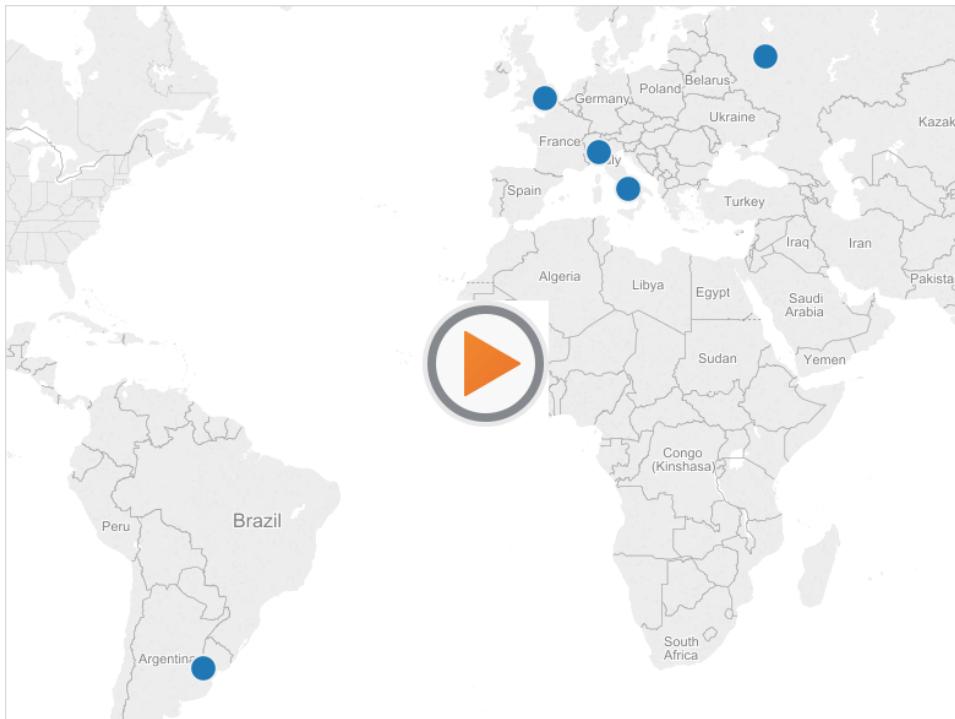
[Prepare Your Geographic Fields](#) on page 735

[Create a schema.ini File](#) on page 733

Blend Geographic Data

This example demonstrates how to blend geographic data in Tableau using two small sample data sources. Follow the steps in this topic to learn how to create a file that defines your geographic data, blend two data sources, and build the map view below using the two data sources.

The embedded view below shows famous theaters around the world, and was created by blending geographic data with another data source. Hover over the marks in the view to see information about each theater. If you would like to follow along with the steps in this topic, click **Download** in the view below, and then select **Tableau Workbook**. This will open the view in Tableau Desktop.



Step 1: Create a file that defines your geographic data

When you have a data source that contains geographic information that is not automatically geocoded in Tableau, the first step is to create a second data source that defines the latitudinal and longitudinal values for that information. You will later connect to this file in Tableau Desktop and blend it with your original data source.

Create a second data source with the following information:

- **Geographic name column:** This column includes any geographic information you want to plot on a map view and typically matches a geographic column name in your original data. For example, if your original data source has a column called Street Address, your second data source should also have a column called Street Address. The data in this column would then be all of the street addresses you want to plot on a map view.

If the column does not match a geographic column name in your original data source, you may need to edit the relationships between your two data sources in Tableau Desktop. For more information, see [Step 3](#).

Finally, this column should not match the names of any geographic roles already in Tableau, such as **County**, **Area Code**, or **CBSA/MSA**.

Note: The data in these columns can be strings or numbers. However, if your locations are numbers, especially numbers with leading zeros (for example, 00501), make sure the data type of the field is set to **String** in Tableau. For more information, see [Data Types](#) on page 262.

- **Latitude and Longitude columns:** The values in these columns need to be in decimal degrees (for example, 47.651808, -122.356881).

Below is an example of an original data source with geographic locations, and a second data source that defines those geographic locations.

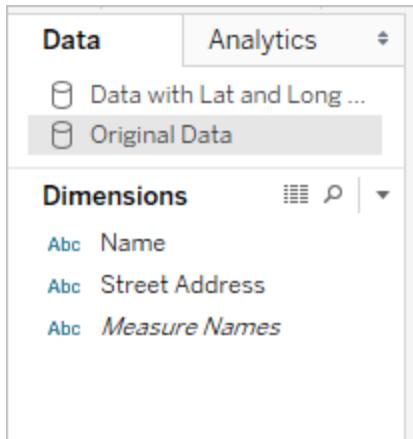
Original data source

Second data source

	A	B		A	B	C
1	Name	Estimated Seats	1	Street Address	latitude	longitude
2	La Scala	3,000	2	Via Filodrammatici, 2 20121 Milan, Italy	45.46731	9.19957
3	Teatro di San Carlo	1,397	3	Via San Carlo, 98 80132 Naples, Italy	40.83832	14.45020
				Cerrito 628 Buenos Aires	-34.60111	-58.42951
4	Teatro Colon	2,500	4	Ciudad Autónoma de Buenos Aires		
5	The Royal Opera House	2,256	5	Bow St London WC2E 9DD	51.5137	-0.1278
6	The Bolshoi	2,200	6	Theatre Square 1 Moscow 125009	55.76016	37.61730

Step 2: Connect to the data sources

1. Connect to the original data source.
2. Connect to the data source that defines your geographic data.

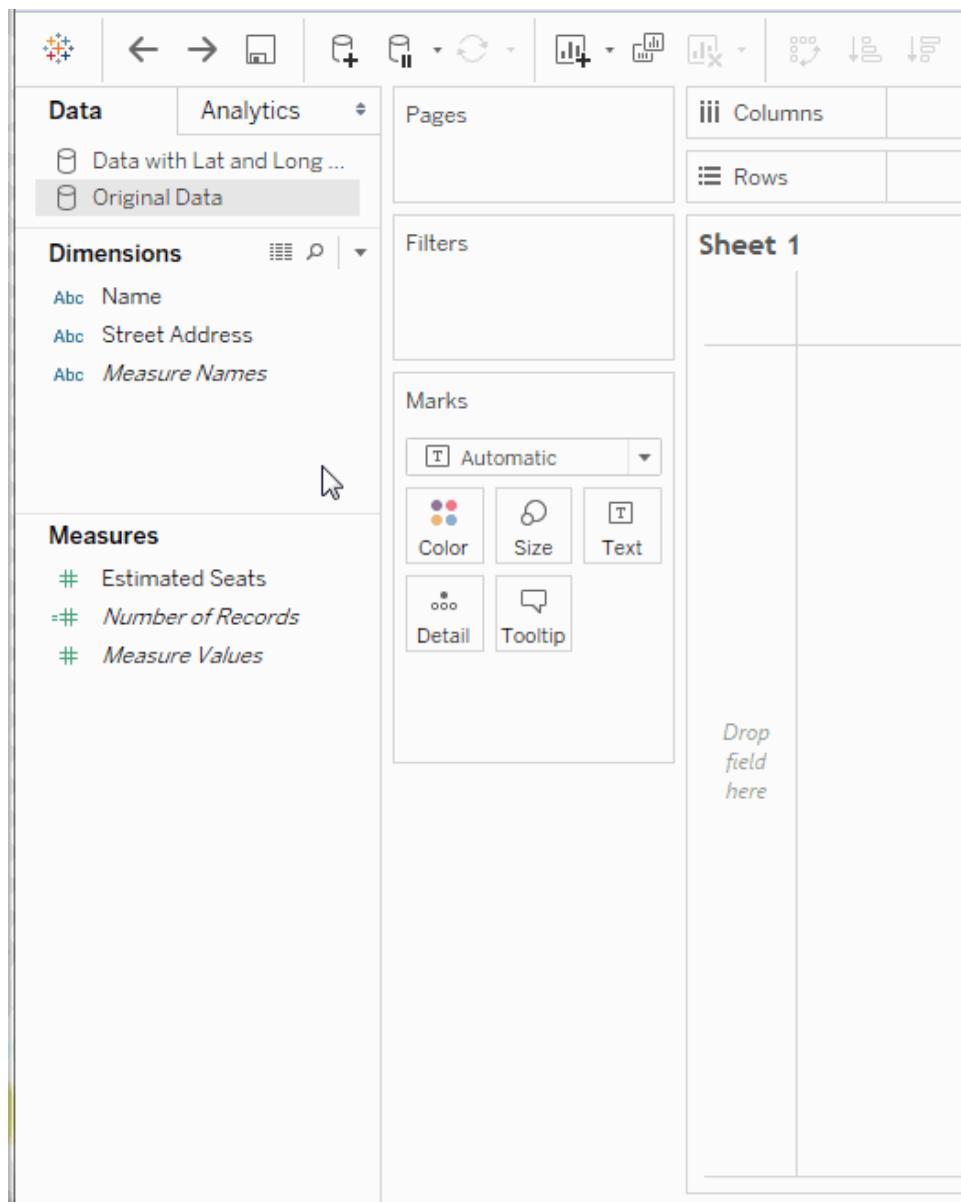


Step 3: Edit Relationships

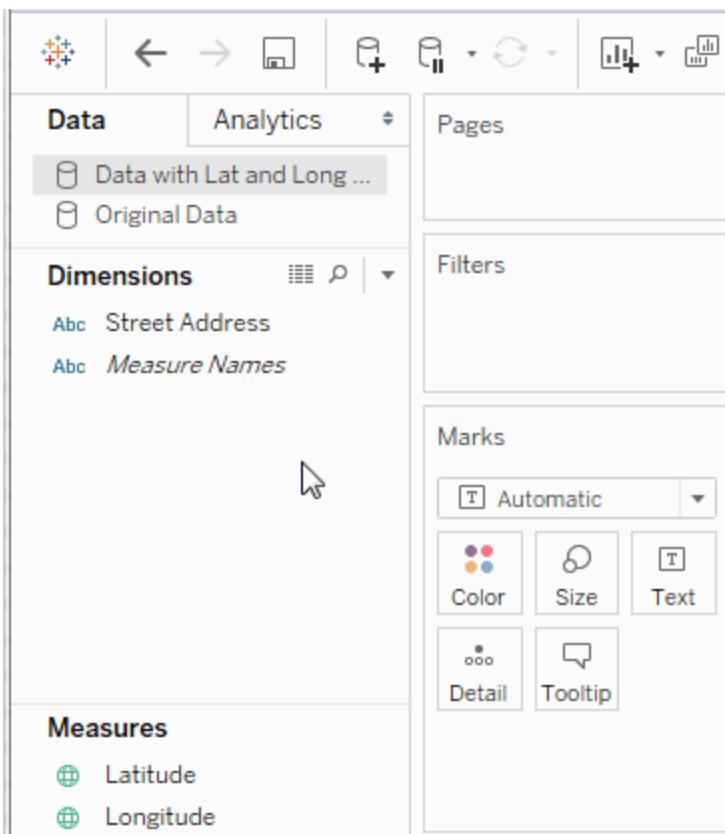
(Optional) If the two data sources share the same column names, your data should blend automatically when you add fields to the view. However, if the file you created to define your geographic data does not share the same column name as the geographic data in your original data source, you will need to create a relationship between those two fields using the **Edit Relationships** option. For information about how to edit relationships, see [Step 4: \(Optional\) Define or edit relationships](#) on page 374.

Step 4: Plot the data on a map view

1. In the **Data** pane, select the second data source that defines your geographic data. Under **Measures**, assign the **Latitude** geographic role to the **Latitude** field and the **Longitude** geographic role to the **Longitude** field. For more information, see [Prepare Your Geographic Fields](#) on page 735.



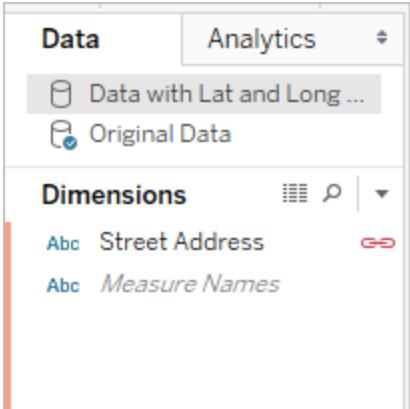
2. In the **Data** pane, select the original data source, and then drag the field you want to plot on a map to **Detail** on the **Marks** card.



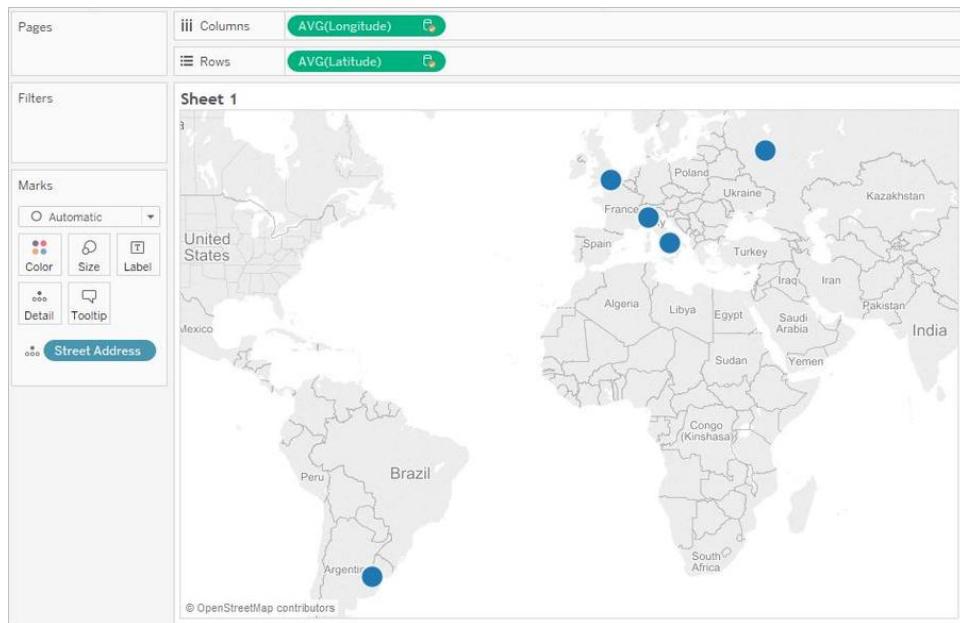
Note: This field should have a data type of String, and should be a dimension. For more information on how to change the data type of a field, see [Data Types](#) on page 262.

3. In the **Data** pane, select the second data source , and then double-click **Latitude** and **Longitude** to add them to the **Columns** and **Rows** shelves.

The link icon indicates that the **Street Address** field is blended with the **Street Address** field in the original data source. A broken link icon indicates that the **Street Address** field is not blended with the **Street Address** field in the original data source.



The view is now complete. Each mark represents an address from the original data source.



See also:

[Data Blending vs. Custom Geocoding on page 725](#)

Create a schema.ini File

Sometimes when you attempt to import custom geographic information in Tableau using a .csv file, you may see a ".csv could not be used because it does not contain a unique column" error message.

This is because your .csv file might contain numerical data, such as numeric postcodes. Tableau will only accept text fields for new geographic roles. However, you can create a

schema.ini file to tell Tableau that the numeric field you want to import should be treated as a text field.

A schema.ini file is a Microsoft configuration file associated with the Microsoft JET engine. It tells JET how to interpret the contents of text files, including data structure, date formats, and other settings. For example, creating a schema.ini file that specifies that the numeric postcodes in your import file should be treated as text will allow you to plot the postcodes on a map.

To create a schema.ini file:

1. Open a text file.
2. In the text file, using the following syntax, specify the names and data types for each column in your import file:

```
[YOURCSVFILENAME.csv]
ColNameHeader=True
Format=CSVDelimited
Col1="Name of 1st Column Header in your .csv file"
Datatype
Col2="Name of 2nd Column Header in your .csv file"
Datatype
Col3="Name of 3rd Column Header in your .csv file"
Datatype
Col4="Name of 4th Column Header in your .csv file"
Datatype
```

For example, if your import (.csv) file contained the columns **German Post Code**, **Latitude**, and **Longitude**, and was named **German Postal Codes.csv**, your schema.ini file would look like the following:

```
[German Postal Codes.csv]
ColNameHeader=True
Format=CSVDelimited
Col1="German Post Code" Text
Col2="Latitude" Double
Col3="Longitude" Double
```

3. Save the file as **schema.ini** in the same folder as your.csv file.

You're now ready to import your .csv and schema.ini files into Tableau.

See also:

[Custom Geocode Your Data on page 718](#)

Prepare Your Geographic Fields

In this topic, you'll learn:

- What to do if Tableau cannot map your location data
 - How to geocode your location data in Tableau (assign geographic roles)
 - What county-equivalents (districts, municipalities, etc.) Tableau recognizes
 - What zip codes and postcodes Tableau recognizes
-

When you connect to location data in Tableau, there are two things that might happen:

1. Tableau recognizes your location data and assigns geographic roles to your geographic fields (e.g. Country, State, City).
2. Tableau *does not* recognize your location data and formats your geographic fields as either numbers or strings.

If Tableau recognizes your location data and automatically assigns geographic roles to your fields (you'll know this has happened because there will be a globe icon  next to them in the **Data** pane), then you're ready to build a map view. Congratulations! Just double-click one of those geographic fields and you've got a map.

If Tableau doesn't recognize your location data, here's what you can do:

If Tableau does not immediately recognize your location data, and you can't build a map view, you'll need to assign geographic roles to your fields. See the [Assign a geographic role to a field on the next page](#) section below to learn how.

Note: This procedure only works if your location data is built in to Tableau.

How do you know if your location data is built in to Tableau?

Tableau recognizes a lot of standard geographic data for many countries, such as state and city names, ZIP codes and postcodes, country borders, and second-level administrative divisions, such as U.S. counties.

For a complete list of location data that's built in to Tableau, see the following sections:

- [Types of geographic roles on page 737](#)
- [County-equivalents recognized by Tableau on page 740](#)
- [ZIP codes and postcodes recognized by Tableau on page 741](#)

Why doesn't Tableau recognize your location data?

Often times Tableau doesn't recognize your location data because it's not built in to the Tableau map server (Tableau Map Service).

The good news is that more and more location data is built-in with every new release of Tableau.

But if your data isn't built-in yet, there are a couple of things you can try in the meantime to map that data in Tableau:

- Custom geocode your data
- Blend your geographic data
- Use shapefile data to build a map view

Assign a geographic role to a field

A *geographic role* associates each value in a field with a latitude and longitude value. When you assign a geographic role to a field, Tableau assigns latitude and longitude values to each location in your data based on data that is already built in to the Tableau map server.

Assigning a geographic role based on the type of location (such as state versus postcode) helps insure that your data is plotted correctly on your map view. For example, you can assign the City geographic role to a field that contains a list of city names.

When a field is assigned a geographic role, Tableau creates a map view when you add the field to **Detail** on the **Marks** card. In other words, Tableau **geocodes** the information in that field.

To assign a geographic role to a field:

- In the Data pane, click the data type icon next to the field, select **Geographic Role**, and then select the geographic role you want to assign to the field.

The screenshot shows the Tableau Data pane with the following details:

- Dimensions:**
 - =Abc Above or Below?
 - Region
 - State, County
 - State
 - County (highlighted in blue)
- Filters:**
 - Region
 - State
- Marks:**
 - Filled Map
 - Color
 - Size
 - Label
 - Detail
 - Tooltip
- Geographic Role:** A dropdown menu is open, showing options like None, Airport, Area Code (U.S.), CBSA/MSA (U.S.), City, Congressional District (U.S.), Country/Region, County (highlighted in blue), State/Province, ZIP Code/Postcode, and Create from.
- Measures:**
 - Extract
 - # Number of Records
 - Sheet1\$ (selected)
 - # Few fruits/veg (% of ...)
 - # High blood pressure (...)
 - # Not Exercise (% of po...)
 - # Obesity Rate (% of po...)
 - # Primary Physicians (...)
 - # Smokers (% of pop)

When you assign a geographic role to a field, Tableau adds two fields to the **Measures** area of the **Data** pane: Latitude (generated) and Longitude (generated).

These fields contain latitude and longitude values and are assigned the Latitude and Longitude geographic roles. If you double-click each of these fields, Tableau adds them to the Columns and Rows shelves and creates a map view using the Tableau background map.

The screenshot shows the Tableau Measures shelf with the following items:

- # Units estimate
- (highlighted in blue) Latitude (generated)
- (highlighted in blue) Longitude (generated)
- # Number of Records
- # Measure Values

Types of geographic roles

The following table describes the geographic roles available in Tableau. Many of the roles are international, but some are limited to the U.S. only.

If your location data does not fit into one of these roles, you may have to import custom geocoding to plot the data on a map.

For more information, see [Custom Geocode Your Data](#) on page 718.

Geographic Role	Assign this role to a field if it contains:
Airport	International Air Transport Association (IATA) or International Civil Aviation Organization (ICAO) airport codes.
Area Code (U.S.)	U.S. telephone area codes; numbers only.
CBSA/MSA (U.S.)	U.S. Core Based Statistical Areas, which includes Metropolitan Statistical Areas, as defined by the U.S. Office of Management and Budget.
City	Worldwide cities with population of 15,000 or more. Names are in English, French, German, Spanish, Brazilian-Portuguese, Japanese, Korean, and Chinese.
Congressional District (U.S.)	U.S. congressional districts.
Country/Region	Worldwide countries, regions, and territories. Names are in English, French, German, Spanish, Brazilian-Portuguese, Japanese, Korean, and Chinese. Tableau also recognizes, FIPS 10, ISO 3166-1 alpha 2, and ISO 3166-1 alpha 3. Names are included in various forms, including long, short, and various abbreviations.
County	Second-level administrative divisions for select countries. For example, U.S. counties, French départements, German kriese, etc. Note: Second-level administrative division definitions vary by country. In Tableau, all second-level administrative divisions are geocoded with the County geographic role. For more information, see the County-equivalents recognized by Tableau on page 740 section.
Latitude	Latitude in decimal degrees. Only available for numeric fields.

Longitude	Longitude in decimal degrees. Only available for numeric fields.
State/Province	Worldwide state, province, and other first-level administrative divisions. Names are in English, French, German, Spanish, Brazilian-Portuguese, Japanese, Korean, and Chinese. Note: Some names are available only in their local form.
ZIP Code/Post-code	ZIP codes and postcodes for select countries. For example, U.S. five-digit zip codes, Australian four-digit postcodes, German five-digit postcodes, etc. For more information, see the ZIP codes and postcodes recognized by Tableau on page 741 section.

For Tableau Desktop version 8.1 and earlier

Role Name	Assign this role to a field if it contains:
Area Code	U.S. telephone area codes; numbers only.
CBSA/MSA	U.S. Core Based Statistical Areas, which includes Metropolitan Statistical Areas, as defined by the U.S. Office of Management and Budget.
City	Worldwide cities with population of 15,000 or more. Names are in English, French, and German.
Congressional District	U.S. congressional districts.
Country/Region	Worldwide countries, regions, and territories. Names are in English, French, and German. Tableau also recognizes, FIPS 10, ISO 3166-1 alpha 2, and ISO 3166-1 alpha 3. Names are included in various forms, including long, short, and various abbreviations.
County	U.S. counties and county-equivalents by name. Independent cities are included with their names followed by city.
Latitude	Latitude in decimal degrees. Only available for numeric fields.
Longitude	Longitude in decimal degrees. Only available for numeric fields.
State/Province	Worldwide state, province, and other first-level administrative divisions. Names are in English, French, and German.

Official abbreviations available for the following countries: Australia, Brazil, Canada, Germany, Poland, Switzerland, U.K., and U.S.

Note: Some names are available only in their local form.

ZIP Code/Post-code	ZIP codes and postcodes for select countries. For example, U.S. five-digit zip codes, Australian four-digit postcodes, German five-digit postcodes, etc.
--------------------	--

For information about map data providers and status, see [About Tableau Maps](#) on the Tableau website.

County-equivalents recognized by Tableau

The following table lists second level administrative divisions (county-equivalents) from several countries that Tableau recognizes.

If your data source contains a field with any of these county-equivalents, you can assign the **County** geographic role to it.

Country	County-equivalent	Version Added	Version Updated
Argentina	Departamentos.	8.2.3	
Australia	Local Government Authorities (LGAs).	8.2.2	8.3.1
Belgium	Arrondissements.	9.0	
Brazil	Municípios.	8.2.3	
Chile	Departamentos.	9.0	
China	Prefectures, Counties, Leagues, County-level Cities, Prefecture-level Cities and Divisions.	9.0	
Denmark	Kommunes.	8.3.1	
Finland	Seutakanta - sub-regions.	9.1	

France	Départements.	8.2
Germany	Kreise, including Landkreise and Kreisfrei Städte.	8.2.4
Hong Kong	Districts.	10.0
India	Districts.	9.3
Italy	Provinces and metropolitan cities.	9.0
Netherlands	Municipalities.	10.1
Singapore	URA Subzones	10.0
South Africa	Municipalities.	8.3.1
South Korea	Sigungu (Metropolitan Areas)	10.0
Spain	Provincias.	8.2
Sweden	Sveriges kommuner - municipalities (including LAU level 2 codes).	9.1
Taiwan	Cities and districts.	10.0
UK	Counties, Unitary Authorities and Local Authority Districts, including ONS and GSS codes.	8.2
U.S	County, including five-digit county codes.	

ZIP codes and postcodes recognized by Tableau

The following table lists zip codes and postcodes from several countries that Tableau recognizes. If your data source contains a field with any of these postcodes, you can assign the **Zip code/Postcode** geographic role to it.

Country	Available in Tableau versions

Australia	7.0 and later
Albania	9.3 and later
Andorra	9.3 and later
Argentina	10.0 and later
<i>Four-digit post codes.</i>	
Austria	9.3 and later
Belarus	9.3 and later
Belgium	9.3 and later
Bosnia and Herzegovina	9.3 and later
Brazil	10.0 and later
<i>Five-digit post codes.</i>	
Bulgaria	9.3 and later
Canada	7.0 and later <i>Forward Sortation Area; first three characters of the six-character Canadian postal code.</i>
Costa Rica	10.0 and later
Croatia	9.3 and later
Cyprus	9.3 and later
Czech Republic	9.3 and later
Denmark	9.3 and later
Estonia	9.3 and later
Finland	9.3 and later
Former Yugoslav Republic of Macedonia	9.3 and later
France	7.0 and later

Germany	7.0 and later
Gibraltar	9.3 and later
Greece	9.3 and later
Hungary	9.3 and later
Iceland	9.3 and later
Italy	9.3 and later
India	10.0 and later
Japan	9.2 and later
<i>Three-digit post codes.</i>	
Note: Japanese seven-digit postcodes are not supported.	
Latvia	9.3 and later
Liechtenstein	9.3 and later
Lithuania	9.3 and later
Luxembourg	9.3 and later
Malta	9.3 and later
Mexico	10.0 and later
Monaco	9.3 and later
Montenegro	9.3 and later
Netherlands	9.3 and later
New Zealand	7.0 and later
Norway	9.3 and later
Poland	9.3 and later
Portugal	9.3 and later

San Marino	9.3 and later
Serbia	9.3 and later
Singapore	10.0 and later
South Africa	10.1 and later
Slovakia	9.3 and later
Slovenia	9.3 and later
Spain	9.3 and later
Sweden	9.3 and later
Switzerland	9.3 and later
Taiwan	10.0 and later
<i>Three-digit post codes</i>	
Thailand	10.1 and later
Turkey	9.3 and later
U.K. <i>Outcodes; first segment of the five- to seven-character U.K. postcode.</i>	7.0 and later
U.S.	7.0 and later
Vatican City	9.3 and later

See also:

[Custom Geocode Your Data](#) on page 718

[Blend Geographic Data](#) on page 727

[Create Tableau Maps from Shapefiles](#) on page 703

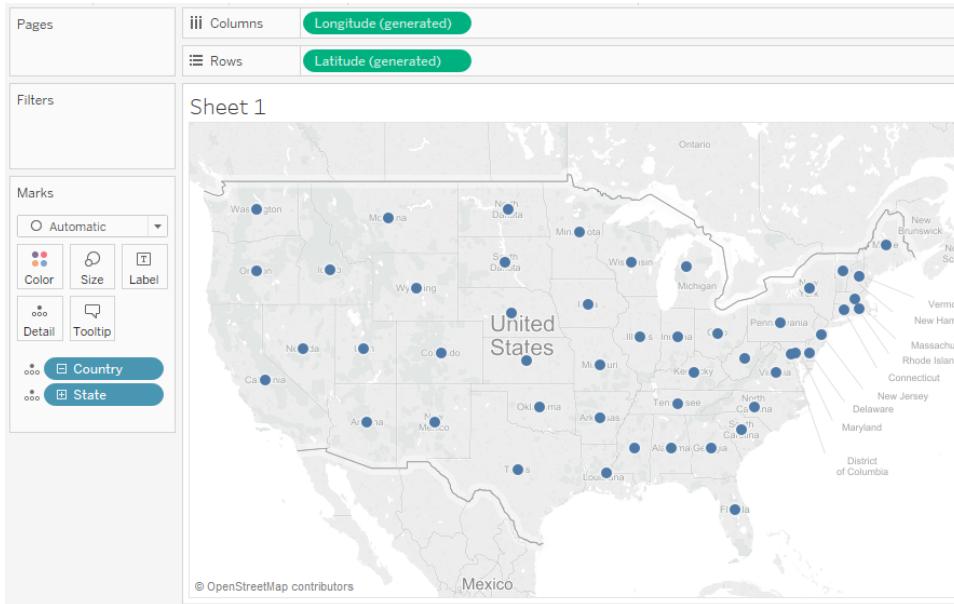
[Edit Locations](#) on page 765

Build a Map View

When creating a map view, Tableau automatically adds the generated **Latitude** and **Longitude** fields to the **Rows** and **Columns** shelves and places a selected geographic field

on **Detail** on the **Marks** card.

The marks in the view correspond to the field on **Detail**. For example, if the **State** field is used in the view, there is a single mark for each state in your data. As you add more geographic fields to **Detail** on the **Marks** card, the marks in the view will be broken down by the members of those fields.



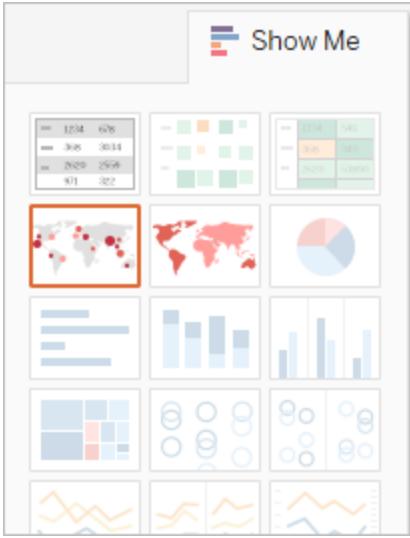
To Build a Map View

To create map views, add geographic fields to the view, and then add measures or continuous dimensions to the **Marks** card. To follow a step by step example of how to create map views using sample data that comes with Tableau, see [Build a Map View on page 118](#).

Add geographic fields to the view

To add geographic fields to the view, do one of the following:

- Double-click a geographic field in the **Data** pane. A globe icon indicates fields that have been geocoded. For more information, see [Prepare Your Geographic Fields on page 735](#).
- Select a geographic field in the **Data** pane, and then select one of the map views in Show Me.

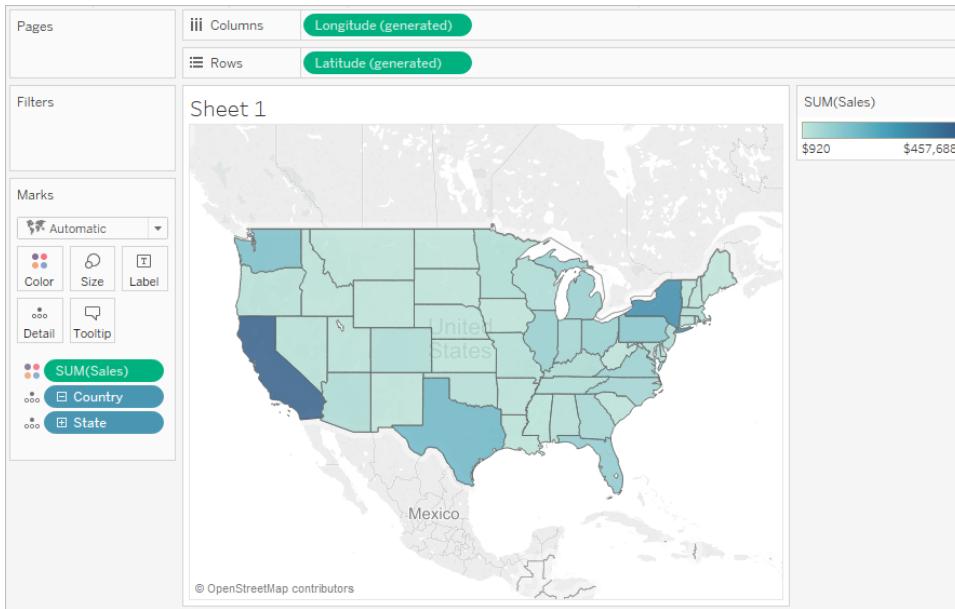


- From the **Data** pane, drag a geographic field to **Detail** on the **Marks** card.

Add fields to the marks card

To complete a map view, from the **Data** pane, drag measures or continuous dimensions to the **Marks** card.

For example, the following map shows the sum of sales for each state. The **Sales** field has been placed on **Color** on the **Marks** card, so each state is colored based on the sum of its sales. When creating a map view, placing a measure on **Color** on the **Marks** card causes the mark type to automatically update to the Filled Map mark type. For more information, see [Filled Map Mark on page 186](#).



See also:

[Build-It-Yourself: Build a map view](#)

[Prepare Your Geographic Fields](#) on page 735

[Customize How Your Map Looks](#) on page 767

Create Spider Maps in Tableau Desktop

You can create spider maps in Tableau Desktop, similar to the examples below.

Spider maps, also known as origin-destination maps, are great for when you're working with hubs that connect to many surrounding points. They are an excellent way to show the path between an origin and one or more destination locations.

There are several ways to create spider maps in Tableau. This topic illustrates how to create a spider map using two examples. Follow the examples below to learn how to set up your data source, and build the view for two different spider maps.

For other examples that might fit closer to your data, see the following workbooks on Tableau Public:

- [Puget Sound Radio Tower Bandwidth](#)
- [Recruitment Strategies in the English Premier League](#)
- [2014 U.S. Flight Departure Delays](#)

Example 1: Metro station traffic in Paris, France

[Learn how to build it!](#)

Your data source

To create this type of spider map in Tableau, your data source should include the following information:

- A path ID for every unique path. See [The Line Group \(Path ID\) column below](#) below for more information.
- Numbers to define the drawing order of each data point (location). See [The Order of Points column on the next page](#) below for more information.
- Latitude and longitude coordinates for every location. See the example table below.

It's also recommended that your data source contain a column with your location names, but it is not required.

The following table is a snippet of the Transports data source, which is included in the [Create Spider Maps in Tableau Example Workbook](#) on Tableau Public. It contains metro traffic data for all metro lines in Paris, France. The first three metro stations for two metro lines are included in this example, and it contains the required columns **Line Group (Path ID)**, **Order of Points**, **Latitude**, and **Longitude**. It also contains the additional columns **Line**, **Station**, and **Traffic** for added clarity and organization.

Line	Line Group (Path ID)	Order of Points	Station	Latitude	Longitude	Traffic
1	1	1	La Défense (Grande Arche)	48.891934	2.237883	14,275,382
1	1	2	Esplanade de la Défense	48.887843	2.250442	9,843,051
1	1	3	Pont de Neuilly	48.884509	2.259892	6,902,931
10 BOUCLE	10	1	Boulogne-Jean-Jaurès	48.842222	2.238836	3,847,782
10 BOUCLE	10	2	Porte d'Auteuil	48.848074	2.258648	687,237
10 BOUCLE	10	3	Michel-Ange-Auteuil	48.847740	2.264297	2,222,709

The Line Group (Path ID) column

In this example, the Line Group (Path ID) column is used to identify each unique path. You will use this column to create your spider map.

For each path, there is a unique key or string, which is added to every location in that path.

For example, in the table above, there are two metro lines (1 and 10 Boucle), and each of those metro lines have a unique path ID listed in the **Line Group (Path ID)** column. For metro line 1, the Line Group is 1. For metro line 10 Boucle, the Line Group is 10. Every location in line one contains the line group 1, and every location in line 10 Boucle contains the line group 10.

The Order of Points column

This column defines the point order and the direction in which the line is drawn from the first data point to the last data point on your map, which, in this example is the start and end of the metro line. This field is very important if you have more than two locations in a path, and can be used to force Tableau to draw your paths in a certain order (you can think of it like connecting the dots).

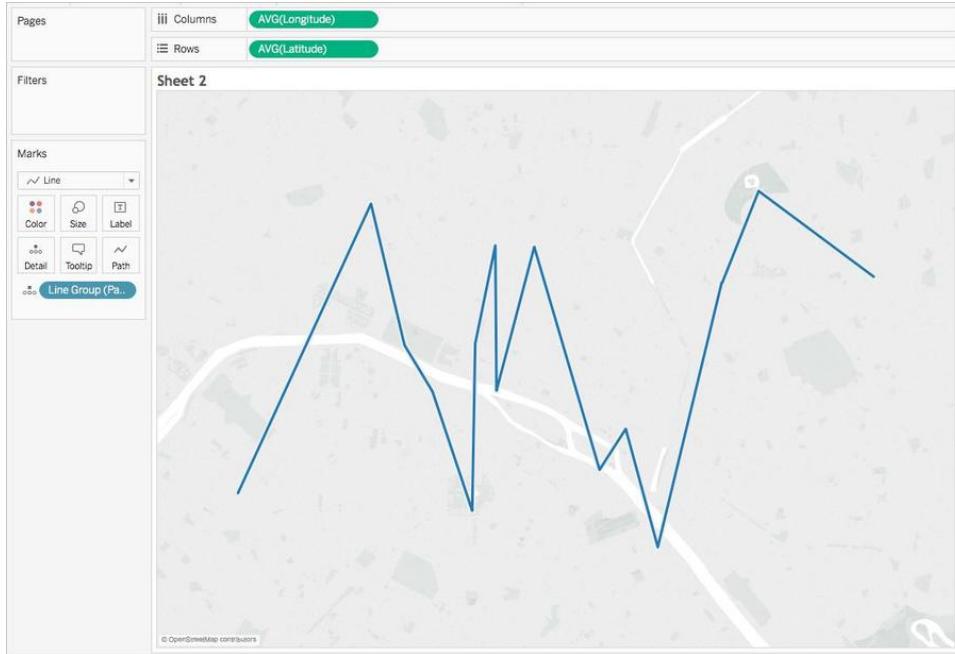
In the example table above, there are three stations on line 1. They are listed in order from the first station (La Défense (Grande Arche)), to the third station (Pont de Neuilly) using the numbers **1** through **3** in the **Order of Points** column. The same is true for line 10 Boucle.

Build the Map view

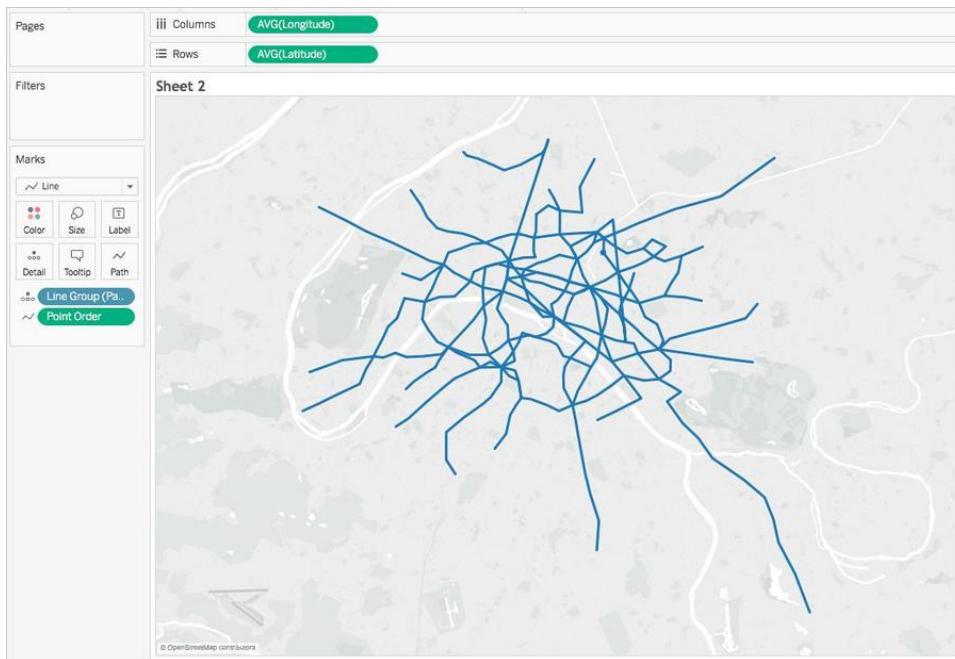
To follow along with this example, download the [Spider Maps Example Workbook](#) from Tableau Public, and open it in Tableau Desktop.

1. In the **Create Spider Maps in Tableau Example Workbook**, click the New Worksheet icon.
2. In the new worksheet, from Measures, drag **Longitude** to the **Columns** shelf, and **Latitude** to the **Rows** shelf.
3. From Dimensions, drag **Line Group (Path ID)** to **Detail** on the Marks card.
4. On the Marks card, click the Mark Type drop-down and select **Line**.

The map view updates with a line connecting all of the points. The Path button should appear on the Marks card.

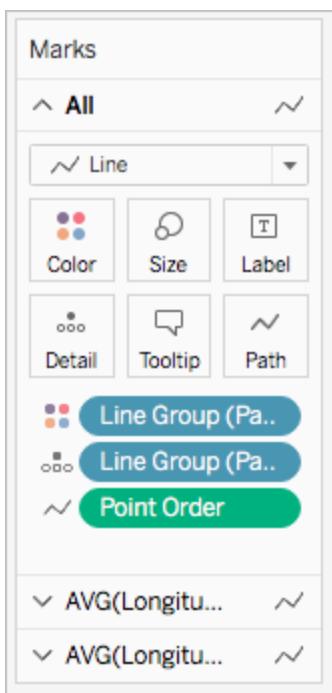


5. From Measures, drag **Point Order** to **Path** on the Marks card.
Point Order is aggregated as a sum.
6. On the Marks card, right-click the **SUM(Point Order)** field and select **Dimension**.
The map view updates with a line for every metro line.



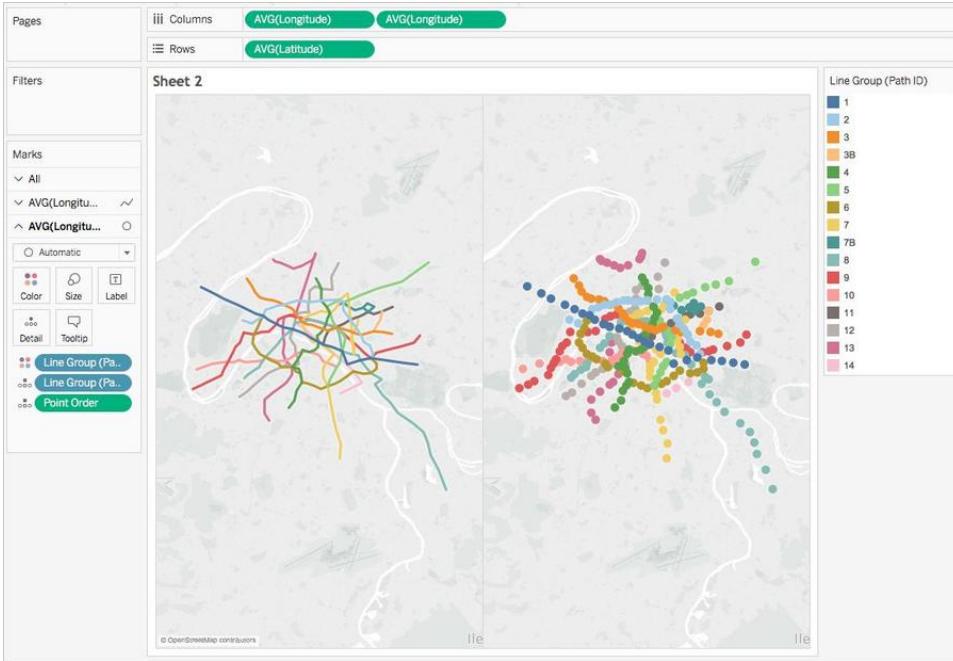
7. From Dimensions, drag **Line Group (Path ID)** to **Color** on the Marks card.
- Each line now has its own color associated with it, and a color legend is added to the view.
8. From Measures, drag **Longitude** to the **Columns** shelf and place it to the right of the first Longitude field.

There are now two Longitude fields on the columns shelf. The view updates with two identical maps. The Marks card updates with two tabs: one for the map on the left, and one for the map on the right. You can customize each of these tabs to change the visual detail of each map view. There is an All tab to control the visual detail in both maps at the same time.

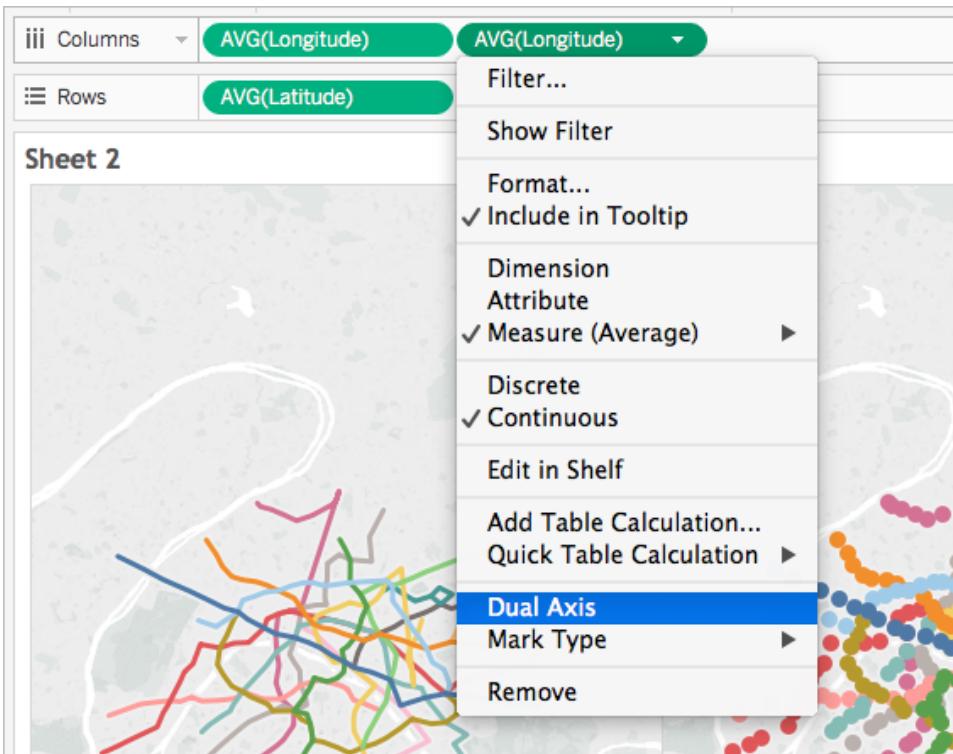


9. On the Marks card, click the bottom **AVG(Longitude)** tab.
10. On the Marks card, under the bottom AVG (Longitude) tab, click the Mark type drop-down and select **Automatic**.

The map view on the right updates to be a point map.



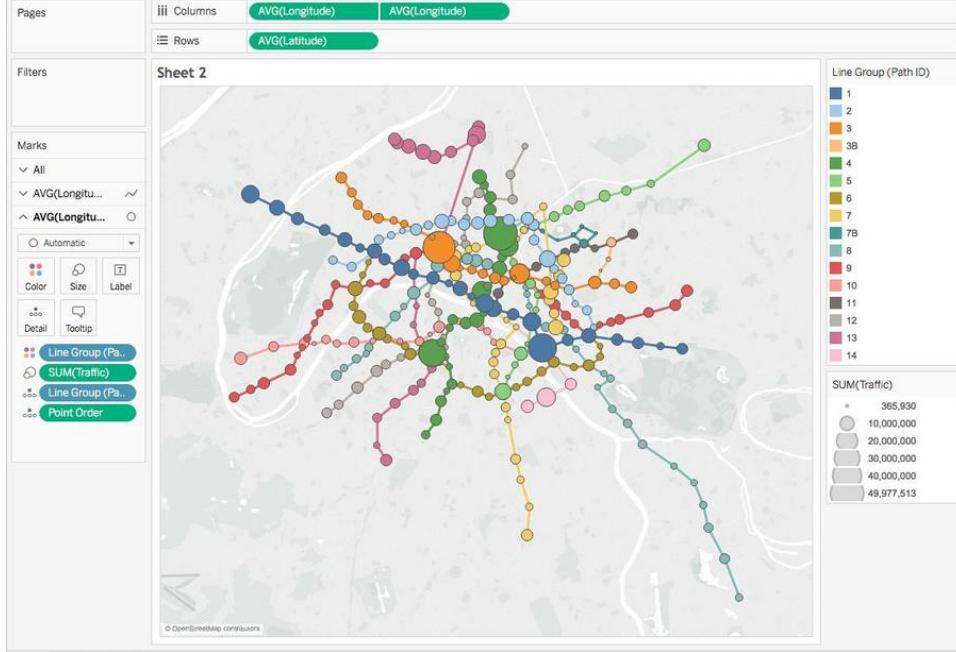
11. On the Columns shelf, right-click the second **AVG (Longitude)** field (on the right), and select **Dual Axis**.



Your map views are now layered on top of one another.

12. From Measures, drag **Traffic** to **Size**, on the bottom AVG (Longitude) Marks card.
The size of the data points update to show the amount of traffic per station.
13. On the Marks card, click **Size** and move the slider to the right.
14. On the Marks card, click **Color**, and then, under Effects, click the **Border** drop-down and select a color.

The view is now complete. You can quickly find the stations on each metro line with the most traffic.



Filter the amount of information in the view

If you want to filter the amount of lines you see in the view:

- From Dimensions, drag **Line Group** to the Filters shelf.

You can also right-click the **Line Group** field and select **Show Filter**, to show a filter card in the view.



Example 2: Bike share data in Seattle, Washington

Learn how to build it!

Your data source

Similar to the first example, for this type of spider map, your data source should contain the following information:

- A Path ID for every unique path. See [The Path ID column](#) on the next page below for more information.
- Latitude and longitude coordinates for every location. See the example table below.

For this type of spider map, there should also be two rows in your data source for each path. That means you should have one row for your origin location data, and one row for your destination location data, repeated for every path. This is a crucial step that enables Tableau to draw your paths correctly. See [Origin-destination rows](#) on the next page below for more information.

It's also recommended that your data source contain a column with your location names, but it is not required.

This example is a snippet of a larger data source that contains bike share data for Seattle, Washington. It has a column for **Origin-Destination**, **Station**, **Path ID**, **Latitude**, and **Longitude**. Only the last three columns in this data source are required to create a spider map, but the Origin-Destination and Location Name columns provide further clarity and organization.

Origin-Destination	Station	Path ID	Latitude	Longitude
Origin	BT-01	BT-01_BT-01	47.61841	-122.35101
Destination	BT-01	BT-01_BT-01	47.61841	-122.35101
Origin	BT-01	BT-01_BT-03	47.61841	-122.35101
Destination	BT-03	BT-01_BT-03	47.61576	-122.34843
Origin	BT-01	BT-01_BT-04	47.61841	-122.35101
Destination	BT-04	BT-01_BT-04	47.61613	-122.34108
Origin	BT-01	BT-01_BT-05	47.61841	-122.35101
Destination	BT-05	BT-01_BT-05	47.61303	-122.34410

Origin-destination rows

For each unique path you want to create, you need a row for your origin location, and a row for your destination location in your data source. This means your origin location will be paired with every destination location.

For example, when showing the path between an origin bike share location and several destination locations in a city, you need a row for the origin location, and a row for the destination location for *every single path*.

In the example above, the origin station BT-01 is paired with several different destination locations (BT-01, BT-03, BT-04, BT-05) to show that bikes were checked out of the BT-01 location and returned either to the same location or to a different location. Each origin-destination pair is color-coded to show that they make up one path. There is a column for Origin-Destination to further illustrate this concept, but this column is not required.

The Path ID column

The Path ID column is used to identify each unique origin-to-destination path. You will use this column to create your spider map.

For each origin and destination location, there is a unique key or string that identifies them as a pair.

In the example below, for the first origin-destination path, the Path ID is BT-01_BT-01. For the second origin-destination path, the Path ID is BT-01_BT-03. Each path ID is listed twice, once for the origin location, and once for the destination location. Again, each pair is color-coded to indicate that they make up one path.

Origin-Destination	Location Name	Path ID	Latitude	Longitude
Origin	BT-01	BT-01_BT-01	47.61841	-122.35101

Destination	BT-01	BT-01_BT-01	47.61841	-122.35101
Origin	BT-01	BT-01_BT-03	47.61841	-122.35101
Destination	BT-03	BT-01_BT-03	47.61576	-122.34843

Note: Your path ID can be anything you want. However, if you want to create complex calculated fields to help filter the locations later on, it helps if your path IDs are consistent across all paths. A great way to do this is to create Path IDs that are a combination of your origin and destination location names, separated by a delimiter. For example, the Path ID for the origin location BT-01 and the destination location BT-03 is BT-01_BT-03. For an example of why you might want to do this, see the [Option 2: Create a dynamic filter: on page 759](#) section.

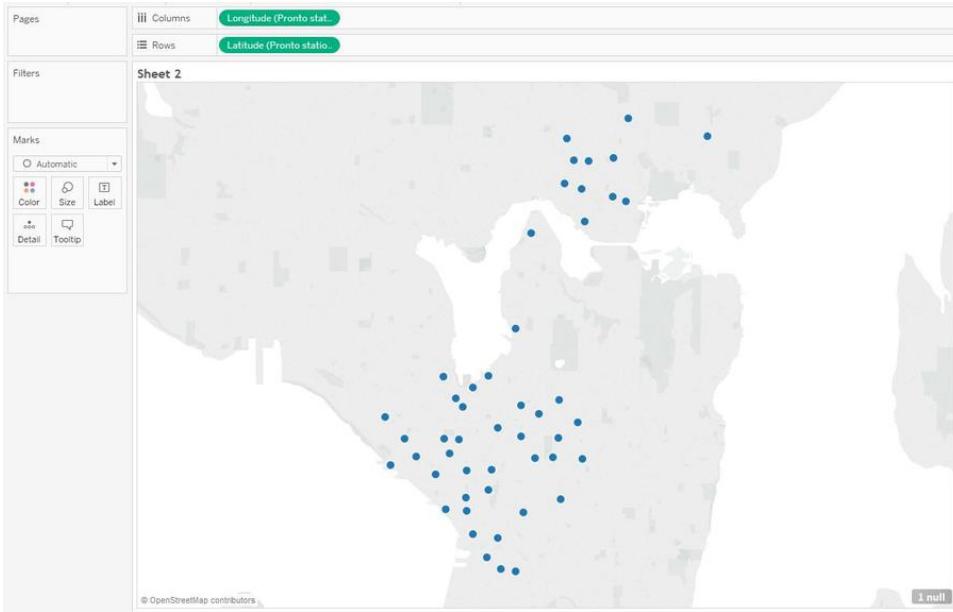
Build the map view

After you set up your data source, you can connect to it in Tableau Desktop and build a spider map.

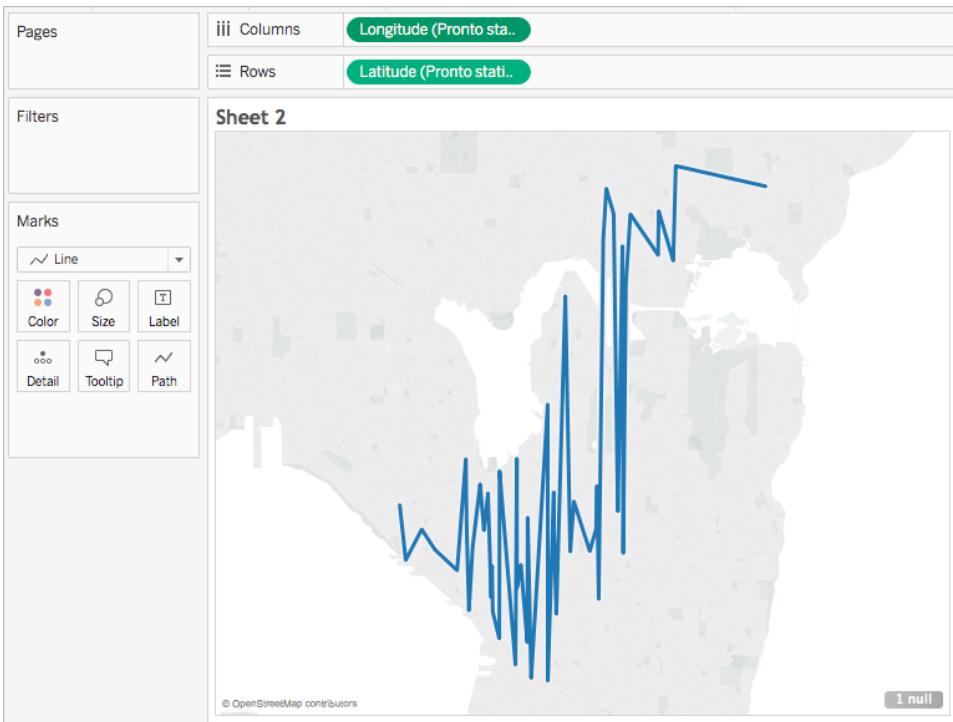
1. From Measures, drag **Longitude** to the **Columns** shelf, and **Latitude** to the **Rows** shelf.
2. On the Columns shelf, right-click the **Longitude** field and select **Dimension**. Do the same for the **Latitude** field on the **Rows** shelf.

This will ensure Tableau does not aggregate your origin and destination locations.

You should see all the locations in the data source as data points on your map. In this example, there is a data point for every bike share location in the data source.

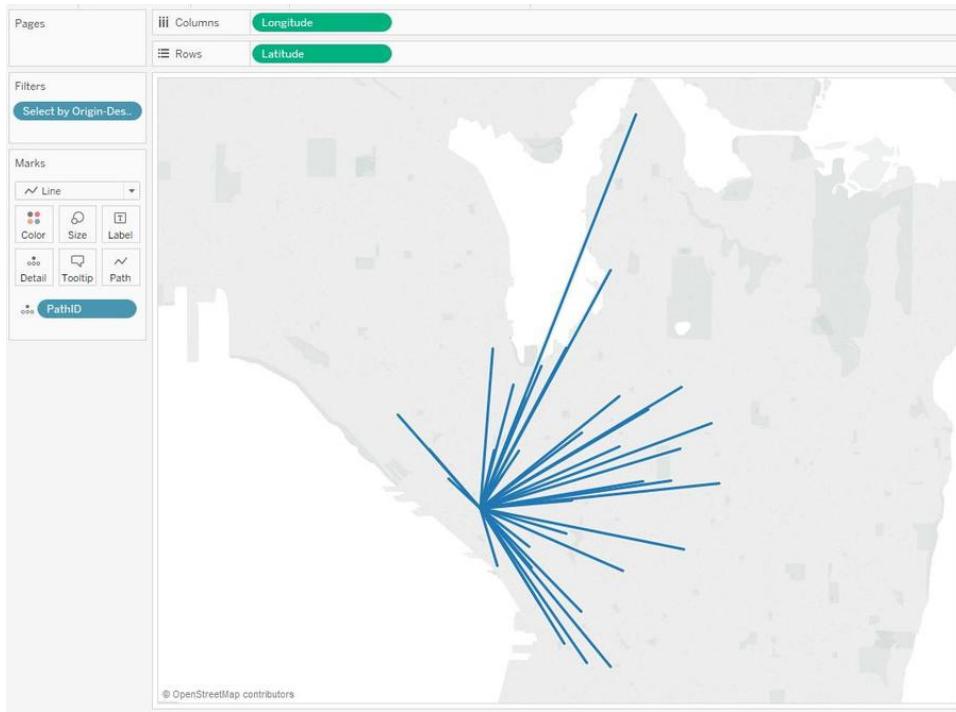


3. On the Marks card, click the Marks Type drop-down, and select **Line**. The view updates to show a line connecting each data point, and the Marks card updates with a Path button.

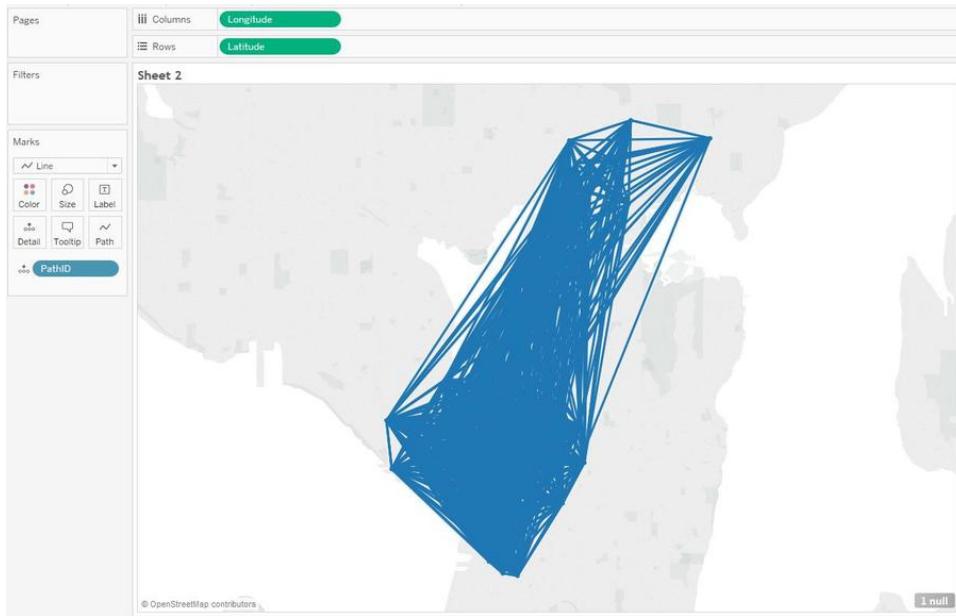


4. From Dimensions, drag **Path ID** to **Detail** on the Marks card.

If you have only a few origin-destination pairs, your view might look something like the following:



However, if you have many origin-destination pairs, your view might look more like this:



This is very common, and can be fixed by filtering a large portion of your paths from the view. Continue to the following section to learn a couple of ways to do so.

Filter the amount of information in the view

If your data source contains a lot of origin-destination pairs, you can filter most of them from the view.

Option 1: Create a simple filter:

1. From Dimensions, drag **Path ID** to the Filters shelf.
2. In the Filter dialog box that opens, do the following:
 - Under the **General** tab, select **None**.
 - Click the **Wildcard** tab.
 - Under the **Wildcard** tab, for **Match Value**, enter **BT-01**, and then click **Starts With**.
 - Click **OK**.

This filters the view down to only the paths that start with BT-01.

Option 2: Create a dynamic filter:

You can also create a calculated field and combine it with a parameter so you can switch between the paths you want to see right in the view. Follow the steps below to learn how.

Step 1: Create the parameter

1. On the Data pane, to the right of Dimensions, click the Data pane drop-down and select **Create Parameter**.
2. In the Create Parameter dialog box, do the following:
 - Name the parameter **StationSelected**.
 - For **Data type**, select **String**
 - For **Allowable values**, click **List**, select **Add from field**, and then select the **Location Names** field.
 - Click **OK**.

Step 2: Create the calculated field

1. Select **Analysis > Create Calculated Field**.
2. In the calculation editor, name the calculated field, **Select by Origin-Destination**, and then enter the following formula:

IF

```

LEFT([Path ID], FIND([Path ID], "_") -1) =
[StationSelected] THEN "Origin"

ELSEIF

    RIGHT([Path ID], LEN([Path ID]) - FIND([Path ID], "_"))
= [StationSelected] THEN "Destination"

ELSE

    "Unselected stations"
END

```

In the **table above**, each Path ID (for example BT-01_BT-03) contains an underscore (_) as a delimiter to separate the origin location name (BT-01) from the destination location name (BT-03). This delimiter is used in the formula to tell Tableau which locations (selected in the parameter you created in step 1 of this procedure) are origin locations, and which are destination locations. The parameter you created above is also used in the formula (StationSelected).

3. Check that the calculation is valid, and then click **OK**.

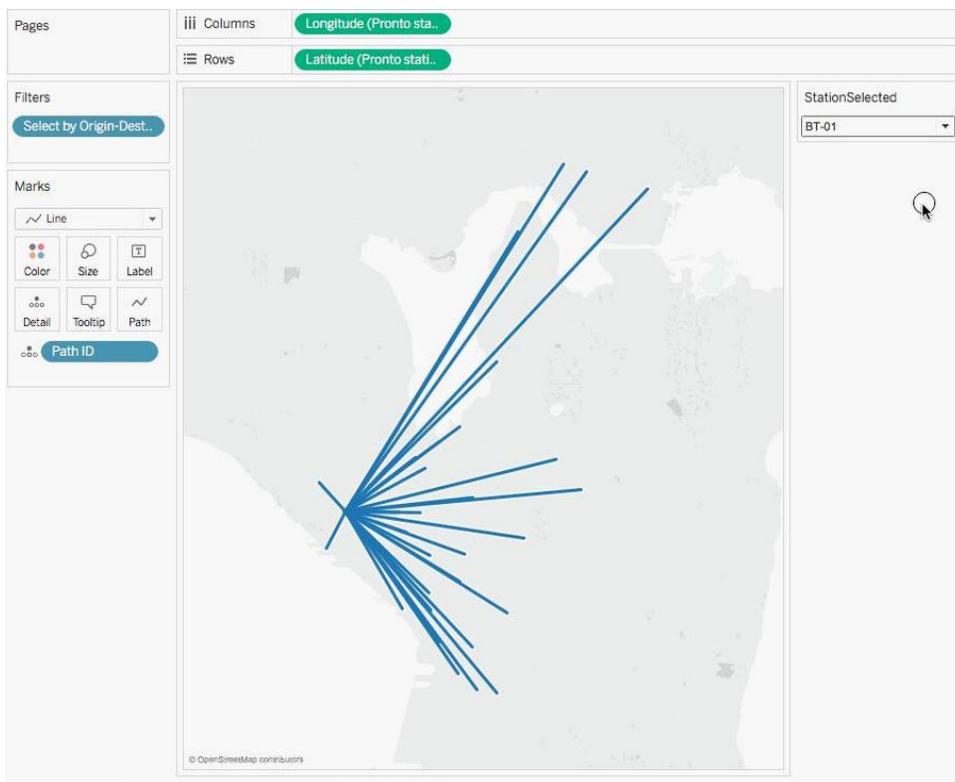
Step 3: Add the calculated field to the Filters shelf

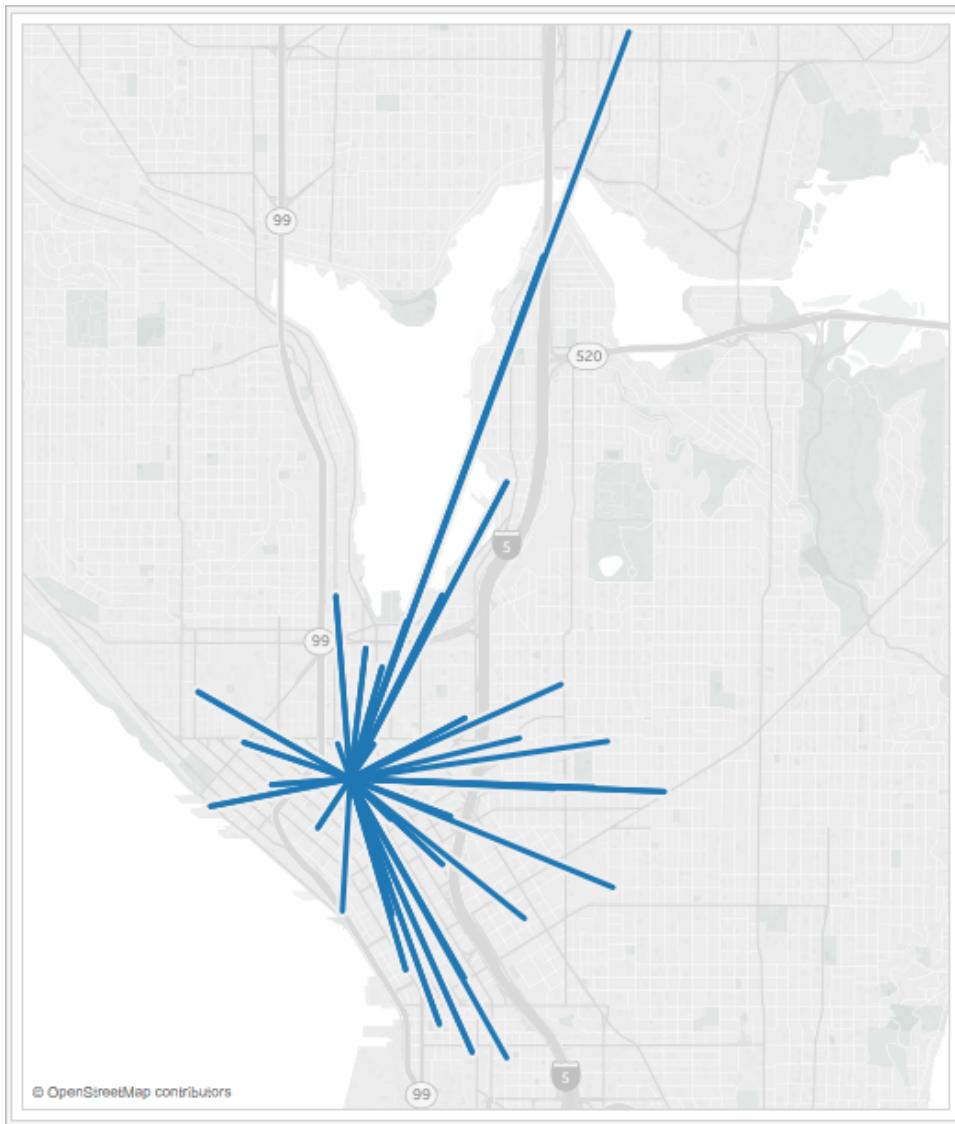
1. From Dimensions, drag the calculated field to the **Filters** shelf.
2. In the Filter dialog box, select **Origin**, and then click **OK**.

Step 4: Show the parameter control

In the Data pane, under Parameters, right-click the parameter you created and select **Show Parameter Control**.

You can now select an origin from the parameter control in the view, and the marks on your map view will update.





See also:

- [Build a Map View](#) on page 744
- [Parameters](#) on page 989
- [Filter Data from Your Views](#) on page 672

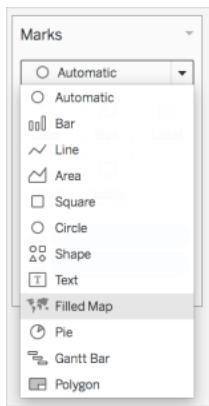
Create Filled Maps with Pie Charts in Tableau

The pie mark type can be useful to show simple proportions to a relative whole. For example, pie marks might be effective when you want to show the percentage of profit for a product by geographic location.

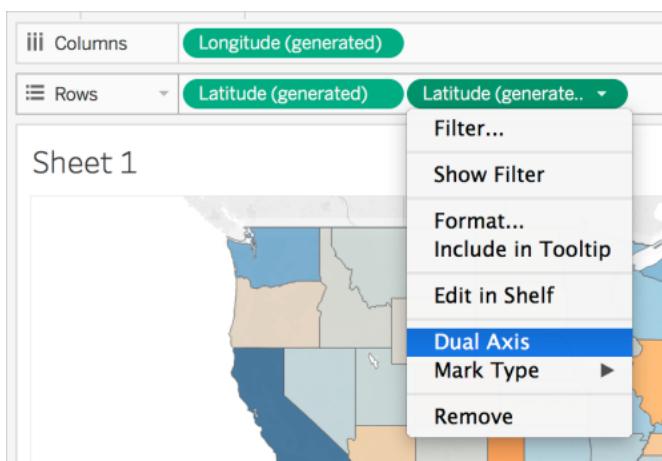
In this example, the pie mark type is used with the filled maps mark type to show the percentage of profit for office supplies, furniture, and technology, relative to the total profits by state. It uses the Sample-Superstore data source that comes with Tableau Desktop.

Build the map view

1. In Tableau, open a new workbook and connect to the Sample-Superstore data source.
 2. On the Data Source page, click **Sheet 1** to go to a new worksheet.
 3. In the Data pane, under Dimensions, double-click **State**.
- Tableau creates a symbol map, with a data point for each state in the Sample-Superstore data source.
4. On the Marks card, click the Mark-type drop-down and select the **Filled Map** mark type.



5. From Measures, drag **Profit** to **Color** on the Marks card.
6. From Measures, drag **Latitude (generated)** to the **Rows** shelf, and place it to the right of the other Latitude field.



You should now have two identical map views.

7. On the Rows shelf, right-click the second **Latitude** field and select **Dual Axis**.

The second map is now layered on top of the first map. There are now three drop-downs on the Marks card: one for each map view, and one for both views (all). These are three separate marks cards that you can use to control the visual detail for each of the map views.

8. On the Marks card, click one of the **Latitude (generated)** tabs, and then click the Mark type drop-down and select the **Pie** mark type.

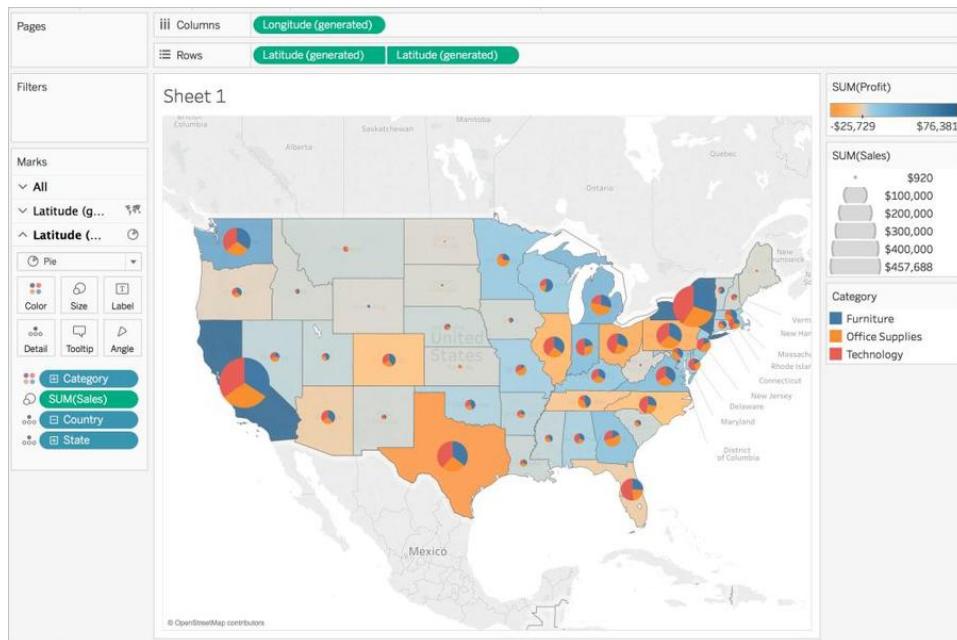
9. From Measures, drag **Sales** to **Size** on the Latitude (generated) Marks card you selected.

The Sum of sales for each state is shown as text.

10. From Dimensions, drag **Category** to **Color** on the same Marks card.

If the size of the pie charts is too small, click **Size** on the Marks card to adjust the size.

The map view now shows the sum of profit, as well as the sum of sales for each category, for each state.



See also:

[Build a Map View on page 744](#)

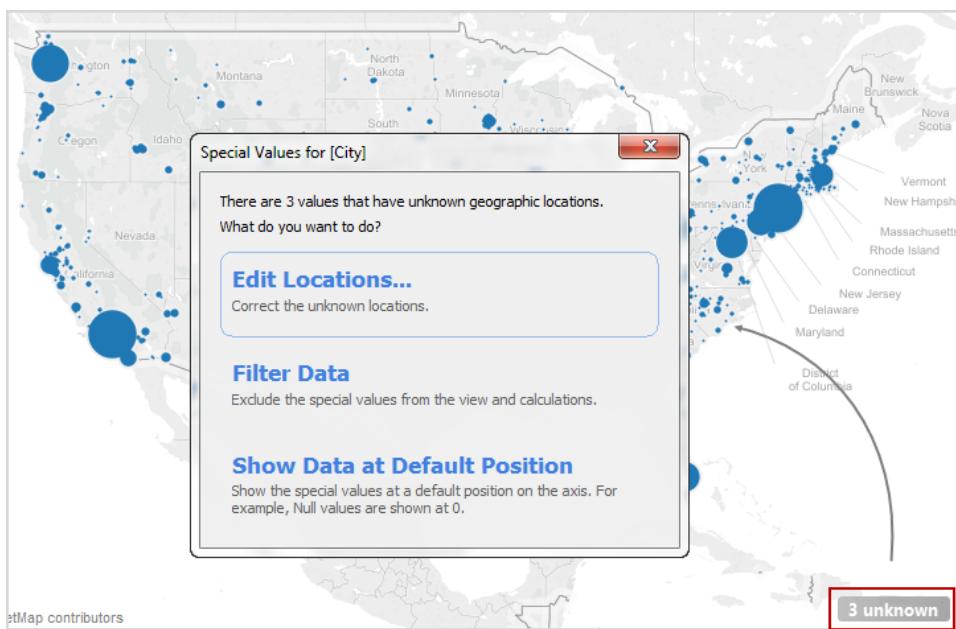
[Build a Map View on page 118](#)

Edit Locations

Sometimes Tableau will not recognize one or more of the location names in your data. When this happens those values are marked as **unknown** in the lower right corner of the view. This often happens because your data values may be spelled incorrectly or use an abbreviation that Tableau does not recognize. When this happens, you can edit the unknown location names to map to known locations.

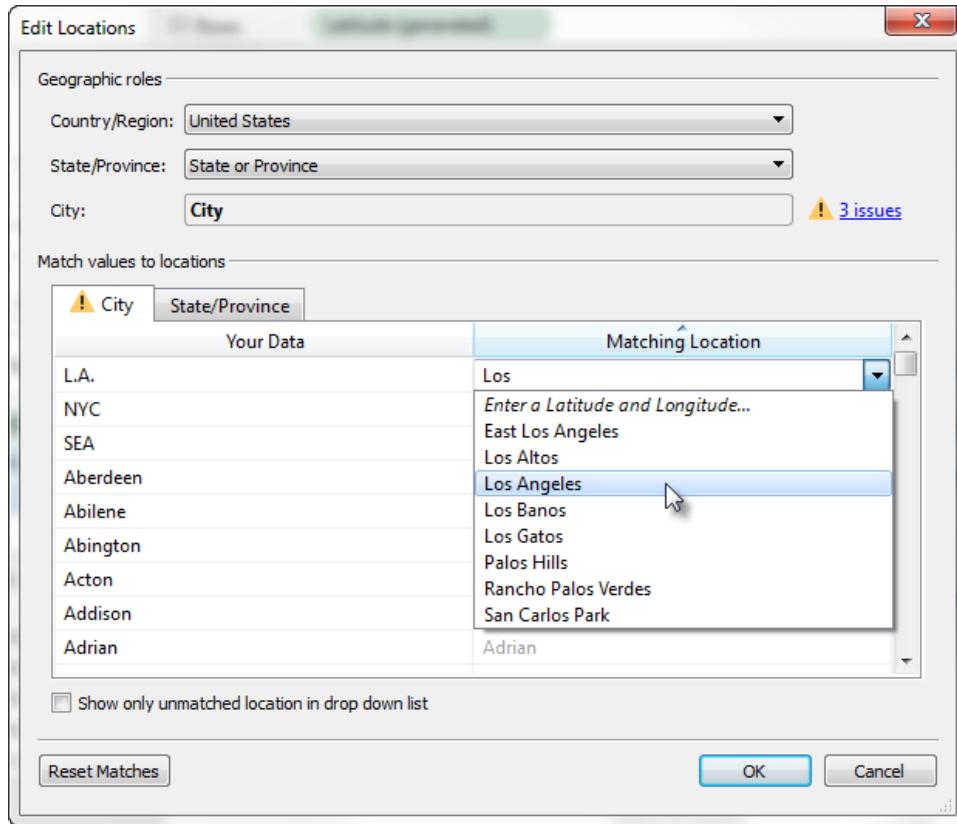
To edit unknown location names

1. In the bottom right corner of the view, click the special values indicator. This opens the Special Values dialog box.
2. In the Special Values dialog box, select **Edit Locations**.



Note: If the special values indicator is not visible, you can also select **Map > Edit Locations**.

3. In the Edit Locations dialog box, click on one of the **Unrecognized** cells to match a known location to your unknown data. When you click on an unrecognized cell, a search box appears. As you begin typing in the search box, Tableau generates a list of possible locations. Select a location from the list.

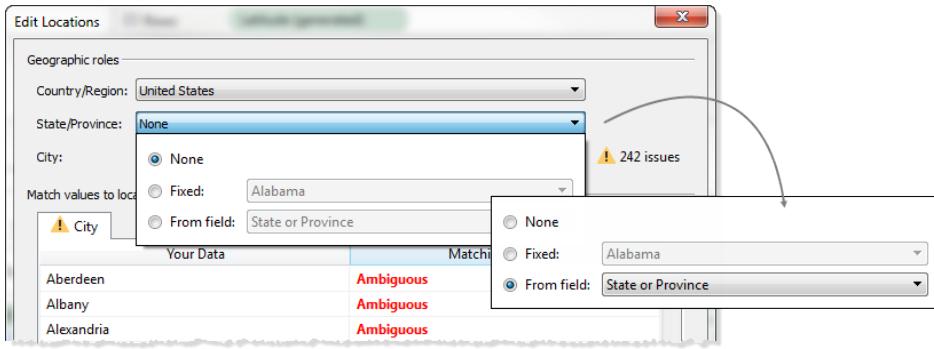


Alternatively, you can select **Enter Latitude and Longitude** from the drop-down menu to manually map the value to a location on the map.

Note: When you type a latitude and longitude, enter the values in decimal format.

Ambiguous Locations

Both unrecognized and ambiguous locations can be listed in the Edit Locations dialog box. You can usually fix ambiguous locations by specifying the **Country/Region** and/or **State/Province** options in the **Geographic Roles** section of the Edit Locations dialog box. For example, if you have several cities that are ambiguous, you can specify a State/Province to fix them.



Restrict Locations in the List

By default, all possible locations are listed in the drop-down menu. To restrict the list to locations that have not yet been matched to your data, select the **Show only unmatched locations in drop-down list** option in the bottom left corner of the Edit Locations dialog box.

See also:

[Prepare Your Geographic Fields](#) on page 735

Customize How Your Map Looks

When you create a map view there are several options to help you control the appearance of the map. These options are available in the **Map Layers** pane.

To open the Map Layers pane:

- Select **Map > Map Layers**.

Use the **Map Layers** pane to modify how the map appears. You can change the map background style, hide and show map layers, such as street names and country borders, and add data layers.

Note: In previous versions of Tableau, the **Map Layers** pane was called the **Map Options** pane.

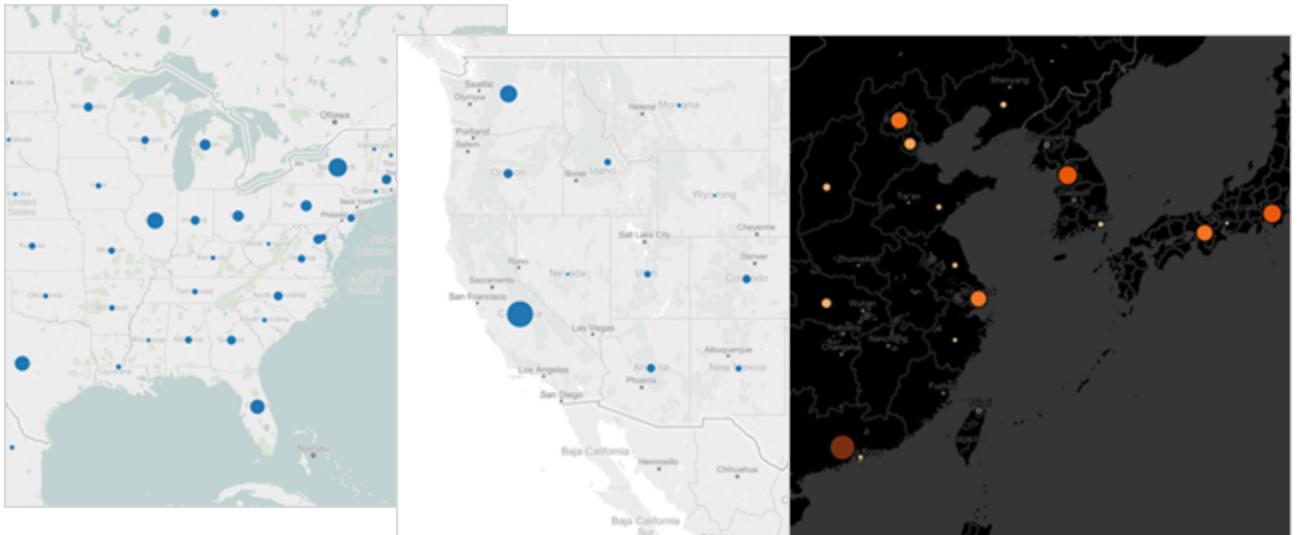
Select a background style

When you are connected to the Tableau background map, you can choose between three background styles (Normal, Light, or Dark) to customize the appearance of your map. You can see the three styles below:

Normal

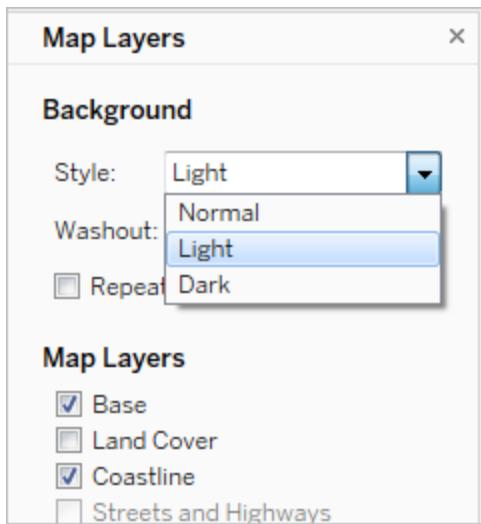
Light

Dark



To change the map background style:

1. Select **Map > Map Layers**.
2. Under Background, click the **Style** drop-down menu , and then select a background style.

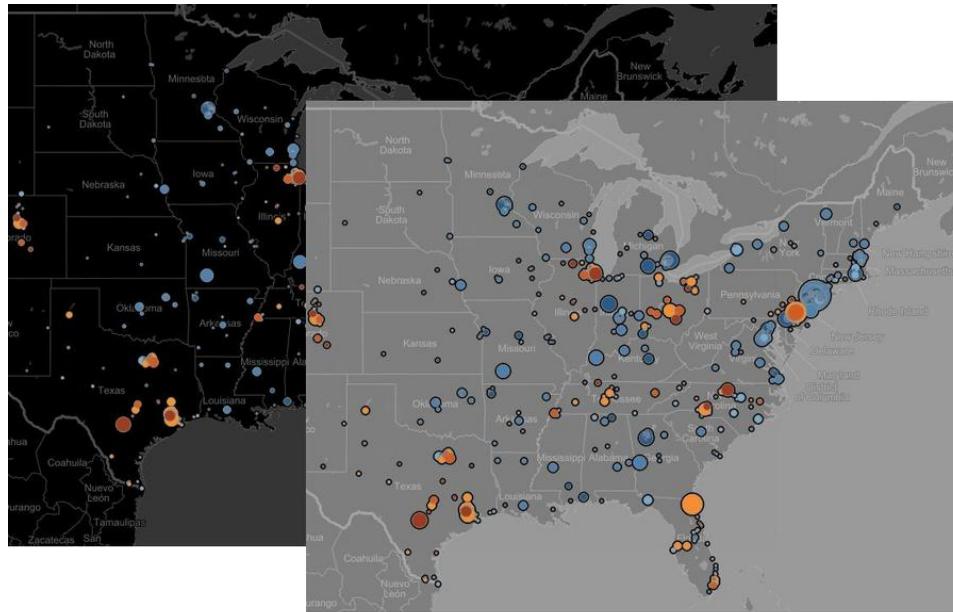


Change the map intensity

In addition to the background map style, you can use the **Washout** slider to control the intensity, or luminance, of the background map.

1. Select Map > Map Layers.
2. Under Background, move the washout slider to the left or right to change the intensity of the background map.

The farther the slider moves to the right, the more faded the map becomes.



Repeat the map background

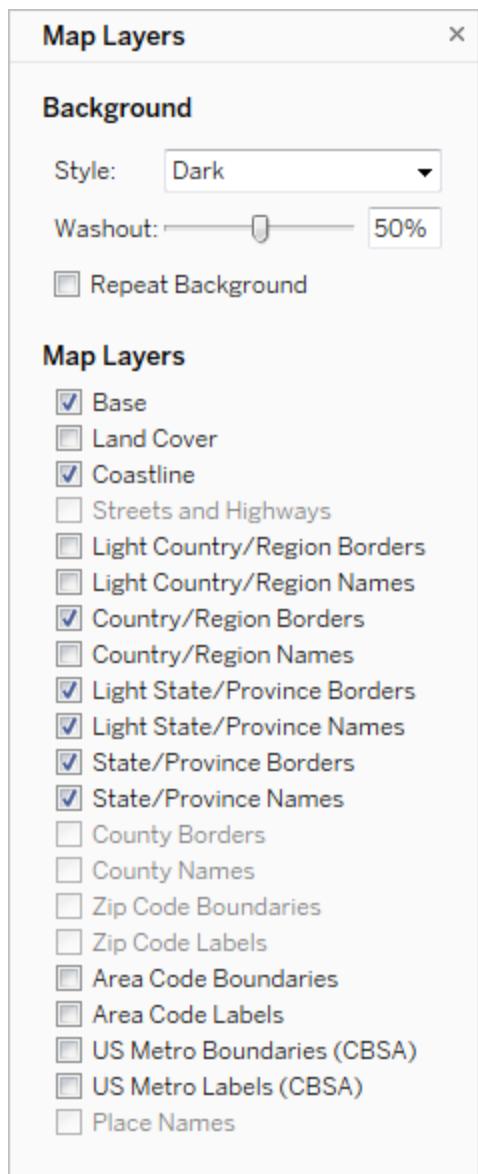
You can select whether to repeat the background. When the **Repeat Background** option is selected, the background map may show the same area multiple times depending on where the map is centered.

Add or Remove Map Layers

Tableau maps provide several layers that can mark points of interest on the map. For example, you can overlay streets and highways, county boundaries, and more on the map to give your data context.

To add or subtract map layers from your map view:

1. Select **Map > Map Layers**.
2. Under Map Layers, select one or multiple map layers to add to the view.



Note: Some map layers are only visible at specific zoom levels. If a map layer is unavailable at your current level of zoom, it will appear grayed out.

To use layers that are not available, zoom further in to the view.

The map layers that come automatically with the Tableau background map are described in the table below.

Layer Name	Description
Base	Shows the base map including water and land areas.
Land Cover	Shades wilderness areas and parks to give the map more depth.
Coastlines	Displays coastlines.
Streets and Highways	Marks freeways and highways as well as small city streets. This layer includes the highway and street names as well.
Light Country/Region Borders	Shows a light gray outline of country/region borders and names.
Light Country/Region Names	Shows country and region names in a light gray.
Country/Region Borders	Highlights country and region borders in a darker gray.
Country/Region Names	Highlights country and region names in a darker gray.
Light State/Province Borders	Shows a light gray outline of state borders and names.
Light State/Province Names	Shows state and province names in a light gray.
State/Province Borders	Highlights state and province borders in a darker gray.
State/Province Names	Highlights state and province names in a darker gray.
US County Borders	Highlights U.S. county borders.
US County Names	Highlights U.S. county names.
Zip Code Boundaries	Marks U.S. zip code boundaries. You must zoom in to see this layer.
Zip Code Labels	Shows labels for U.S. zip codes. You must zoom in to see this layer.
Area Code Boundaries	Marks the U.S. area code boundaries. You must zoom in to see this layer.

Layer Name	Description
Area Code Labels	Shows labels for the U.S. area codes. You must zoom in to see this layer.
US Metro Boundaries (CBSA)	Marks U.S. Metropolitan Statistical Areas and Micropolitan area boundaries.
US Metro Labels (CBSA)	Shows labels for the U.S. Metropolitan Statistical Areas and Micropolitan areas.
Place Names	Displays the names of places from country/region names and borders to city names, bodies of water, parks, universities, and more. This layer is dependent on the zoom level.

Add Data Layers

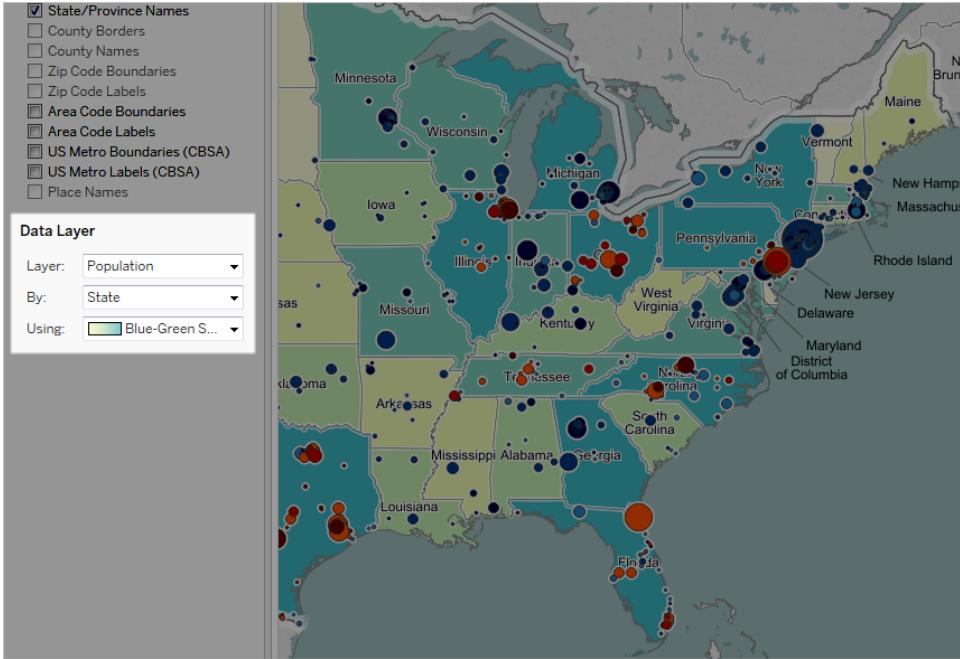
In addition to map layers, you can turn on a variety of predefined data layers that show U.S. census information.

Note: Map data layers are only available for locations in the U.S.

If you would like to add data layers for locations outside the U.S. you can connect to a WMS server that contains the desired demographic information. For more information, see [Use Web Map Service \(WMS\) Servers](#) on page 782.

To add a data layer to your map view:

1. Select **Map > Map Layers**
2. In the **Data Layer** section at the bottom of the **Map Layers** pane, do the following:
 - Click the **Layer** drop-down menu and select a data layer.
 - Click the **By** drop-down menu to layer the data by **State, County, Zip Code** or **Block Group**.
 - Click the **Using** drop-down menu to select a color scheme.



Once you select a data layer, it is added as shading to the map and a legend is shown to explain the colors of the layers.

To hide or show the map legend in a worksheet:

- Select **Map > Map Legend**.

Set Default Map Layer Options

If you find that you have a set of map or data layers you would always like to show, or that you always want a certain background map style when you create new workbooks or sheets, you can set the selected map layer options as the default.

To set default map layer options:

1. Select **Map > Map Layers**.
2. In the Map Layers pane, select the options you want to show by default every time you create a new map view.
3. At the bottom of the **Map Layers** pane, click **Make Default**.

Reset Map Layer Options

If you find that you want to reset the map layer options to their default settings, you can clear any selected map layer options.

To reset map layer options:

- Select **Map > Map Layers**, and then, at the bottom of the Map Layers pane, click **Reset**.

Note: Resetting map layer options returns the Map Layers pane to the default settings you configured. If you have not configured any default settings, the Map Layers pane will return to the original settings.

See also:

[Build a Map View on page 744](#)

[Use Web Map Service \(WMS\) Servers on page 782](#)

[Use Mapbox Maps on page 788](#)

[Select Background Maps on page 800](#)

Create Territories on a Map

When you're analyzing data on a map view, you might want to group *existing* locations together to create your own territories or regions, such as sales territories for your organization.

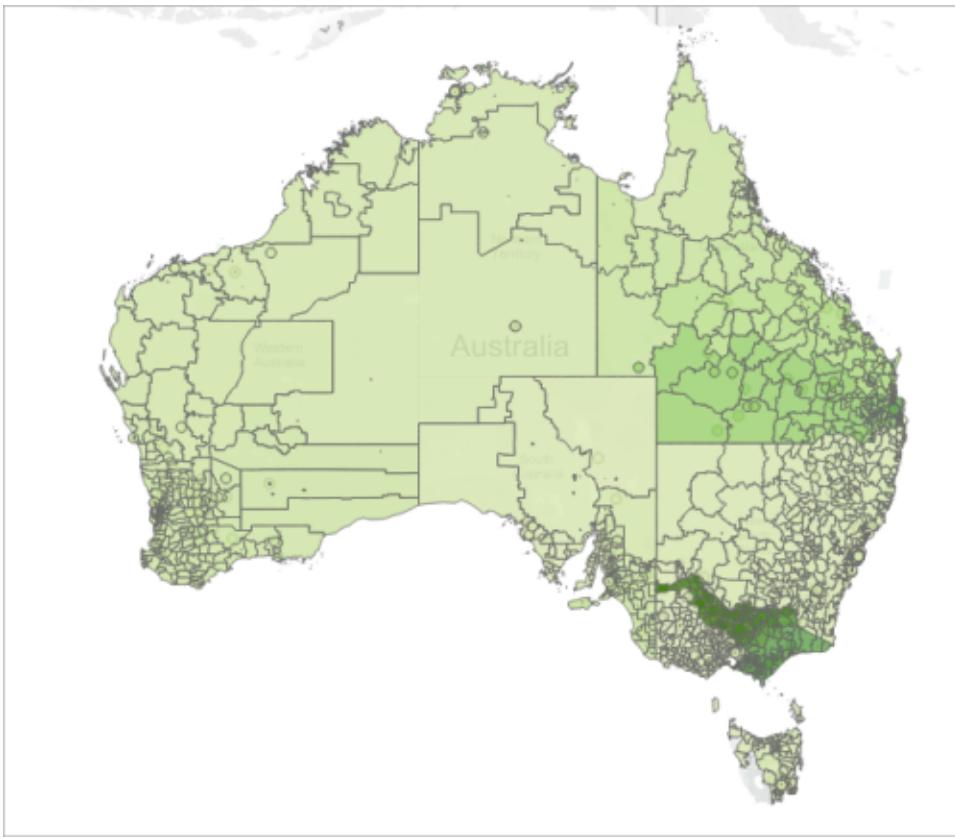
In Tableau, there are several ways to create territories:

- [Option 1: Select and group locations on a map](#) below
- [Option 2: Create a territory from a geographic field](#) on page 778
- [Option 3: Geocode a territory field using another geographic field](#) on page 778

Option 1: Select and group locations on a map

If you have created a map view, you can select and group locations (marks) on the map to create your territories.

For example, the following map view shows the total sales by postcode for a company with sales across Australia.



The first thing you might be thinking is: That's a lot of postcodes! In fact, there is a mark in this map view for every single 4-digit postcode in Australia, and each of those marks has an aggregation for SUM of Sales. If you're looking for sales information for every single postcode in Australia, than this is the view for you.

However, you might not want to analyze every individual location in your map view. Instead you might have your own areas, regions, or territories that you want to analyze as a whole. For example, maybe your organization splits Australia into five different sales territories, and you want to analyze the total sales for each of them.

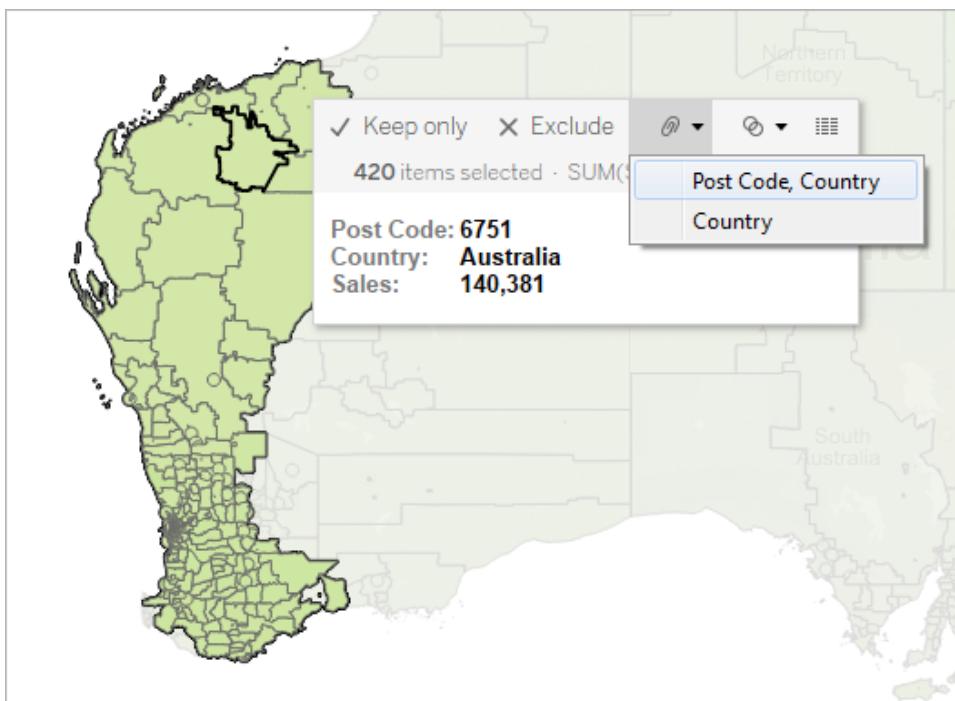
In this case, because you've already created a map view, you can select locations directly on the map and group them to create your own territories. Follow the steps below to learn how.

: Select locations on the map to group into a territory

On the map view, select the locations (marks) that you want to group into your own territory. For more information about how to select marks, see [Select Marks on page 566](#).

: Create the first group (your first territory)

When you select marks in the view, or hover over a selection, a tooltip appears. On that tooltip, click the Group icon, and then select the level of detail you want to group the locations by.

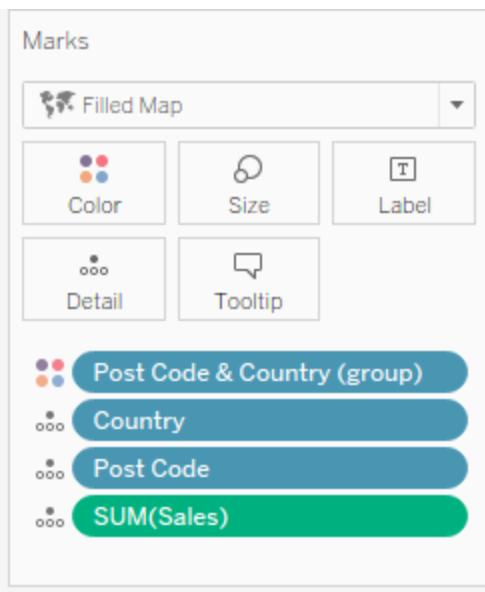


A new group field is added in the **Data** pane with a group icon next to it. To change the locations within the group at any time, right-click the group field in the **Data** pane and select **Edit group**.

Dimensions ▼

- =🌐 Country
- =🌐 Post Code
- 🌐 Post Code & Country (group)

The new field is also added to **Color** on the **Marks** card.



Notice that the marks you selected are now all the same color.

: Continue grouping data to add territories

Repeat steps 1 and 2 to create additional territories. Each territory is added as a group within the group field you created in step 2. You can create as many territories as you want.

: Add the group field to the view

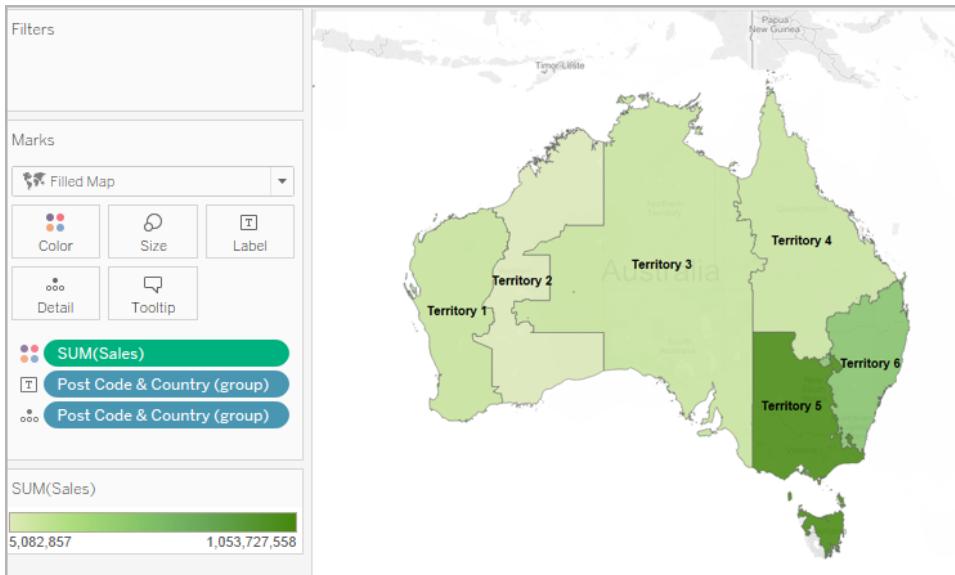
From the **Data** pane, drag the newly created group field to **Detail** on the **Marks** card. Next, to make each territory appear as a single mark, you must also remove the related geographic field(s) from the view.

For example, if you grouped postcodes at the **Post Code, Country** level, you must remove the **Post Code** and **Country** fields from the view.

In the image below, notice that the **Post Code** and **Country** fields are no longer on **Detail** on the **Marks** card because they have been replaced by the **Post Code & Country (group)** field.

In this example, there are five groups within the Post Code & Country (group) field, so five territories appear on the map. Aggregations, such as SUM (Sales), are computed at the level of the territory, rather than separately for each location within the territory.

You can now see the SUM (Sales) for each territory by color. Territory 5 has the most sales.



Option 2: Create a territory from a geographic field

You can also create territories by creating groups in the **Data** pane.

1. In the **Data** pane, right-click a geographic field (such as City or State) and select **Create > Group**.
 2. In the Create Group dialog box, select the locations you want in your first territory and click **Group**. Each group you create represents a territory.
 3. Repeat step 2 until you've created all of your territories, and then click **OK**.
- The new group field is added to the **Data** pane.
4. From the **Data** pane, drag the newly created group field to **Detail** on the **Marks** card.

You can also drag the field to **Color** or **Label** on the **Marks** card to help differentiate between each territory.

Change locations in your territory groups

If you want to change the locations in your territory groups at any time, right-click the group field in the **Data** pane and select **Edit group**.

Option 3: Geocode a territory field using another geographic field

If your data source already contains a custom territory field , you can quickly geocode it to make a map in Tableau using an existing geographic field in your data source.

For example, the following data source contains a custom territory field for geographic areas called **Zone géographique**. It contains the values, **North**, **Central** and **South**.

If you were to bring this field into Tableau by itself, Tableau would not be able to recognize it as geographic data.

However, this data source also contains three geographic fields that Tableau *does* recognize: **Country, Region, and City**.

Each row in this data source contains a value for country, region, city and geographic zone. Therefore, each location is assigned a value from the custom territory field.

 Achats Ville	 Achats Région	 Achats Pays	 Achats Zone géographique
Leeds	Angleterre	Royaume-U...	Nord
Leeds	Angleterre	Royaume-U...	Nord
West Bromwich	Angleterre	Royaume-U...	Nord
West Bromwich	Angleterre	Royaume-U...	Nord
West Bromwich	Angleterre	Royaume-U...	Nord
Le Bouscat	Aquitaine	France	Centre
Le Bouscat	Aquitaine	France	Centre
Le Bouscat	Aquitaine	France	Centre

Because the Zone géographique field is included in the workbook with other geographic fields that contain locations that Tableau recognizes, you can geocode it at the level of detail of one of those fields.

Follow the steps below to geocode a territory field using another geographic field:

: [Assign a geographic role to the territory field](#)

On the data source page, click the data type icon for the custom territory field and select **Geographic Role > Create from**, and then select a level of detail to aggregate the field.

 Achats Pays	 Achats Zone géographique
Royaume-U...	Nord

The options you see are based on the available geographic fields in your data source. Choose the level of aggregation you want in the view. If you're unsure how they will affect your view, try several different levels of detail. You can always undo and try again.

After you assign a geographic role to the territory field, it's added to its respective place in the location hierarchy. You can see it in the **Data** pane when you go to any worksheet.

The screenshot shows the 'Dimensions' section of the Tableau Data pane. The 'Lieu' category is expanded, displaying four levels of geography: Zone géographique, Pays, Région, and Ville. Other dimensions listed include Date d'expédition, Date de commande, ID commande, Mode d'expédition, Produit (with sub-categories Catégorie, Sous-catégorie, Fabricant, Nom du produit, Profit (compartiment), and Noms de mesures), and Date.

Note: You can also click the data type icon next to the field in the **Data** pane to geocode a territory field using the above procedure.

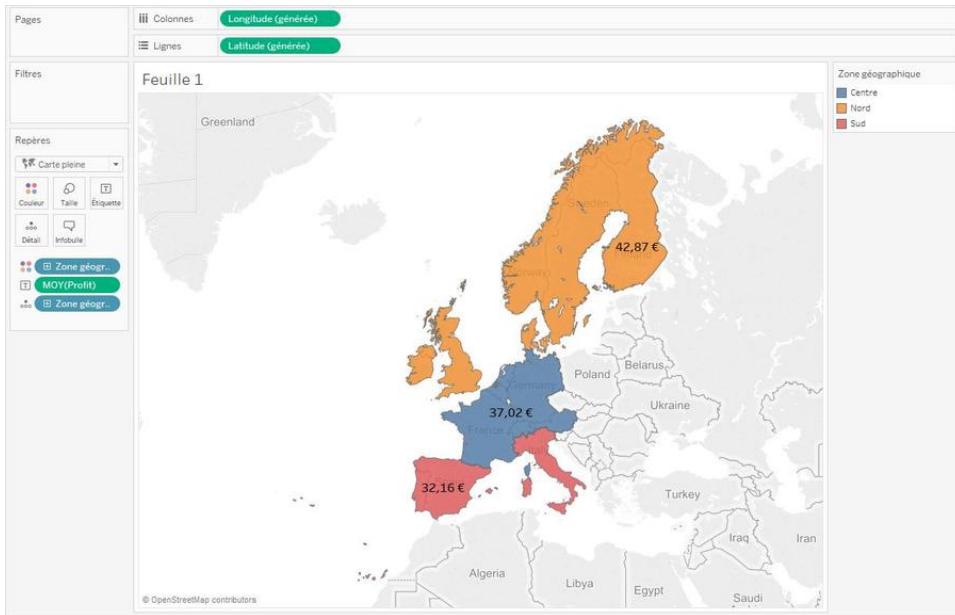
: Add the territory field to the view

Create a map view and add the newly geocoded territory field to the **Marks** card. You can place it on **Detail**, **Color**, **Label**, or **Tooltip**. Note that fields on **Detail** determine the level of aggregation in the view.

The custom territory field functions similar to your other geographic fields.

In the example below, the custom territory field (Zone géographique) is on **Detail** and **Color** on the **Marks** card. The average profit is shown for each territory (since the lowest level of detail is the territory field).

If you add another geographic field to **Detail** on the **Marks** card, such as the country field (Pays), the aggregation will update to show information for that level of detail.



Blend custom territory data

If your data source does not contain a territory field, you can blend a territory field with another geographic field in your data source. For more information, see [Blend Geographic Data](#) on page 727. After you have blended your custom territory field with another field, follow the procedure in [Step 1: Assign a geographic role to the custom territory field](#) to geocode your territory field.

Customize How People Interact with your Map

When you create a map view, there are several default ways you can explore and interact with the view. You can zoom in and out of the view, pan, select marks, and even search for locations worldwide with map search.

However, sometimes you want to limit some of the ways your audience can interact with your map.

You can customize how your audience interacts with your view in the following ways using the **Map Options** dialog box.

Hide map search

You can hide the map search icon so your audience cannot search for locations in your map view.

To hide the map search icon:

1. Select **Map > Map Options**.
2. In the Map Options dialog box that appears in the view, clear **Show Map Search**.

Note: When you hide the map search icon, you also hide the locate me button for views published on the web. For more information about the locate me button, see [Interact with Maps](#) in the Tableau Online Help.

Hide the view toolbar

You can hide the view toolbar in a map view so your audience cannot lock the map in place or zoom the map to all of your data.

To hide the view toolbar:

1. Select **Map > Map Options**.
2. In the Map Options dialog box that appears in the view, clear **Show View Toolbar**.

Note: When you hide the view toolbar, you can still use keyboard shortcuts to zoom in and out of the view, pan, and select marks. For more information, see [View Toolbar](#) on page 565.

Turn off pan and zoom

You can turn off pan and zoom in your map view, as well as in background images, so your audience cannot pan, or zoom in or out of the view.

To turn off pan and zoom:

1. Select **Map > Map Options**.
2. In the Map Options dialog box that appears in the view, clear **Allow Pan and Zoom**.

When you turn off pan and zoom, the pan tool and all zoom controls are removed from the view toolbar, and the rectangular tool becomes the default tool. Keyboard shortcuts for zooming in and out of the view, or panning, no longer work. If the map search icon is not hidden, you can still navigate the view by searching for locations with the map search box.

See also:

[Search for Locations in Your Map](#) on page 793

[View Toolbar](#) on page 565

[Select Marks](#) on page 566

[Pan and Zoom](#) on page 569

Use Web Map Service (WMS) Servers

Tableau can connect to map servers with the Web Map Service (WMS) protocol. WMS is a standard protocol for requesting and receiving geographically referenced imagery.

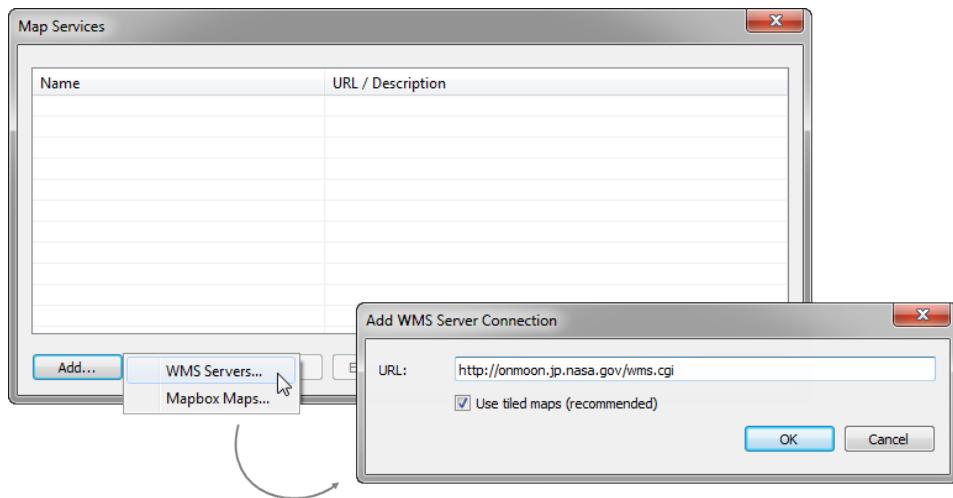
In Tableau Desktop, you can connect to any WMS server that supports the WMS 1.0.0, 1.1.0, or 1.1.1 standards.

In this topic

- Connect to a WMS server in Tableau Desktop
- Use a WMS background map to build a map view in Tableau
- Save a WMS server as a Tableau map source
- Performance considerations for WMS servers in Tableau
- Supported Spatial Reference Systems for WMS servers

Connect to a WMS server

1. Select **Map > Background Maps > Map Services**.
2. In the Map Services dialog box, click **Add > WMS Servers**.
3. In the Add WMS Servers dialog box, type the URL for the server you want to connect to in Tableau, and then click **OK**.



You can add as many map servers as you want to a workbook. Each WMS server you add appears as a background map in the **Background Maps** menu.

Save a WMS server as a Tableau map source

After you add a WMS server to your workbook, it is saved with the workbook and available to anyone you share the workbook with. You can also save a WMS server as a Tableau Map Source (.tms) file, which you can share with others so they can quickly connect to it and use it in their own workbooks.

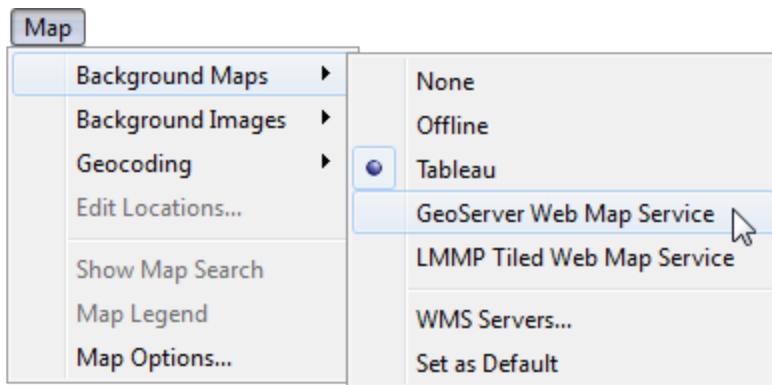
For more information about how to save a map as a Tableau map source, see [Save a Map Source](#) on page 799.

Use a WMS background map

After you connect to a WMS server, you can create a map view using the WMS background map that Tableau creates.

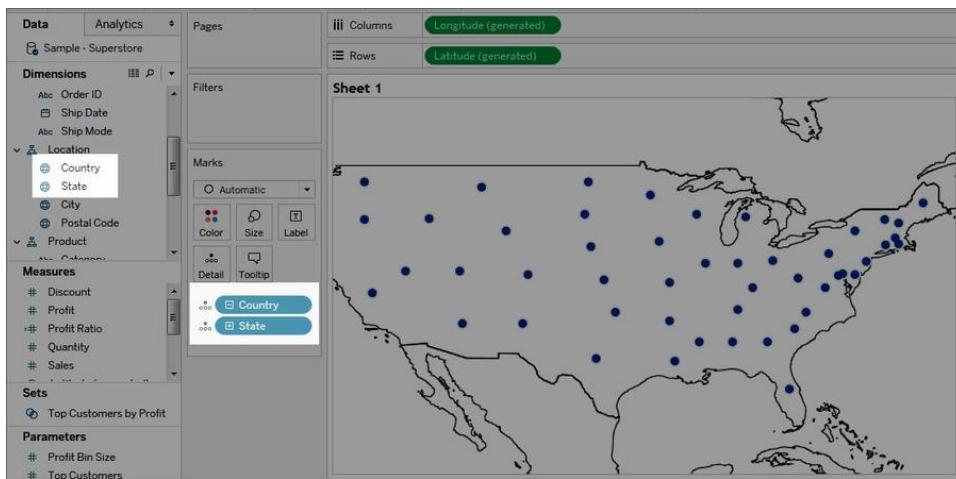
To create a map view using a WMS server:

1. In Tableau, select **Map > Background Maps**, and then select a WMS background map to use in the view.



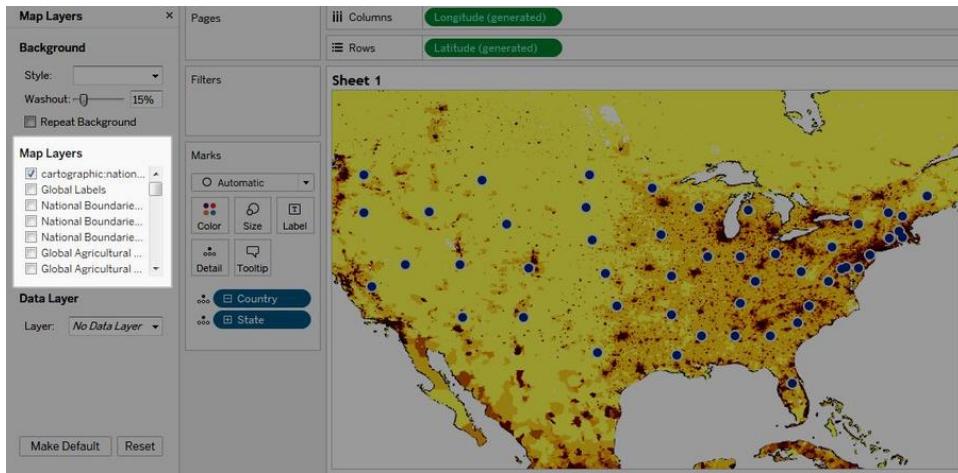
2. Add a geographic field to the view.

For more information, see [Build a Map View](#) on page 744.



3. Select **Map > Map Layers**, and then select the map layers you want to show in the view.

For more information, see [Customize How Your Map Looks](#) on page 767.



Performance considerations

The content, speed, and performance of a WMS server is reliant on the network and WMS provider. If your Internet connection is interrupted or if you're working behind a restrictive firewall, you can use the Offline background map that comes installed with Tableau Desktop to avoid any performance issues while you build your map view. You can switch back to your WMS server at any time.

Note: With the offline background map, you might only be able to access up to four levels of zoom. For more information, see [Select Background Maps](#) on page 800.

Supported Spatial Reference Systems

Tableau automatically supports the following list of Spatial Reference Systems (SRS) / European Petroleum Survey Group (EPSG) codes. WMS servers must support at least one of the following spatial reference systems to be compatible with Tableau mapping features.

Supported SRS/EPSC Codes in Tableau:

- 3857 - WGS 84 / Pseudo-Mercator [DEFAULT]
- 4326 - WGS 84
- 4269 - NAD83
- 3824 - TWD97
- 3889 - IGRS
- 4019 - Unknown datum based on the GRS 80 ellipsoid

4023 - MOLDREF99
4030 - Unknown datum based upon the WGS 84 ellipsoid
4031 - Unknown datum based upon the GEM 10C ellipsoid
4046 - RGRDC 2005
4075 - SREF98
4081 - REGCAN95
4126 - LKS94 (ETRS89)
4130 - Moznet
4140 - NAD83(CSRS98)
4148 - Hartebeesthoek94
4151 - CHTRF95
4152 - NAD83(HARN)
4163 - Yemen NGN96
4166 - Korean 1995
4167 - NZGD2000
4170 - SIRGAS 1995
4171 - RGF93
4172 - POSGAR
4173 - IRENET95
4176 - Australian Antarctic
4180 - EST97
4189 - REGVEN
4190 - POSGAR 98
4258 - ETRS89
4283 - GDA94
4319 - KUDAMS
4612 - JGD2000
4617 - NAD83(CSRS)
4619 - SWEREF99
4627 - RGR92

4640 - RRAF 1991
4645 - RGNC 1991
4659 - ISN93
4661 - LKS92
4667 - IKBD-92
4669 - LKS94
4670 - IGM95
4674 - SIRGAS 2000
4686 - MAGNA-SIRGAS
4687 - RGPF
4693 - Nakhl-e Ghanem
4694 - POSGAR 94
4702 - Mauritania 1999
4737 - Korea 2000
4742 - GDM2000
4747 - GR96
4749 - RGNC91-93
4755 - DGN95
4756 - VN-2000
4757 - SVY21
4758 - JAD2001
4759 - NAD83(NSRS2007)
4761 - HTRS96
4762 - BDA2000
4763 - Pitcairn 2006
4764 - RSRGD2000
4765 - Slovenia 1996
102100 - WGS 84 Web Mercator (Auxiliary Sphere)

See also:

[Use Mapbox Maps](#) on the next page

[Save a Map Source](#) on page 799

[Import a Map Source](#) on page 800

[Select Background Maps](#) on page 800

[Customize How Your Map Looks](#) on page 767

Use Mapbox Maps

If you have access to Mapbox maps, you can add them to your workbooks or use them to create map views in Tableau Desktop.

When you publish a view that uses Mapbox maps to Tableau Server, Tableau Online, or Tableau Public, your audience can view your data and your Mapbox map without having a Mapbox account.

Add a Mapbox map to your workbook

You can add a Mapbox map to your workbook and use it as a background map.

After you add a Mapbox map to your workbook, the map is saved with the workbook and available to anyone with whom you share the workbook. You can also save a Mapbox map as a Tableau Map Source (.tms) file that you can share with others so they can quickly connect to it and use it in their own workbooks. For more information, see [Save a Map Source](#) on page 799.

1. Select **Map > Background Maps > Map Services**.
2. In the Map Services dialog box, select **Add > Mapbox Map**.
3. In the Add Mapbox Map dialog box, you can add a Mapbox GL map or a classic Mapbox map. To do so, you need the following from your Mapbox account:
 - API access token
 - Username
 - One or more map IDs
 - Style URL (Mapbox GL maps Only)

For more information about any of the above items, see the [Access Tokens, Maps](#), and [Styles](#) sections of the Mapbox API Help.

Add a Mapbox GL map

To add a Mapbox GL map, click **Mapbox GL** in the Add Mapbox Map dialog box, and then do the following:

Note: Mapbox GL is selected by default.

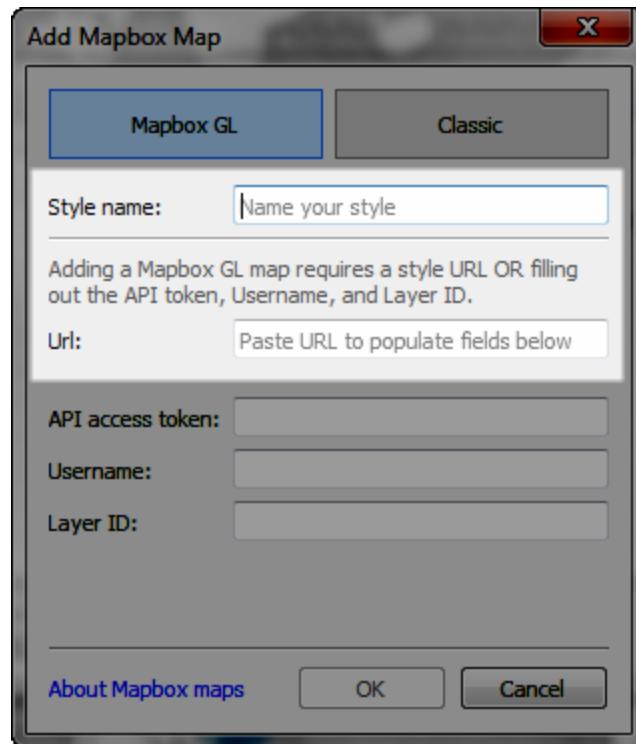
- For **Style name**, enter a name for the Mapbox map. This name can be anything you want, and will appear in the Background maps menu after you add the map.
- For **Url**, enter a style URL for the Mapbox map you want to add.

This URL contains the style ID for your Mapbox map, your access token, and your username. It might look similar to the following:

```
https://api.mapbox.com/styles/v1/<username>/<styleid>
?access_token=<access token>
```

When you add the correct style URL to this field, the **API access token**, **Username**, and **Layer ID** fields automatically populate.

If you don't have a style URL for the Mapbox map, you must enter a Mapbox access token, username, and layer (map) ID to add your Mapbox map.

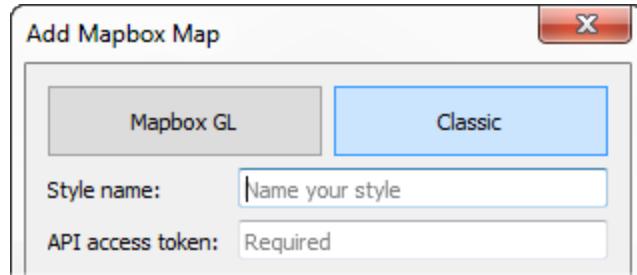


Add a classic Mapbox map

To add a classic Mapbox map, click **Classic** in the Add Mapbox Map dialog box, and then do the following:

- For **Style name**, enter a name for the Mapbox map. This name can be anything you want, and will appear in the Background maps menu after you add the map.

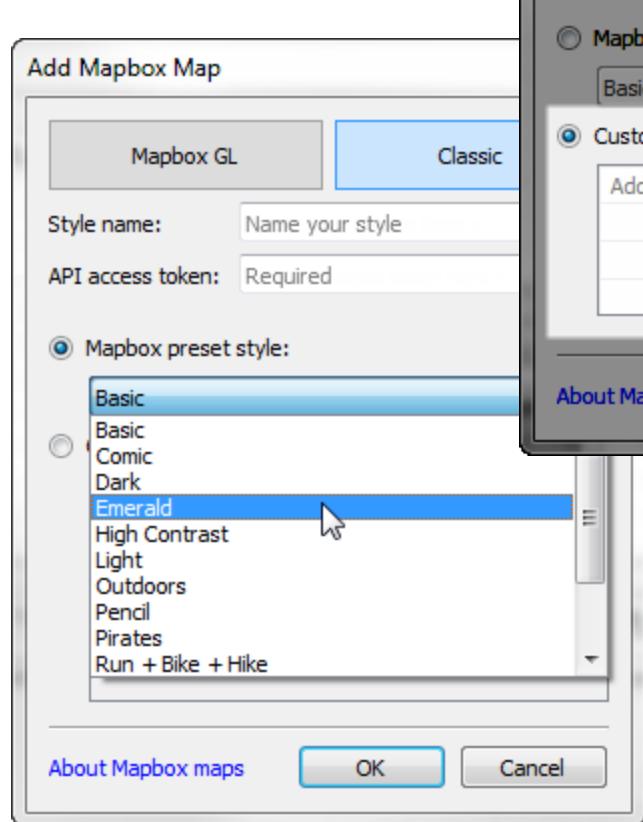
- For **API access token**, enter the API access token for the Mapbox map you want to add.



- Choose a Mapbox preset style, or add one or more custom map layers:

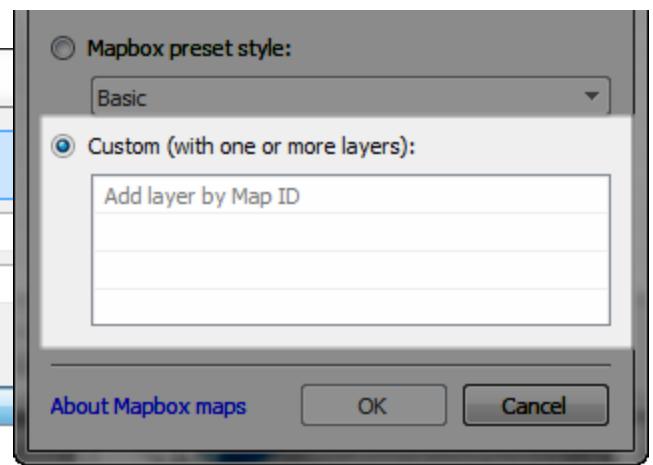
To use a Mapbox preset style:

Click **Mapbox preset style**, and then select a style from the drop-down menu.



To add one or more map layers:

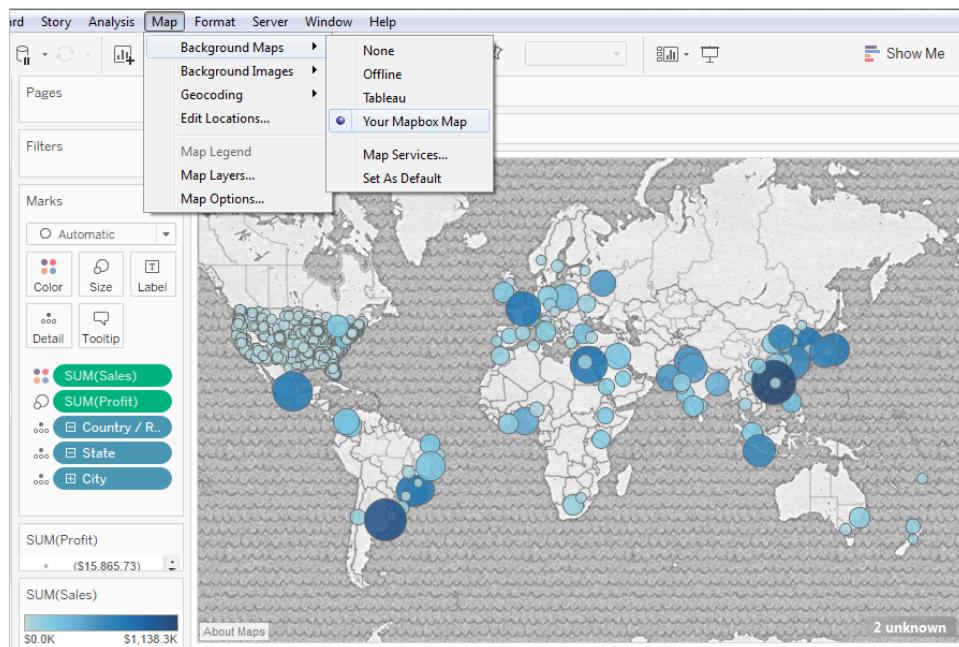
Click **Custom**, and then enter one or more map IDs in the space indicated.



- When finished, click **OK** to exit the Add Mapbox Map dialog box, and then click **Close** to return to the view.

Create a map view using a Mapbox map

To create a map view using a Mapbox map, select **Map > Background Maps**, and then select the Mapbox map you want to use. Next, build the map view. For information, see [Build a Map View on page 744](#).



Add or subtract map layers from the view

If you chose to add one or more custom layers by map ID when you connected to your Mapbox map in Tableau, you can add or subtract those layers from the view using the **Map Layers** pane. For more information, see [Add or Remove Map Layers on page 769](#) in the Customize How Your Map Looks topic.

By default, all map layers appear in the view when you first add a Mapbox map to a workbook.

Note: When you use a Mapbox preset style, you can't toggle the layers within the map.

For example, if you use the Mapbox Streets map, you can't toggle streets, labels, building footprints, or administrative boundaries like you can with the Tableau map service. This is because Tableau only receives the image tiles that make up the Mapbox map.

Map layers are different from data layers in Tableau. Data Layers are pre-built tiles that include demographic information by various levels, like State, County, and Block Group. Data layers

and Mapbox are independent of one another in Tableau. For more information about how data layers work, see [Customize How Your Map Looks](#) on page 767.

See also:

[Use Web Map Service \(WMS\) Servers](#) on page 782

[Save a Map Source](#) on page 799

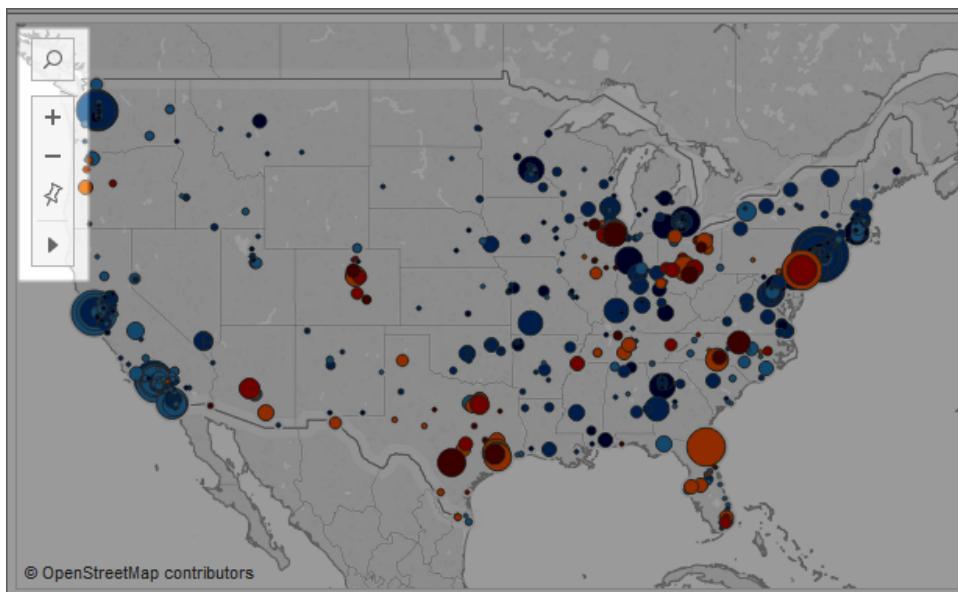
[Import a Map Source](#) on page 800

[Select Background Maps](#) on page 800

[Customize How Your Map Looks](#) on page 767

Explore Data in Maps

Tableau maps can help you quickly find locations and analyze data worldwide. There are many ways you can explore and interact with map views. You can zoom in and out, pan, and select marks with the view toolbar, and even search for locations worldwide with map search.



If you don't see the view toolbar or the map search icon, or if you can't pan or zoom, it's probably because the workbook author has chosen to customize how others can interact with the view.

The workbook author can choose to hide the view toolbar and the map search icon so they do not appear in the view. Similarly, the workbook author can choose to turn off pan and zoom to control how you interact with the view.

For more information, see [Customize How People Interact with your Map](#).

Select marks, pan, and zoom

You can use the view toolbar in the upper left corner of the view to select marks, pan, and zoom in and out of a map view. For more information, see [Select Marks and Pan and Zoom](#).

Search for locations

You can use the map search icon  to search for locations in your map view. For more information, see [Set Map Search Options](#).

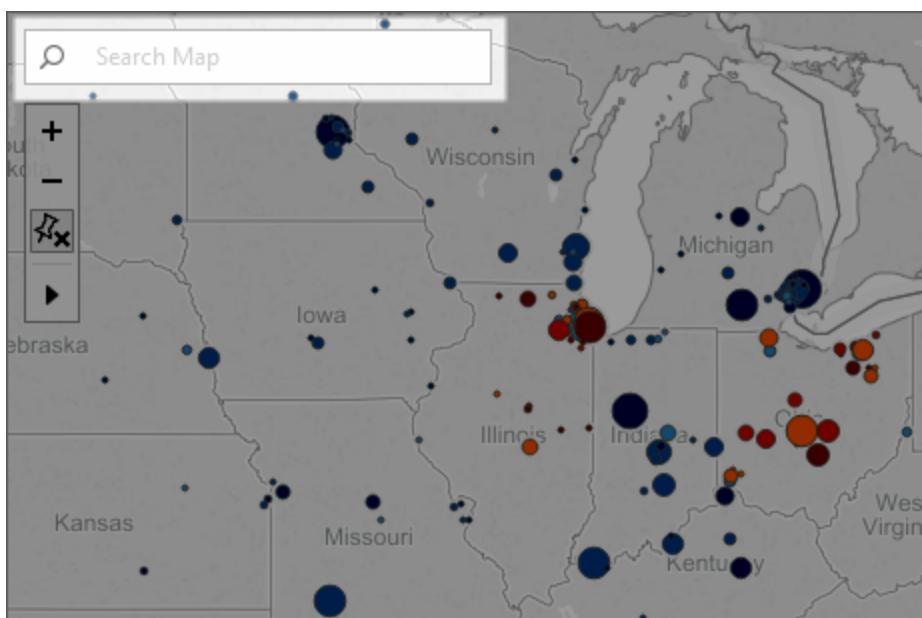
Return to the initial view

After exploring you can return to the initial view of your map. To do so, click the reset axes button  on the view toolbar.

Search for Locations in Your Map

Map search helps you find locations in a map view so you can quickly explore and inspect data.

The map search icon appears in the top left corner of the view. When you click the search icon, a search box appears.

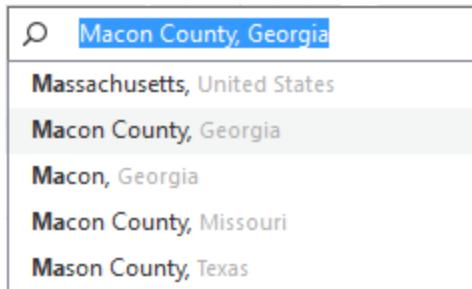


When you begin to type in the search box, map search suggests possible locations that are in your map view. The suggestions are based on location names and text in your data source.

You can search for the following location types:

- Continent
- Country
- State or province
- County
- City
- Postcode

Select a location from the list of suggestions to pan and zoom to that location on the map.



Hide Map Search

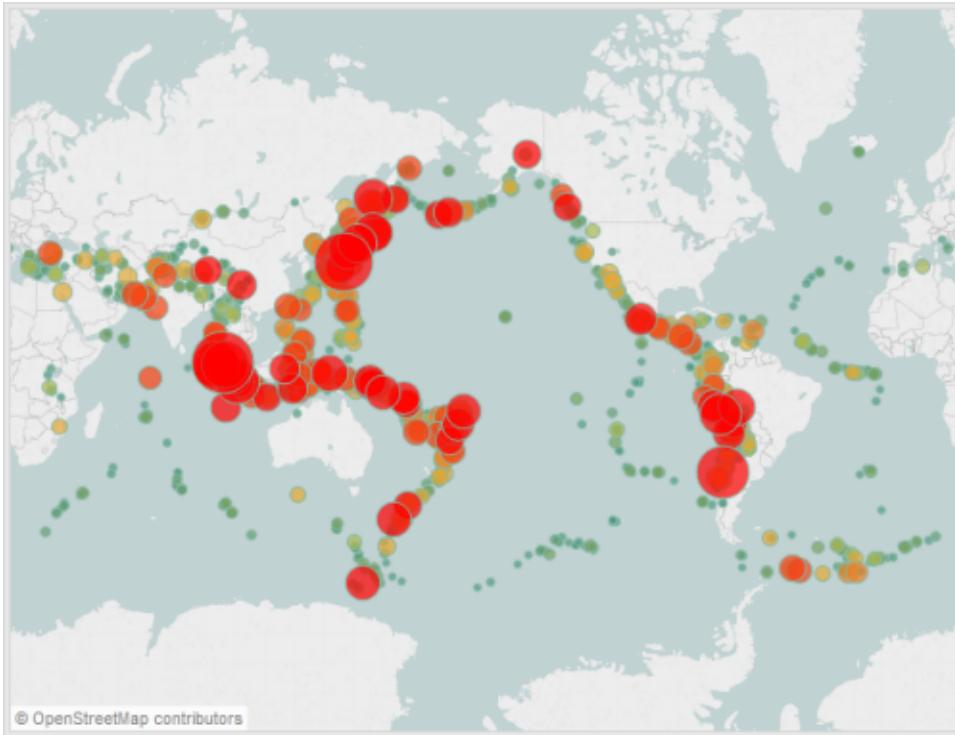
By default, the map search icon appears in the top left corner of the view. When you publish a view to Tableau Server or Tableau Online, or share the view through Tableau Reader, the map search icon remains in the view. If you do not want viewers to search for locations in your map view, you can hide the map search icon.

For more information about showing and hiding map search, see [Customize How People Interact with your Map](#) on page 781.

Measure Distances Between Data Points and Locations in a Map

While exploring data in a map view, you might have questions about how that data relates to its surrounding geography, locations, or landmarks. To answer these types of questions, you can use the Radial tool to measure approximate distances in your map view.

The following view shows the number of earthquakes of magnitude 6.0 and higher that have occurred around the world between January, 1994 and February, 2014.



At a global scale, this view is very powerful, but you or your audience might want to zoom in and explore certain areas more closely. For example, maybe you want to see how many earthquakes have occurred in Taiwan in the past 10 years, specifically within 100 miles of its capital, Taipei.

To do so, you can use the Radial tool to find all the earthquakes within approximately 100 miles of the capital city.

Follow the steps below to learn how to measure distance with the Radial tool.

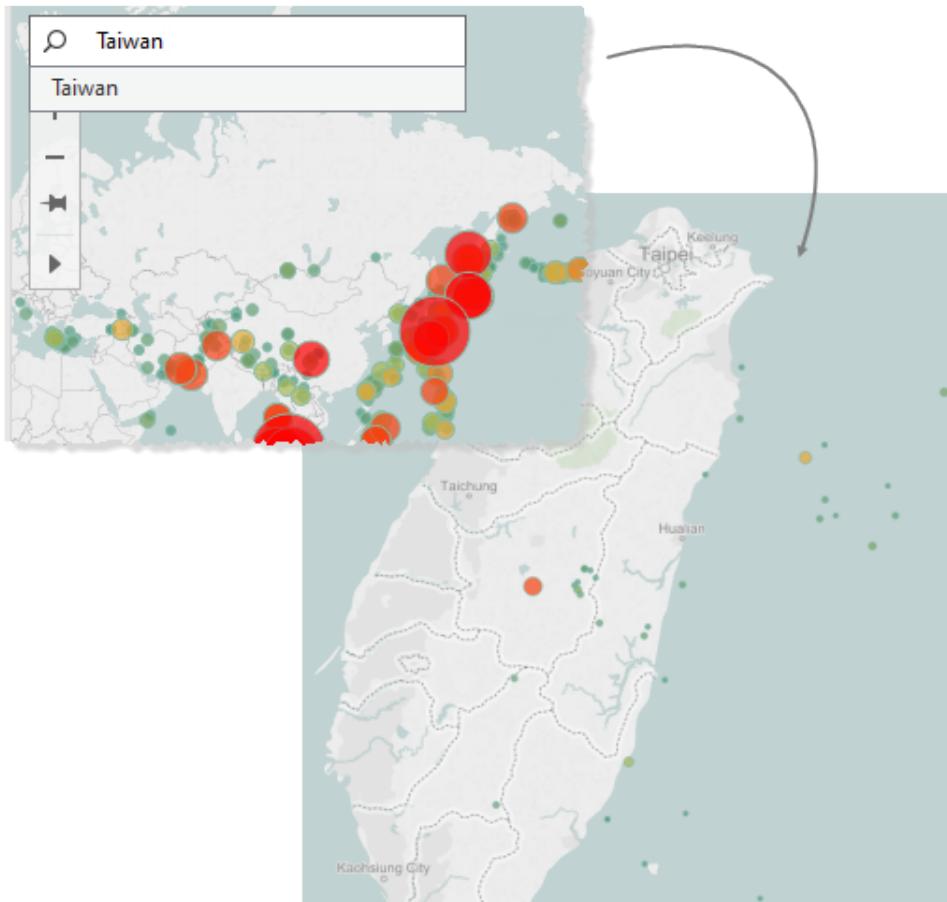
Step 1: Zoom in to an area or location

The first step to measuring distance in maps with the Radial tool is to zoom in to an area or location in the map view. For more information about how to zoom in and out of the view, see [Pan and Zoom on page 569](#).

You can also use map search to quickly navigate to a location in your map.

For more information about how to use map search, see [Search for Locations in Your Map on page 793](#).

Note: Due to the map projection, you can only measure distance at higher zoom levels. This means that you must zoom in several times before you can measure distance with the Radial tool. For more information, see the [Measurement accuracy on page 798](#) section.

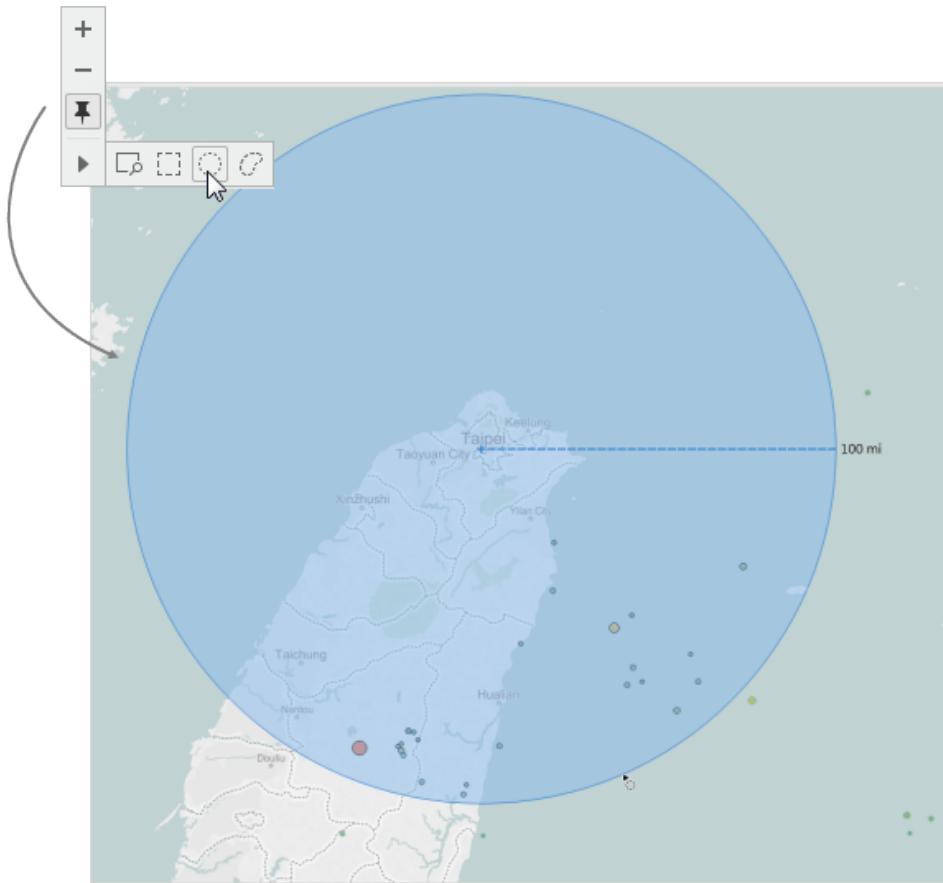


Step 2: Select the Radial tool

After you have zoomed in to a particular area or location in your view, select the Radial tool on the view toolbar, and then click and drag across the view. The measured distance appears to the right of the circle that appears when you drag across the view.

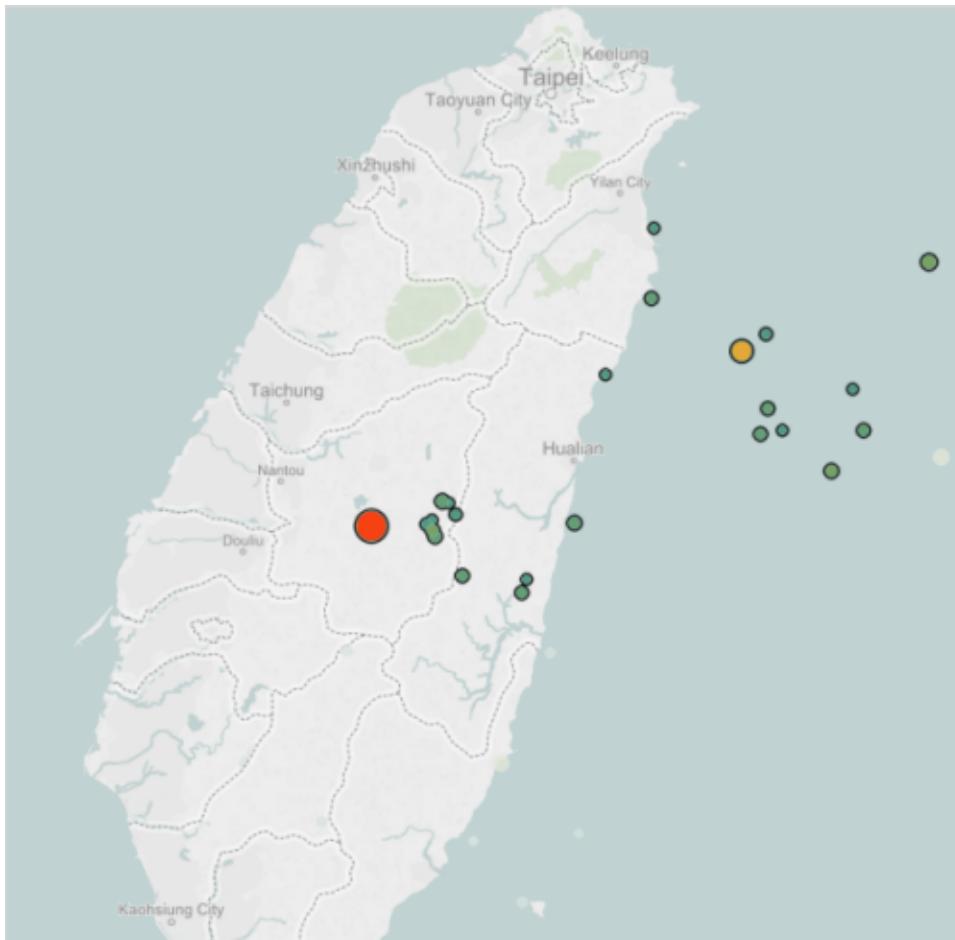
For more information about how to use the Radial tool, see [Select Marks](#) on page 566.

Note: If you do not see a measured distance, you must zoom in further to a location or area in the view.



Note: If the view toolbar is hidden, press **S** on your keyboard to use the Radial tool.

As you drag, the Radial tool selects marks that are located within the radius of the circle. In this example, the radius is 100 miles, and it is centered over Taipei. This means that, according to this data, all of the selected earthquakes (25) have occurred within approximately 100 miles of Taipei in the past 10 years.



Measurement accuracy

By default, Radial tool measurements have a small margin of error because, in the map projection, distances become exaggerated and stretch as you move away from the equator. This means that the Radial tool can only measure approximate distances.

The Radial tool can measure distance more accurately the closer you are to the equator, and the further you zoom in to the view.

Note: By design, the Radial tool does not display a measured distance when you are zoomed too far out of the view, because the measurement might be inaccurate.

Change the units of measurement

By default, your workbook locale determines which units the Radial tool uses to measure distance. If your workbook locale is set to a country that uses the Imperial system, the Radial tool measures distance in feet and miles. If your workbook locale is set to a country that uses the Metric system, the Radial tool measures distance in meters and kilometers.

You can change the units the Radial tool uses to measure distance for any map view in your workbook.

To do so, select **Map > Map Options**. Next, under **Units**, click the drop-down menu and choose from the following:

- If you want to measure distances in meters and kilometers, select **Metric**.
- If you want to measure distances in feet and miles, select **U.S.**.
- If you want the units of measurement to be determined by your workbook locale, select **Automatic**.

The units you choose will be saved with the workbook and will remain if you publish the map view to Tableau Server, Tableau Online, or Tableau Public.

Save a Map Source

After you add a Mapbox map or WMS server to your workbook, you can save it as a Tableau map source (.tms).

Saving a Mapbox map or WMS server as a Tableau Map Source allows you to share your map with others so they can quickly import it into their own workbooks and use it to create new map views. For more information about importing a Tableau Map Source into a workbook, see

[Import a Map Source](#) on the next page.

1. Select **Map > Background Maps > Map Services**. This opens the Map Services dialog box.
2. Select the map that you want to save as a Tableau map source, and then click **Export**. This opens the Export Connection dialog box.
3. Type a name for the file, choose a location, and then click **Save**.

The Tableau map source includes any default map layer settings you have specified in the workbook. For example, the map source will include any custom set of map layers you have specified to show by default. For more information, see [Set Default Map Layer Options](#) on page 773 in the Set Map Layers topic.

Note: If you change the default settings for the map layer options, you should export the map again to include the new settings in the Tableau map source file.

See also:

[Select Background Maps](#) on the next page

Import a Map Source

You can import a Tableau Map Source (.tms) that someone has shared with you into a workbook, and then use it to create custom map views.

1. Select **Map > Background Maps > Map Services**. This opens the Map Services dialog box.
2. Click **Import**. This opens the Import Connection dialog box.
3. Navigate to the saved Tableau map source file (.tms) that you want to import, select it, and then click **Open**. This closes the Import Connection dialog box and adds the Tableau map source to the list of maps in the Map Services dialog box.

If you want to edit the newly added Tableau map source, select it from the list, and then click **Edit**. Otherwise, click **Close** to return to the view.

You can import as many Tableau map sources as you want into a workbook. Each Tableau map source you add appears as a background map in the **Background Maps** menu and is automatically selected as the default background map until another Tableau map source is imported, until you connect to a WMS server or Mapbox map, or until you select to use a different background map.

Note: When you create a new map view, the background map that is selected in the **Background Maps** menu is used to create the map for that view.

See also:

[Save a Map Source](#) on the previous page

[Select Background Maps](#) below

Select Background Maps

Tableau comes with a set of online and offline maps that you can access to create map views.

In addition, Tableau supports connecting to a Web Map Server (WMS) or to a Mapbox map so you can create custom maps that are specific to your industry.

You can specify the map background, import new Tableau map sources, or export a Tableau map source to share with others. For more information, see [Save a Map Source](#) on the previous page and [Import a Map Source](#) above.

To select a new background map:

- Select **Map > Background Maps** and then select the map you want to use as a background map in the view.

By default, Tableau Desktop connects to an online map provider. You can also select to connect to an offline map that ships with the product, or you can add a WMS server or Mapbox map.

The three background maps that come with Tableau are described below.

None	Displays data between latitude and longitude axes.
Offline	Stores the images that make up the map in a cache with your Internet Explorer Temporary Internet Files for improved performance and offline access. For more information, see Use the Offline Background Map on the next page below.
Tableau	Connects to the Tableau background map. By default, all map views connect to this background map, unless you specify otherwise.

About the Tableau background map

The Tableau background map includes updated map and demographic data, as well as stylistic improvements. It has taken the place of the Tableau classic background map. Any map view created with an earlier version of Tableau Desktop, and with a connection to the Tableau classic background map, will automatically connect to the Tableau background map when opened in Tableau Desktop version 9.0 or later.

Note: If you create a map view with data layers in Tableau Desktop version 8.3 or earlier using the Tableau classic background map, and then open that view in Tableau Desktop 9.0 or later, the color of the data layers may appear differently due to data layering and color ramp (color palette) improvements made to the Tableau background map.

Set a Default Background Map

By default, all map views connect to the Tableau background map. If you do not want your map view to connect to the Tableau background map, you can specify a different background map as the default.

To specify a default background map:

1. Select **Map > Background Maps**, and then select the background map that you want to make the default.
2. Select **Map > Background Maps > Set as Default** to set the selected background map to the default.

The background map is automatically saved as a Tableau Map Source (.tms) and placed in the Mapsources folder of your My Tableau Repository. It is now the default background map for all new workbooks.

Use the Offline Background Map

You can create and inspect data in a map view offline using the offline background map that comes with Tableau.

To use the offline background map:

- Select **Map > Background Maps > Offline**

The offline background map uses map images stored in the cache. There are several actions, however, that require Tableau to retrieve a map image that may not be in the cache. If the new map image is not stored in the cache, you won't be able to load the map until you reconnect to the online map that comes with Tableau. For more information, see [Map Storing](#) below.

You may need to reconnect to the online map if you would like to do one or more of the following:

- **Turn layers on or off** - if you decide to turn on a layer that isn't stored in the cache, Tableau will need to connect to retrieve the necessary information.
- **Zoom in or out** - zooming in or out on a map requires different map images. If the images at the specified zoom level don't exist in the cache, Tableau will need to retrieve the updated maps.
- **Pan** - panning sometimes requires new map images. If you are working offline and don't have the necessary map images and legends stored in the cache, the new images and legends will not load.

To reconnect to the Tableau online map:

- Select **Map > Background Maps > Tableau**

Map Storing

When you create map views using the online map provider, Tableau stores the images that make up the map in a cache. That way, as you continue your analysis you don't have to wait for the maps to be retrieved. In addition, by storing the maps, you can do a certain amount of work when you are offline. For more information, see [Select Background Maps](#) on page 800.

The cache for the maps are stored with your temporary internet files and can be cleared at any time by deleting the temporary files from your browser.

Stored map images and legends remain valid for about thirty days. After that time, Tableau will not use the stored image; instead, it will require you to reconnect and fetch an updated map. This is to prevent the map images from becoming outdated.

Use Background Images in Your Views

In this topic:

- [Add background images to your workbook](#) below
 - [Build a view with a background image](#) on page 806
 - [Edit a background image](#) on page 807
 - [Enable or disable background images](#) on page 807
 - [Add show and hide conditions to background images](#) on page 808
 - [Remove a background image](#) on page 810
-

Background images are images that you display underneath your data in order to add more context to the marks in the view. A common use of background images is adding custom map images that correspond to a coordinate system in your data.

For example, you might have data that corresponds to several floors in a building. You can use background images to overlay that data on the actual floor plan of the building to give more context. Other examples of using background images include showing a model of the sea floor, images of web pages for analyzing web logs, and even levels from video games to visualize player statistics.

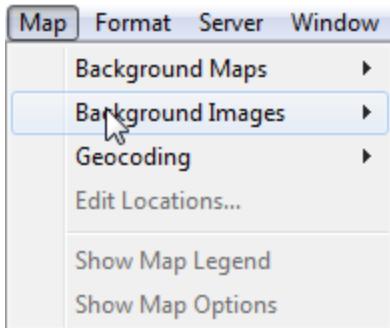
While Tableau allows you to load dynamic maps from the online and offline provider, background images allow you to use your own custom images whether they are special maps or any other image that corresponds to your data.

Add background images to your workbook

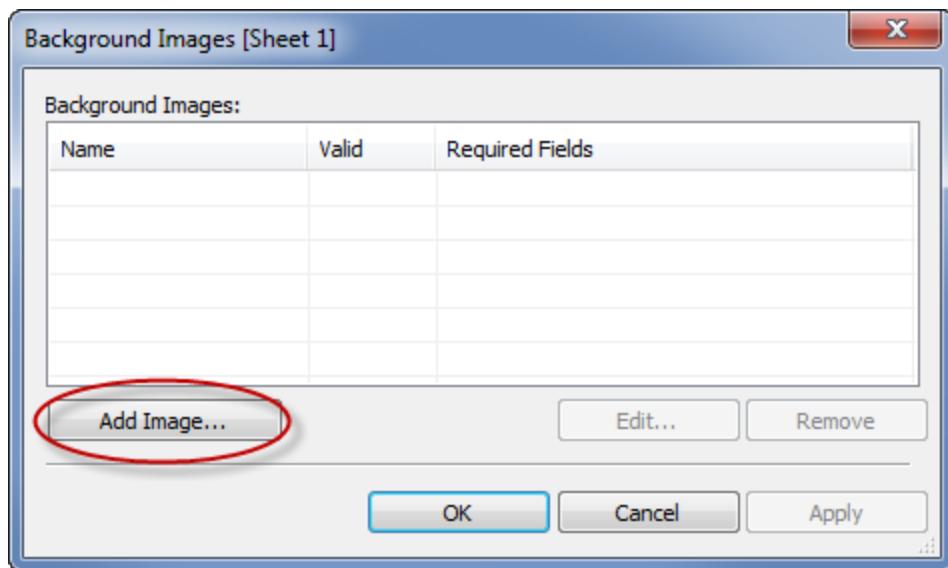
When you add a background image to the view, you need to specify a coordinate system by mapping both the X and Y axes to the values of fields in your database. If you are adding a map, the X and Y axes should be longitude and latitude expressed as a decimal. However, you can map the axes to any relevant fields based on your own coordinate system.

To add a background image:

1. Select **Map > Background Images** and then select a data source.

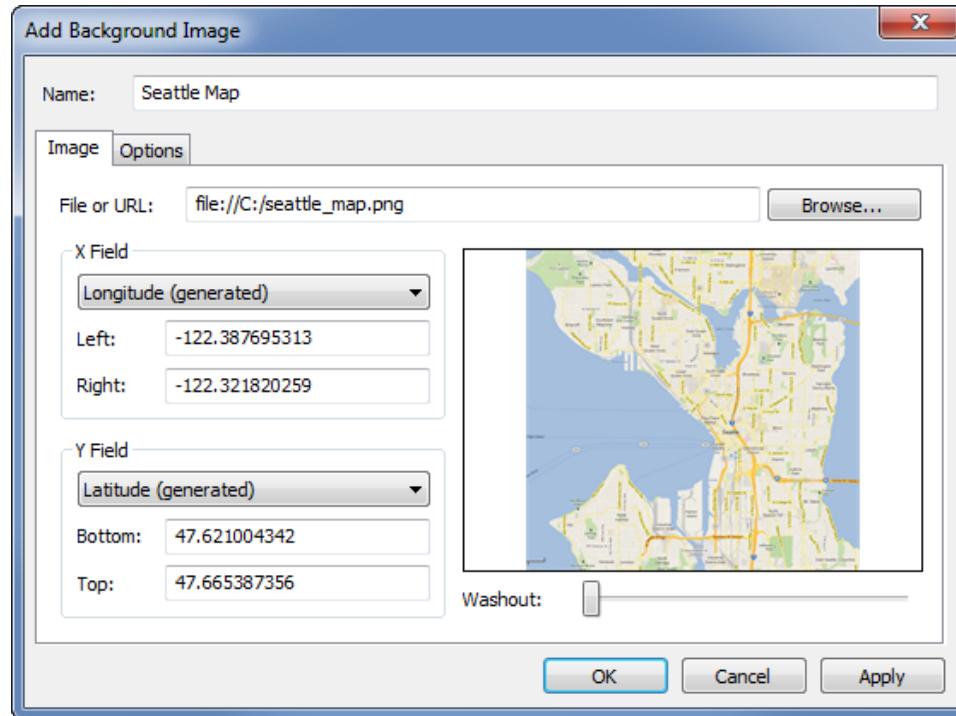


2. In the Background Images dialog box, click **Add Image**.



3. In the Add Background Image dialog box do the following:
 - Type a name for the image into the **Name** text box.
 - Click **Browse** to navigate to and select the image you want to add to the background. You can also type a URL to link to an image hosted online.
 - Select the field to map to the x-axis of the image and specify the left and right values. When adding a map, the longitude values should be mapped to the x-axis using decimal values (instead of degrees/minutes/seconds or N/S/E/W).
 - Select the field to map to the y-axis of the image and specify the top and bottom values. When adding a map, the latitude values should be mapped to the y-axis using decimal values (instead of degrees/minutes/seconds or N/S/E/W).
 - You can adjust the intensity of the image using the Washout slider. The farther the

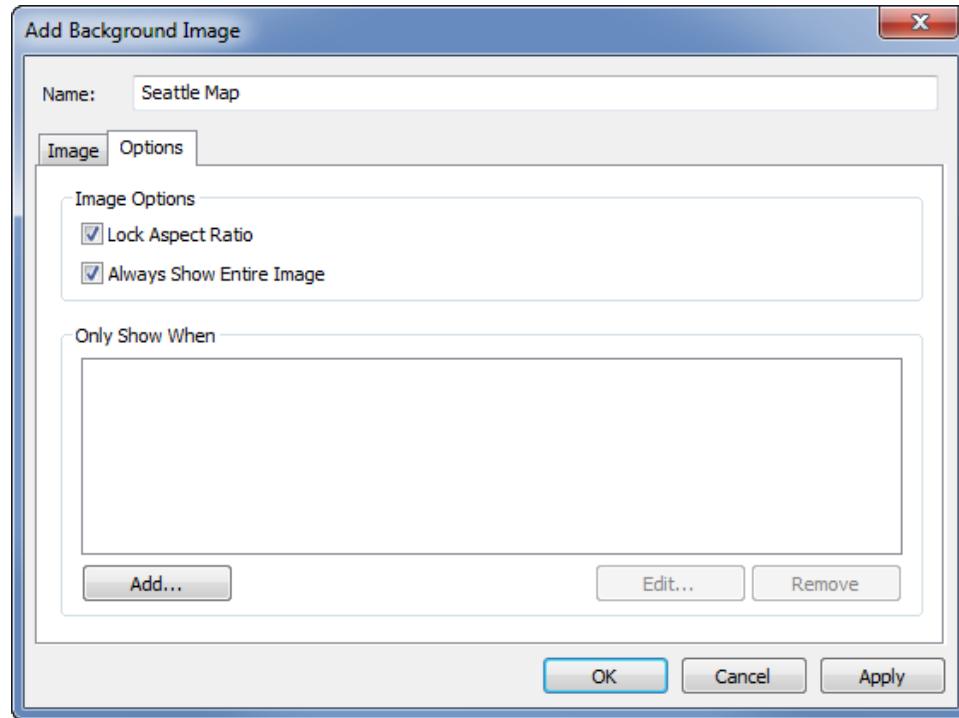
slider moves to the right, the more faded the image will appear behind your data.



4. You can specify the following options using the **Options** tab:

- **Lock Aspect Ratio** - select this option to maintain the original dimensions of the image for any manipulations of the axes. Deselecting this option allows the image's shape to be distorted.
- **Always Show Entire Image** - select this option to avoid cropping the image when the data encompasses only a portion of the image. If you lock both the axis in a view, this option may be negated.
- Add conditions for when to show the image. Refer to [Filter Data from Your](#)

[Views](#) on page 672 to learn more about defining conditions.



5. Click **OK**.

When you add the x and y fields to the Rows and Columns shelf in the view, the background image displays behind the data. If the background image does not display, make sure that you are using the disaggregated measures for the x and y fields. To disaggregate all measures, select **Analysis > Aggregate Measures**. To change each measure individually, right-click the field on the shelf and select **Dimension**. Finally, if you've used the generated Latitude and Longitude fields for the x and y fields, you'll need to disable the built in maps before your background image will display. Select **Map > Background Maps > None** to disable the built-in maps.

In order to make the marks in a view more visible when placed on top of a background image, each mark is surrounded by a solid contrasting color called a halo. You can turn mark halos off by selecting **Format > Show Mark Halos**.

Build a view with a background image

After you add a background image, you need to build the view in a way that matches the x and y mappings you specified for the image. That is, the fields you specified as x and y must be on the proper shelves. Follow the steps below to set up the view correctly:

1. Place the field mapped to the x-axis on the **Columns** shelf.

If you are working with maps, the longitude field should be on the columns shelf. It may seem backward at first, however, the fields on the columns shelf determine the values distributed across the x-axis.

2. Place the field mapped to the y-axis on the **Rows** shelf.

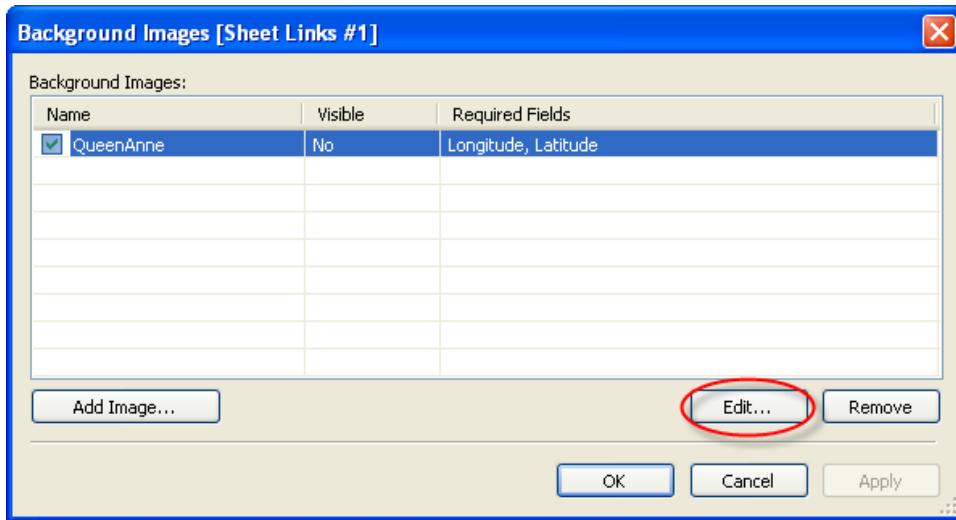
If you are working with maps, the latitude field should be on the rows shelf. It may seem backward at first, however, the fields on the rows shelf determine the values distributed across the y-axis.

Edit a background image

After adding a background image, you can always go back and edit the x and y field mappings as well as any of the options on the Options tab.

To edit an image:

1. Select **Map > Background Images**.
2. In the Background Images dialog box, select the image you want to edit and click **Edit** (you can also just double-click the image name).



3. In the Edit Background Image dialog box, make the changes to the image and click **OK**.

Enable or disable background images

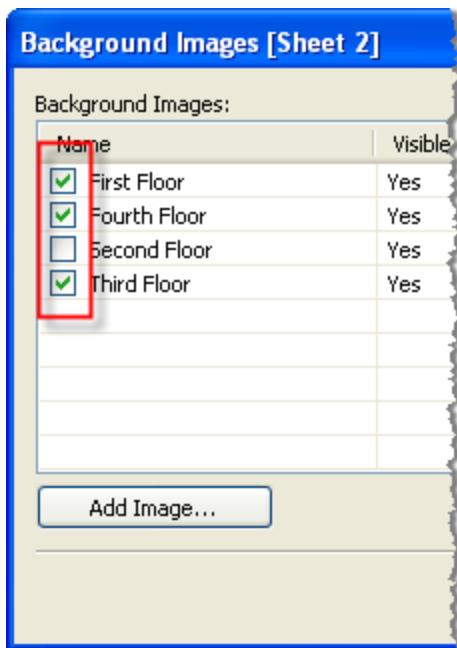
Although you can add multiple images to a workbook, you may want to only use a subset of the images for a particular set of worksheets. For example, you may want to show a map of the entire United States of America on one view, and maps of individual states in other views.

Use the check boxes in the Background Images dialog box to enable and disable the images for the current worksheet. You can show several images by enabling multiple images on a single

worksheet. For example, you may have several images that you want to tile in the background to make a larger background image.

To enable or disable a background image:

1. Select **Map > Background Images**.
2. In the Background Images dialog box, select the check boxes next to the images you want enabled.



3. Click **OK**.

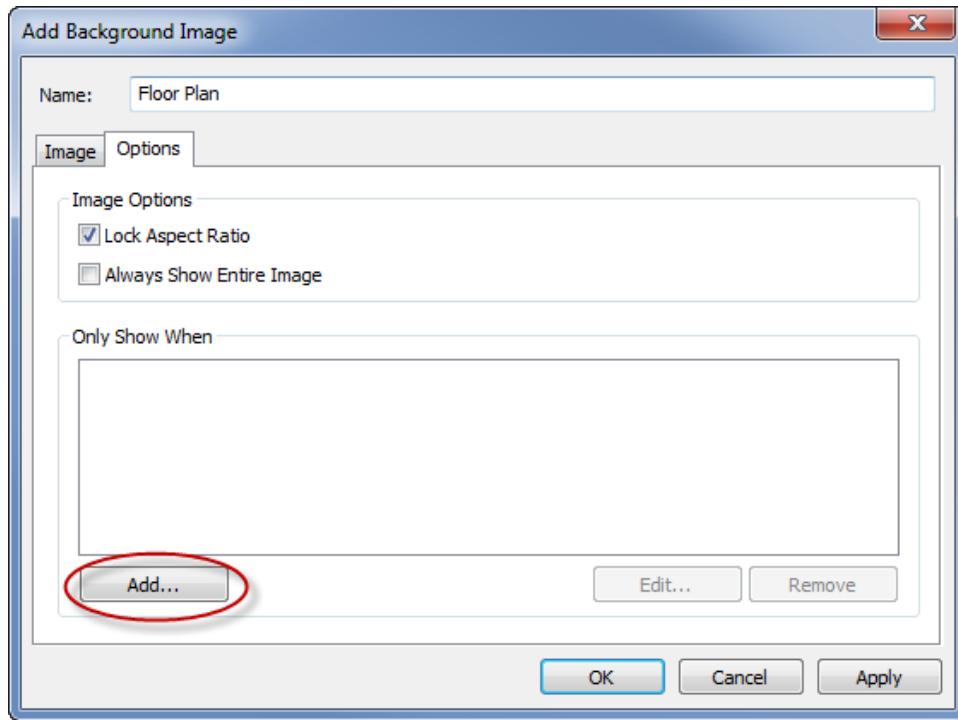
Add show and hide conditions to background images

When you add a background image and enable it, the image will be shown automatically on any worksheet that has the required fields used in the view. To avoid showing an image on all the worksheets, you can specify show/hide conditions. Show/Hide conditions are conditional statements that you define to specify when to show the image. For example, you may have a floor plan image for a multi-story building. While each image is associated with the same coordinates (the corners of the building), you do not want to show the first floor map when looking at the third floor information. In this case, you can specify a condition to only show the first floor image when the Floor field is equal to one.

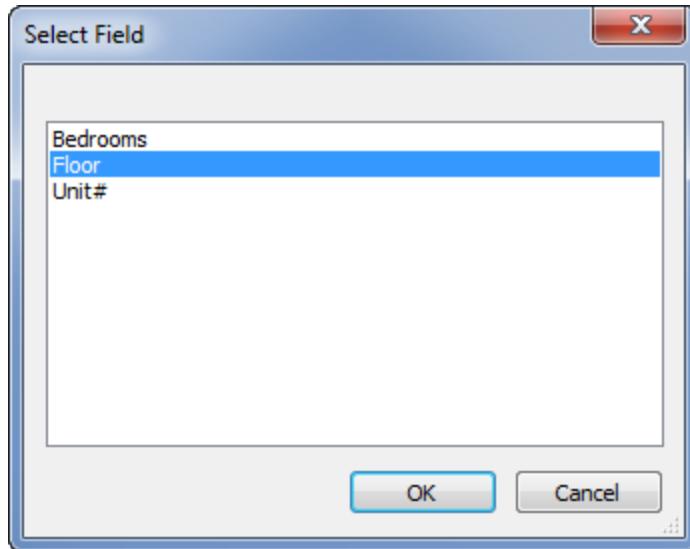
To specify show/hide conditions:

1. Select **Map > Background Images** and then select a data source.
2. In the Background Images dialog box, select the image you want to add a condition to and click **Edit**.

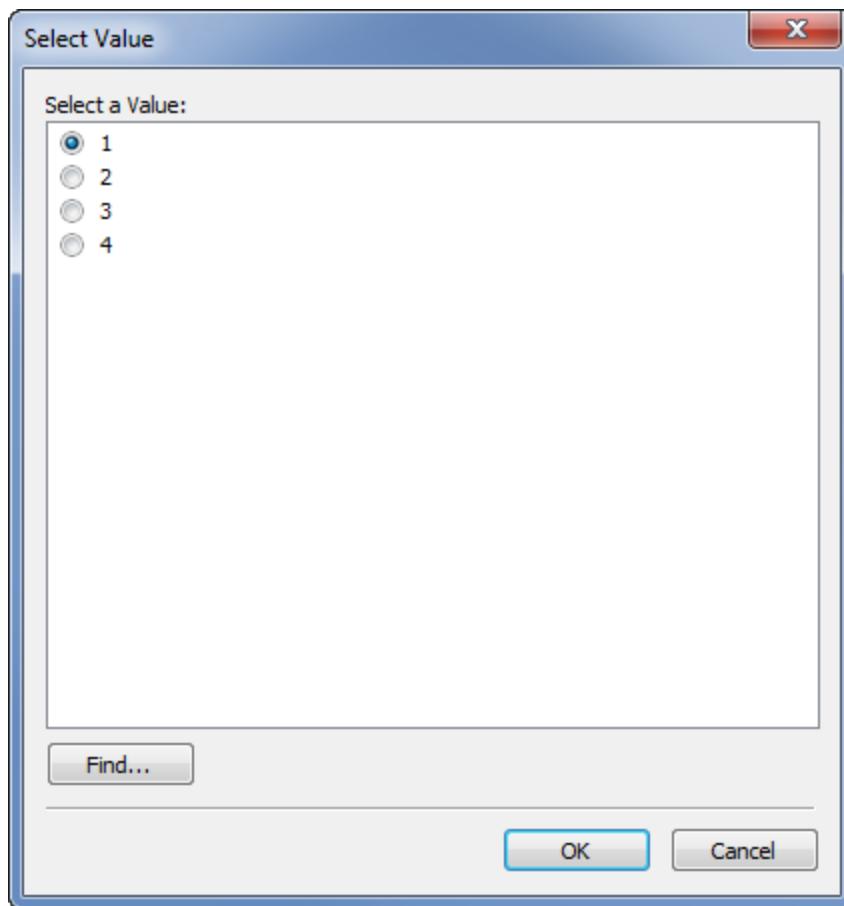
3. In the subsequent dialog box, select the **Options** tab.
4. Click the **Add** button at the bottom of the dialog box.



5. Select a field to base the condition on. In the example described above, the field is Floor.



6. Specify when you want to show the image by selecting a value of the field. For this example, one is selected.



7. Click **OK**.

A condition statement is added to the image. In the building floor plan example, the condition statement is Only show the image when Floor is equal to One.

8. Click **OK** twice to close the Background Image dialog boxes and apply the changes.

When you add multiple conditions, the background image will only show when all conditions are met. For example, if a background image has two conditions on Property Name and Floor, it will only show when Property is Greenwood Estates and Floor is 3.

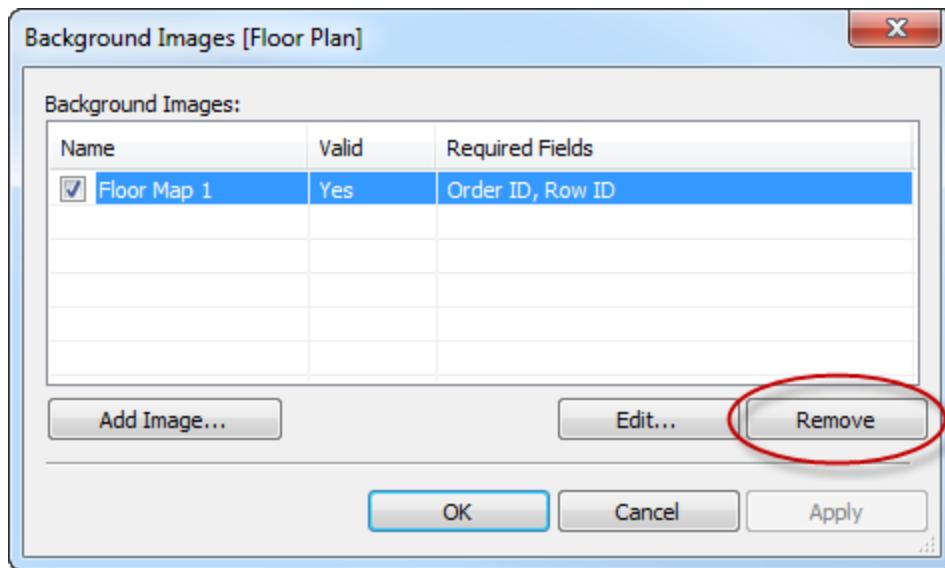
Remove a background image

When you no longer want to use a background image you can either disable it or remove it, making it unavailable to all worksheets.

To remove an image:

1. Select **Map > Background Images**.
2. In the Background Images dialog box, select the image you want to remove and click

Remove.



3. Click **OK**.

Find Background Image Coordinates

In many scenarios, you might want to map your data onto a background image instead of on a Tableau map. This article explains how to plot data points on a background image.

There are several steps to this process:

1. Create a table in your data source for X and Y coordinates.
2. Connect to your data source in Tableau Desktop and join the coordinate table with the rest of your data.
3. Import your background image and build the view.
4. Annotate points on the background image.
5. Add coordinates to the coordinates table in your data source.
6. Refresh your data source in Tableau.

: Create the coordinate table

Create a table in your data source that contains columns to identify a unique mark. In this example, there are three columns:

- The first column holds the unique identifier for each mark.
- The other two columns are for the X and Y coordinates.

After you create your columns, you join the new coordinate table to the original data source. The join is based on the unique identifier used in both the original data source and the coordinate table (the identifier that represents a single mark).

1. Select a scale that is appropriate for your background image. If the image is wider than it is tall, use X: 0-100 and Y: 0-50.
2. For this example, open the Sample Superstore Excel file from your Tableau repository in Excel and select a new sheet tab.
3. On the new tab, put a single row in a table that contains one of the unique identifiers and the X and Y end points you selected for your two scales.

In this case, **Product Sub-category** is a column in Sample Superstore, and one of its members is **Tables**. The **X** endpoint is **100**, and the **Y** endpoint is **50** because that is the scale for the background image.

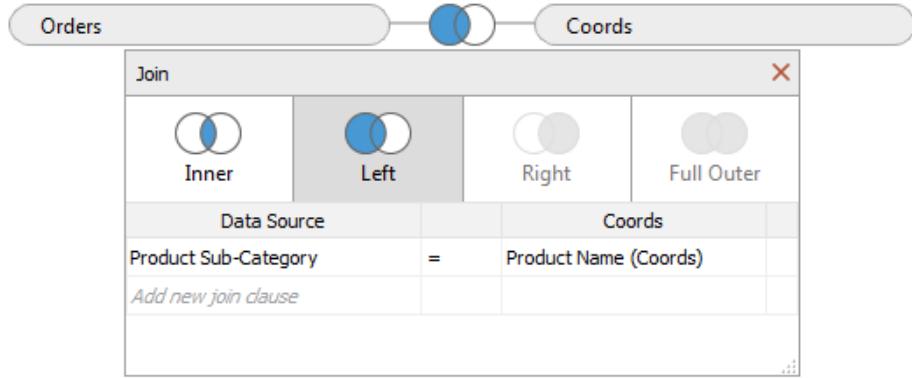
	A	B	C	D
1	Product Sub-category	X	Y	
2	Tables	100	50	
3				
4				
5				

4. Change the tab name to **Coords** and save the file.

: Connect to the coordinate table

1. In a new Tableau workbook, select **Connect to Data**.
2. In the **Connect to Data** dialog box, select **Microsoft Excel** and then click **Next**.
3. On the data source page, do the following:
 - Drag the Orders sheet to the Join area.
 - Drag the Coords sheet to the Join area.
 - Click the join icon to modify the existing join.
 - In the Join dialog box that opens, select **Left**, and then select **Product Sub-Category** for the **Data Source** column, and **Product Name (Coords)** for the

Coords column.



4. Close the Join dialog box, and then select the sheet tab to go to a worksheet.

: Import the background image

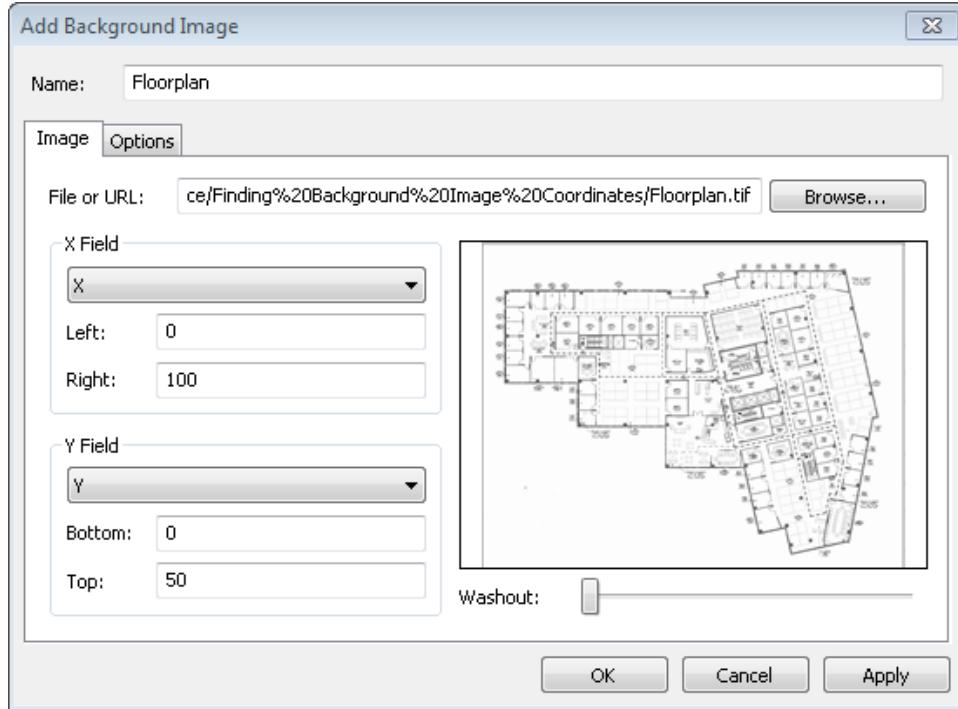
1. In the new worksheet, select **Maps > Background Images, Sample - Superstore**. This opens the Background Images dialog box.
2. In the **Background Images** dialog box, click **Add Image**.
3. In the **Add Background Image** dialog box, browse to and select the file.
4. For **X Field**, select **X** in the list.

Note: If you do not see the X field in the list, it means that you need to change the X field to be a continuous number.

To change the field to be a number, right-click the **X** field in the Data pane, and select **Change Data Type > Number (Whole)**. To convert the field to be continuous, right-click the **X** field in the Data pane, and select **Convert to Continuous**.

Repeat these steps for the **Y** field.

5. In the **Right** text box, type **100** (100 is the number you used in the coordinate table).
6. For **Y Field**, select **Y** in the list.
7. In the **Top** text box, type **50** (50 is the number you used in the coordinate table). Be sure to type in the correct text box.



8. Click **OK**, and then click **OK** in the **Background Images** dialog box. (Disregard the **Valid** setting of **No** in the **Background Images** dialog box.)

: Find the coordinates

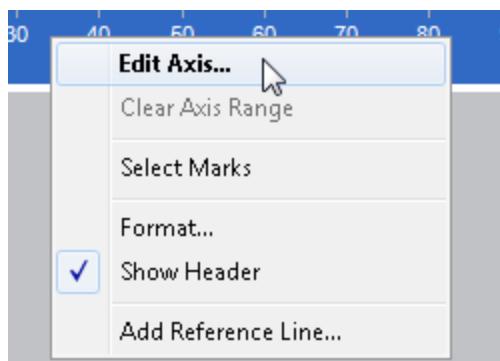
1. From the Data pane, drag **X** to the **Columns** shelf and **Y** to the **Rows** shelf.
2. On the **Columns** shelf, right-click **X** and select **Dimension**.
3. On the **Rows** shelf, right-click **Y** and select **Dimension**.

Note: This step is only necessary if the X and Y fields were brought in to Tableau as Measures.

The background image appears in the view.



4. Right-click the X axis and select **Edit Axis**.



5. In the **Edit Axis** dialog box, select **Fixed**, and then in the **Start** text box, type **0**, and in the **End** text box, type **100**.
6. Click **OK**.
7. Right-click the Y axis and select **Edit Axis**.
8. In the **Edit Axis** dialog box, select **Fixed**, and then in the **Start** text box, type **0**, and in

the **End** text box, type **50**.

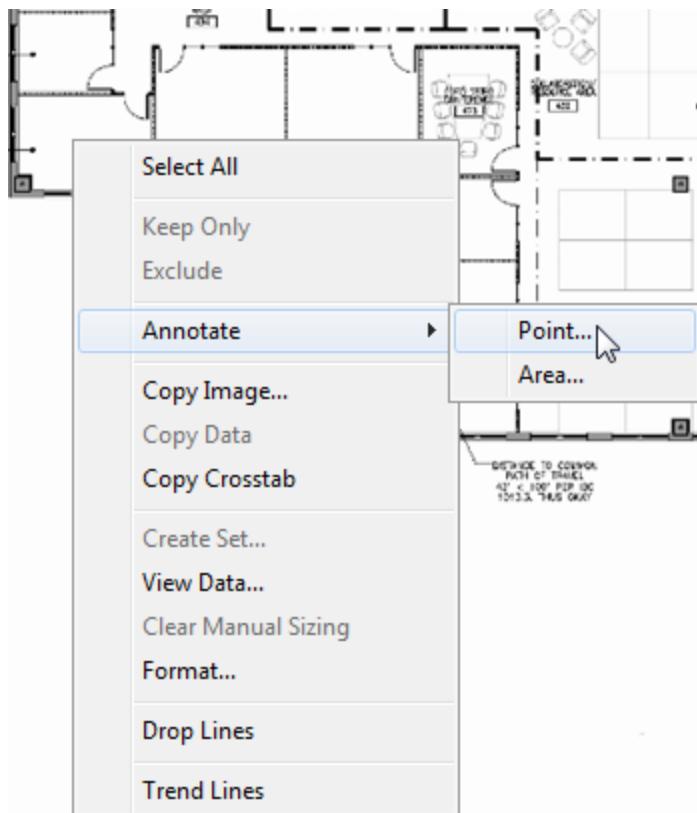
9. Click **OK**.

Now the axes start at zero.



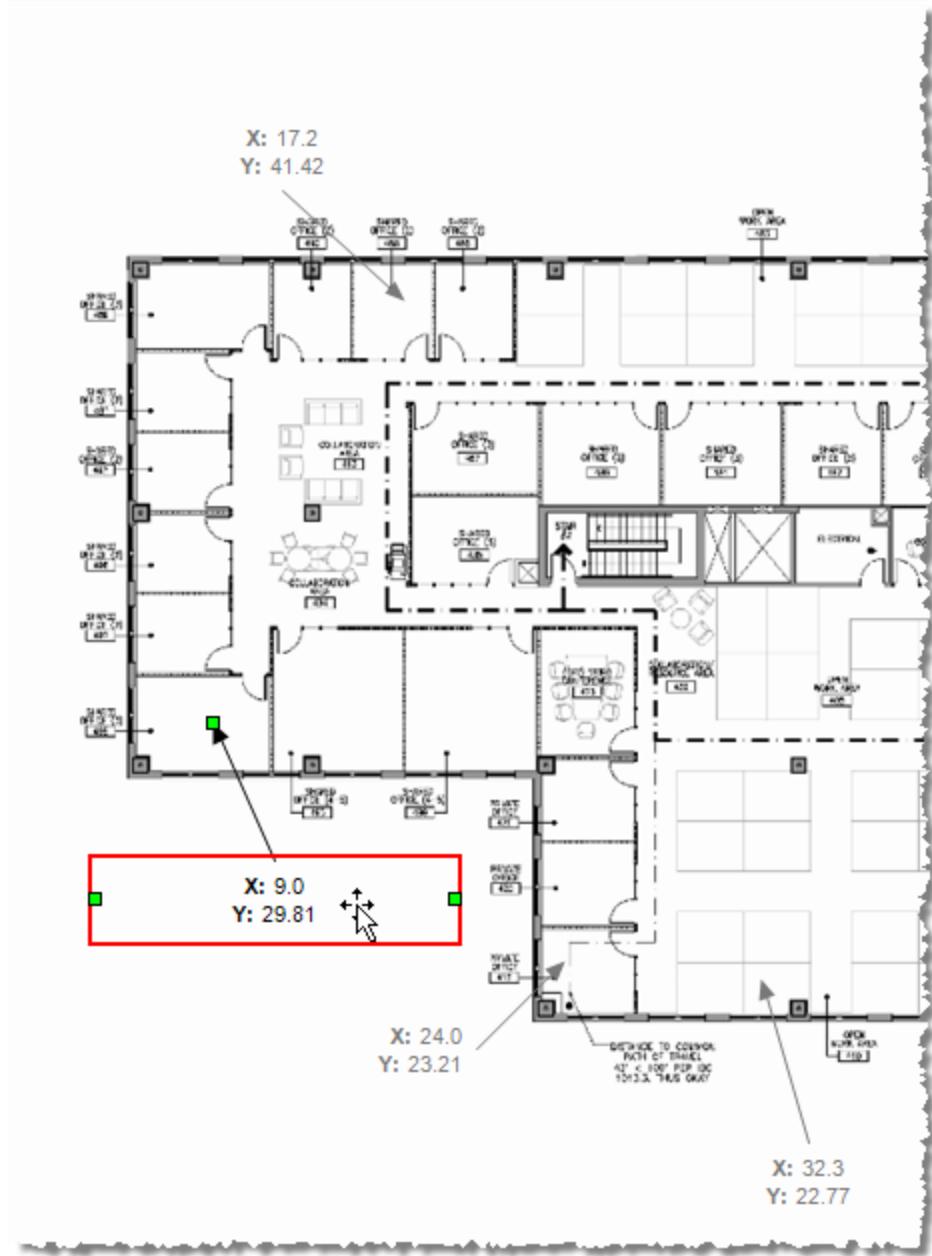
10. Right-click anywhere on the image where you want to find the coordinates, and select **Annotate > Point**.

For more information about annotating marks or points in the view, see the [Annotations on page 536](#) topic in the Tableau Desktop Help.



11. In the **Edit Annotation** dialog box, click **OK**.
12. Repeat steps 10 and 11 for each point you want to annotate.

The coordinates appear on the background image as a callout. You can move a callout to a location where it shows up better; just click and drag it. You can also resize the callout by dragging one of the size handles. You can move a coordinate point to a new location by clicking and dragging the arrowhead; the coordinates in the callout change to reflect the new location.



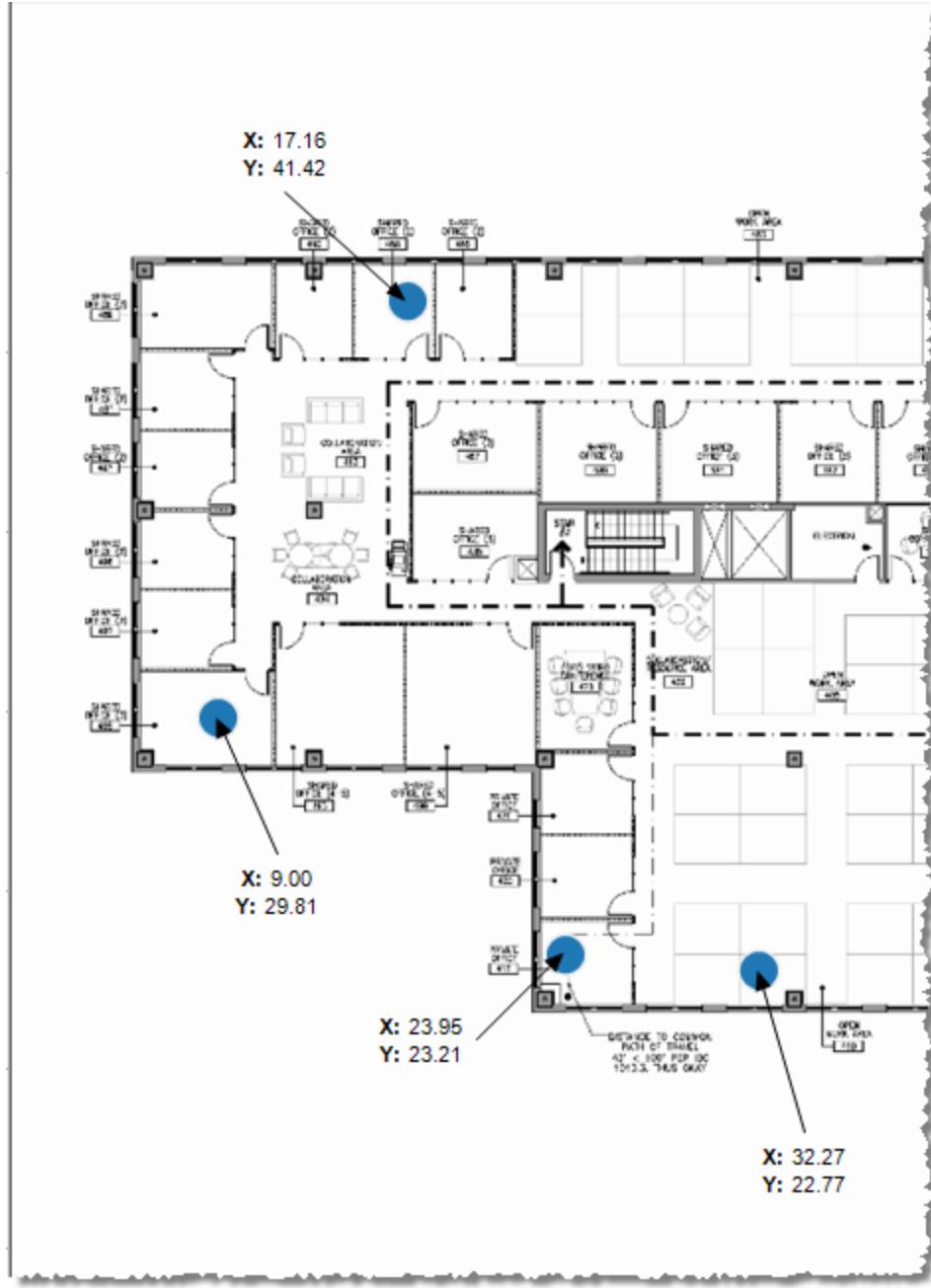
13. Add these coordinates to the Coords sheet in the Sample - Superstore Excel file and save the changes.

	A	B	C	D
1	Product Sub-category	X	Y	
2	Tables	100	50	
3	Labels	17.16	41.42	
4	Paper	9.00	29.81	
5	Bookcases	23.95	23.21	
6	Office machines	32.27	22.77	
7				
8				

14. In Tableau, select **Data > Sample - Superstore > Refresh**.

The refreshed view displays marks at the coordinate locations.

If you don't want to continue to display the callouts, you can select and remove them on the background image.



Other Ideas

Try placing another Measure on the Size shelf and put either a Dimension or Measure on Color on the Marks card. Add your own custom shapes. Develop powerful and unique analyses by creating Actions between this view and your other analyses.

If you want to hide the X and Y scales, right-click the respective field on the Columns or Rows shelf and clear **Show Header**.

See also:

- [Use Background Images in Your Views](#) on page 802
- [Build and Use Maps](#) on page 702

Advanced Analysis

Now that you understand the basics of building views in Tableau, become an advanced user by learning how to create custom calculations, use the built in statistics tools, leverage dynamic parameters, and more.

Calculated Fields

If your underlying data doesn't include all of the fields you need to answer your questions, you can create new fields in Tableau and then save them as part of the data source. For example, you could create a new calculated field called **Profit** that calculates the difference between the **Sales** and the **Cost** fields, or you could create a formula that sorts values for the **Sales Budget** field depending on whether they are under budget or over budget.

You created calculated fields in Tableau by defining a formula that is based on existing fields and other calculated fields, using standard functions and operators. You can create calculated fields using the calculation editor, or by double-clicking a field on a shelf and building an ad-hoc calculation.

You can create calculated field with Tableau Desktop or in Tableau web editing environments in products like Tableau Server and Tableau Online.

Note: You can use all types of calculations if you are using a relational data source, however multidimensional data sources do not support aggregations and binned data.

Create or Edit a Calculated Field

Use the calculation editor to create or modify a calculated field.

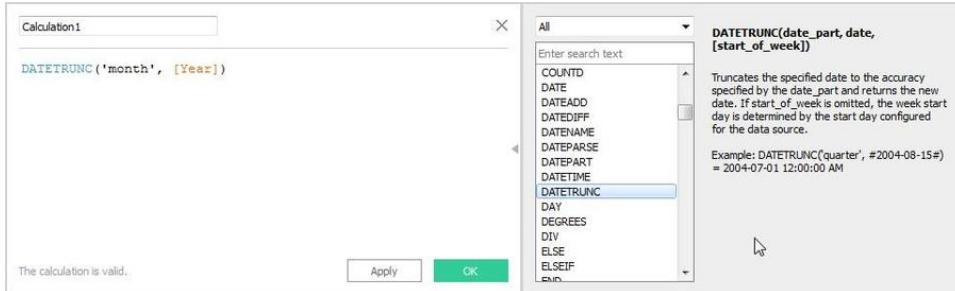
To display all collapsed content, click the  (**Expand all**) button at the top of the page.

Create a Calculated Field

To open the calculation editor, click the drop down to the right of Dimensions on the **Data** pane and choose **Create Calculated Field**.

You can also choose **Analysis > Create Calculated Field**, or right-click (control-click on Mac) in the **Data** pane and choose **Create Calculated Field**.

Click the triangle at the right edge of the editor to open a pane where you can select functions by category or see help for the selected function.



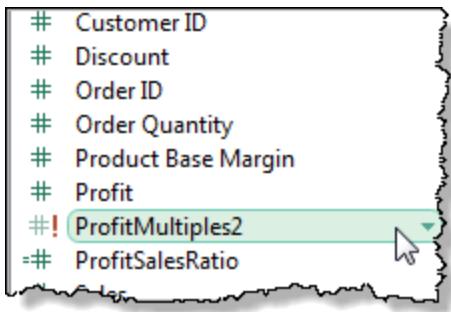
See [Create Formulas in Tableau on page 827](#) for information on how to bring the different element that make up a formula into the calculation editor.

When the workbook is connected to multiple data sources, the current data source is listed next to the calculation name box in the upper left--in the image above, the current data source is **Sample - Superstore**. As you work in the editor, any fields that you that are not from the current data source are shown with the data source prepended to the field name. For example: [DS1].[Sales].

Click **Apply** in the calculation editor to save the field as currently defined and add it to the **Data** pane without closing the editor; click **OK** to save the calculation and close the editor.

Calculations that return a string or date are saved as dimensions, and calculations that return a number are saved as measures.

Tableau will allow you to save an invalid calculation; however, a red exclamation point appears next to it in the **Data** pane:



Until you correct an invalid calculated field, you will not be able to drag it into the view.

Edit a Calculated Field

To edit a calculated field, right-click (control-click on Mac) it in the **Data** pane and select **Edit**.

You can only edit calculated fields--that is, named fields created in Tableau (as opposed to named fields that were part of the original data source).

Drag and Drop Options with the Calculation Editor

You can work on a calculation in the editor and do other things in Tableau at the same time.

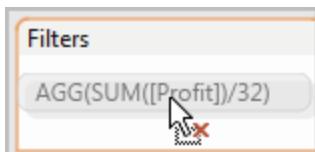
Here is an example of the kind of workflow that is possible with the calculation editor:

1. Start by creating or editing a view.
2. Open the calculation editor and begin working on a calculated field.
3. Drag all or part of your formula to a shelf, dropping it on an existing field, to see how it changes the view.
4. Double-click the field you just dropped on the shelf to open it as an ad-hoc calculation (see [Ad-Hoc Calculations on page 831](#) for details). Then tweak the calculation.
5. Drag the ad-hoc calculation back to the calculation editor and drop it on the original formula in the calculation editor, thereby replacing the original formula.

You can also drag all or part of a formula to the Data pane to create a new field.

As you drag formula content to a shelf, card, or pane, Tableau provides visual feedback:

- A red X indicates that you cannot drop the expression at the current location:



- If you drop an expression that you have dragged from the calculation editor onto a field in the view, the expression replaces that field. If the resulting expression is not valid, the resulting field will be red. You can then edit the expression to make it a valid field.
- If you drag in a field from a secondary data source, it comes with fully qualified name and default aggregation. For example, with a secondary data source named DS1, here is how fields might appear when dragged into the calculation editor:

```
[DS1].[Sales]  
SUM( [DS1].[Sales] )  
ATTR( [DS1].[Customer Name] )
```

For purposes of calculations, the primary data source is whichever data source is currently displayed in the **Data** pane.

It is not possible to drag numeric bins, generated latitude and longitude fields, **Measure Names**, or **Measure Values** into the calculation editor.

Copy and Paste Calculated Fields

Calculated fields are available to all sheets that use the same data source in a single workbook. To copy and paste calculated fields between workbooks, right-click the field in the **Data** window of the source workbook and choose **Copy**. Then right-click in the **Data** window in the

destination workbook and choose **Paste**. You can copy and paste all custom defined fields, including calculated fields, ad hoc groups, user filters, and sets.

Working in the Calculation Editor

When you use the calculation editor, the result must always be a calculated field—that is, a user-created named field (as opposed to a named field that was part of the original data source). If you are editing an existing calculated field, then that field is updated when you click **OK** or **Apply**. If you are creating a new calculated field, or wish to save your modifications as a new field, type a new field name in the box at the top of the calculation editor before clicking **OK** or **Apply**. Ad-hoc calculations, by contrast, do not need to be named. See [Ad-Hoc Calculations](#) on page 831.

Customizing the Calculation Editor

You can customize the calculation editor in the following ways:

- **Expand or collapse the function list and help area**

You can expand (open) or collapse (close) the function list and the help area on the right side of the calculation editor by clicking the angle control that is between the working area and the function list:

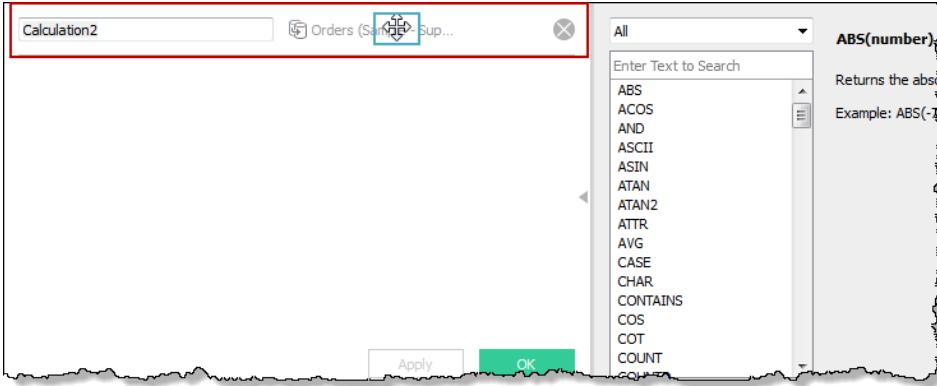


Click the same control (which now faces left) to reopen the function list and help area.

- **Resize or move the calculation editor**

You can resize the calculation editor by dragging from any of its corners.

You can move the editor by clicking in the upper-left section (anywhere in the box shown in red, below) until you see the four-way arrow cursor (outlined in blue), and then dragging:



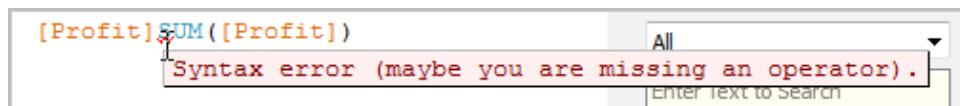
The editor is initially displayed within the view, but you can move it outside the view--for example, to a second monitor. Moving the calculation editor does not move Tableau Desktop, and moving Tableau Desktop does not move the editor.

- **Resize text**

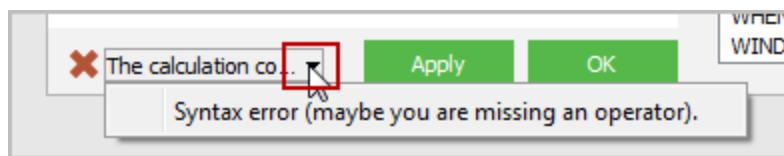
To increase text size in the calculation editor, hold down the Ctrl key and scroll the mouse button upward. To decrease text size, hold down Ctrl and scroll the mouse button downward. All text in the editor is resized—you cannot resize only selected text. Text size persists until you close the editor. The next time you open the editor, text is at the default size.

Visual Feedback in the Calculation Editor

The calculation editor has built-in coloring and validation to help you avoid syntax errors. As you create a formula, syntax errors are underlined with a red line. Hover over the error to see a suggestion for resolving it:



Feedback on formula validity is also displayed at the bottom of the calculation editor. A green check mark indicates that the formula is valid; a red X indicates that it is not. You can click the drop down box to see details:

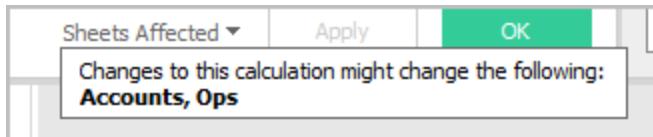


When you are writing formulas, any part that displays in bold indicates that it will be computed locally within Tableau on the aggregated results. Any normal weight text will be computed at the database level.

Note: If you are connected to a Microsoft Analysis Services data source, the calculation editor contains a "Run before SSAS" check box. Choose this option to execute the Tableau calculation before any Microsoft Analysis Services calculations. For information on connecting to Microsoft Analysis Services data sources, see [Microsoft Analysis Services](#) on page 1426

Sheets Affected

As you edit a calculated field, you can click **Sheets Affected** in the editor's status bar to see which other sheets are using the field, which will be updated when you commit your changes:



The **Sheets Affected** drop-down is only shown if the field you are editing is also being used in other sheets.

Create Formulas in Tableau

You create formulas from the following elements:

- **Functions**

Tableau functions are listed on the right in the calculation editor. Use the drop-down list above the function list to filter functions by category. You can also type in the **Enter Text to Search** area to find functions that match the characters that you type. By default all functions are displayed. Certain functions are only available with specific data sources. See [Additional Functions](#) on page 1603 for details. For complete help on Tableau functions and other keywords, see [Functions, Operators, & Data Types](#) on page 1549. The function help provides examples that demonstrate the correct syntax for functions. For more practical, real-world examples, the [Tableau Community](#) and internet blogs may be useful.

Click a function in the function list to view a brief description and an example on the right. Double-click a function in the list to include it at the cursor location in the current formula.

Functions are colored light blue in formulas.

Note: Autocomplete is available in the calculation editor. For details, see [Auto-Completion for Formulas](#) on page 829.

- **Fields**

You can incorporate a field in a formula by dragging it from the **Data** pane and dropping it in the calculation editor.

You can also drag a field from a shelf in the view to the calculation editor. The result will not necessarily be the same as when you drag a field from the **Data** pane. For example, if you drag the **Sales** field from the **Data** pane to the calculation editor, this is what you will see:

```
[Sales]
```

But if you drag **Sales** to the **Rows** shelf and then drag it from there to the calculation editor, the field will have changed as a result of being aggregated in the view:

```
SUM([Sales])
```

You may also choose to modify a field on a shelf before dragging it to the calculation editor. For example, if you have dragged **Sales** to the **Rows** shelf and then added a quick table calculation to the field, you can drag the modified field to the calculation editor to see how the field is expressed in the Tableau calculation language, and to make further modifications. This is what you might see:



You can also drag part or all of an ad-hoc calculation from a shelf to the calculation editor. In addition to the various drag-and-drop options, you can also just type a field name in the calculation editor. Field names that include special characters or spaces must be delimited with square brackets--for example, `SUM([Store Profit])`. For field names that actually include bracket characters, just type two additional brackets. For example, to specify a field named "Store Profit" you would type `[Store Profit]]`; two of the brackets say "include the bracket character in the field name," and the third bracket delimits the field name. For information on ad-hoc calculations, see [Ad-Hoc Calculations on page 831](#).

Fields are colored orange in formulas.

- **Operators**

Type any necessary operators into your formula. All standard operators such as addition (+), subtraction (-), multiplication (*), and division (/) are supported.

Operators are colored black in the formula.

- **Parameters**

Parameters are placeholder variables that can be inserted into calculations to replace constant values. When you use a parameter in a calculation, you can then expose a parameter control in a view or dashboard to allow users to dynamically change the value.

For details, see [Parameters on page 989](#).

Parameters are colored purple in formulas.

- **Comments**

To add a comment to a calculation, type two forward slash characters into the formula pane.

For example:

```
Sales * Profit //John's calculation
```

In this example //John's calculation is a comment.

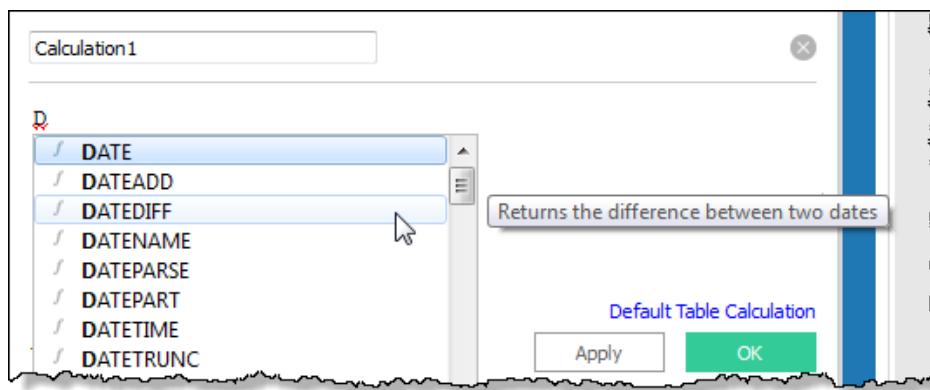
A comment starts at the two forward slashes (//) and goes to the end of the line. A multiline comment can be written by starting each line with two forward slashes (//).

Comments are colored green in the formula.

Auto-Completion for Formulas

As you type a formula, either in the calculation editor or in an ad-hoc calculation, Tableau displays a list of options for completing the formula.

As you scroll the list, using mouse or keyboard, Tableau shows a short description when the current item is a function:



When the current item is a field, set, or bin, and that keyword has a comment attached, that comment will appear as the description.

For information on adding field comments, see [Comments](#) on page 317.

Click a keyword in the list or press Enter to select it. If the keyword is a function, Tableau displays syntax information when you select it:



As you type, the list of suggested keywords is organized into the following categories, in the following order:

- Functions, dimensions, measures, parameters, sets, and bins that *begin* with the string you have typed so far, from the current data source, sorted alphabetically.
- Functions, dimensions, measures, parameters, sets, and bins that *contain* the string you have typed so far, from the current data source, sorted alphabetically.
- If the workbook connects to multiple data sources, dimensions, measures, sets, and bins from the secondary data sources that *begin* with or *contain* the string you have typed so far are displayed, sorted by data source and sorted alphabetically within each data source.

Typing certain characters will either turn off auto-completion or cause it to behave differently:

- Auto-completion will not offer suggestions as you type a quoted string.
- Auto-completion will not offer suggestions if you begin by typing a number.
- If you begin by typing an open square bracket, [, auto-completion shows fields, parameters, sets, and bins, but not functions.

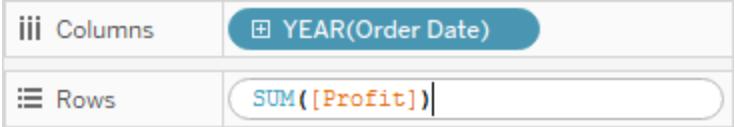
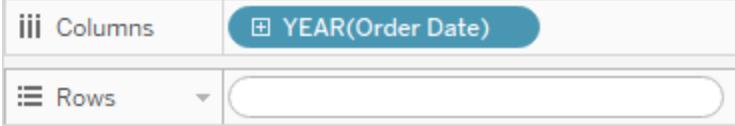
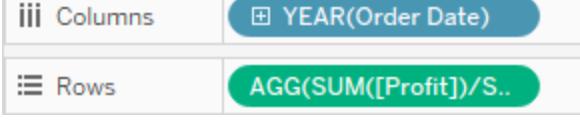
[Auto-Completion with Multiple Data Sources](#)

If the workbook is using multiple data sources, auto-completion behaves as follows:

- If the selected field is from a secondary source, auto-completion adds the field with its aggregation and fully qualified name. For example:
`ATTR ([secondaryDataSource] . [sate])`
- Matches from secondary sources are only shown if there is an explicit blend relationship set with the currently active sheet.
- Fields that are being used to blend the two data sources are only shown once in the search results (the field shown is from the primary data source).

Ad-Hoc Calculations

Ad-hoc calculations are calculations that you can create and update as you work with a field on a shelf in the view. Ad-hoc calculations are also known as type-in or in-line calculations.

Double-click on an existing field to start editing.	
You can also double-click on an empty shelf or on an empty part of a shelf to create a new calculation.	
Type to update the expression, or drag new fields into the expression from the Data pane or elsewhere in the view.	
Press Enter, Tab, or click outside the expression to commit the expression, close the calculation, and update the view. Press Ctrl+Enter (or Command+Enter on a Mac) to commit the expression and update the view without closing the calculation.	

Ad-hoc calculations are supported on the **Rows**, **Columns**, **Marks**, and **Measure Values** shelves; they are not supported on the **Filters** or **Pages** shelves.

Errors in ad-hoc calculations are underlined in red. Hover over the error to see a suggestion for resolving it.

Note: Ad-hoc calculations are not named, but are saved when you close the workbook. If you want to save an ad-hoc calculation for use in other workbook sheets, copy it to the

Data pane. You will be prompted to name the calculation. {On server it just deposits it w/o rename prompt.} Once you name an ad-hoc calculation, it is the same as a calculation you created with the calculation editor, and is available on other sheets in the workbook. See [Create or Edit a Calculated Field](#) on page 822

Multi-Line Ad-Hoc Calculations

As you're typing an ad-hoc calculation, you can press Shift+Enter to start a new line. However, only the current line is ever visible in an ad-hoc calculation, so this can be confusing for anyone viewing or editing the calculation who has no way of knowing that it contains multiple lines.

The first line of a multi-line ad-hoc calculation can be a comment that serves as a title for the calculation. This is the only line that is visible on the shelf after the calculation is committed:

```
//City and State  
[City] + ', ' + [State]
```

Ad-Hoc Calculations and Aggregation

If Tableau determines that the expression you enter is a measure (that is, returns a number), it automatically adds an aggregation to the expression when you commit the expression. For example, if you type `DATEDIFF('day', [Ship Date], [Order Date])` in an ad-hoc calculation and then press Enter, what you will see is the following:

```
SUM(DATEDIFF('day', [Ship Date], [Order Date]))
```

If you use a field that is already an aggregated field (for example, `SUM([Profit])`) in an ad-hoc calculation, the result is an aggregate calculation. For example, when you commit an ad-hoc calculation `SUM([Profit])/SUM([Sales])`, the result is:

```
AGG(SUM([Profit])/SUM([Sales]))
```

For more information on aggregate calculations, see [Aggregate Calculations](#) on page 838.

Ad-Hoc Calculations for Insight and Experimentation

Typically you create ad-hoc calculations on-the-fly to do things like

- Test a hunch
- Try a what-if scenario
- Debug a complex calculation

Managing Ad-Hoc Calculations

Use the following keystrokes to manage ad-hoc calculations.

- Double-click in an existing field on the **Rows**, **Columns**, **Marks**, or **Measure Values** shelf to open it as an ad-hoc calculation.
- Double-click elsewhere on any of these shelves to create a new ad-hoc calculation from scratch.
- Press Esc to cancel an ad-hoc calculation.
- Press Enter to commit an ad-hoc calculation, which updates the view and closes the ad-hoc calculation. Press Ctrl+Enter to commit the change and update the view without closing the ad-hoc calculation.
- If there is a field to the right of the current ad-hoc calculation on the same shelf, pressing Tab opens that field as an ad-hoc calculation. If there is no field to the right of the current ad-hoc calculation on the shelf, pressing Tab opens a new ad-hoc calculation. Shift+Tab has the same functionality, except it moves to the left.
- When you double-click a named field on a shelf to edit it, you are not changing the original named field.

In addition, the following rules govern the use of ad-hoc calculations.

- Only one ad-hoc calculation can be open at a time.
- If a field that has a geographic role or a fiscal year setting associated with it is added to an ad-hoc calculation, the ad-hoc calculation inherits that role or setting.
- The right-click context menu for an ad-hoc calculation supports the same options that would be available for any other field in the view of the same type, including the ability to change aggregation, show a filter, or format.
- Ad-hoc calculations are not available when you create groups, sets, hierarchies, or parameters.
- Ad-hoc calculations are valid for creating trend lines, forecasts, and reference lines, bands, and distributions.

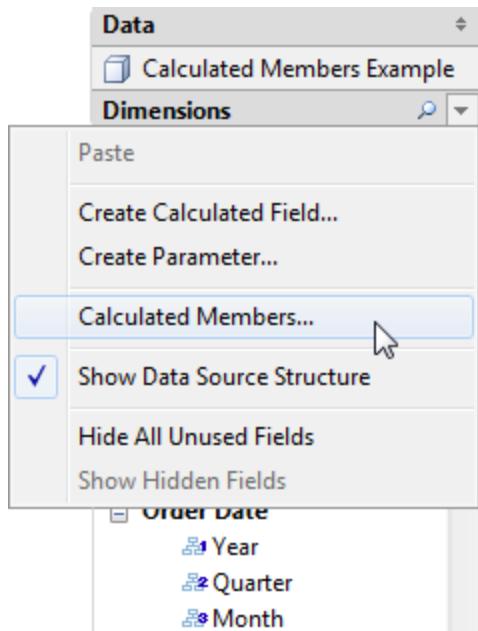
How to Create a Calculated Member

If you are using a multidimensional data source, you can create calculated members using MDX formulas instead of Tableau formulas. A calculated member can be either a calculated measure, which is a new field in the data source just like a calculated field, or a calculated dimension member, which is a new member within an existing hierarchy. For example, if a dimension Product has three members (Soda, Coffee, and Crackers) you can define a new calculated member Beverages that sums the Soda and Coffee members. When you then place the Products dimension on the **Rows** shelf it displays four rows: Soda, Coffee, Crackers, and Beverages.

Note: In Tableau, multidimensional data sources are supported only in Windows.

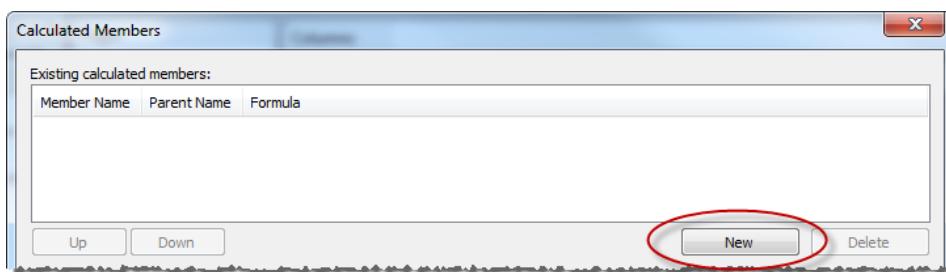
Defining Calculated Members

You can define a calculated dimension member by selecting **Calculated Members** from the Data pane menu. In the Calculated Members dialog box that opens, you can create, delete, and edit calculated members.

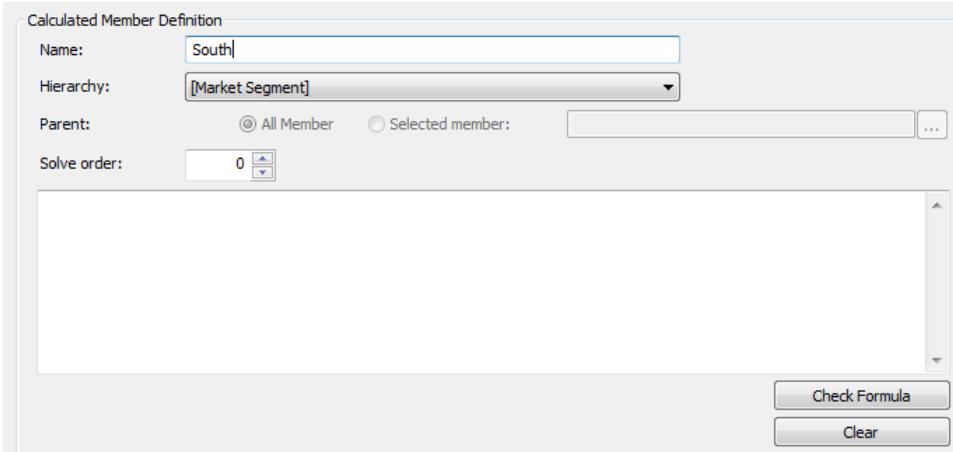


To create a new calculated member do the following:

1. Click New to add a new row to the list of calculated members at the top of the dialog box.



2. Type a Name for the new calculated member in the Member Definition area of the dialog box.



3. Specify the **Parent** member for the new calculated member. **All Member** is selected by default, however, you can choose **Selected Member** to browse the hierarchy and select a specific parent member.

4. Give the new member a solve order.

Sometimes a single cell in your data source can be defined by two different formulas. The solve order defines the precedence given to each formula. Formulas with a lower solve order are solved first. The default solve order is zero.

5. If you are connected to a Microsoft Analysis Services data source, the calculation editor contains a **Run before SSAS** check box. Choose this option to execute the Tableau calculation before any Microsoft Analysis Services calculations. For information on connecting to Microsoft Analysis Services data sources, see [Microsoft Analysis Services](#) on page 1426

6. Type or paste an MDX expression into the large white text box.

7. Click **Check Formula** to verify that the formula is valid.

8. When finished, click **OK**.

The new member displays in the **Data** pane either in the Measures area, if you chose [Measures] as the parent member, or in the Dimensions area under the specified parent member. You can use the new member just like any other field in the view.

Example – Creating a Calculated Field

This example shows how to create a calculated field using Tableau formulas and then use the new field in a data view.

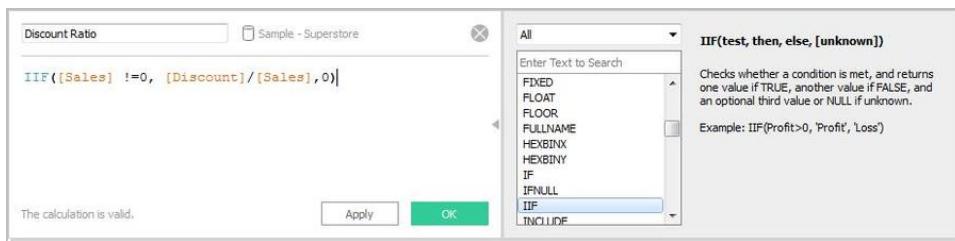
1. Connect to the **Sample - Superstore** data source, which is included with Tableau Desktop.
2. Click the drop down to the right of Dimensions on the **Data** pane and select **Create Calculated Field** to open the calculation editor.

See [Create Formulas in Tableau](#) on page 827 for information on how to bring the different element that make up a formula into the calculation editor.

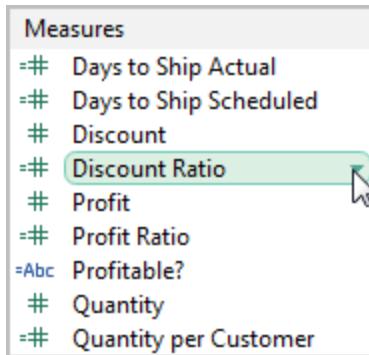
3. Name the new field **Discount Ratio** and create the following formula:

```
IIF([Sales] !=0, [Discount]/[Sales],0)
```

Build the formula by first double clicking the **IIF** statement from the function list and then either dragging the fields from the **Data** pane or typing them in the editor. You must type the operators (**!=** and **/**) manually. The **IIF** statement is used to avoid dividing by zero. For information on the **IIF** statement, see [Logical Functions](#) on page 1571.



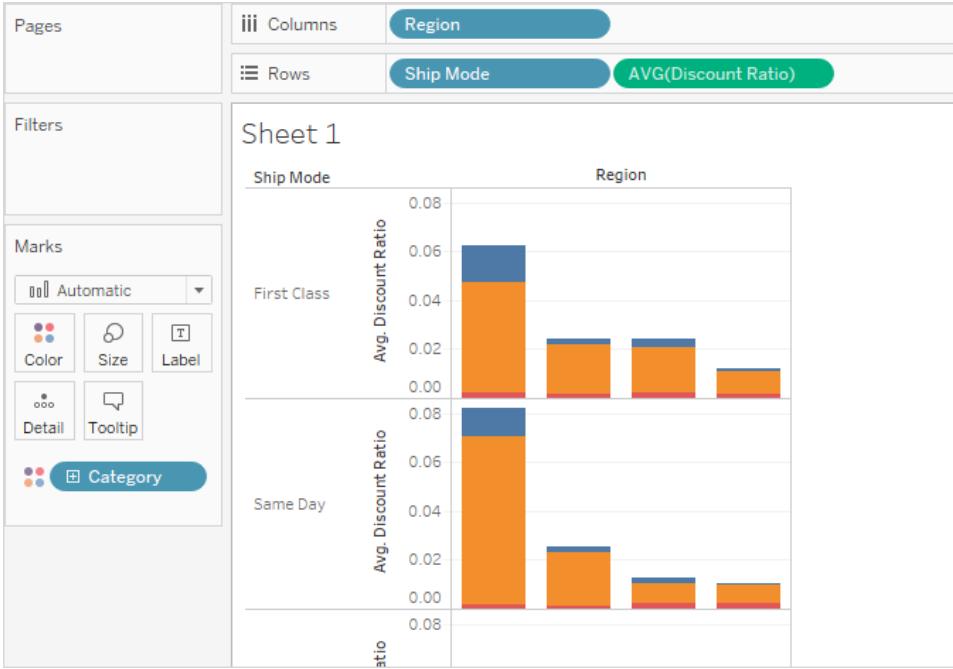
4. Click **OK** to add the new field to the Measures area in the **Data** pane. The new field is listed under Measures instead of Dimensions because the calculation returns a number.



5. Use the calculation in a new view.

Place **Region** on the **Columns** shelf, **Ship Mode** on the **Rows** shelf, and **Category** on Color in the Marks card. Then place the new calculated field, **Discount Ratio**, on the **Rows** shelf.

6. Change the aggregation for **Discount Ratio** for the field in the view from Sum to Average. To do this, right-click (Control-click on a Mac) the **Discount Ratio** field on the **Rows** shelf and choose **Measure > Average**. Here is what the view should now look like:



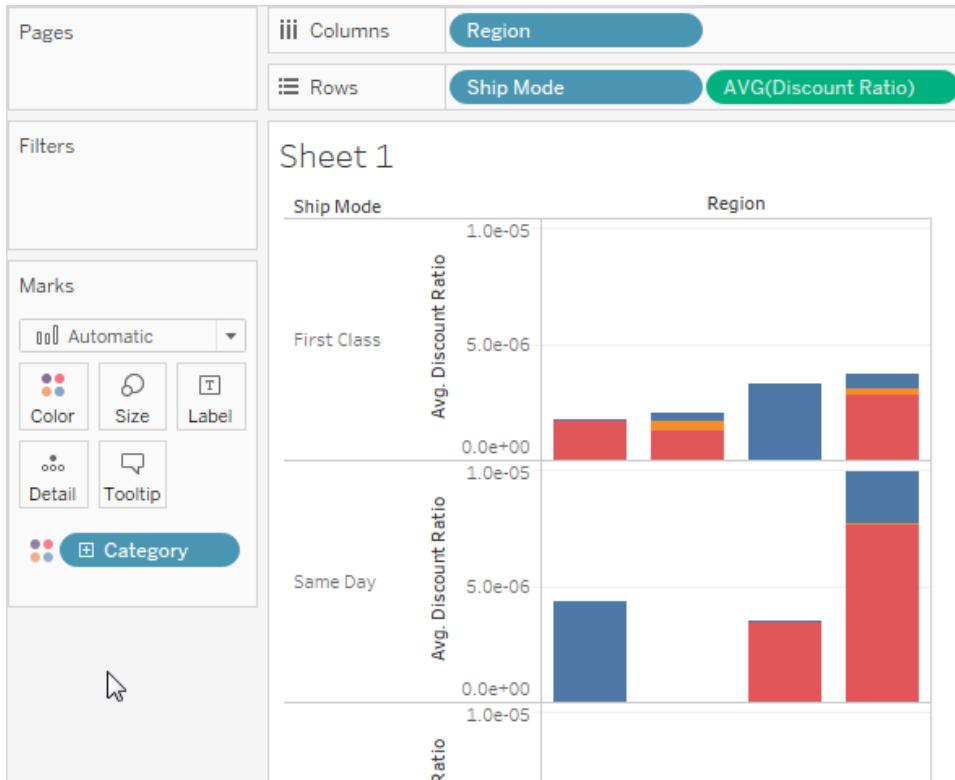
- But now you learn that there has been a change in company policy. You must edit the calculation to compute discount ratio only for sales over \$2000.

Right-click **Discount Ratio** in the **Data** pane and select **Edit**.

In the calculation editor, change the formula to compute discount ratio only for sales over \$2000:

```
IIF([Sales] > 2000, [Discount]/[Sales], 0)
```

The view automatically updates after you click **OK** to dismiss the calculation editor.



Aggregate Calculations

Aggregate functions allow you to summarize data. As described in [Aggregations on page 329](#), Tableau includes a variety of predefined aggregations such as summation and variance. An aggregate calculation allows you to define aggregations other than these predefined choices.

About Aggregate Calculations

Suppose you want to analyze the overall gross margin for every product in your data source. One way to do this is to create a new calculated field called **Margin** that is equal to the profit divided by the sales. Then you could place this measure on a shelf and use the predefined summation aggregation. In this scenario, **Margin** is defined as follows:

Margin = SUM([Profit]/ [Sales])

This formula calculates the ratio of profit and sales for every row in the data source, and then sums the numbers. That is, the division is performed before the aggregation. However, this is almost certainly not what you would have intended because summing ratios is generally not useful.

Instead, you probably want to know the sum of all profits divided by the sum of all sales. That formula is shown below.

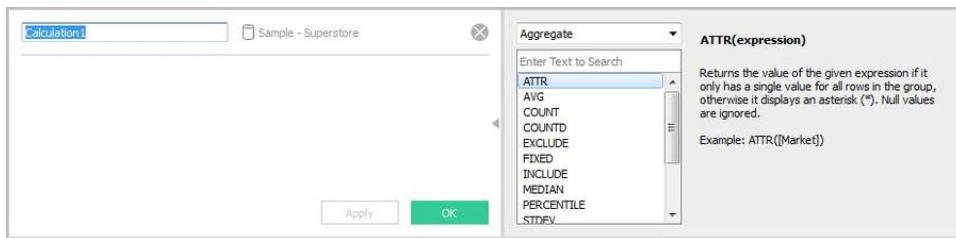
Margin = SUM([Profit]) / SUM([Sales])

In this case, the division is performed after each measure is aggregated. An aggregate calculation allows you to create formulas like this.

How to Create an Aggregate Calculation

When a calculation uses an aggregate function, it's called an aggregate calculation. You create an aggregate calculation by defining a new calculated field as described in [Create or Edit a Calculated Field on page 822](#). The formula will contain one or more aggregate functions. You can easily pick an aggregate function in the calculation editor by selecting the **Aggregate** category from the Functions menu as shown below.

These functions are identical to the predefined aggregate functions listed in [Aggregations on page 329](#).



The aggregate calculation appears with the letters **AGG** in front of it when it is placed on a shelf.



When you create an aggregate calculation, no further aggregation of the calculation is possible. Therefore, the field's context menu does not offer any aggregation choices. However, you can disaggregate the field.

The rules that apply to aggregate calculations are:

- For any aggregate calculation, you cannot combine an aggregated value and a disaggregated value. For example, `SUM(Price) * [Items]` is not a valid expression because `SUM(Price)` is aggregated and `Items` is not. However, `SUM(Price*Items)` and `SUM(Price) * SUM(Items)` are both valid.
- Constant terms in an expression act as aggregated or disaggregated values as appropriate. For example: `SUM(Price*7)` and `SUM(Price) * 7` are both valid expressions.
- All of the functions can be evaluated on aggregated values. However, the arguments to

any given function must either all be aggregated or all disaggregated. For example: MAX(SUM(Sales), Profit) is not a valid expression because Sales is aggregated and Profit is not. However, MAX(SUM(Sales), SUM(Profit)) is a valid expression.

- The result of an aggregate calculation is always a measure.
- Like predefined aggregations, aggregate calculations are computed correctly for grand totals. Refer to [Grand Totals on page 979](#) for more information.

Aggregate Calculations in a Disaggregated State

If an aggregate calculation is disaggregated, the calculation is modified in a way that depends on the functions used. Every function has a disaggregated substitute, as shown below.

Aggregation Function	Disaggregated Substitute
AVG(data)	data
COUNT(data)	IIF(ISNULL(data), 0, 1)
COUNTD(data)	IIF(ISNULL(data), 0, 1)
MAX(data)	data
MIN(data)	data
STDEV(data)	Null
STDEVP(data)	IIF (ISNULL (data), Null, 0)
SUM(data)	data
VAR(data)	Null
VARP(data)	IIF (ISNULL (data), Null, 0)

Note that STDEV and VAR are Null because those functions return Null if there are fewer than two elements in a group that are not Null, and each group has size 1 when it is disaggregated.

See [Aggregations on page 329](#) for descriptions of the aggregation functions.

Therefore, if you define an aggregate calculation called Margin that is equal to SUM(Profit)/SUM(Sales) and then disaggregate the data, it is interpreted as Profit/Sales.

Example – Aggregate Calculation

This example shows how to create an aggregate calculation called Margin, and use the new field in a data view.

1. Connect to the **Sample - Superstore** data source, which is included with Tableau Desktop.

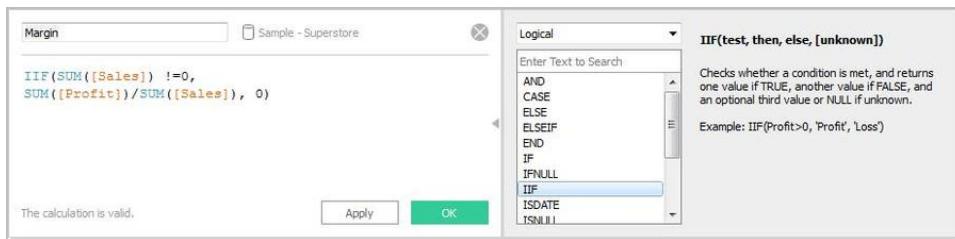
- Click the drop down to the right of Dimensions on the **Data** pane and choose **Create Calculated Field** to open the calculation editor.

See [Create Formulas in Tableau](#) on page 827 for information on how to bring the different element that make up a formula into the calculation editor.

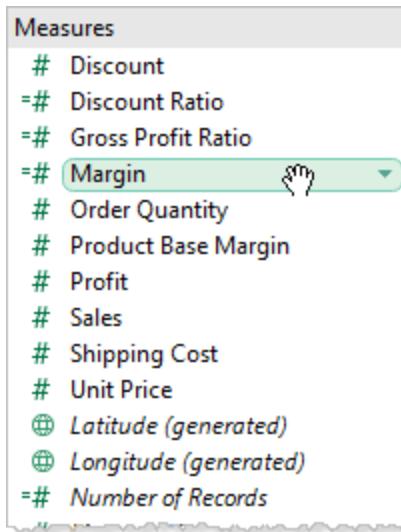
- Name the new field **Margin** and create the following formula:

```
IIF(SUM([Sales]) !=0, SUM([Profit])/SUM([Sales]), 0)
```

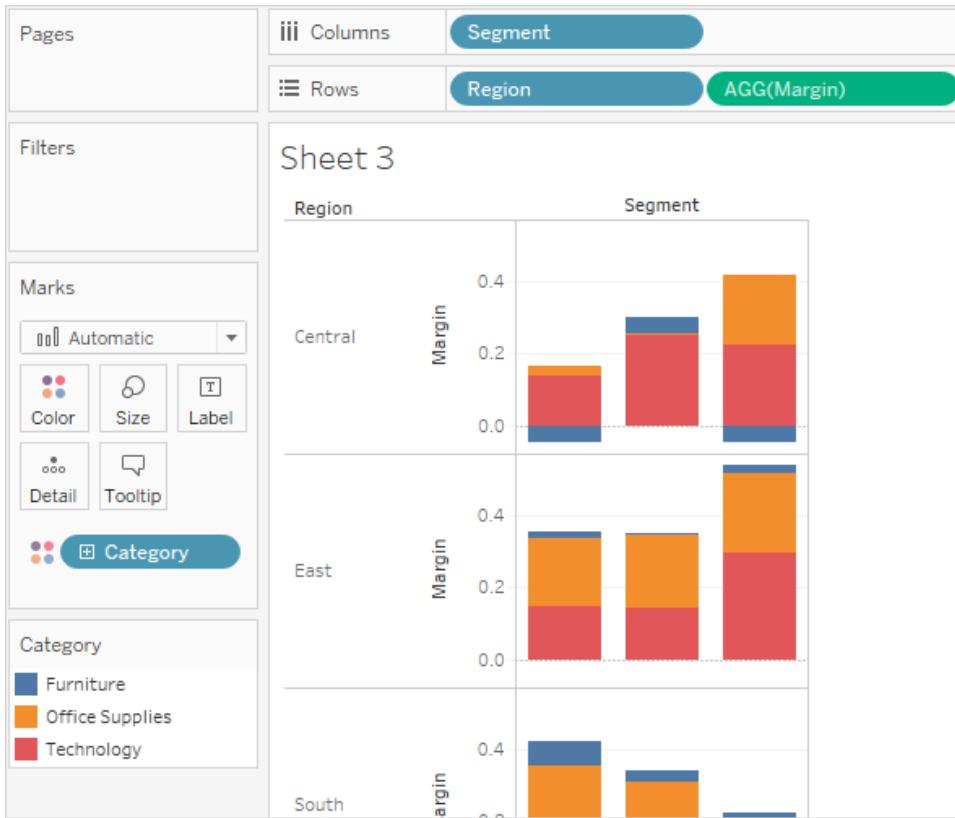
- Build the formula by first double clicking the **IIF** statement from the function list and then either dragging the fields from the **Data** pane or typing them in the editor. You must type the operators (**!=** and **/**) manually. The **IIF** statement is used to avoid dividing by zero. For information on the **IIF** statement, see [Logical Functions](#) on page 1571.



The new calculated field displays in the **Measures** area of the **Data** pane where you can use it like any other measure.

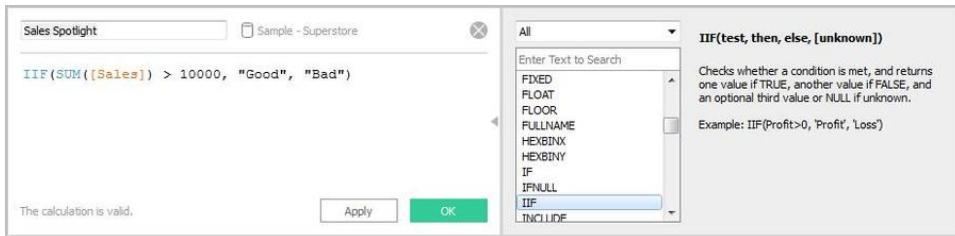


When **Margin** is placed on a shelf, its name is automatically changed to **AGG(Margin)**, which indicates that it is an aggregate calculation. As a result, the field's context menu does not include any aggregation choices because it is not possible to aggregate a field that is already aggregated.

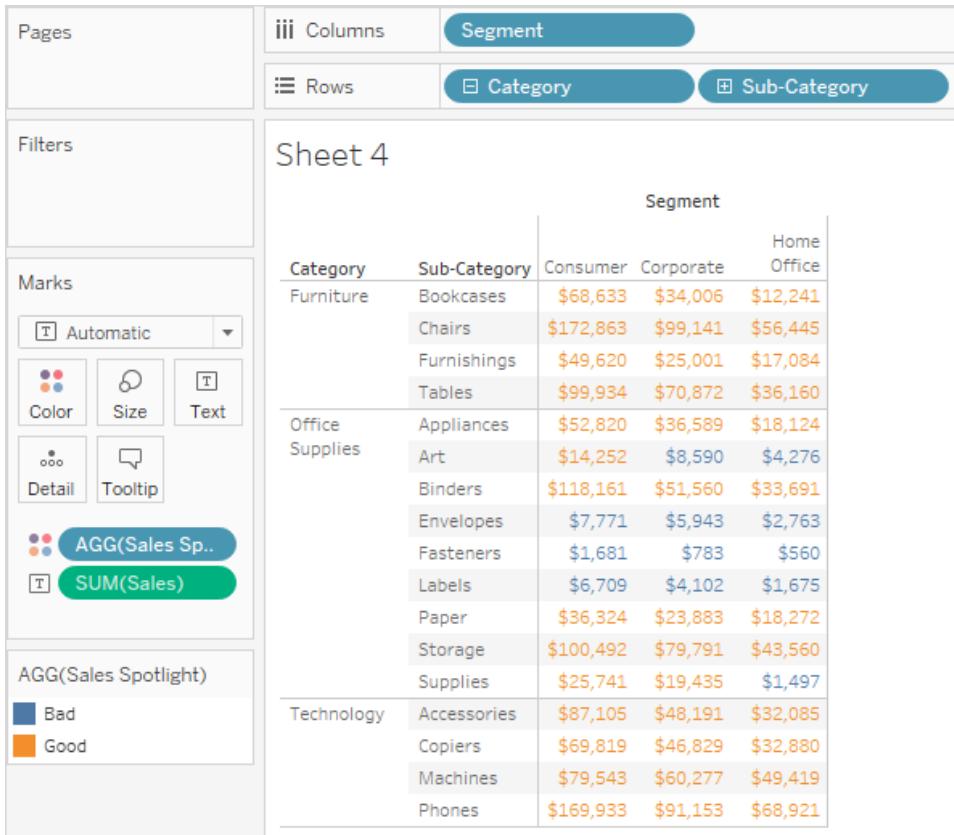


Example - Spotlighting Using Calculations

Spotlighting is a technique for showing discrete thresholds based on the values of a measure. For instance, you might want to color-code sales so that those over 10,000 appear green and those below 10,000 appear red. A spotlighting calculation is just a special case of a calculation that results in a discrete measure. A discrete measure is a calculation that is a dependent variable (and therefore a measure), but which results in a discrete result (as opposed to a continuous result). Thus the name discrete measure. Here is an example:



The formula in this example defines a discrete measure named **Sales Spotlight**. Discrete measures always appear with a blue abc icon in the **Data** pane. **Sales Spotlight** is classified as a measure in Tableau because it is a function of another measure; it is discrete because it produces discrete values ("Good" and "Bad"). Here is an example of this categorical measure in use:



Here, **Sales Spotlight** is on Color in the Marks card. It appears with the **AGG** prefix because it is an aggregate calculation. Values above 10,000 and below 10,000 are assigned different colors.

Transform Values with Table Calculations

A table calculation is a transformation you apply to the values for a measure in the view.

You can use table calculations for a variety of purposes, including:

- Transforming values to rankings
- Transforming values to show running totals
- Transforming values to show percent of total

Table calculations are available as you work with views in Tableau Desktop and also as you edit views in Tableau Server or Tableau Online.

Table calculations are a special type of calculated field that computes on the local data in Tableau. They are calculated based on what is currently in the view and do not consider any measures or dimensions that are filtered out of the view.

Note: The “table” is determined by the dimensions in the view—not the tables in your data source. Also, a “table calculation” is an entirely different thing than a calculated field. For information on calculated fields, see Calculated Fields.

Table Calculation Enhancements in Tableau 10

Tableau 10 provides some key enhancements to the experience of creating table calculations:

- As you work in the Table Calculation dialog box to configure a calculation, the view updates in real time to show the results of your choices.
- Highlighting** shows you the scope of your calculation by setting off part of the view with a colored background as you change the **Compute Using** option in the Table Calculation dialog box:

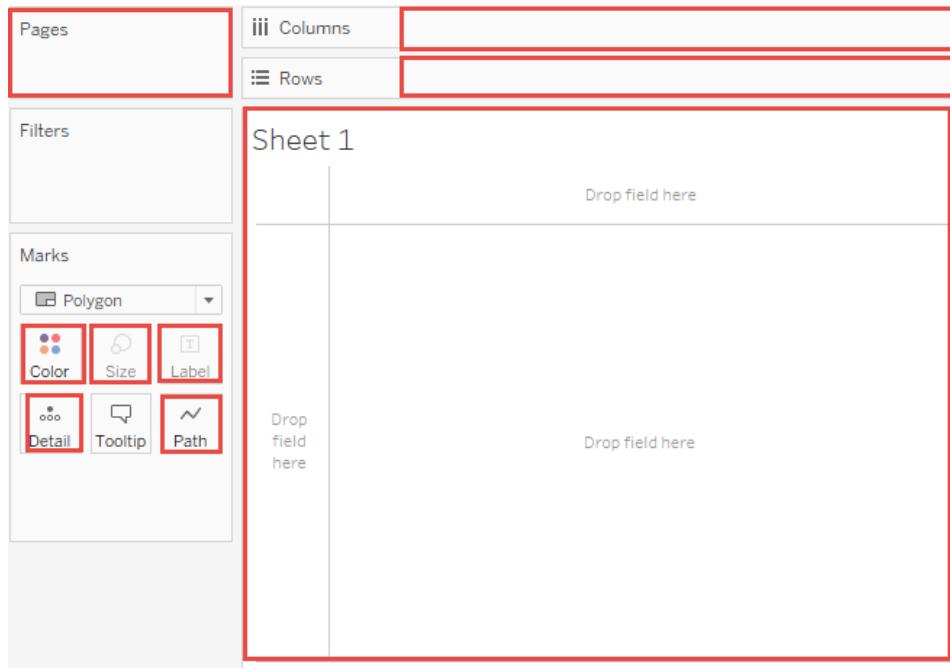
State	Order Date / Segment								
	2011			2012			2013		
	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..
Alabama	\$2,184	\$6,139	\$6,139	\$2,418	\$3,892	\$3,892	\$2,668	\$7,646	\$7,651
Arizona	\$4,658	\$7,846	\$8,295	\$5,632	\$9,232	\$9,611	\$2,184	\$4,077	\$6,242
Arkansas	\$5,196	\$6,303	\$6,303	\$437	\$437	\$444	\$1,989	\$1,989	\$2,224
California	\$53,365	\$75,971	\$91,304	\$53,741	\$74,158	\$88,444	\$64,507	\$109,659	\$131,211

If you select a different mark in the view while the Table Calculations dialog box is still open, highlighting will switch to the partition containing that mark.

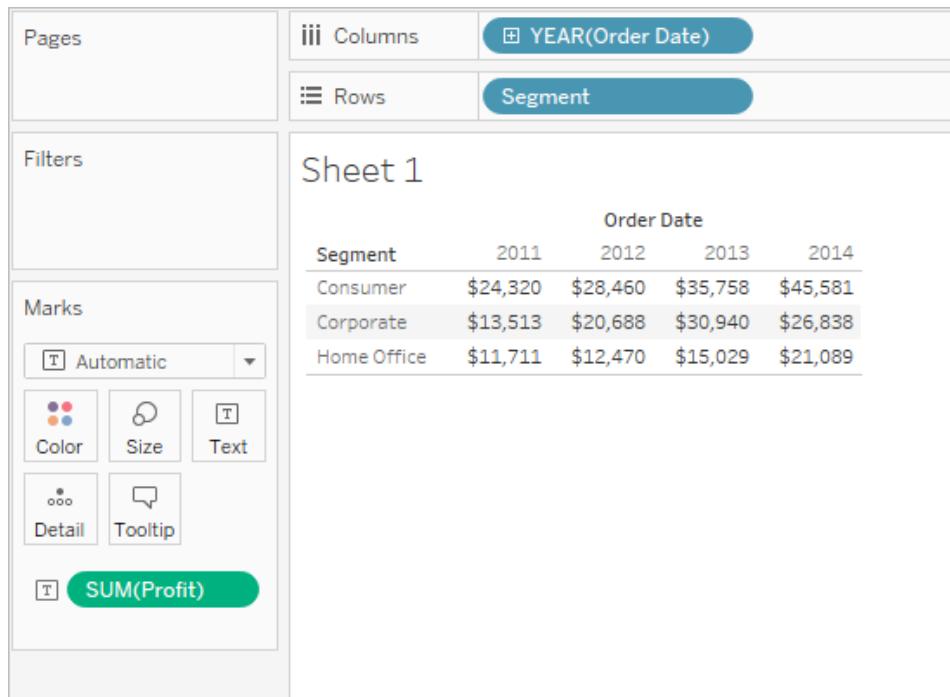
What is a Table Calculation?

A table calculation is a transformation you apply to the values of a single measure in your view, based on the dimensions in the level of detail.

For any Tableau view, there is a virtual table that is determined by the dimensions in the view. This table is not to be confused with the tables in your data source. Specifically, the virtual table is determined by the dimensions within the “level of detail,” which means the dimensions on any of the following shelves or cards in a Tableau worksheet:



Consider this simple view:



In this view, the dimensions are **Order Date** (with values aggregated up to **YEAR**) on **Columns**, and **Segment** on **Rows**. The individual cells in the table show the value of the **Profit** measure for each combination of **Order Date** and **Segment**. There are four years' worth of

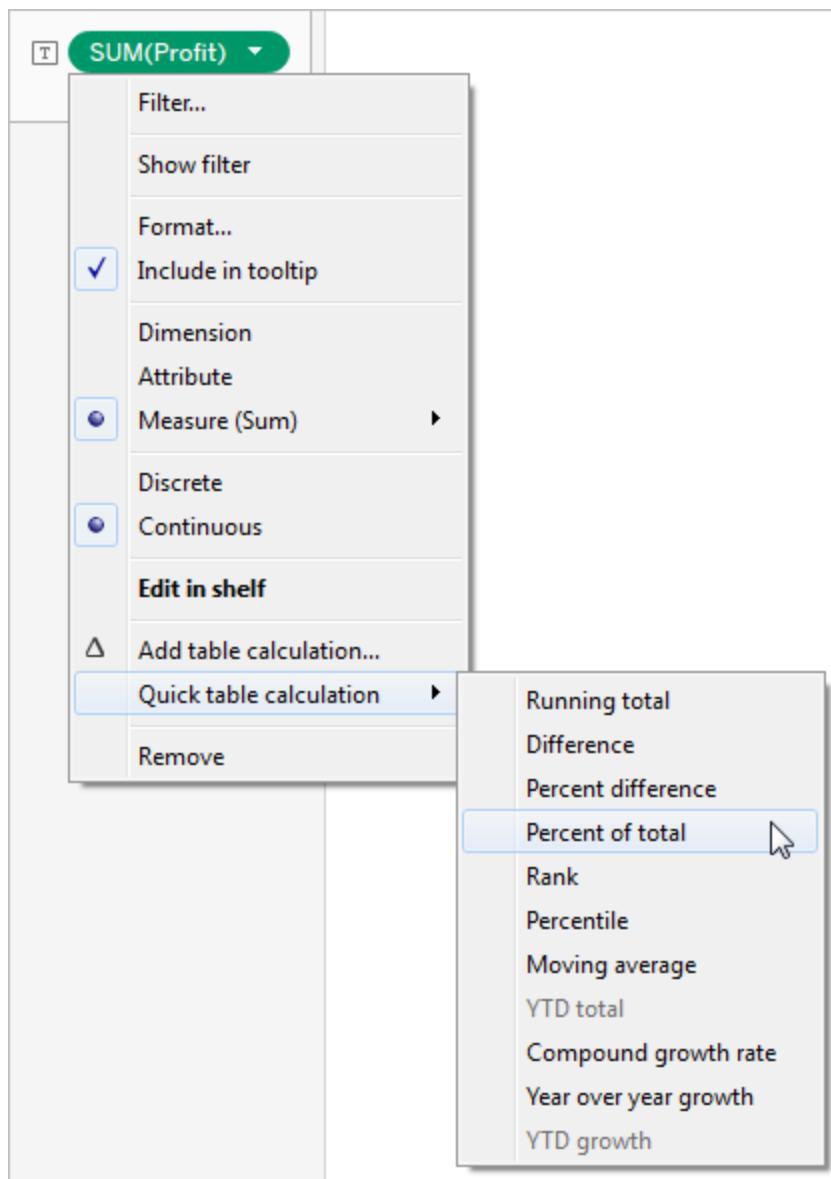
Order Date data, and three **Segments**; multiply these numbers to get 12 individual cells, each showing a **Profit** value.

Suppose you want to see not absolute dollar values, but the percentage of the total profit that each of these 12 individual profit values contributes to the total profit, so that when all the cell values are added, they total 100%. To do this, you can add a table calculation. Here's how you could add a table calculation to show this.

1. You always add a table calculation to a measure in the view. In this case there is only one measure in the view—**SUM(Profit)**—so when you right-click that measure on the Marks card (or Control-click on a Mac) you see two options that mention table calculations:

- **Add Table Calculation**
- **Quick Table Calculation**

If you choose **Quick Table Calculation**, you will see a range of options:



2. **Percent of Total** looks right. Choose that, and the view updates to show percentages, instead of absolute dollar values:

The screenshot shows the Tableau Data Explorer interface. In the top right, there's a table view with the following structure:

Segment	2011	2012	2013	2014
Consumer	18.13%	21.22%	26.66%	33.99%
Corporate	14.69%	22.49%	33.64%	29.18%
Home Office	19.42%	20.68%	24.92%	34.97%

In the bottom left, the Marks card is visible, showing various mark types like Color, Size, Text, Detail, and Tooltip. The **SUM(P)** button is selected, and it has a small triangle icon next to it, indicating a table calculation is applied.

Notice the triangle icon that now appears next to **SUM(P)** on the Marks card:

A zoomed-in view of the Marks card from the previous screenshot. The **SUM(P)** button is highlighted with a red box, and the small triangle icon to its right is also highlighted with a red box, emphasizing the applied table calculation.

This indicates that a table calculation is currently being applied to this measure.

3. But what about when the view is not a text table? Are table calculations still an option? They are.

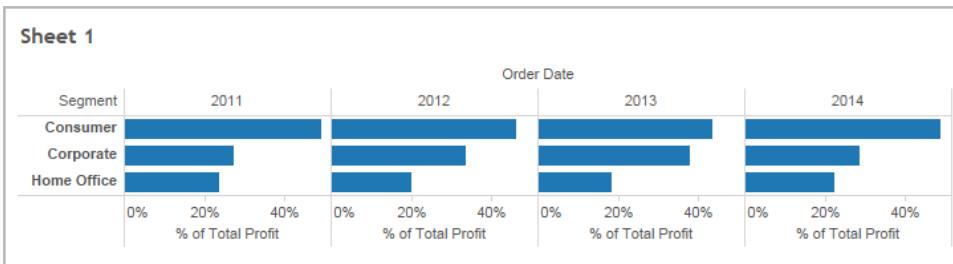
Back up one step (click the Undo button on the toolbar) to remove the quick table calculation.

4. Now use Show Me to change the chart type to horizontal bars:



You might not be as likely to refer to this view as a “table,” but the dimensions in the level of detail are the same, so the view’s virtual table is the same.

5. You can now apply the exact same table calculation as before—right-click **SUM(Profit)**, choose **Quick Table Calculation**, and then **Percent of Total**. The view changes to show percentages along the horizontal axis, instead of dollar values:



Note: When you are learning about table calculations, or experimenting with different options, a text table is usually going to provide more intuitive insight than other chart types.

Quick Table Calculations

A quick table calculation is a one-step process where you choose a common table calculation type from a list. Tableau automatically applies the most typical settings for that calculation type.

To apply a quick table calculation, do the following:

1. Click a measure in the view and choose **Quick Table Calculation** from the context menu.
2. Choose a Calculation type from the sub-menu.

If the results are satisfactory, you’re done. If not, you can continue working with the calculation by clicking the measure again and choosing **Edit table calculation**.

The list of available quick calculation types does not exactly match the list of default calculation types you see in the Table Calculation dialog box. For example, a **Compound growth rate**

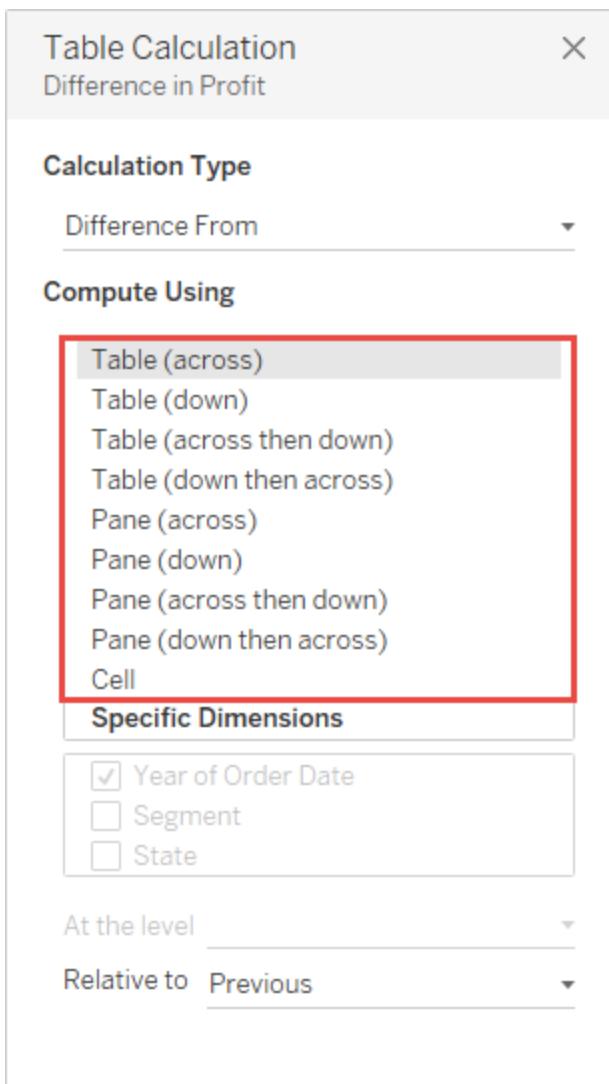
quick table calculation is a **Percent Difference From** table calculation, but with the **Compute compounded rate** option selected. If you want to know how any given quick table calculation is defined, add the calculation and then go back and edit it to see how it is configured in the Table Calculation dialog box.

Choose Your Approach to Table Calculations

The easiest way to add a table calculation is to use a quick table calculation. See [Quick Table Calculations on the previous page](#). But if you need more flexibility than a quick table calculation affords, you can base your calculation on the visual structure of your view, or you can reference the specific dimensions in the view.

Table Calculations: Use the Visual Structure of Your View

When you configure a table calculation using the **Compute Using** options in the Table Calculation dialog box, you define a table calculation using the visual structure of your view.



As you add or edit a table calculation in the Table Calculation dialog box, Tableau highlights the effects of the **Compute Using** options you choose in the view.

For example, a **Table (Across)** calculation moves from left to right across the entire width of the view:

Region	2010				Order Date
	Q1	Q2	Q3	Q4	
Central	\$7,306	\$12,201	\$19,208	(\$34,893)	
East		\$17,023	\$4,287	\$64,183	(\$75,536)
International		\$60,022	\$99,595	\$359,091	(\$525,164)
South		\$20,821	\$11,251	\$13,935	(\$22,633)
West		\$21,013	(\$205)	\$13,010	(\$19,048)

Table Calculation X

Difference in Profit

Calculation Type

Difference From ▼

Compute Using

- Table (across) ▼
- Table (down)
- Table (across then down)
- Table (down then across)
- Pane (across)
- Pane (across then down)
- Pane (down then across)
- Cell

Specific Dimensions

- Year of Order Date
- Quarter of Order Date
- Region

At the level ▼

Relative to ▼ Previous

A Pane (Down then Across) calculation moves down the first column in a pane, then down the second column in a pane, and so on:

The screenshot shows a Tableau interface with a data view and a calculation dialog.

Data View:

Region	2010				Order Date Q1
	Q1	Q2	Q3	Q4	
Central	\$6,397	(\$2,876)	(\$2,415)	\$16,998	
East	(\$12,593)		(\$10,790)	\$34,185	(\$6,459)
International	\$35,599	\$78,598	\$173,907	\$468,815	\$19,187
South	(\$54,297)	(\$93,498)	(\$181,841)	(\$526,998)	(\$24,467)
West	\$32,199	\$32,391	\$20,935	(\$6,010)	(\$2,425)

Table Calculation Dialog:

Table Calculation
Difference in Profit

Calculation Type
Difference From

Compute Using

- Table (across)
- Table (down)
- Table (across then down)
- Table (down then across)
- Pane (across)
- Pane (across then down)
- Pane (down then across)** (selected)
- Cell
- Specific Dimensions

Quarter of Order Date

Region

Year of Order Date

At the level

Relative to Previous

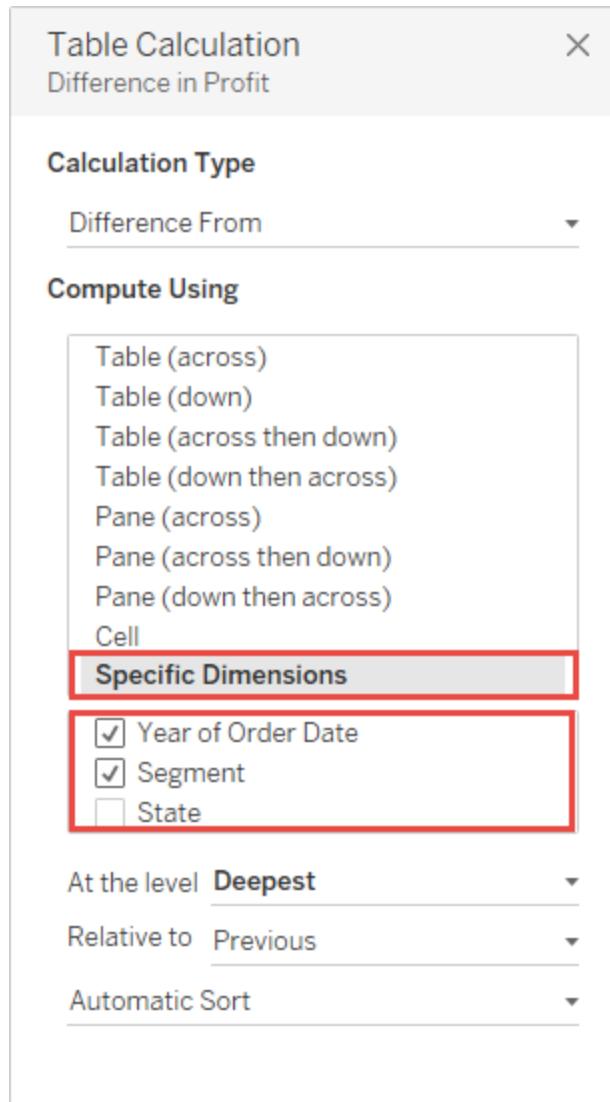
The **Compute Using** approach should be convenient and intuitive for most users because the calculation aligns with what you see in your view.

The list of options under **Compute Using** changes according to the content of your view. For example, if you have only two dimensions in your view (one on Rows and one on Columns), there will be no Pane options because your view is not complex enough for there to be multiple panes—in this case, your entire view is single pane

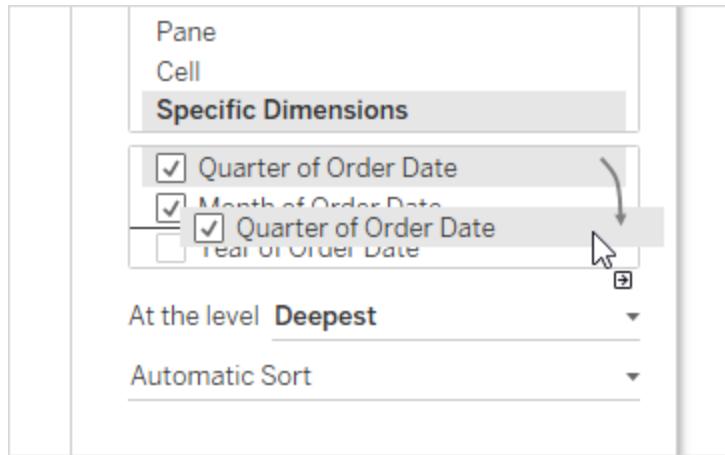
Table Calculations: Reference Specific Dimensions in Your View

With this approach, you define a table calculation by referencing the dimensions in the view.

Choose **Specific Dimensions** in the Table Calculation dialog box and then select or clear fields in the box below.



To configure a table calculation using specific dimensions, click **Specific Dimensions** and then select dimensions in the box below. You can also drag dimensions up or down in the list to set the sequence for the calculation:



You can still see the effect of your choices in the view, but because the visual structure and the table calculation are not necessarily aligned (as they always are with the **Compute Using** options), the results can sometimes look a bit exotic. For example:

State	Order Date / Segment											
	2011			2012			2013			2014		
	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..
Alabama												
Arizona	\$-1,003	\$-1,151	\$-1	\$-727	\$-297	\$61	\$-510	\$1,896	\$-589	\$36	\$35	\$325
Arkansas	\$2,400	\$534	\$1	\$76	\$151	\$-56	\$-776	\$-2,687	\$-862	\$-629	\$-302	\$-842
California	\$4,834	\$3,007	\$2,765	\$8,862	\$3,582	\$1,825	\$992	\$479	\$928	\$1,107	\$633	\$495
Colorado	\$-6,176	\$-3,261	\$-2,763	\$-10,272	\$-3,417	\$-1,824	\$10,2	\$-11,0				
Connecticut	\$-24	\$334	\$2	\$1,507	\$56	\$5	\$2					
Delaware	\$-246	\$773	\$66	\$551	\$368	\$262	\$6					
District of Columbia				\$226	\$-589	\$-271	\$-8					
Florida	\$-20	\$-894	\$-1,837	\$-2,032	\$-868	\$167	\$					
Georgia	\$822	\$233	\$1,849	\$5,137	\$1,456	\$-70	\$1,2					
Idaho	\$-878	\$-473	\$-78	\$-3,978	\$-403	\$-97	\$-1,2					
Illinois	\$-1,711	\$-188	\$-208	\$-1,188	\$735	\$-842	\$-1,3					
Indiana	\$2,319	\$279	\$314	\$2,302	\$-560	\$1,357	\$2,2					
Iowa	\$-502	\$-149	\$-108	\$-1,139	\$-259	\$-102	\$-7					
Kansas	\$-110	\$75	\$191	\$-69	\$0	\$-346						
Kentucky	\$278	\$613	\$1,152	\$2,403	\$297	\$146	\$1,0					
Louisiana	\$-256	\$-283	\$-1,163	\$-2,123	\$-297	\$-215	\$-1,1					
Maine	\$-22	\$-148	\$-180	\$-288	\$0	\$32	\$-					
Maryland	\$171	\$54	\$4	\$1,721	\$763	\$197	\$8					
Massachusetts	\$326	\$906	\$344	\$-1,416	\$-294	\$952	\$1					
Michigan	\$213	\$-619	\$163	\$884	\$1,026	\$1,300	\$5,5					
Minnesota	\$4,275	\$-580	\$726	\$-240	\$-1,440	\$-1,764	\$-6,4					
Mississippi	\$-4,482	\$26	\$-1,236	\$-948	\$67	\$717	\$5					
Missouri	\$-347	\$3	\$248	\$291	\$2,478	\$0	\$-8					
Montana	\$-31	\$-37	\$-246	\$-139	\$-2,602	\$85						

Table Calculation X

Calculation Type

Difference From

Compute Using

- Table (across)
- Table (down)
- Table (across then down)
- Table (down then across)
- Pane (across)
- Pane (across then down)
- Pane (down then across)
- Cell

Specific Dimensions

Year of Order Date

State

Segment

At the level Deepest

Relative to Previous

Automatic Sort

If you are configuring table calculations using specific dimensions, then you should become familiar with the concepts of addressing and partitioning. See **Table Calculations: Addressing and Partitioning** below.

Table Calculations: Addressing and Partitioning

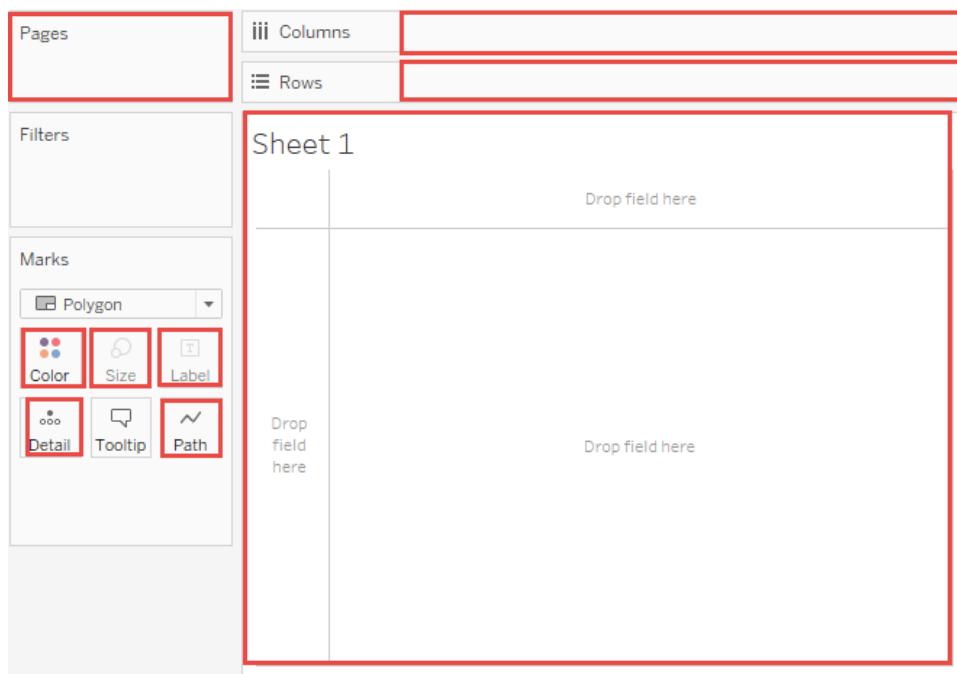
If you are configuring table calculations using specific dimensions, then you should become familiar with the concepts of *addressing* and *partitioning*.

Background: How Level of Detail Affects the Direction and Scope of Your Calculations

Your view contains marks; these marks are typically aggregated. This means that an individual mark collects data not from a single row in your data source, but from all rows that have the same values for the dimensions in your view.

See [Aggregating Data](#) on page 334 for more on aggregation in Tableau.

The number of marks in your view is determined by the view's level of detail. Dimensions located on any of the shelves or cards highlighted in the following image contribute to the level of detail:



So if your view has **State** and **Segment** on Columns, and **Region** on Rows, then a single mark represents all rows in your data source that share the same value for these three dimensions. For example, one mark would aggregate individual values for rows where the value of State is Indiana, the value of Segment is Consumer, and the value of Region is Central. The measure that Tableau is aggregating—it might be Profit, Sales, or any other measure in the data source—is for you to decide, as is the aggregation itself—it might be Sum, Average or another aggregation.

When you add a table calculation, you must use all dimensions in the level of detail either for partitioning (scoping) or for addressing (direction):

- The dimensions that define how to group the calculation, that is, define the scope of data it is performed on, are called partitioning fields. The table calculation is performed

separately within each partition.

- The remaining dimensions, upon which the table calculation is performed, are called addressing fields, and determine the direction of the calculation.

Partitioning fields break the view up into multiple sub-views (or sub-tables), and then the table calculation is applied to the marks within each such partition. The direction in which the calculation moves (for example, in calculating a running sum, or computing the difference between values) is determined by the addressing fields. So when you order the fields in the Specific Dimensions section of the Table Calculation dialog box from top to bottom, you are specifying the direction in which the calculation moves through the various marks in the partition.

When you add a table calculation using the **Compute Using** options, Tableau identifies some dimensions as addressing and others as partitioning automatically, as a result of your selections. But when you use Specific Dimensions, then it's up to you to determine which dimensions are for addressing and which for partitioning.

The following example demonstrates how experimenting with the options in the Table Calculations dialog box can help you understand how your calculation is working.

Example - Addressing and Partitioning

In this example, using the Sample – Superstore data source provided with Tableau Desktop, create the following simple view:

1. Drag **Order Date** and **Segment** to Columns.
2. Drag **State** to Rows.
3. Drag **Sales** to Text.

This gives you a basic view:

State	Order Date / Segment											
	2011			2012			2013					
	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..
Alabama	\$2,184	\$3,955		\$2,418	\$1,474		\$2,668	\$4,978	\$5	\$267		
Arizona	\$4,658	\$3,188	\$449	\$5,632	\$3,600	\$379	\$2,184	\$1,893	\$2,165	\$3,950		
Arkansas	\$5,196	\$1,106		\$437		\$6	\$1,989		\$235	\$1,179		
California	\$53,365	\$22,605	\$15,333	\$53,741	\$20,418	\$14,286	\$64,507	\$45,152	\$21,552	\$58,023		
Colorado	\$5,422	\$1,065	\$15	\$3,122	\$1,468	\$49	\$2,487	\$5,102	\$3,077	\$4,763		
Connecticut	\$1,121	\$1,673		\$527	\$735	\$22	\$1,902	\$1,824	\$273	\$2,383		
Delaware	\$189	\$4,399	\$198	\$2,915	\$2,468	\$808	\$2,372	\$348		\$11,486		
District of Columbia				\$2,670			\$83		\$34			
Florida	\$5,875	\$2,948	\$25,426	\$5,905	\$7,822	\$1,451	\$4,832	\$6,306	\$2,394	\$16,090		
Georgia	\$2,753	\$1,528	\$259	\$9,377	\$1,701	\$260	\$5,504	\$3,082	\$5,471	\$6,813		
Idaho	\$10	\$456		\$496	\$1,005		\$35	\$1,149		\$904		
Illinois	\$11,685	\$1,923	\$2,594	\$9,890	\$4,624	\$4,064	\$12,661	\$3,444	\$4,928	\$10,946		

4. Click **SUM(Sales)** on the Marks card and then select **Add table calculation** from the context menu.

This opens the Table Calculation dialog box.

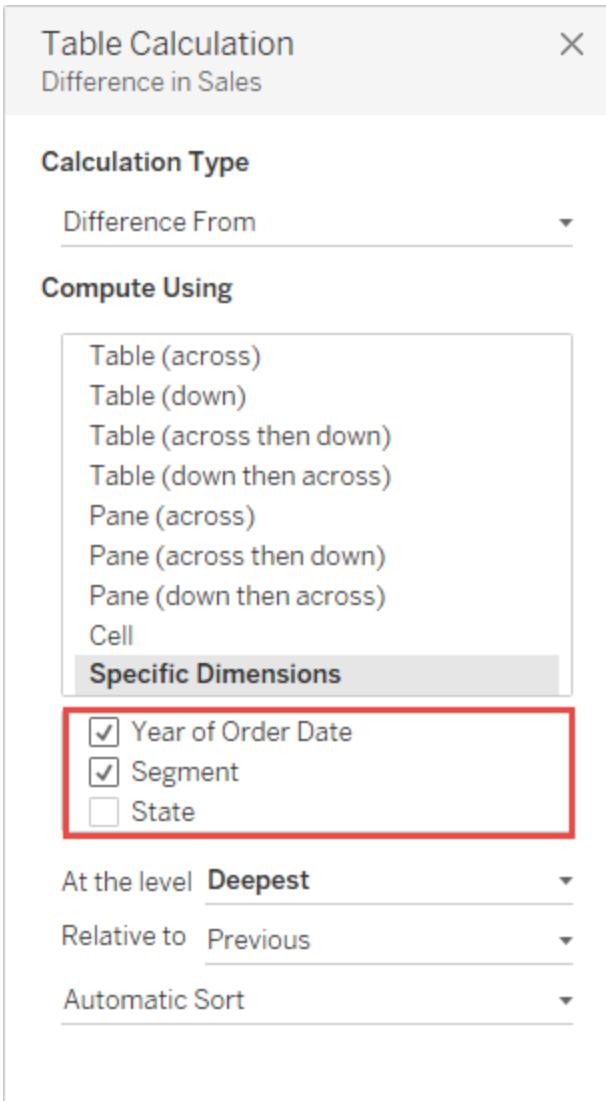
Note the yellow highlighting running across the top row of the view. Highlighting shows you exactly how the current **Compute Using** option partitions the view.

State	Order Date / Segment								
	2011			2012			2013		
	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..
Alabama	\$1,770	-\$3,955		\$2,418	-\$943	-\$1,474	\$2,668	\$2,310	-\$4,973
Arizona	-\$1,470	-\$2,740		\$5,184	-\$2,033	-\$3,221	\$1,805	-\$291	\$272
Arkansas	-\$4,090	-\$1,106		\$437	-\$437	\$6	\$1,983	-\$1,989	\$235
California	-\$30,760	-\$7,272		\$38,408	-\$33,323	-\$6,132	\$50,221	-\$19,355	-\$23,600
Colorado	-\$4,357	-\$1,051		\$3,107	-\$1,653	-\$1,419	\$2,438	\$2,615	-\$2,025
Connecticut	\$551	-\$1,673		\$527	\$208	-\$713	\$1,880	-\$78	-\$1,550
Delaware	\$4,210	-\$4,201		\$2,717	-\$448	-\$1,660	\$1,564	-\$2,024	-\$348
District of Columbia	\$0	\$0		\$2,670	-\$2,670	\$0	\$83	-\$83	\$34
Florida	-\$2,927	\$22,478		-\$19,521	\$1,917	-\$6,370	\$3,381	\$1,474	-\$3,912
Georgia	-\$1,224	-\$1,269		\$9,118	-\$7,676	-\$1,441	\$5,244	-\$2,422	\$2,389
Idaho	\$446	-\$456		\$496	\$509	-\$1,005	\$35	\$1,114	-\$1,149
Illinois	-\$9,762	\$671		\$7,296	-\$5,266	-\$560	\$8,597	-\$9,217	\$1,484

5. Set **Calculation Type** to **Running Total**.

6. Click **Specific Dimensions**, the last item in the **Compute Using** list.

The list box below the **Compute Using** list, which lists all the dimensions in the view's level of detail, is now available for edit:



Any field with a check mark is an addressing field. **Year of Order Date** and **Segment** are checked, so this leaves **State** as the only partitioning field. This configuration matches **Table (Across)**, which was the **Compute Using** value that was selected when you clicked **Specific Dimensions**.

Note: You can select any of the **Compute Using** options and then click **Specific Dimensions** to see how your selected option translates in terms of specific dimensions.

Year of Order Date and **Segment** are both on Columns in the view, so these dimensions define the columns in the view. Notice that the values running from left to

right in the view are ascending—the running total increases across the entire width of the view.

But when you get to the second row (Arizona) you enter a new partition, so the calculation restarts.

To gain insight, try changing things up in the list of specific dimensions.

7. Clear the check mark from **Year of Order Date**.

Year of Order Date is now a partition field, leaving only **Segment** for addressing. **Year of Order Date** and **State** are both now partitioning fields, which means the partitions are more plentiful—and smaller, because you are using two fields to partition (think “divide up”) the view. There are only three marks per partition now, so as you continue moving left-to-right across the view, you enter a new partition with the fourth value, and the calculation restarts.

State	Order Date / Segment								
	2011			2012			2013		
	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..
Alabama	\$2,184	\$6,139	\$6,139	\$2,418	\$3,892	\$3,892	\$2,668	\$7,646	\$7,651
Arizona	\$4,658	\$7,846	\$8,295	\$5,632	\$9,232	\$9,611	\$2,184	\$4,077	\$6,242
Arkansas	\$5,196	\$6,303	\$6,303	\$437	\$437	\$444	\$1,989	\$1,989	\$2,224
California	\$53,365	\$75,971	\$91,304	\$53,741	\$74,158	\$88,444	\$64,507	\$109,659	\$131,211

By default, marks are highlighted in the first partition in the view. The first headers for these fields are also highlighted—in the current case, 2011 and Alabama.

8. Reselect **Year of Order Date** and then drag **Segment** to the top of the list—above **Year of Order Date**. Highlighting shows that the partition is the same as in the default view, stretching left to right across the entire width of the view. This makes sense because the same two fields—**Segment** and **Year of Order Date**—are used for addressing.

But notice that the values are not the same as in the previous view:

State	2011			2012			2013		
	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..	Consumer	Corporate	Home Of..
	\$2,184	\$11,492	\$18,507	\$4,602	\$12,966	\$18,507	\$7,270	\$17,945	\$18,512
Alabama	\$2,184	\$11,492	\$18,507	\$4,602	\$12,966	\$18,507	\$7,270	\$17,945	\$18,512
Arizona	\$4,658	\$19,613	\$28,610	\$10,290	\$23,213	\$28,988	\$12,474	\$25,106	\$31,154
Arkansas	\$5,196	\$9,908	\$11,266	\$5,634	\$9,908	\$11,272	\$7,623	\$9,908	\$11,507
California	\$53,365	\$252,241	\$392,144	\$107,106	\$272,659	\$406,429	\$171,613	\$317,811	\$427,981

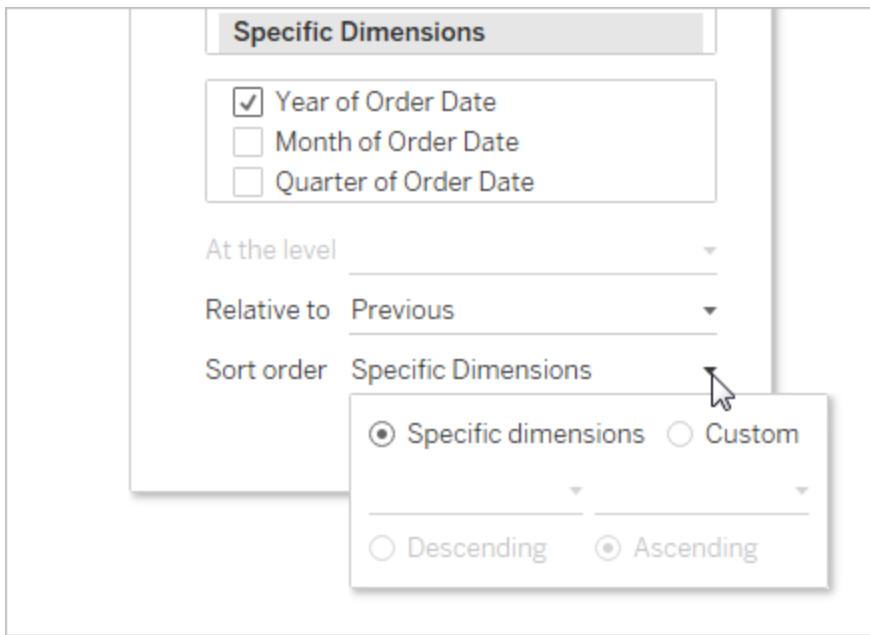
Because you reversed the order in which the fields are listed for **Specific Dimensions**, Tableau computes the running total using values for **Segment** first and then **Year of Order Date**. The arrows in the image show the order in which Tableau is computing the running sum. The logic of the calculation is now somewhat at odds with the visual alignment of the view, but that is what can happen when you configure a table calculation using **Specific Dimensions**.

Customize How Tableau Sorts the Results of Table Calculations

When you add a table calculation using specific dimensions from your view, you can also customize the way Tableau sorts the values. This option is available for all table calculation types except Rank and Percentile.

With custom sorting, you can sort the results of a table calculation using values from a field in the data source. With **Specific Dimensions** selected in the Table Calculations dialog box, do the following:

1. Click to open the **Specific dimensions** drop-down in the Custom Sort area at the bottom of the Table Calculations dialog box.



2. Click **Custom**.
3. From the drop-down list on the left, choose the field with the values you want to sort on.
4. From the drop-down list on the right, choose the aggregation you want to use with the field.
5. Specify whether to sort by **Descending** or **Ascending** values.

Create a Table Calculation

You can add a table calculation to a measure in the view. To add a table calculation:

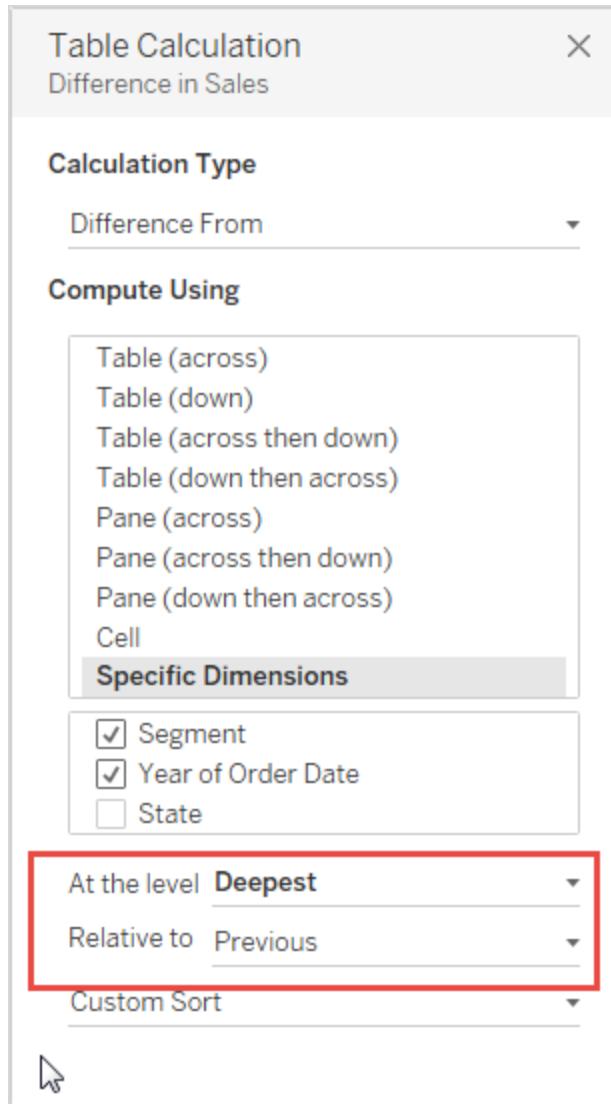
1. Click a measure in the view and choose **Add table calculation** from the field menu.

Tableau opens the Table Calculation dialog box and also adds highlighting to the view. As you work in the Table Calculation dialog box, highlighting shows the scope and direction of the calculation.

2. Choose a Calculation Type.

For details on available calculation types, see [Table Calculation Types](#) on page 865.

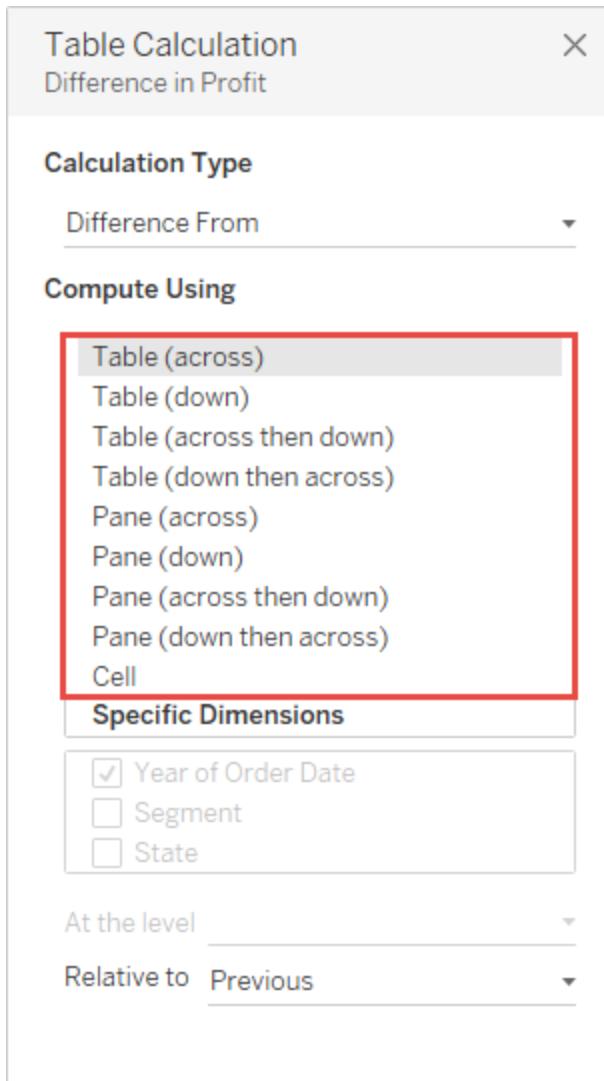
For each calculation type there is a specific set of options for you to consider. For example, for a **Difference From** calculation, there are two associated fields:



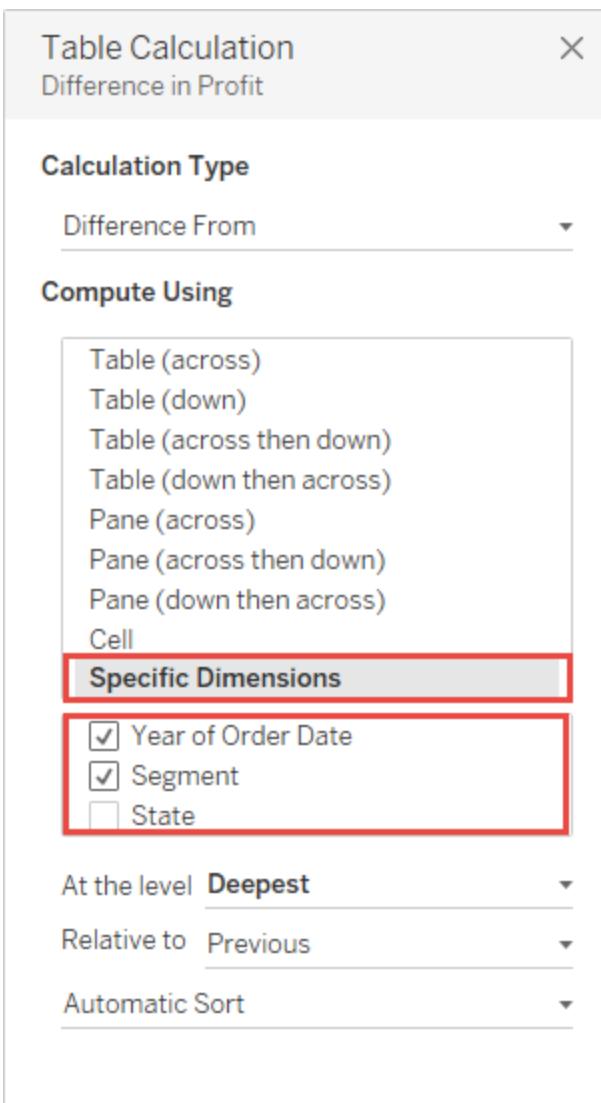
For information on these fields, go to [Table Calculation Types](#) on page 865, and then click the link for **Difference From Calculation**.

3. To define the table calculation using the visual structure of your view, choose one of the

Compute Using options.



Alternatively, to define the table calculation by referencing specific dimensions in the view, choose **Specific Dimensions** and then select and order dimensions in the box further down in the Table Calculation dialog box.



For information on these two alternative approaches, see [Choose Your Approach to Table Calculations](#) on page 850. **Compute Using** is the recommended option for most users.

4. When you are satisfied with your table calculation, click the X in the upper-right corner of the Table Calculations dialog box to close it. Highlighting disappears. To indicate that the field now includes a table calculation, Tableau displays a small triangle next to the field name. Hover the cursor over this triangle to see a description of the table calculation:

The screenshot shows a Tableau interface with a calculated field named "SUM(Profit)" selected. A tooltip or context menu is displayed, explaining the calculation: "Using SUM(Profit). Calculates the current value as a difference from the previous value. Results are computed along Segment, State for each Year of Order Date." The background shows a small portion of a data view with columns for Florida, Georgia, and Kentucky.

Table Calculation Types

As you add or modify a table calculation, you must choose a calculation type. Each type has a specific set of options.

Table Calculation Type: Difference From

For each mark in the view, a **Difference From** table calculation computes the difference between the current value and another value in the table.

Suppose you are starting with the following text view, which shows sales totals broken out by year (from left to right) and by quarter and month (from top to bottom):

		Order Date			
Quarter of O..	Month of Or..	2011	2012	2013	2014
Q1	January	\$13,946	\$18,174	\$18,542	\$44,703
	February	\$4,811	\$12,211	\$22,868	\$20,284
	March	\$55,691	\$38,467	\$51,186	\$53,909
Q2	April	\$28,295	\$34,195	\$39,249	\$40,112
	May	\$23,648	\$30,132	\$56,691	\$45,651
	June	\$34,595	\$24,797	\$39,430	\$48,260
Q3	July	\$33,946	\$28,765	\$38,441	\$48,428
	August	\$27,909	\$36,898	\$33,266	\$61,516
	September	\$81,777	\$64,596	\$72,908	\$90,489
Q4	October	\$31,453	\$31,405	\$56,463	\$77,794
	November	\$78,629	\$75,973	\$82,192	\$112,326
	December	\$69,546	\$74,920	\$97,237	\$90,475

Instead of absolute sales values, you want to see by how much values go up or down from month-to-month during the course of each year.

Click for Information on How to Create the Basic View

1. Connect to the **Sample - Superstore** data source.

2. Click and drag the **Order Date** field in the Data pane and drag it to the Columns shelf.
The default date level is YEAR(Order Date).
3. Click and drag **Order Date** again and drop it this time on the Rows shelf.
4. Click on the right side of the field to open the context menu. Then choose **Quarter**.

The screenshot shows the Tableau Data pane with the following structure:

- Columns** shelf: Contains the field **YEAR(Order Date)**.
- Rows** shelf: Contains the field **YEAR(Order Date)**.
- Sheet 5** view: Shows a table with columns "Year of Order" and "Sales". The rows are 2011, 2012, 2013, and 2014.
- Context menu for the **YEAR(Order Date)** field in the Rows shelf:
 - Filter...
 - Show Filter
 - Show Highlighter
 - Sort...
 - Format...
 - Show Header
 - Include in Tooltip**
 - Show Missing Values
 - Year: 2015
 - Quarter: Q2** (highlighted with a blue border and a cursor icon)
 - Month: May
 - Day: 8
 - More
 - Year: 2015
 - Quarter: Q2 2015

You will see two options named **Quarter**. Be sure to choose the first one.

The field should now read **QUARTER(Order Date)**.

Note: If you are creating the view on the web, the menu looks a bit different.

5. Click and drag **Order Date** a third time and drop it on the Rows shelf to the right of **QUARTER(Order Date)**.
6. Click on the right side of the field to open the context menu and this time choose **Month** (again, choose the first of two options named **Month**). The field should now read **MONTH(Order Date)**.
7. Drag **Sales** from the Data pane and drop it on Text on the Marks card.

You now have the basic view, showing Sales by Order Date over a four-year period, by month, quarter, and year.

Modify the basic view to show how by how much values go up or down from month-to-month during the course of each year.

1. Click the **SUM(Sales)** field on the Marks card and then select **Add table calculation**.
2. In the Table Calculation dialog box, verify that **Difference From** is the **Calculation Type**.

This is the default Calculation Type, so you really shouldn't have to change this field value.

3. Choose **Table (Down)** from the **Compute Using** list.

The highlighting shows how this **Compute Using** value sets the scope of the calculation in the view:

		Order Date			
Quarter of O..	Month of Or..	2011	2012	2013	2014
Q1	January	\$50,880	-\$5,963	\$4,325	-\$24,420
	February	-\$9,136			
	March	\$50,880	\$26,256	\$28,319	\$33,625
Q2	April	-\$27,396	-\$4,272	-\$11,938	-\$13,797
	May	-\$4,647	-\$4,064	\$17,442	\$5,539
	June	\$10,947	-\$5,334	-\$17,261	\$2,609
Q3	July	-\$649	\$3,968	-\$990	\$169
	August	-\$6,037	\$8,133	-\$5,175	\$13,088
	September	\$53,868	\$27,698	\$39,643	\$28,973
Q4	October	-\$50,324	-\$33,191	-\$16,445	-\$12,695
	November	\$47,175	\$44,568	\$25,729	\$34,533
	December	-\$9,083	-\$1,053	\$15,045	-\$21,852

Comparing the values in the original text view with the values in this view, you can verify that this is the result that you want.

4. Click the X in the upper-right corner of the Table Calculations dialog box to close it.

Difference from what?

With a **Difference From**, **Percent Difference From**, or **Percent From** calculation, there are always two values to consider: the current value, and the value from which the difference should be calculated. In most cases, you want to calculate the difference between the current value and the previous value, as in the procedure above. But in some cases you may want something different:

Previous	Calculates the difference between the current value and the previous value in the partition. This is the default value.
Next	Calculates the difference between the current value and the next value in the partition.
First	Calculates the difference between the current value and the first value in the partition.
Last	Calculates the difference between the current value and the last value in the partition.

At the level

Use this setting to set a break (that is, restart of the calculation) in the view, based on a particular dimension. How is this different from just using that dimension for partitioning? In fact, it is partitioning, but it's partitioning by position rather than by value, which is how partitioning is defined with the **Compute Using** options.

The **At the level** option is only available when you select **Specific Dimensions** in the Table Calculations dialog box, and when more than one dimension is selected in the field immediately below the **Compute Using** options —that is, when more than one dimension is defined as an addressing field.

This option is not available when you're defining a table calculation with **Compute Using**, because those values establish partitions by position. But with **Specific Dimensions**, because the visual structure and the table calculation are not necessarily aligned, the **At the level** option is available to let you fine-tune your calculation.

For example, if you take the result of the **Difference From** calculation you added above, you can see the effect of **At the level** by doing the following:

1. Click the **SUM(Sales)** field on the Marks card and then select **Edit table calculation**.
2. In the Table Calculation dialog box, choose **Specific Dimensions**.

Notice that two dimensions are now checked in the list box of dimensions: **Quarter of Order Date** and **Month of Order Date**. These are the addressing fields, and because more than one field is being used for addressing, **At the level** is now available.

The choices available from the At the level drop-down list are:

Deepest	Specifies that the calculation should be performed at the level of finest granularity. This is the default option.
Quarter of Order Date	Specifies that the calculation should be performed at the quarter level.
Month of Order Date	Specifies that the calculation should be performed at the month level.

--	--

3. With this particular view, **Deepest** and **Month of Order Date** are equivalent options, and they are both equivalent to just choosing the **Table (Down)** option from **Compute Using**. The only option that yields a different result is **Quarter of Order Date**:

Quarter of O..	Month of Or..	Order Date			
		2011	2012	2013	2014
Q1	January				
	February				
	March				
Q2	April	\$14,349	\$16,021	\$20,706	-\$4,591
	May	\$18,838	\$17,921	\$33,823	\$25,368
	June	-\$21,096	-\$13,670	-\$11,756	-\$5,649
Q3	July	\$5,651	-\$5,430	-\$808	\$8,316
	August	\$4,261	\$6,767	-\$23,426	\$15,865
	September	\$47,182	\$39,799	\$33,478	\$42,229
Q4	October	-\$2,493	\$2,640	\$18,022	\$29,365
	November	\$50,719	\$39,074	\$48,927	\$50,810
	December	-\$12,232	\$10,324	\$24,329	-\$14

The difference is now calculated between the first month in a given quarter and the first month in the previous quarter.

Table Calculation Type: Percent Difference From

For each mark in the view, a **Percent Difference From** table calculation computes the difference between the current value and another value in the table, as a percentage.

Suppose you are starting with the following text view, which shows sales totals broken out by year (from left to right) and by quarter and month (from top to bottom):

Quarter of Order	Month of Order	Order Date			
		2011	2012	2013	2014
Q1	January	\$13,946	\$18,174	\$18,542	\$44,703
	February	\$4,811	\$12,211	\$22,868	\$20,284
	March	\$55,691	\$38,467	\$51,186	\$53,909
Q2	April	\$28,295	\$34,195	\$39,249	\$40,112
	May	\$23,648	\$30,132	\$56,691	\$45,651
	June	\$34,595	\$24,797	\$39,430	\$48,260
Q3	July	\$33,946	\$28,765	\$38,441	\$48,428
	August	\$27,909	\$36,898	\$33,266	\$61,516
	September	\$81,777	\$64,596	\$72,908	\$90,489
Q4	October	\$31,453	\$31,405	\$56,463	\$77,794
	November	\$78,629	\$75,973	\$82,192	\$112,326
	December	\$69,546	\$74,920	\$97,237	\$90,475

Instead of absolute sales values, you want to see how values change, in terms of percentage, from month to month during the course of each year.

Click for Information on How to Create the Basic View

1. Connect to the **Sample - Superstore** data source.
2. Click and drag the **Order Date** field in the Data pane and drag it to the Columns shelf.
The default date level is YEAR(Order Date).
3. Click and drag **Order Date** again and drop it this time on the Rows shelf.
4. Click on the right side of the field to open the context menu. Then choose **Quarter**.

You will see two options named **Quarter**. Be sure to choose the first one.

The field should now read **QUARTER(Order Date)**.

Note: If you are creating the view on the web, the menu looks a bit different.

5. Click and drag **Order Date** a third time and drop it on the Rows shelf to the right of **QUARTER(Order Date)**.
6. Click on the right side of the field to open the context menu and this time choose **Month** (again, choose the first of two options named **Month**). The field should now read **MONTH(Order Date)**.
7. Drag **Sales** from the Data pane and drop it on Text on the Marks card.

You now have the basic view, showing Sales by Order Date over a four-year period, by month, quarter, and year.

Follow these steps to add a **Percent Difference From** table calculation to the basic view.

1. Click the **SUM(Sales)** field on the Marks card and then select **Add table calculation**.
2. In the Table Calculation dialog box, choose **Percent Difference From** as the

Calculation Type.

3. Choose **Table (Down)** from the **Compute Using** list.

The highlighting in the view shows how this Compute Using value sets the scope of the calculation in the view:

Quarter of Or...	Month of Or...	Order Date			
		2011	2012	2013	2014
Q1	January				
	February	-66%	-33%	23%	-55%
	March	1,058%	215%	124%	166%
Q2	April	-49%	-11%	-23%	-26%
	May	-16%	-12%	44%	14%
	June	46%	-18%	-30%	6%
Q3	July	-2%	16%	-3%	0%
	August	-18%	28%	-13%	27%
	September	193%	75%	119%	47%
Q4	October	-62%	-51%	-23%	-14%
	November	150%	142%	46%	44%
	December	-12%	-1%	18%	-19%

Comparing the values in the original text view with the values in this view, you can see that the result is correct. For example, in the original view we see that the January 2010 value is 530,288 and the February 2010 value is 318,309. In the updated view, the value for February 2010 is -39.9%, which is 39.9% less than 530,288.

4. Click the X in the upper-right corner of the Table Calculations dialog box to close it.

You also have the option to click Compute compounded rate if you want the calculation to be compounded.

Difference from what?

With a **Difference From**, **Percent Difference From**, or **Percent From** calculation, there are always two values to consider: the current value, and the value from which the difference should be calculated. In most cases, you want to calculate the difference between the current value and the previous value, as in the procedure above. But in some cases you may want something different:

Previous	Calculates the difference between the current value and the previous value in the partition. This is the default value.
Next	Calculates the difference between the current value and the next value in the partition.

First	Calculates the difference between the current value and the first value in the partition.
Last	Calculates the difference between the current value and the last value in the partition.

At the level

Use this setting to set a break (that is, restart of the calculation) in the view, based on a particular dimension. How is this different from just using that dimension for partitioning? In fact, it is partitioning, but it's partitioning by position rather than by value, which is how partitioning is defined with the **Compute Using** options.

The **At the level** option is only available when you select **Specific Dimensions** in the Table Calculations dialog box, and when more than one dimension is selected in the field immediately below the **Compute Using** options—that is, when more than one dimension is defined as an addressing field.

This option is not available when you're defining a table calculation with **Compute Using**, because those values establish partitions by position. But with **Specific Dimensions**, because the visual structure and the table calculation are not necessarily aligned, the **At the level** option is available to let you fine-tune your calculation.

For an example of how the **At the level** option can affect a table calculation, see [Table Calculation Type: Difference From](#) on page 865.

Table Calculation Type: Percent From

For each mark in the view, a **Percent From** table calculation computes a value as a percentage of some other value—typically, as a percentage of the previous value in the table.

Suppose you are starting with the following text view, which shows sales totals broken out by year (from left to right) and by quarter and month (from top to bottom):

		Order Date			
Quarter of Or..	Month of Or..	2011	2012	2013	2014
Q1	January	\$13,946	\$18,174	\$18,542	\$44,703
	February	\$4,811	\$12,211	\$22,868	\$20,284
	March	\$55,691	\$38,467	\$51,186	\$53,909
Q2	April	\$28,295	\$34,195	\$39,249	\$40,112
	May	\$23,648	\$30,132	\$56,691	\$45,651
	June	\$34,595	\$24,797	\$39,430	\$48,260
Q3	July	\$33,946	\$28,765	\$38,441	\$48,428
	August	\$27,909	\$36,898	\$33,266	\$61,516
	September	\$81,777	\$64,596	\$72,908	\$90,489
Q4	October	\$31,453	\$31,405	\$56,463	\$77,794
	November	\$78,629	\$75,973	\$82,192	\$112,326
	December	\$69,546	\$74,920	\$97,237	\$90,475

Instead of absolute sales values, you want to see how each Year/Quarter/Month value differs from the previous month's value, as a percentage.

Click for Information on How to Create the Basic View

1. Connect to the **Sample - Superstore** data source.
2. Click and drag the **Order Date** field in the Data pane and drag it to the Columns shelf.
The default date level is YEAR(Order Date).
3. Click and drag **Order Date** again and drop it this time on the Rows shelf.
4. Click on the right side of the field to open the context menu. Then choose **Quarter**.

You will see two options named **Quarter**. Be sure to choose the first one.

The field should now read **QUARTER(Order Date)**.

Note: If you are creating the view on the web, the menu looks a bit different.

5. Click and drag **Order Date** a third time and drop it on the Rows shelf to the right of **QUARTER(Order Date)**.
6. Click on the right side of the field to open the context menu and this time choose **Month** (again, choose the first of two options named **Month**). The field should now read **MONTH(Order Date)**.
7. Drag **Sales** from the Data pane and drop it on Text on the Marks card.

You now have the basic view, showing Sales by Order Date over a four-year period, by month, quarter, and year.

Follow these steps to add a **Percent From** table calculation to the basic view.

1. Click the **SUM(Sales)** field on the Marks card and then select **Add table calculation**.
2. In the Table Calculation dialog box, choose **Percent From** as the **Calculation Type**.

- Choose **Table (Down)** from the **Compute Using** list.

The highlighting in the view shows how this **Compute Using** value sets the scope of the calculation in the view:

			Order Date			
Quarter of O..	Month of Or..	2011	2012	2013	2014	
Q1	January					
	February	34%	67%	123%	45%	
	March	1,158%	315%	224%	266%	
Q2	April	51%	89%	77%	74%	
	May	84%	88%	144%	114%	
	June	146%	82%	70%	106%	
Q3	July	98%	116%	97%	100%	
	August	82%	128%	87%	127%	
	September	293%	175%	219%	147%	
Q4	October	38%	49%	77%	86%	
	November	250%	242%	146%	144%	
	December	88%	99%	118%	81%	

Comparing the values in the original text view with the values in this view, you can verify that the result is correct. For example, in the original view we see that the January 2011 value is 13,946 and the February 2011 value is 4,811. In the updated view, the value for February 2010 is 34%; you can verify that 4,811 is about 34% of 13,946.

- Click the X in the upper-right corner of the Table Calculations dialog box to close it.

Difference from what?

With a **Difference From**, **Percent Difference From**, or **Percent From** calculation, there are always two values to consider: the current value, and the value from which the difference should be calculated. In most cases, you want to calculate the difference between the current value and the previous value, as in the procedure above. But in some cases you may want something different:

Previous	Calculates the difference between the current value and the previous value in the partition. This is the default value.
Next	Calculates the difference between the current value and the next value in the partition.
First	Calculates the difference between the current value and the first value in the partition.
Last	Calculates the difference between the current value and the last

value in the partition.

At the level

Use this setting to set a break (that is, restart of the calculation) in the view, based on a particular dimension. How is this different from just using that dimension for partitioning? In fact, it is partitioning, but it's partitioning by position rather than by value, which is how partitioning is defined with the **Compute Using** options.

The **At the level** option is only available when you select **Specific Dimensions** in the Table Calculations dialog box, and when more than one dimension is selected in the field immediately below the **Compute Using** options—that is, when more than one dimension is defined as an addressing field.

This option is not available when you're defining a table calculation with **Compute Using**, because those values establish partitions by position. But with **Specific Dimensions**, because the visual structure and the table calculation are not necessarily aligned, the **At the level** option is available to let you fine-tune your calculation.

For an example of how the **At the level** option can affect a table calculation, see [Table Calculation Type: Difference From on page 865](#).

Table Calculation Type: Percent of Total

For each mark in the view, a **Percent of Total** table calculation computes a value as a percentage of all values in the current partition.

Suppose you are starting with the following text view, which shows sales totals broken out by year (from left to right) and by quarter and month (from top to bottom):

		Order Date			
Quarter of O..	Month of Or..	2011	2012	2013	2014
Q1	January	\$13,946	\$18,174	\$18,542	\$44,703
	February	\$4,811	\$12,211	\$22,868	\$20,284
	March	\$55,691	\$38,467	\$51,186	\$53,909
Q2	April	\$28,295	\$34,195	\$39,249	\$40,112
	May	\$23,648	\$30,132	\$56,691	\$45,651
	June	\$34,595	\$24,797	\$39,430	\$48,260
Q3	July	\$33,946	\$28,765	\$38,441	\$48,428
	August	\$27,909	\$36,898	\$33,266	\$61,516
	September	\$81,777	\$64,596	\$72,908	\$90,489
Q4	October	\$31,453	\$31,405	\$56,463	\$77,794
	November	\$78,629	\$75,973	\$82,192	\$112,326
	December	\$69,546	\$74,920	\$97,237	\$90,475

Instead of absolute sales values, you want to see the percentage that each month contributes to each year's total sales. So, for example, you want to see what percent of 2010 sales were realized in January of 2010, what percent in February, and so on.

Click for Information on How to Create the Basic View

1. Connect to the **Sample - Superstore** data source.
2. Click and drag the **Order Date** field in the Data pane and drag it to the Columns shelf.
The default date level is **YEAR(Order Date)**.
3. Click and drag **Order Date** again and drop it this time on the Rows shelf.
4. Click on the right side of the field to open the context menu. Then choose **Quarter**.

The screenshot shows the Tableau Data pane with the 'Sheet 5' tab selected. In the 'Columns' shelf, there is a field named 'YEAR(Order Date)'. In the 'Rows' shelf, there is a field named 'YEAR(Order Dat..)' with a dropdown arrow. A context menu is open next to this field, listing options: Filter..., Show Filter, Show Highlighter, Sort..., Format..., Show Header, Include in Tooltip, Show Missing Values, and a section for date hierarchy. This section contains 'Year' (2015), 'Quarter' (Q2), 'Month' (May), 'Day' (8), and 'More'. Below this, another 'Year' row (2015) and a 'Quarter' row ('Q2 2015') are shown. The 'Quarter' option in the hierarchy is highlighted with a blue selection bar and has a cursor pointing at it.

You will see two options named **Quarter**. Be sure to choose the first one.

The field should now read **QUARTER(Order Date)**.

Note: If you are creating the view on the web, the menu looks a bit different.

5. Click and drag **Order Date** a third time and drop it on the Rows shelf to the right of

QUARTER(Order Date).

6. Click on the right side of the field to open the context menu and this time choose **Month** (again, choose the first of two options named **Month**). The field should now read **MONTH(Order Date)**.
7. Drag **Sales** from the Data pane and drop it on Text on the Marks card.

You now have the basic view, showing Sales by Order Date over a four-year period, by month, quarter, and year.

Follow these steps to add a **Percent of Total** table calculation to the basic view.

1. Click the **SUM(Sales)** field on the Marks card and then select **Add table calculation**.
2. In the Table Calculation dialog box, choose **Percent of Total** as the **Calculation Type**.
3. Choose **Table (Down)** from the **Compute Using** list. The highlighting in the view shows how this **Compute Using** value sets the scope of the calculation in the view:

		Order Date			
Quarter of O..	Month of Or..	2011	2012	2013	2014
Q1	January	2.88%	3.86%	3.05%	6.09%
	February	0.99%	2.60%	3.76%	2.76%
	March	11.50%	8.18%	8.41%	7.35%
Q2	April	5.84%	7.27%	6.45%	5.47%
	May	4.88%	6.40%	9.32%	6.22%
	June	7.14%	5.27%	6.48%	6.58%
Q3	July	7.01%	6.11%	6.32%	6.60%
	August	5.76%	7.84%	5.47%	8.38%
	September	16.89%	13.73%	11.98%	12.33%
Q4	October	6.50%	6.67%	9.28%	10.60%
	November	16.24%	16.15%	13.51%	15.30%
	December	14.36%	15.92%	15.98%	12.33%

Comparing the values in the original text view with the values in this view verifies that the result is correct. For example, the 12 values in the view for 2011 add up to 100%.

4. Click the X in the upper-right corner of the Table Calculations dialog box to close it.

Choose **Compute across all pages** if you want the calculation to take account of any dimension on the Pages shelf.

At the level

Use this setting to set a break (that is, restart of the calculation) in the view, based on a particular dimension. How is this different from just using that dimension for partitioning? In fact,

it is partitioning, but it's partitioning by position rather than by value, which is how partitioning is defined with the **Compute Using** options.

The **At the level** option is only available when you select **Specific Dimensions** in the Table Calculations dialog box, and when more than one dimension is selected in the field immediately below the **Compute Using** options —that is, when more than one dimension is defined as an addressing field.

This option is not available when you're defining a table calculation with **Compute Using**, because those values establish partitions by position. But with **Specific Dimensions**, because the visual structure and the table calculation are not necessarily aligned, the **At the level** option is available to let you fine-tune your calculation.

Table Calculation Type: Rank

For each mark in the view, a **Rank** table calculation computes a ranking for each value in a partition.

Suppose you are starting with the following text view, which shows sales totals broken out by year (from left to right) and by quarter and month (from top to bottom):

		Order Date			
Quarter of Or..	Month of Or..	2011	2012	2013	2014
Q1	January	\$13,946	\$18,174	\$18,542	\$44,703
	February	\$4,811	\$12,211	\$22,868	\$20,284
	March	\$55,691	\$38,467	\$51,186	\$53,909
Q2	April	\$28,295	\$34,195	\$39,249	\$40,112
	May	\$23,648	\$30,132	\$56,691	\$45,651
	June	\$34,595	\$24,797	\$39,430	\$48,260
Q3	July	\$33,946	\$28,765	\$38,441	\$48,428
	August	\$27,909	\$36,898	\$33,266	\$61,516
	September	\$81,777	\$64,596	\$72,908	\$90,489
Q4	October	\$31,453	\$31,405	\$56,463	\$77,794
	November	\$78,629	\$75,973	\$82,192	\$112,326
	December	\$69,546	\$74,920	\$97,237	\$90,475

Instead of absolute sales values, you want to see how each month within a given year ranks in total sales, from most sales (ranked 1) to least sales (ranked 12).

Click for Information on How to Create the Basic View

1. Connect to the **Sample - Superstore** data source.
2. Click and drag the **Order Date** field in the Data pane and drag it to the Columns shelf.

The default date level is YEAR(Order Date).

3. Click and drag **Order Date** again and drop it this time on the Rows shelf.
4. Click on the right side of the field to open the context menu. Then choose **Quarter**.

The screenshot shows the Tableau interface with a context menu open over a 'Year' field on the Rows shelf. The menu options include:

- Filter...
- Show Filter
- Show Highlighter
- Sort...**
- Format...
- Show Header
- Include in Tooltip
- Show Missing Values
- Year** 2015
- Quarter** Q2 May
- Month
- Day
- More ▾
- Year** 2015
- Quarter** Q2 2015

The 'Quarter' option under 'Year' is highlighted with a blue selection bar and has a cursor icon pointing at it.

You will see two options named **Quarter**. Be sure to choose the first one.

The field should now read **QUARTER(Order Date)**.

Note: If you are creating the view on the web, the menu looks a bit different.

5. Click and drag **Order Date** a third time and drop it on the Rows shelf to the right of **QUARTER(Order Date)**.
6. Click on the right side of the field to open the context menu and this time choose **Month** (again, choose the first of two options named **Month**). The field should now read **MONTH(Order Date)**.
7. Drag **Sales** from the Data pane and drop it on Text on the Marks card.

You now have the basic view, showing Sales by Order Date over a four-year period, by month, quarter, and year.

Follow these steps to add a **Rank** table calculation to the basic view.

1. Click the **SUM(Sales)** field on the Marks card and then select **Add table calculation**.
2. In the Table Calculation dialog box, choose **Rank** as the **Calculation Type**.
3. Choose **Table (Down)** from the **Compute Using** list.

The highlighting in the view shows how this **Compute Using** value sets the scope of the calculation in the view:

		Order Date			
Quarter of O..	Month of Or..	2011	2012	2013	2014
Q1	January	11	11	12	10
	February	12	12	11	12
	March	4	4	6	6
Q2	April	8	6	8	11
	May	10	8	4	9
	June	5	10	7	8
Q3	July	6	9	9	7
	August	9	5	10	5
	September	1	3	3	2
Q4	October	7	7	5	4
	November	2	1	2	1
	December	3	2	1	3

Comparing the values in the original text view with the values in this view shows that the result is correct. The values run from 1 to 12, and the highest sales total in 2011, for September (81,777), is ranked 1 in the updated view, while the second highest value, for November (78,629) is ranked 2 in the updated view.

4. Click the X in the upper-right corner of the Table Calculations dialog box to close it.

Descending vs. Ascending

Ascending order ranks values from least to most. **Descending** order ranks values from most to least. For Rank table calculation, the default value is **Descending**.

Rank Type

One issue with **Rank** calculations is that there may be more than one mark with the same value. What would happen, for example, if Tables in the Central region and Appliances in the South region both had sales of exactly \$36,729? Tableau lets you specify how to handle such cases by including an additional field in the Table Calculation dialog box when you set **Calculation Type to Rank**.

The choices are listed below. The number sequence at the beginning of each option show how each option would rank a hypothetical set of four values where two of the values are identical:

Option	Result
Competition (1, 2, 2, 4)	Identical values are assigned an identical rank. The highest value is ranked 1 and then the next two, identical values, are both ranked 2. The next value is then ranked 4.
Modified Competition (1, 3, 3, 4)	Identical values are assigned an identical rank. The highest value is ranked 1 and then the next two, identical values, are both ranked 3. The next value is then ranked 4.
Dense (1, 2, 2, 3)	Duplicate values are all given the same rank, which is the next number in the ranking sequence. The next value after the duplicate values is computed as though the duplicate values were a single value.
Unique (1, 2, 3, 4)	Duplicate values are given unique rankings, according to the direction in which the ranking is being computed.

Table Calculation Type: Percentile

For each mark in the view, a **Percentile** table calculation computes a percentile rank for each value in a partition.

Suppose you are starting with the following text view, which shows sales totals broken out by year (from left to right) and by quarter and month (from top to bottom):

		Order Date			
Quarter of O..	Month of Or..	2011	2012	2013	2014
Q1	January	\$13,946	\$18,174	\$18,542	\$44,703
	February	\$4,811	\$12,211	\$22,868	\$20,284
	March	\$55,691	\$38,467	\$51,186	\$53,909
Q2	April	\$28,295	\$34,195	\$39,249	\$40,112
	May	\$23,648	\$30,132	\$56,691	\$45,651
	June	\$34,595	\$24,797	\$39,430	\$48,260
Q3	July	\$33,946	\$28,765	\$38,441	\$48,428
	August	\$27,909	\$36,898	\$33,266	\$61,516
	September	\$81,777	\$64,596	\$72,908	\$90,489
Q4	October	\$31,453	\$31,405	\$56,463	\$77,794
	November	\$78,629	\$75,973	\$82,192	\$112,326
	December	\$69,546	\$74,920	\$97,237	\$90,475

Instead of absolute sales values, you want to see the values for each month in a given year expressed as a percentile. With this kind of table calculation, the lowest ranked value is assigned the 0 percentile and the highest ranked value is assigned the 100 percentile (assuming the calculation is working in ascending order, which is the default). All other values are assigned a percentile based on their ranking. So for a set of four values, 1, 3, 22, 67, the percentiles would be 0%, 33%, 67%, and 100%, respectively.

Click for Information on How to Create the Basic View

1. Connect to the **Sample - Superstore** data source.
2. Click and drag the **Order Date** field in the Data pane and drag it to the Columns shelf.
The default date level is **YEAR(Order Date)**.
3. Click and drag **Order Date** again and drop it this time on the Rows shelf.
4. Click on the right side of the field to open the context menu. Then choose **Quarter**.

The screenshot shows the Tableau Data pane with the following structure:

- Columns** shelf: Contains a field labeled **YEAR(Order Date)**.
- Rows** shelf: Contains a field labeled **YEAR(Order Date)**.
- Sheet 5** view: Shows a list of years from 2011 to 2014.
- A context menu is open on the **YEAR(Order Date)** field in the Rows shelf. The menu items are:
 - Filter...
 - Show Filter
 - Show Highlighter
 - Sort...
 - Format...
 - Show Header
 - Include in Tooltip
 - Show Missing Values
 - Year: 2015
 - Quarter: Q2 (highlighted with a blue border)
 - Month: May
 - Day: 8
 - More: ▾
 - Year: 2015
 - Quarter: Q2 2015

You will see two options named **Quarter**. Be sure to choose the first one.

The field should now read **QUARTER(Order Date)**.

Note: If you are creating the view on the web, the menu looks a bit different.

5. Click and drag **Order Date** a third time and drop it on the Rows shelf to the right of **QUARTER(Order Date)**.
6. Click on the right side of the field to open the context menu and this time choose **Month** (again, choose the first of two options named **Month**). The field should now read **MONTH(Order Date)**.
7. Drag **Sales** from the Data pane and drop it on Text on the Marks card.

You now have the basic view, showing Sales by Order Date over a four-year period, by month, quarter, and year.

Follow these steps to add a **Percentile** table calculation to the basic view.

1. Click the **SUM(Sales)** field on the Marks card and then select **Add table calculation**.
2. In the Table Calculation dialog box, choose **Percentile** as the **Calculation Type**.
3. Choose **Table (Down)** from the **Compute Using** list.

The highlighting in the view shows how this Compute Using value sets the scope of the calculation in the view:

		Order Date			
Quarter of O..	Month of Or..	2011	2012	2013	2014
Q1	January	9.1%	9.1%	0.0%	18.2%
	February	0.0%	0.0%	9.1%	0.0%
	March	72.7%	72.7%	54.5%	54.5%
Q2	April	36.4%	54.5%	36.4%	9.1%
	May	18.2%	36.4%	72.7%	27.3%
	June	63.6%	18.2%	45.5%	36.4%
Q3	July	54.5%	27.3%	27.3%	45.5%
	August	27.3%	63.6%	18.2%	63.6%
	September	100.0%	81.8%	81.8%	90.9%
Q4	October	45.5%	45.5%	63.6%	72.7%
	November	90.9%	100.0%	90.9%	100.0%
	December	81.8%	90.9%	100.0%	81.8%

Comparing the values in the original text view with the values in this view verifies that the result is correct. The lowest value in the original text table was February (4,711); in the new text view, February is 0.0%; the next lowest value in the original table was January (13,946); in the new view, January is 9.1%.

Descending vs. Ascending

Ascending order ranks values from least to most. **Descending** order ranks values from most to least. For a Percentile table calculation, the default value is **Ascending**.

Table Calculation Type: Running Total

For each mark in the view, a **Running Total** table calculation aggregates values cumulatively in a partition. It can do this by summing values, averaging values, or replacing all values with either the lowest or highest actual value.

Suppose you are starting with the following text view, which shows sales totals broken out by year (from left to right) and by quarter and month (from top to bottom):

		Order Date			
Quarter of Or..	Month of Or..	2011	2012	2013	2014
Q1	January	\$13,946	\$18,174	\$18,542	\$44,703
	February	\$4,811	\$12,211	\$22,868	\$20,284
	March	\$55,691	\$38,467	\$51,186	\$53,909
Q2	April	\$28,295	\$34,195	\$39,249	\$40,112
	May	\$23,648	\$30,132	\$56,691	\$45,651
	June	\$34,595	\$24,797	\$39,430	\$48,260
Q3	July	\$33,946	\$28,765	\$38,441	\$48,428
	August	\$27,909	\$36,898	\$33,266	\$61,516
	September	\$81,777	\$64,596	\$72,908	\$90,489
Q4	October	\$31,453	\$31,405	\$56,463	\$77,794
	November	\$78,629	\$75,973	\$82,192	\$112,326
	December	\$69,546	\$74,920	\$97,237	\$90,475

Instead of absolute sales values, you want to see a running total of sales for each year, such that each month's sales are added to all previous months' sales.

Click for Information on How to Create the Basic View

1. Connect to the **Sample - Superstore** data source.
2. Click and drag the **Order Date** field in the Data pane and drag it to the Columns shelf.
The default date level is YEAR(Order Date).
3. Click and drag **Order Date** again and drop it this time on the Rows shelf.
4. Click on the right side of the field to open the context menu. Then choose **Quarter**.

You will see two options named **Quarter**. Be sure to choose the first one.

The field should now read **QUARTER(Order Date)**.

Note: If you are creating the view on the web, the menu looks a bit different.

5. Click and drag **Order Date** a third time and drop it on the Rows shelf to the right of **QUARTER(Order Date)**.
6. Click on the right side of the field to open the context menu and this time choose **Month** (again, choose the first of two options named **Month**). The field should now read **MONTH(Order Date)**.
7. Drag **Sales** from the Data pane and drop it on Text on the Marks card.

You now have the basic view, showing Sales by Order Date over a four-year period, by month, quarter, and year.

Follow these steps to add a **Running Total** table calculation to the basic view.

1. Click the **SUM(Sales)** field on the Marks card and choose **Add table calculation**.
2. In the Table Calculation dialog box, choose **Running Total** as the **Calculation Type**.

- Choose **Table (Down)** from the **Compute Using** list.

The highlighting in the view shows how this **Compute Using** value sets the scope of the calculation in the view:

		Order Date			
Quarter of O..	Month of Or..	2011	2012	2013	2014
Q1	January	\$13,946	\$18,174	\$18,542	\$44,703
	February	\$18,757	\$30,385	\$41,410	\$64,987
	March	\$74,448	\$68,852	\$92,596	\$118,896
Q2	April	\$102,743	\$103,047	\$131,845	\$159,008
	May	\$126,391	\$133,179	\$188,536	\$204,659
	June	\$160,987	\$157,976	\$227,967	\$252,919
Q3	July	\$194,933	\$186,741	\$266,407	\$301,347
	August	\$222,842	\$223,640	\$299,673	\$362,863
	September	\$304,620	\$288,236	\$372,581	\$453,352
Q4	October	\$336,073	\$319,640	\$429,044	\$531,146
	November	\$414,702	\$395,613	\$511,236	\$643,472
	December	\$484,247	\$470,533	\$608,474	\$733,947

Comparing the values in the original text view with the values in this view shows that the result is correct. The monthly values ascend steadily and the December value (484,247) is the same value you see if you show column grand totals (from the Analysis menu, select **Totals > Show column grand totals**).

- Click the X in the upper-right corner of the Table Calculations dialog box to close it.

The Running Total Doesn't Have to Be a Sum

For a **Running Total** table calculation, Tableau can update values cumulatively in other ways than summing. Choose one of the options from the drop-down list just below the **Calculation Type** field:

Option	Meaning
Sum	Each value is added to the previous value.
Average	The running total averages the current and all previous values.
Minimum	All values are replaced with the lowest value in the original partition.
Maximum	All values are replaced with the highest value in the original partition.

Restarting Every

Use this setting to set a break (that is, restart of the calculation) in the view, based on a particular dimension.

Restarting every can be useful in the following situations:

- With dates or other hierarchies, if you restart every month, as you bring in Year or Quarter, Tableau knows to partition automatically.
- With non-hierarchies, **Restarting every** affects the sorting. If you want to address on **Products** and partition by **State**, but you want the products sorted by **SUM(Sales)** within each state, you need to include **States** as an addressing field under Specific Dimensions, but then restart every state. Otherwise, the sort by **SUM(Sales)** would be based on each product's sum of sales across all states.

The **Restarting every** option is only available when you select **Specific Dimensions** in the Table Calculations dialog box and when more than one dimension is selected in the field immediately below the **Compute Using** options—that is, when more than one dimension is defined as an addressing field.

This option is not available when you're defining a table calculation with **Compute Using**.

For example, if you take the result of the **Running Total** calculation you added above, you can see the effect of **Restarting every** by doing the following:

- Click the **SUM(Sales)** field on the Marks card and choose **Edit table calculation**.
- In the Table Calculation dialog box, choose **Specific Dimensions**.

Notice that two dimensions are now checked in the list box of dimensions: **Quarter of Order Date** and **Month of Order Date**. These are the addressing fields, and because more than one field is being used for addressing, **Restarting every** is now available.

The choices available from the At the level drop-down list are:

None	Specifies that the calculation should be performed at the level of greatest granularity. This is the default option. This option does not change the view.
Quarter of Order Date	Specifies that the calculation should be performed at the quarter level.

- If you choose Quarter of Order Date, the view updates to show the effect of this change:

		Order Date			
Quarter of Or..	Month of Or..	2011	2012	2013	2014
Q1	January	\$13,946	\$18,174	\$18,542	\$44,703
	February	\$18,757	\$30,385	\$41,410	\$64,987
	March	\$74,448	\$68,852	\$92,596	\$118,896
Q2	April	\$28,295	\$34,195	\$39,249	\$40,112
	May	\$51,944	\$64,327	\$95,940	\$85,764
	June	\$86,539	\$89,124	\$135,370	\$134,023
Q3	July	\$33,946	\$28,765	\$38,441	\$48,428
	August	\$61,856	\$65,664	\$71,706	\$109,944
	September	\$143,633	\$130,260	\$144,614	\$200,433
Q4	October	\$31,453	\$31,405	\$56,463	\$77,794
	November	\$110,082	\$107,377	\$138,655	\$190,120
	December	\$179,628	\$182,297	\$235,893	\$280,595

The calculation now restarts after every quarter. If you click out of the Table Calculations dialog box (to dismiss the highlighting) you can see this more clearly.

- Click the X in the upper-right corner of the Table Calculations dialog box to close it.

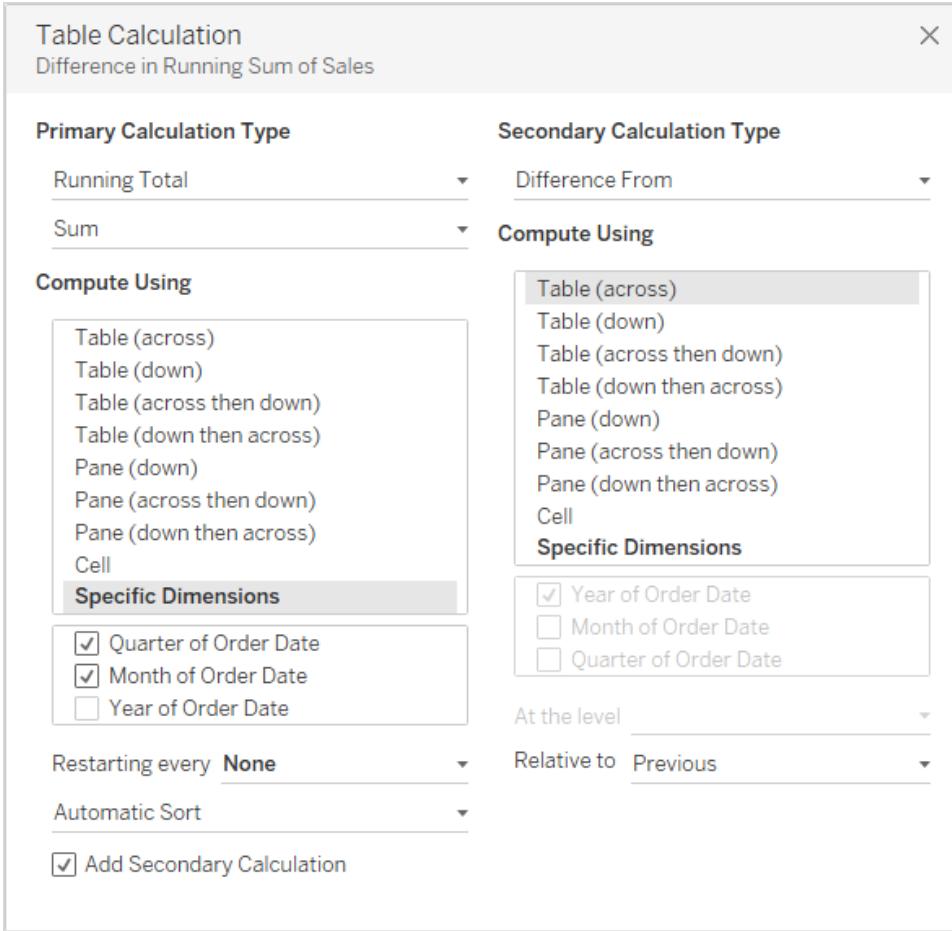
Add Secondary Calculation

With **Running Total** and **Moving Calculation** table calculations, you have the option to transform values twice to obtain the result you want—that is, to add a secondary table calculation on top of the primary table calculation. For example, you could add an initial table calculation to calculate the running total for sales per month within each individual year, and then a secondary calculation to calculate the year-over-year percent difference for each month from one year to the next.

To do this, first add the primary table calculation, as shown above. Then continue as follows:

- Click the **SUM(Sales)** field on the Marks card and select **Edit table calculation**.
- In the Table Calculation dialog box, click **Add Secondary Calculation**.

The Table Calculation dialog box expands to show a second panel:



3. In the second panel, choose **Percent Difference From** as the **Secondary Calculation Type**.
4. You do not need to change the **Compute Using** selection: **Table (Across)** is the right option.
5. Click the X in the upper-right corner to dismiss the Table Calculation dialog box.

Now your view shows what you needed: a year-over-year percent difference of a running total:

		Order Date			
Quarter of O..	Month of Or..	2011	2012	2013	2014
Q1	January		\$4,228	\$368	\$26,161
	February		\$11,628	\$11,025	\$23,576
	March		-\$5,596	\$23,745	\$26,299
Q2	April		\$304	\$28,798	\$27,163
	May		\$6,787	\$55,357	\$16,123
	June		-\$3,011	\$69,991	\$24,952
Q3	July		-\$8,192	\$79,666	\$34,940
	August		\$797	\$76,033	\$63,191
	September		-\$16,384	\$84,345	\$80,771
Q4	October		-\$16,433	\$109,404	\$102,102
	November		-\$19,089	\$115,623	\$132,236
	December		-\$13,715	\$137,941	\$125,473

Table Calculation Type: Moving Calculation

For each mark in the view, a **Moving Calculation** table calculation determines the value for a mark in the view by performing an aggregation (sum, average, minimum, or maximum) across a specified number of values before and/or after the current value.

A moving calculation is typically used to smooth short-term fluctuations in your data so that you can see long-term trends. For example, with securities data there are so many fluctuations every day that it is hard to see the big picture through all the ups and downs. You can use a moving calculation to define a range of values to summarize using an aggregation of your choice.

Suppose you are starting with the following text view, which shows sales totals broken out by year (from left to right) and by quarter and month (from top to bottom):

		Order Date			
Quarter of Or..	Month of Or..	2011	2012	2013	2014
Q1	January	\$13,946	\$18,174	\$18,542	\$44,703
	February	\$4,811	\$12,211	\$22,868	\$20,284
	March	\$55,691	\$38,467	\$51,186	\$53,909
Q2	April	\$28,295	\$34,195	\$39,249	\$40,112
	May	\$23,648	\$30,132	\$56,691	\$45,651
	June	\$34,595	\$24,797	\$39,430	\$48,260
Q3	July	\$33,946	\$28,765	\$38,441	\$48,428
	August	\$27,909	\$36,898	\$33,266	\$61,516
	September	\$81,777	\$64,596	\$72,908	\$90,489
Q4	October	\$31,453	\$31,405	\$56,463	\$77,794
	November	\$78,629	\$75,973	\$82,192	\$112,326
	December	\$69,546	\$74,920	\$97,237	\$90,475

Instead of absolute sales values, you want to find out how sales totals are trending over time. To do this, you transform each monthly total so that it averages the monthly total for the two previous months and the current month. So instead of seeing just raw sales totals for October, 2010 (for example), you want to replace that number with a number that averages sales for August, September, and October.

Click for Information on How to Create the Basic View

1. Connect to the **Sample - Superstore** data source.
2. Click and drag the **Order Date** field in the Data pane and drag it to the Columns shelf.
The default date level is YEAR(Order Date).
3. Click and drag **Order Date** again and drop it this time on the Rows shelf.
4. Click on the right side of the field to open the context menu. Then choose **Quarter**.

The screenshot shows the Tableau Data pane with a context menu open over the 'Year' field. The menu includes options like Filter..., Show Filter, Show Highlighter, Sort..., Format..., Show Header, and Include in Tooltip. Below these, there's a section for 'Show Missing Values'. The main part of the menu shows hierarchical levels: Year (2015), Quarter (Q2), Month (May), Day (8), and More. At the bottom, it shows 'Year' again (2015) and 'Quarter' (Q2 2015).

You will see two options named **Quarter**. Be sure to choose the first one.

The field should now read **QUARTER(Order Date)**.

Note: If you are creating the view on the web, the menu looks a bit different.

5. Click and drag **Order Date** a third time and drop it on the Rows shelf to the right of **QUARTER(Order Date)**.
6. Click on the right side of the field to open the context menu and this time choose **Month** (again, choose the first of two options named **Month**). The field should now read **MONTH(Order Date)**.
7. Drag **Sales** from the Data pane and drop it on Text on the Marks card.

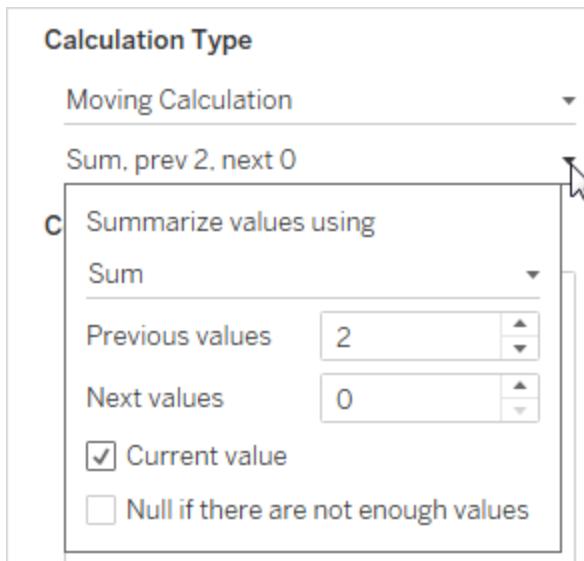
You now have the basic view, showing Sales by Order Date over a four-year period, by month, quarter, and year.

Follow these steps to add a **Moving Calculation** table calculation to the basic view.

1. Click the **SUM(Sales)** field on the Marks card and then click **Add table calculation**.
2. In the Table Calculation dialog box, choose **Moving Calculation** as the **Calculation**

Type.

3. Click in the drop-down list below the **Calculation Type** option to display the options that you use to configure the calculation:



4. Choose from the drop-down list of available aggregations: **Sum**, **Average**, **Minimum**, or **Maximum**. In this case, you want **Average**.
5. Using the **Previous values** option, specify how many marks preceding the current mark should be included in calculating the average. In this case, you want 2.
6. Using the **Next values** option, specify how many marks following the current mark should be included in calculating the average. In this case, you want 0.
7. Click **Current value** to indicate that you want the current value to be included in the calculation.
8. Do not click **Null if there are not enough values**. You can select this option if you want the current value to be null if there are not enough previous or next values to perform the calculation as specified. When you do not click this option, Tableau will use as many of the values within the specified range as are available.

When you close the drop-down list, Tableau shows a summary of the options you have chosen:

Calculation Type

Moving Calculation

Sum, prev 2, next 0

- Choose **Table (Down then Across)** from the **Compute Using** list.

This will cause the calculation to run through the entire time range—from January 2010 to December 2013.

Your view is now “smoothed” to show how sales numbers are trending over time:

		Order Date			
Quarter of Or..	Month of Or..	2011	2012	2013	2014
Q1	January	\$13,946	\$166,348	\$169,435	\$224,133
	February	\$18,757	\$99,931	\$116,330	\$162,224
	March	\$74,448	\$68,852	\$92,596	\$118,896
Q2	April	\$88,797	\$84,873	\$113,303	\$114,305
	May	\$107,635	\$102,794	\$147,126	\$139,673
	June	\$86,539	\$89,124	\$135,370	\$134,023
Q3	July	\$92,190	\$83,694	\$134,562	\$142,339
	August	\$96,451	\$90,461	\$111,137	\$158,204
	September	\$143,633	\$130,260	\$144,614	\$200,433
Q4	October	\$141,140	\$132,899	\$162,637	\$229,799
	November	\$191,859	\$171,973	\$211,564	\$280,609
	December	\$179,628	\$182,297	\$235,893	\$280,595

- Click the X in the upper-right corner of the Table Calculations dialog box to close it.

Add Secondary Calculation

With **Running Total** and **Moving Calculation** table calculations, you have the option to transform values twice to obtain the result you want—that is, to add a secondary table calculation on top of the primary table calculation. For example, you could add an initial table calculation to calculate the running total for sales per month within each individual year, and then a secondary calculation to calculate the year-over-year percent difference for each month from one year to the next.

For an example showing how to create a secondary calculation, see **Table Calculation Type: Running Total** on page 886.

Customize Table Calculations

You can always customize a table calculation by editing it in the Table Calculations dialog box, but there are other, more specialized ways to customize a table calculation.

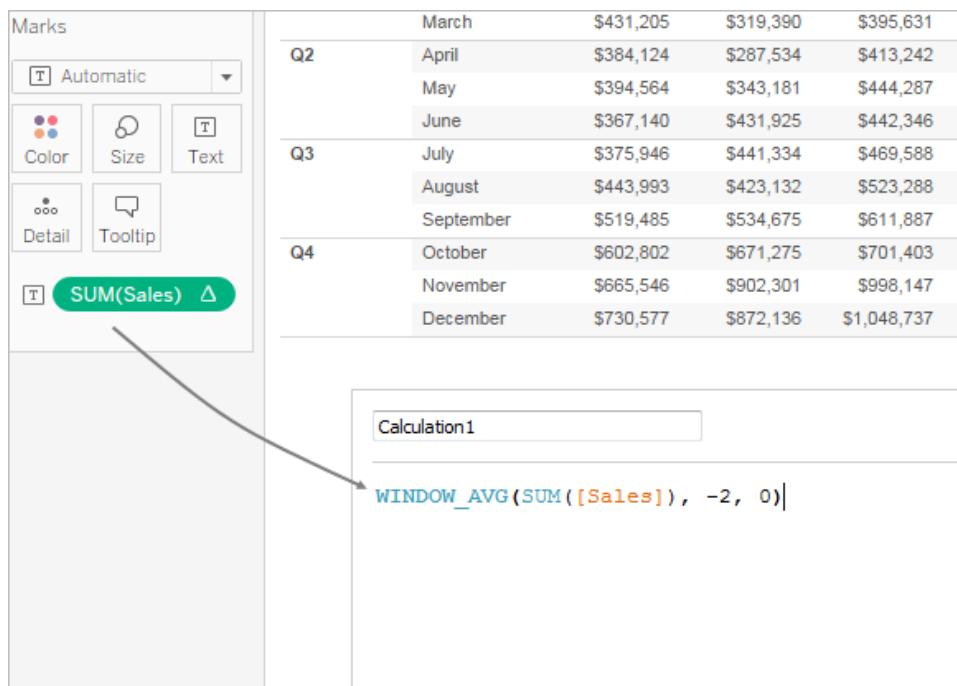
Customizing a Table Calculation Using Its Context Menu

Click any field in the view to see a context menu listing ways to customize the field. For a field in the view that has a table calculation, you can change the **Compute Using** option—that is, the option that determines the direction and scope of the calculation relative to the visual structure of the view. Do this by clicking the field and then choosing an option from the **Compute Using** list.

For **Difference From**, **Percent Difference From**, and **Percent From** table calculations, you can also specify a different field from which the difference should be computed. Do this by clicking the field and then choosing an option from the **Relative to** list. The options are **Previous**, **Next**, **First**, and **Last**.

Customizing a Table Calculation Using the Calculation Editor

You can customize a table calculation by dragging it into the calculation editor:



When you edit a table calculation in the calculation editor, you can click **Default Table Calculation** in the lower-right corner of the editor to open the calculation in the Table Calculation dialog box. This will allow you to create a new named calculated field that uses the same table calculation as the one you are basing the calculation on.

Nested Table Calculations

A nested table calculation can be one of two types of calculated fields:

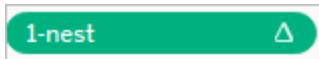
- A calculated field that includes more than one calculated field with a table calculation (as in the example below), or
- A calculated field that itself has a table calculation and includes at least one calculated field with a table calculation.

With nested table calculations, you can set **Compute Using** configurations for individual calculations independently.

Here is a scenario you can try, using the Sample - Superstore data source that is included with Tableau Desktop, which results in a nested table calculation.

1. Drag **Sub-Category** to Columns and **Region** to Rows.
2. Create a calculated field, *1-nest*, with the definition `TOTAL(SUM([Sales]))`.

`TOTAL` is a table calculation function, so this calculated field automatically has a table calculation—when you use it in the view, the field will have the tell-tale table triangle, indicating a table calculation:

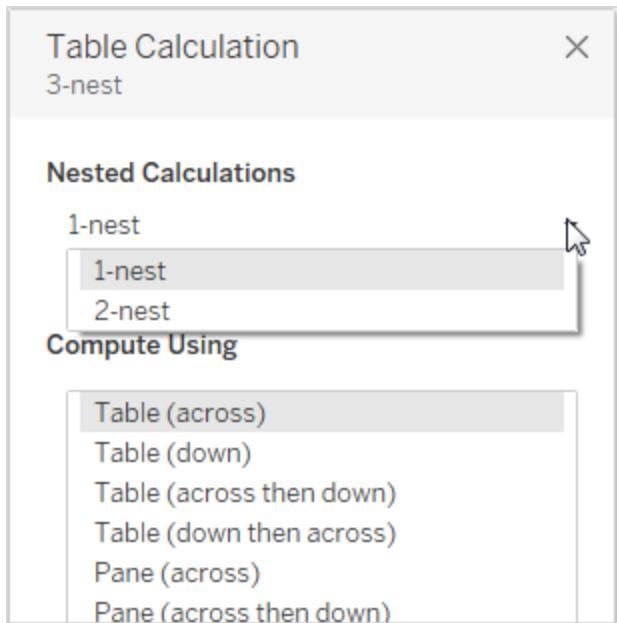


For information on table calculation functions, see [Table Calculation Functions on page 1587](#).

3. Create a second calculated field, *2-nest*, with the definition `TOTAL(SUM([Profit]))`.
4. Create a third calculated field, *3-nest*, with the definition `[1-nest] + [2-nest]`.
5. Drag *3-nest* and drop it to the right of **Sub-Category** on Columns.
6. Click *3-nest* on Columns and choose **Edit Table Calculation**.

In the Table Calculations dialog box, you can now separately configure the underlying

table calculations:



Level of Detail Expressions

Level of detail expressions support aggregation at dimensionalities other than the view level. With level of detail expressions, you can attach one or more dimensions to any aggregate expression.

Unlike table calculations, totals, or reference lines, level of detail expressions are computed in the data source. On the plus side, this lets you avoid the overhead of bringing all the data from the database to your computer over the network. With large data sources, this can be a huge performance gain. On the minus side, this can cause Tableau to run more complicated queries (for example, containing many joins), and if the underlying data source is slow, performance can suffer.

Overview: Level of Detail Expressions

Level of detail expressions (which are sometimes also referred to as "LOD Expressions" or "LOD Calculations") are useful for a variety of use cases, including:

- Cohort analysis – comparing data for different subgroups
- Totals or averages across segments

- Aggregates of aggregates
- Binning aggregates

Background: Row Level Expressions and View Level Expressions

In Tableau, expressions referencing **unaggregated** datasource columns are computed for each row in the underlying table. In this case, the dimensionality of the expression is *row level*. An example of a row-level expression is:

`[Sales] / [Profit]`

This calculation will be evaluated in each row of the database. For each row, the Sales value in that row will be divided by the Profit value in that row, producing a new column with the result of the multiplication (a profit ratio).

If you create a calculation with this definition, save it with the name **[ProfitRatio]**, and then drag it from the **Data** pane to a shelf, Tableau typically aggregates the calculated field for the view:

`SUM[ProfitRatio]`

By contrast, expressions referencing **aggregated** data source columns are computed at the dimensionality defined by the dimensions in the view. In this case, the dimensionality of the expression is *view level*. An example of a view-level expression is:

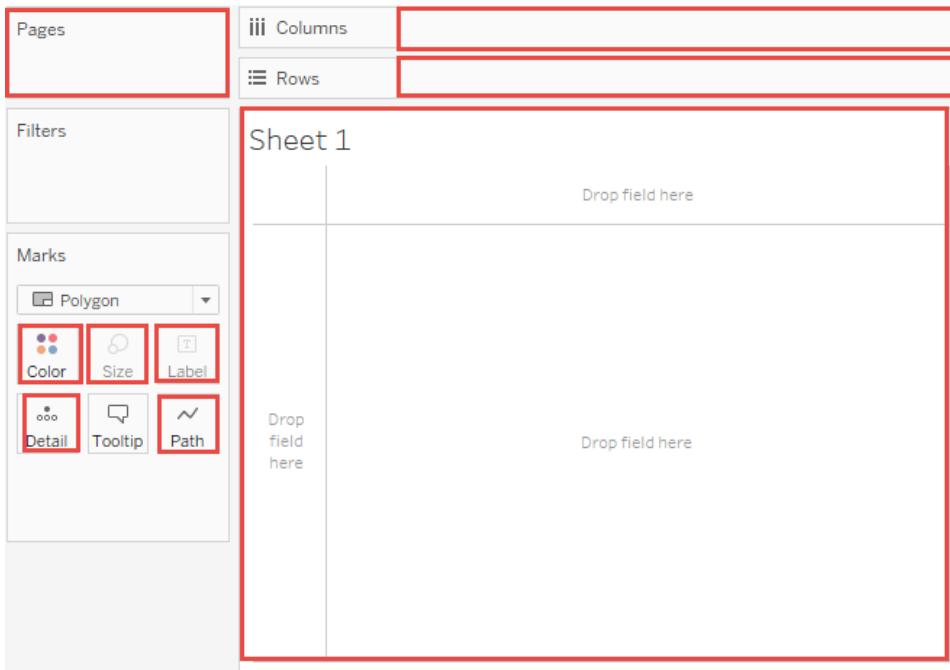
`SUM(Sales) / SUM(Profit)`

If you drag this calculation to a shelf (or type it directly on a shelf as an ad-hoc calculation), Tableau encloses it in an AGG function:

AGG(SUM(Sales) / SUM(Profit))

This is what is known as an aggregate calculation. For details, see [Aggregate Calculations on page 838](#).

Dimension and set fields placed on any of the locations highlighted in the following image contribute to the view level of detail:



Before level of detail expressions were supported in Tableau, it was not possible to create calculations at a level of detail other than the view level. For example, if you attempt to save the following expression, Tableau displays the error message: “Cannot mix aggregate and non-aggregate arguments with this function”:

```
[Sales] - AVG([Sales])
```

The user’s intent in this case was to compare store sales for each individual store to the average of sales for all stores. This can now be accomplished with a level of detail expression:

```
[Sales] - {AVG([Sales])}
```

This is what is known as a table-scoped level of detail expression. See [Table-Scoped Level of Detail Expressions](#) on page 905

Level of Detail Expression Syntax

A level of detail expression has the following structure:

```
{[FIXED | INCLUDE | EXCLUDE] <dimension declaration> : <aggregate expression>}
```

The elements in a level of detail expression are described in the following table.

Element	Description
{}	The entire level of detail expression is enclosed in curly braces.
[FIXED INCLUDE EXCLUDE]	The first element after the opening curly brace is one of the following scoping

	<p>keywords:</p> <ul style="list-style-type: none"> • FIXED <p>FIXED level of detail expressions compute values using the specified dimensions without reference to the view level of detail—that is, without reference to any other dimensions in the view.</p> <p>FIXED level of detail expressions also ignore all the filters in the view other than context filters, data source filters, and extract filters.</p> <p>Example: { FIXED [Region] : SUM([Sales]) }</p> <p>See FIXED Level of Detail Expressions on page 906.</p> <ul style="list-style-type: none"> • INCLUDE <p>INCLUDE level of detail expressions compute values using the specified dimensions in addition to whatever dimensions are in the view.</p> <p>INCLUDE level of detail expressions are most useful when including a dimension that isn't in the view.</p> <p>Example: { INCLUDE [Customer Name] : SUM([Sales]) }</p> <p>See INCLUDE Level of Detail Expressions on page 908.</p> <ul style="list-style-type: none"> • EXCLUDE <p>EXCLUDE level of detail expressions explicitly remove dimensions from the expression—that is, they subtract dimensions from the view level of detail.</p> <p>EXCLUDE level of detail expressions are most useful for eliminating a dimension in the view.</p>
--	--

	<p>Example: {EXCLUDE [Region] : SUM([Sales])}</p> <p>See EXCLUDE Level of Detail Expressions on page 912.</p> <p>In the case of a table-scoped level of detail expression, no scoping keyword is required.</p> <p>See Table-Spaced Level of Detail Expressions on page 905</p> <div style="background-color: #f0f0f0; padding: 10px;"> <p>The examples in the respective sections on the different types of level of detail expressions are simple examples. For more creative and advanced examples, aligned with real-world use cases, see the Tableau blog. Search for "level of detail expressions" or "LOD."</p> </div>
<dimension declaration>	<p>Specifies one or more dimensions to which the aggregate expression is to be joined. Use commas to separate dimensions. For example:</p> <pre>[Segment], [Category], [Region]</pre> <p>For level of detail expressions, you can use any expression that evaluates as dimension in a dimensionality declaration, including Date expressions.</p> <p>This example will aggregate the sum of Sales at the Year level:</p> <pre>{FIXED YEAR([Order Date]) : SUM(Sales)}</pre> <p>This example will aggregate the sum of Sales for the [Order Date] dimension, truncated to the day date part. Because it is an INCLUDE expression, it will also use the dimensions in the view to aggregate the value:</p>

```
{ INCLUDE DATETRUNC('day',
[Order Date]) : AVG(Profit) }
```

Note: It is strongly recommended that you drag fields into the calculation editor when creating dimension declarations, instead of typing them. For example, if you see **YEAR([Order Date])** on a shelf and then type that as the dimension declaration, it will not match the field on the shelf. But if you drag the field from the shelf into the expression, it will become **DATEPART('year', [Order Date])**, and that will match the field on the shelf.

With named calculations (that is, calculations that you save to the Data pane, as opposed to ad-hoc calculations, which you do not name), Tableau cannot match the name of a calculation to its definition. So if you create a named calculation, MyCalculation, defined as follows:

```
MyCalculation = YEAR([Order Date])
```

And then you created the following EXCLUDE level of detail expression and used it in the view:

```
{ EXCLUDE YEAR([Order Date]) : SUM(Sales) }
```

Then MyCalculation would not be excluded.

Similarly, if the EXCLUDE expression specified MyCalculation:

```
{ EXCLUDE MyCalculation : SUM(Sales) }
```

Then YEAR([Order Date]) would not be excluded.

:

A colon separates the dimension declaration

	from the aggregate expression.
<aggregate expression>	The aggregate expression is the calculation performed to define the target dimensionality.

Limitations for Level of Detail Expressions

The following limitations and constraints apply for level of detail expressions. Also see [Data Source Constraints for Level of Detail Expressions](#) on page 917.

- Level of detail expressions that reference floating-point measures can behave unreliably when used in a view that requires comparison of the values in the expression. For details, see [Data Types](#) on page 1549.
- Level of detail expressions are not shown on the Data Source page. See [Data Source Page](#) on page 130.
- When referencing a parameter in a dimensionality declaration, always use the parameter name, and not the parameter value.
- With data blending, the linking field from the primary data source must be in the view before you can use a level of detail expression from the secondary data source. See [Troubleshoot Data Blending](#) on page 382.

In addition, some data sources have complexity limits. Tableau will not disable calculations for these databases, but query errors are a possibility if calculations become too complex.

Level of Detail Expressions Can Be Dimensions or Measures

When you save a level of detail expression, Tableau adds it to either Dimensions or Measures.

FIXED level of detail expressions can result in measures or dimensions, depending on the underlying field in the aggregate expression. So `MIN([Date])` will be a dimension because [Date] is a dimension, and `{fixed Store : SUM([Sales])}` will be a measure because [Sales] is a measure. When a FIXED level of detail expression is saved as a measure you have the option of moving it to dimensions.

INCLUDE and EXCLUDE level of detail expressions are always measures.

Table-Spaced Level of Detail Expressions

It is possible to define a level of detail expression at the table level without using any of the scoping keywords. For example, the following expression returns the minimum (earliest) order date for the entire table:

```
{MIN( [Order Date] )}
```

This is equivalent to a FIXED level of detail expression with no dimension declaration:

```
{FIXED : MIN( [Order Date] )}
```

FIXED Level of Detail Expressions

FIXED level of detail expressions compute a value using the specified dimensions, without reference to the dimensions in the view.

The following examples can be recreated using the **Sample - Superstore** data source.

Example 1

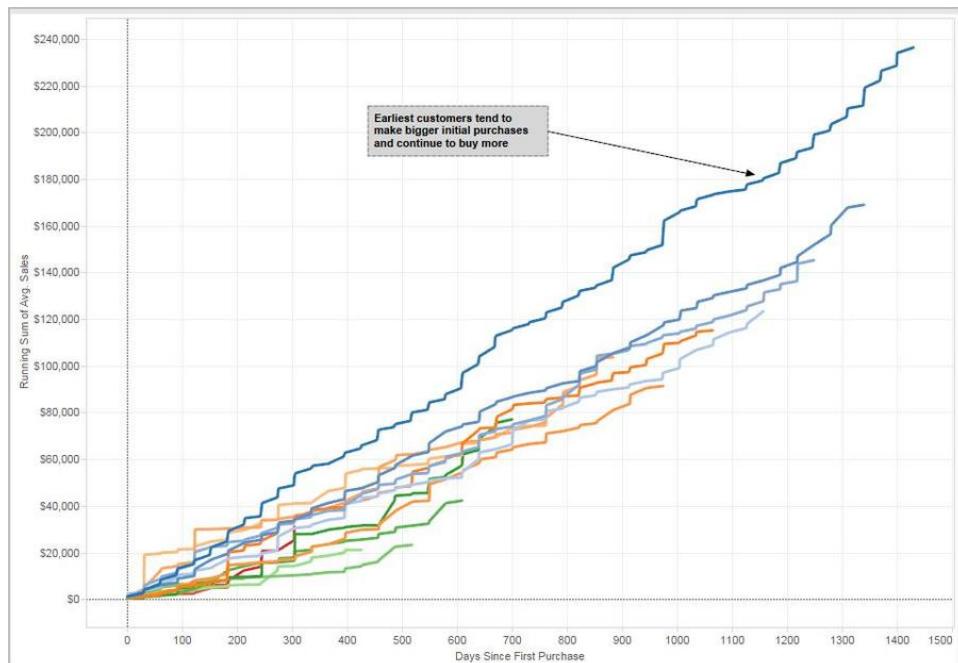
The following FIXED level of detail expression computes the first purchase date for each customer in the data source:

```
{FIXED [Customer Name] : MIN([Order Date])}
```

The resulting field, saved as **[First Purchase Date]**, is then used in a second calculated field to compute the interval between a customer's first purchase date and any subsequent purchase:

```
DATETRUNC('month', [Order Date])-DATETRUNC('month', [First Purchase Date])
```

By placing this calculated field, saved as **[Days Since First Purchase]**, on the **Columns** shelf, and a running total of sales on the **Rows** shelf, you can show customer buying behavior over time, as measured by days since first purchase:



Example 2

The following FIXED level of detail expression computes the sum of sales per region:

```
{FIXED [Region] : SUM([Sales])}
```

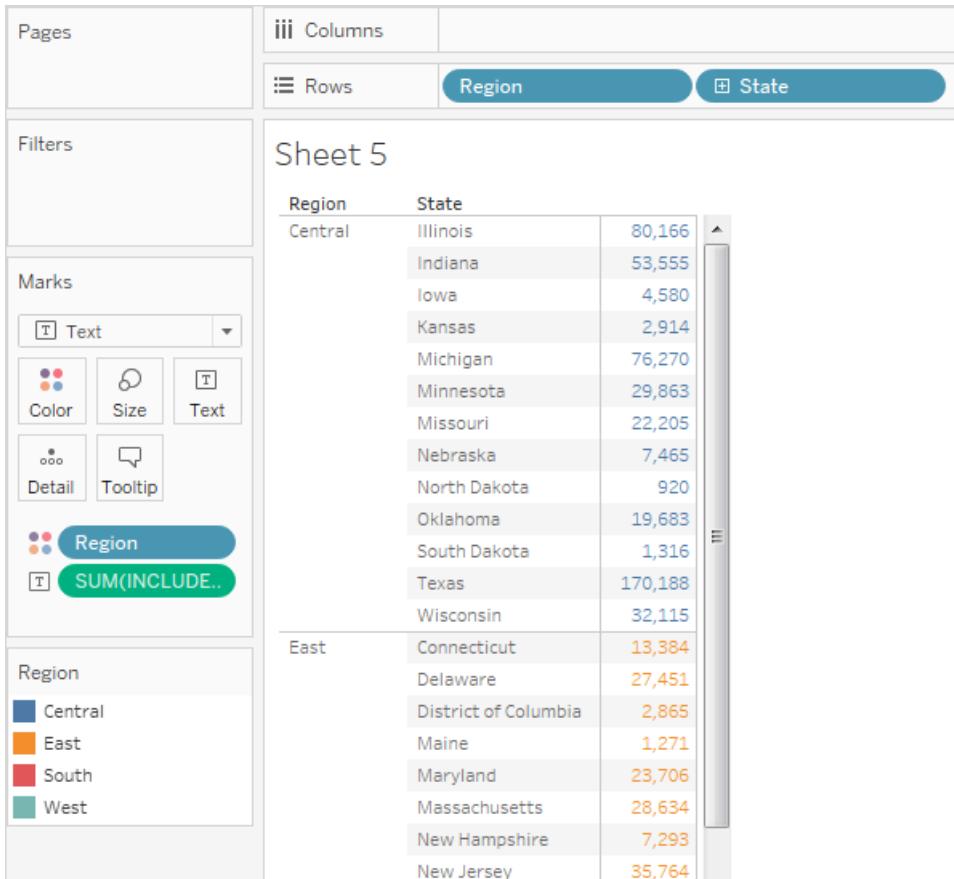
This level of detail expression, named [**Sales by Region**], is then placed on Text to show total sales per region:

The screenshot shows the Tableau Data Editor interface. On the left, there's a sidebar with 'Pages' and 'Filters'. Under 'Filters', there's a 'Marks' section with options for Text, Color, Size, Detail, and Tooltip. Below that, there's a 'Region' button and a 'SUM(Sales By ...)' button. To the right of the sidebar is a table titled 'Sheet 5' with two columns: 'Region' and 'State'. The data is grouped by region: Central and East. The Central region contains 11 states, all with the same value of 501,240. The East region contains 9 states, all with the same value of 678,781.

Region	State	
Central	Illinois	501,240
	Indiana	501,240
	Iowa	501,240
	Kansas	501,240
	Michigan	501,240
	Minnesota	501,240
	Missouri	501,240
	Nebraska	501,240
	North Dakota	501,240
	Oklahoma	501,240
	South Dakota	501,240
Texas	501,240	
Wisconsin	501,240	
East	Connecticut	678,781
	Delaware	678,781
	District of Columbia	678,781
	Maine	678,781
	Maryland	678,781
	Massachusetts	678,781
	New Hampshire	678,781
	New Jersey	678,781

The view level of detail is [**Region**] plus [**State**], but because FIXED level of detail expressions do not consider the view level of detail, the calculation only uses the [**Region**] dimension, and so the values for the individual states in each region are identical. See [Level of Detail Expressions and Aggregation on page 914](#) for an explanation of why this is so.

If the INCLUDE keyword had been used in the level of detail expression instead of FIXED, the values would be different for each state, because Tableau would add the dimension in the expression ([**Region**]) with any additional dimensions in the view ([**State**]) in determining values for the expression. The result would be:



INCLUDE Level of Detail Expressions

INCLUDE level of detail expressions compute values using the specified dimensions in addition to whatever dimensions are in the view.

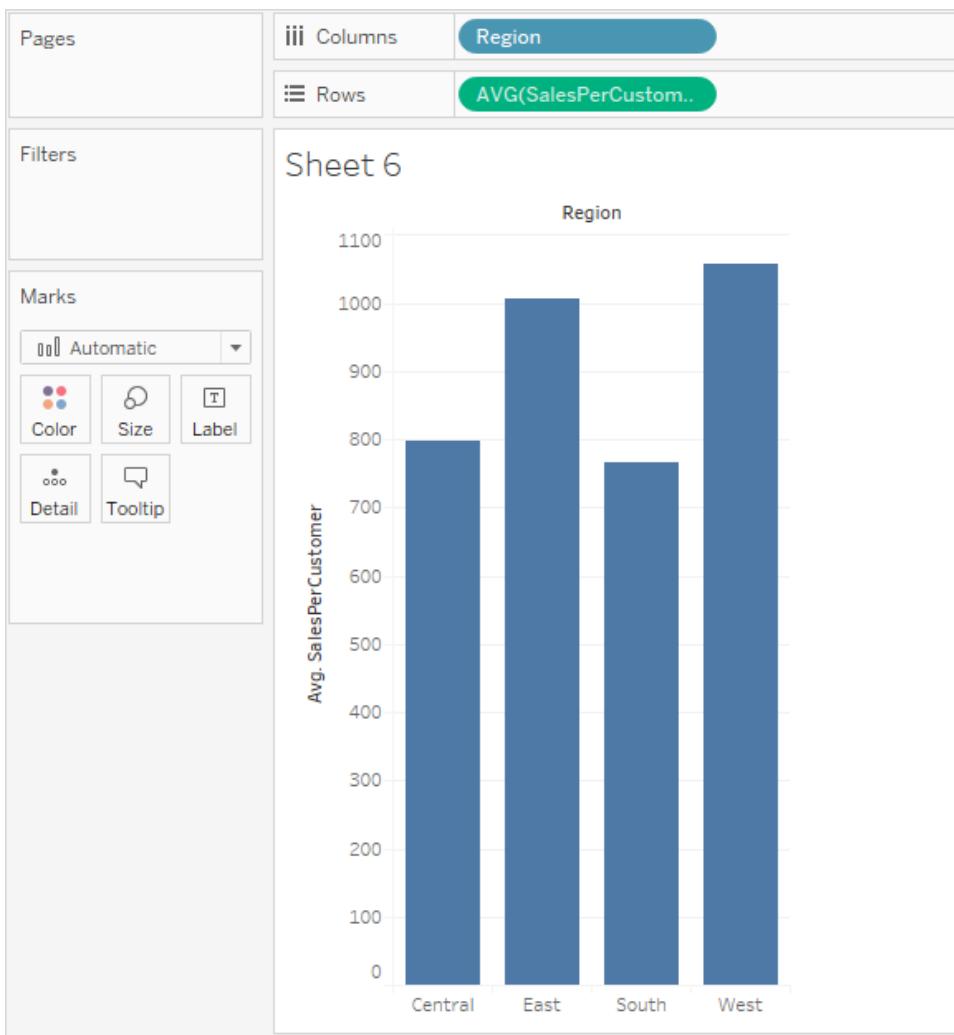
INCLUDE level of detail expressions can be useful when you want to calculate at a fine level of detail in the database and then re-aggregate and show at a coarser level of detail in your view. Fields based on INCLUDE level of detail expressions will change as you add or remove dimensions from the view.

Example 1

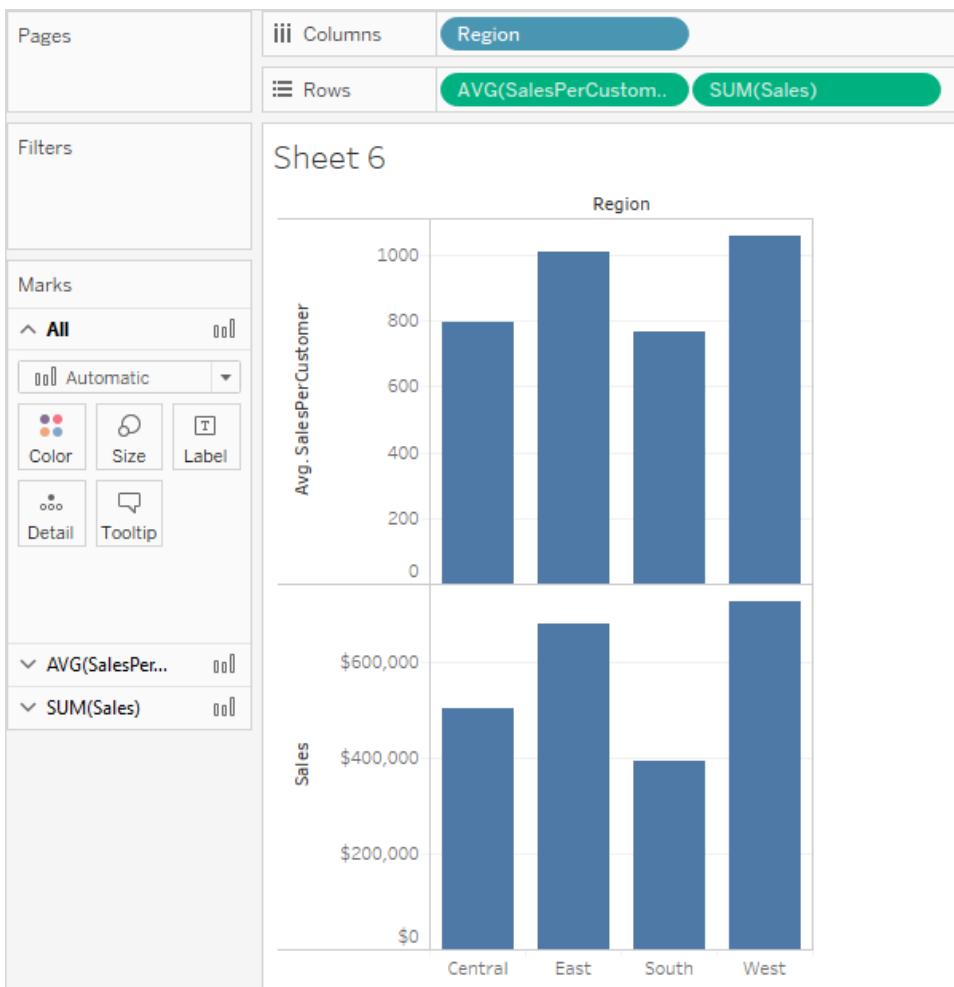
The following INCLUDE level of detail expression computes total sales per customer:

```
{ INCLUDE [Customer Name] : SUM([Sales]) }
```

When that calculation is placed on the **Rows** shelf, aggregated as AVG, and the **[Region]** dimension is placed on the **Columns** shelf, the view shows the average customer sales amount per region:



If the **[Sales]** measure is then dragged to the **Rows** shelf, the result illustrates the difference between the average sale (somewhere between \$200 and \$250 per region) and the average sale per customer (between \$750 and \$1100 per region):

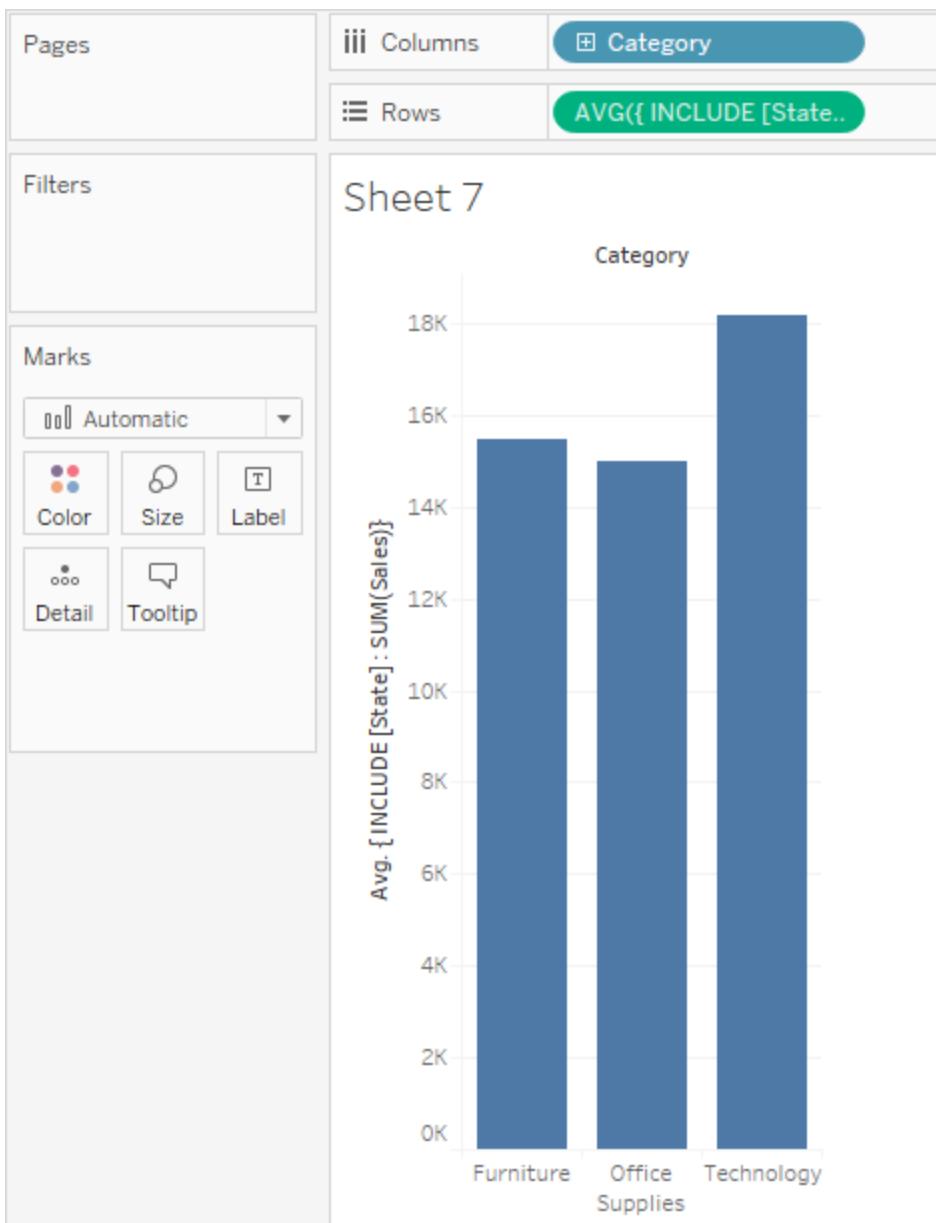


Example 2

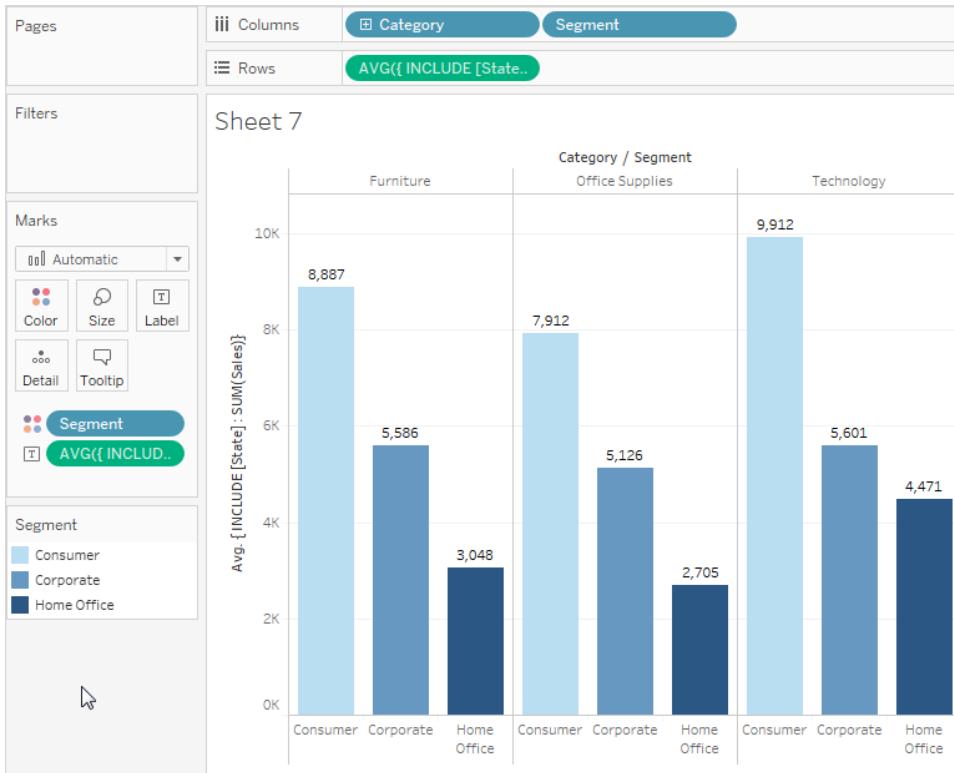
The following INCLUDE level of detail expression calculates sum of sales on a per-state basis:

```
{ INCLUDE [State] : SUM(Sales) }
```

Type the calculation directly on the **Rows** shelf, and then change the aggregation to AVG using the field's context menu. The resulting view averages the sum of sales by state across categories.



The view can be further enhanced by adding **[Segment]** to **Columns** and then copying the ad-hoc calculation from **Rows** to **Label** (using Ctrl+drag). In the image below we also dragged **Segment** to **Color** and then edited the colors for aesthetic effect. Now we can see how the average sum of sales per state varies across categories and segments.



EXCLUDE Level of Detail Expressions

EXCLUDE level of detail expressions declare dimensions to omit from the view level of detail.

EXCLUDE level of detail expressions are useful for 'percent of total' or 'difference from overall average' scenarios. They are comparable to such features as Totals and Reference Lines.

EXCLUDE level of detail expression cannot be used in row-level expressions (where there are no dimensions to omit), but can be used to modify either a view level calculation or anything in between (that is, you can use an EXCLUDE calculation to remove dimension from some other level of detail expression).

Example 1

The following EXCLUDE level of detail expression computes the average sales total per month and then excludes the month component:

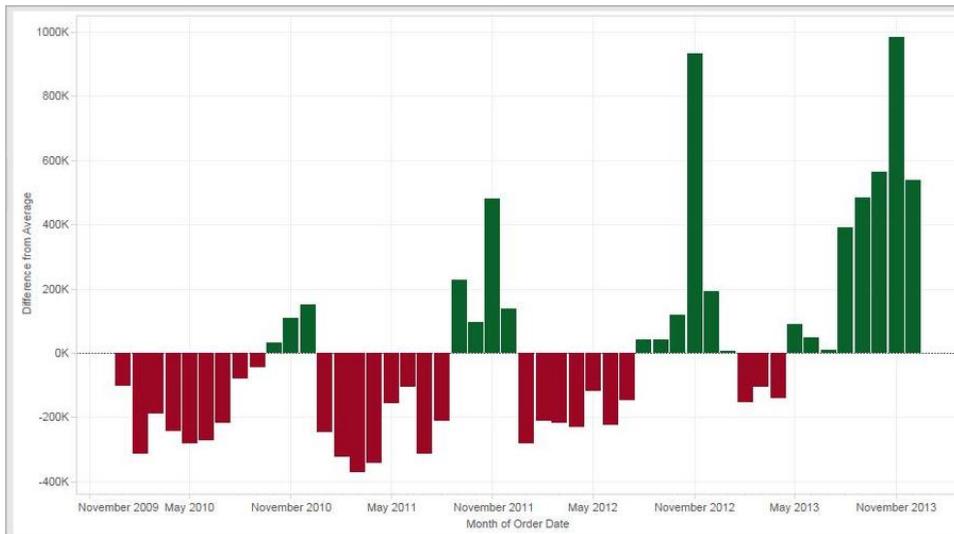
```
{EXCLUDE [Order Date (Month / Year)] : AVG({FIXED [Order Date (Month / Year)] : SUM([Sales])})}
```

Notice that this is a nested level of detail expression—that is, a level of detail expression within another level of detail expression.

Saved as **[average of sales by month]**, the calculation can then be subtracted from the sum of sales per month by means of an ad-hoc calculation on the **Rows** shelf:



With **Month([Order Date])** on the **Columns** shelf, this creates a view that shows the difference between actual sales per month over a four-year period and the average monthly sales for the entire four-year period:



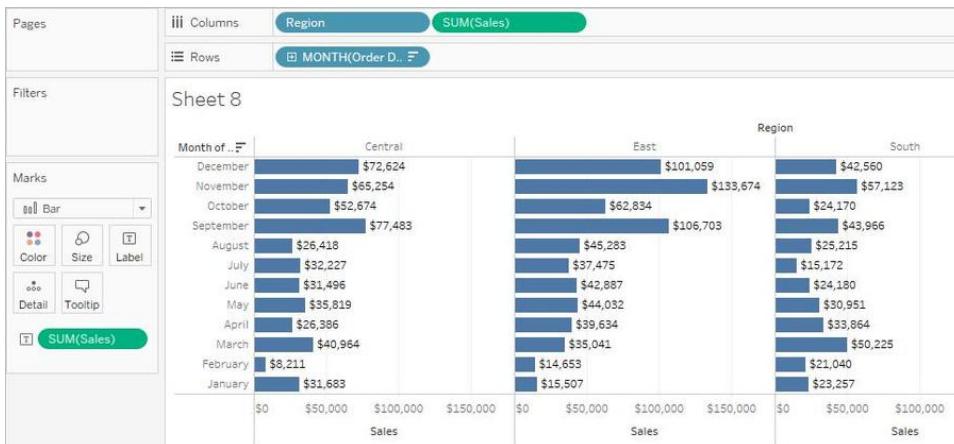
Example 2

The following level of detail expression excludes **[Region]** from a calculation of the sum of **[Sales]**:

```
{EXCLUDE [Region]: SUM([Sales])}
```

The expression is saved as **[ExcludeRegion]**.

To illustrate how this expression might be useful, first consider the following view, which breaks out the sum of sales by region and by month:



Dropping [**ExcludeRegion**] on Color shades the view to show total sales by month but without the regional component:



Level of Detail Expressions and Aggregation

The level of detail of the view determines the number of marks in your view. When you add a level of detail expression to the view, Tableau must reconcile two levels of detail—the one in the view, and the one in your expression.

The behavior of a level of detail expression in the view varies depending on whether the expression's level of detail is coarser, finer, or the same as the level of detail in the view. What do we mean by "coarser" or "finer" in this case?

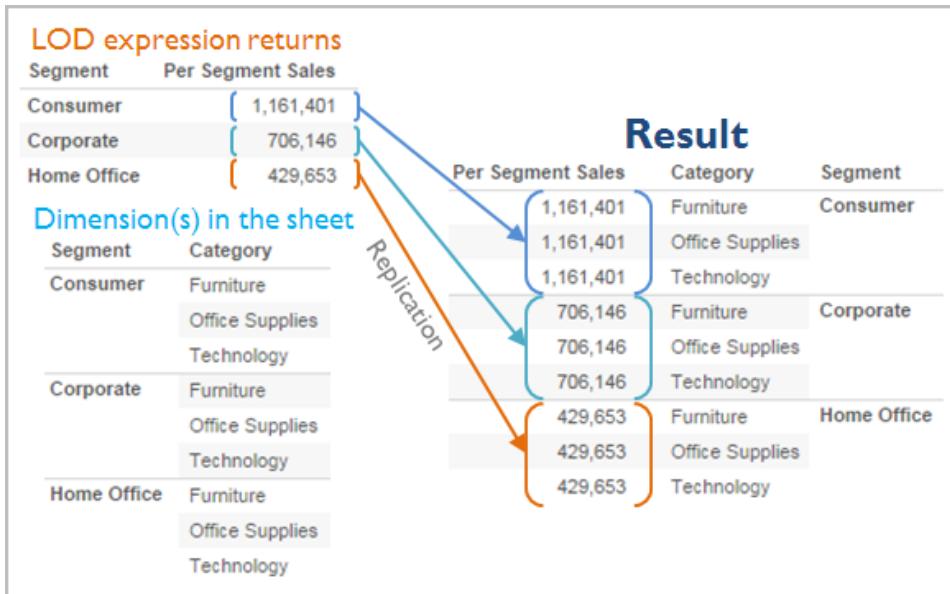
Level of Detail Expression is Coarser Than View Level of Detail

An expression has a coarser level of detail than the view when it references a subset of the dimensions in the view. For example, for a view that contained the dimensions [**Category**] and [**Segment**], you could create a level of detail expression that uses only one of these dimensions:

```
{FIXED [Segment] : SUM( [Sales] )}
```

In this case, the expression has a coarser level of detail than the view. It bases its values on one dimension ([**Segment**]), whereas the view is basing its view on two dimensions ([**Segment**] and [**Category**]).

The result is that using the level of detail expression in the view causes certain values to be replicated—that is, to appear multiple times.



Replicated values are useful for comparing specific values against average values within a category. For example the following calculation subtracts average sales for a customer from the average sales overall:

```
[Sales] - {FIXED [Customer Name] : AVG([Sales])}
```

When values are being replicated, changing the aggregation for the relevant field in the view (for example, from AVG to SUM) will not change the result of the aggregation.

Level of Detail Expression is Finer Than View Level of Detail

An expression has a finer level of detail than the view when it references a superset of the dimensions in the view. When you use such an expression in the view, Tableau will aggregate results up to the view level. For example, the following level of detail expression references two dimensions:

```
{FIXED [Segment], [Category] : SUM([Sales])}
```

When this expression is used in a view that has only [Segment] as its level of detail, the values must be aggregated. Here's what you would see if you dragged that expression to a shelf:

```
AVG({{FIXED [Segment]}, [Category] : SUM([Sales])})
```

An aggregation—in this case, average—is automatically assigned by Tableau. You can change the aggregation as needed.

Adding a Level of Detail Expression to the View

Whether a level of detail expression is aggregated or replicated in the view is determined by the expression type (FIXED, INCLUDE, or EXCLUDE) and whether the expression's granularity is coarser or finer than the view's.

- INCLUDE level of detail expressions will have either the same level of detail as the view or a finer level of detail than the view. Therefore, values will never be replicated.
- FIXED level of detail expressions can have a finer level of detail than the view, a coarser level of detail, or the same level of detail. The need to aggregate the results of a FIXED level of detail depends on what dimensions are in the view.
- EXCLUDE level of detail expressions always cause replicated values to appear in the view. When calculations including EXCLUDE level of detail expressions are placed on a shelf, Tableau defaults to the ATTR aggregation (as opposed to SUM or AVG) to indicate that the expression is not actually being aggregated and that changing the aggregation will have no effect on the view.

Level of detail expressions are always automatically wrapped in an aggregate when they are added to a shelf in the view unless they're used as dimensions. So if you double-click on a shelf and type

```
{FIXED[Segment], [Category] : SUM([Sales])}
```

and then press Enter to commit the expression, what you now see on the shelf is

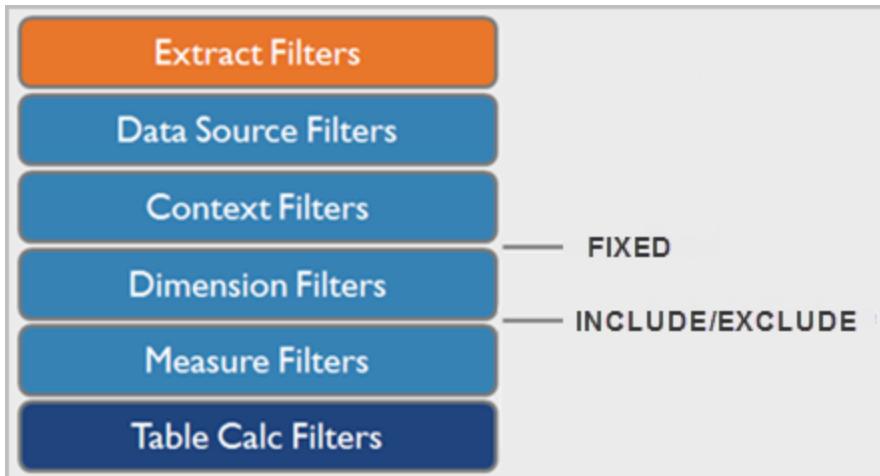
```
SUM({FIXED[Segment], [Category] : SUM([Sales])})
```

But if you double-click into the shelf to edit the expression, what you see in edit mode is the original expression.

If you wrap a level of detail expression in an aggregation when you create it, Tableau will use the aggregation you specified rather than assigning one when any calculation including that expression is placed on a shelf. When no aggregation is needed (because the expression's level of detail is coarser than the view's), the aggregation you specified is still shown when the expression is on a shelf, but it is ignored.

Filters and Level of Detail Expressions

There are several different kinds of filters in Tableau and they get executed in the following order from top to bottom.



The text on the right shows where level of detail expressions are evaluated in this sequence.

Extract Filters (in orange) are only relevant if you're creating a Tableau Extract from a data source. Table calculations filters (dark blue) are applied after calculations are executed and therefore hide marks without filtering out the underlying data used in the calculations.

If you're familiar with SQL, you can think of measure filters as equivalent to the HAVING clause in a query, and dimension filters as equivalent to the WHERE clause.

FIXED calculations are applied before dimension filters, so unless you promote the fields on your Filter shelf to [Improve View Performance with Context Filters on page 694](#), they will be ignored. For example, consider if you have the following calculation on one shelf in a view, along with **[State]** on a different shelf:

```
SUM([Sales]) / ATTR({FIXED : SUM([Sales])})
```

This calculation will give you the ratio of a state's sales to total sales.

If you then put **[State]** on the Filters shelf to hide some of the states, the filter will affect only the numerator in the calculation. Since the denominator is a FIXED level of detail expression, it will still divide the sales for the states still in the view against the total sales for all states—including the ones that have been filtered out of the view.

INCLUDE and EXCLUDE level of detail expressions are considered after Dimension filters. So if you want filters to apply to your FIXED level of detail expression but don't want to use Context Filters, consider rewriting them as INCLUDE or EXCLUDE expressions.

Data Source Constraints for Level of Detail Expressions

For some data sources, only more recent versions support level of detail expressions. Some data sources do not support level of detail expressions at all.

In addition, some data sources have complexity limits. Tableau will not disable calculations for these databases, but query errors are a possibility if calculations become too complex.

Data Source	Support
Actian Vectorwise	Not supported.
Amazon EMR	Supported for Hive 0.13 and later.
Amazon Redshift	Supported.
Aster Database	Supported for version 4.5 and later.
Cloudera Hadoop	Supported for Hive 0.13 and later.
Cloudera Impala	Supported for Impala 1.2.2 and later.
Cubes (multidimensional data sources)	Not supported.
DataStax Enterprise	Not supported.
EXASolution	Supported.
Firebird	Supported for version 2.0 and later.
Generic ODBC	Limited. Depends on the specific data source.
Google Big Query	Supported.
Hortonworks Hadoop Hive	Supported for Hive 0.13 and later. On version 1.1 of HIVE level of detail expressions that produce cross joins are not reliable. Cross join occur when there is no explicit field to join on. For example, for a level of detail expression {fixed [Product Type] : sum(sales)} when the view only contains one dimension [Ship Mode], Tableau creates a cross-join. A cross join produces rows which combine each row from the first table with each row from the second table.
HP Vertica	Supported for version 6.1 and later.
IBM BigInsights	Supported.

IBM DB2	Supported for version 8.1 and later.
MarkLogic	Supported for version 7.0 and later.
Microsoft Access	Not supported.
Microsoft Jet-based connections (legacy connectors for Microsoft Excel, Microsoft Access, and text)	Not supported.
Microsoft SQL Server	SQL Server 2005 and later.
MySQL	Supported.
IBM Netezza	Supported version 7.0 and later.
Oracle	Supported version 9i and later.
ParAccel	Supported version 3.1 and later.
Pivotal Greenplum	Supported for version 3.1 and later.
PostgreSQL	Supported version 7 and later.
Progress OpenEdge	Supported.
SAP HANA	Supported.
SAP Sybase ASE	Supported.
SAP Sybase IQ	Supported version 15.1 and later.
Spark SQL	Supported.
Splunk	Not supported.
Tableau Data	Supported.

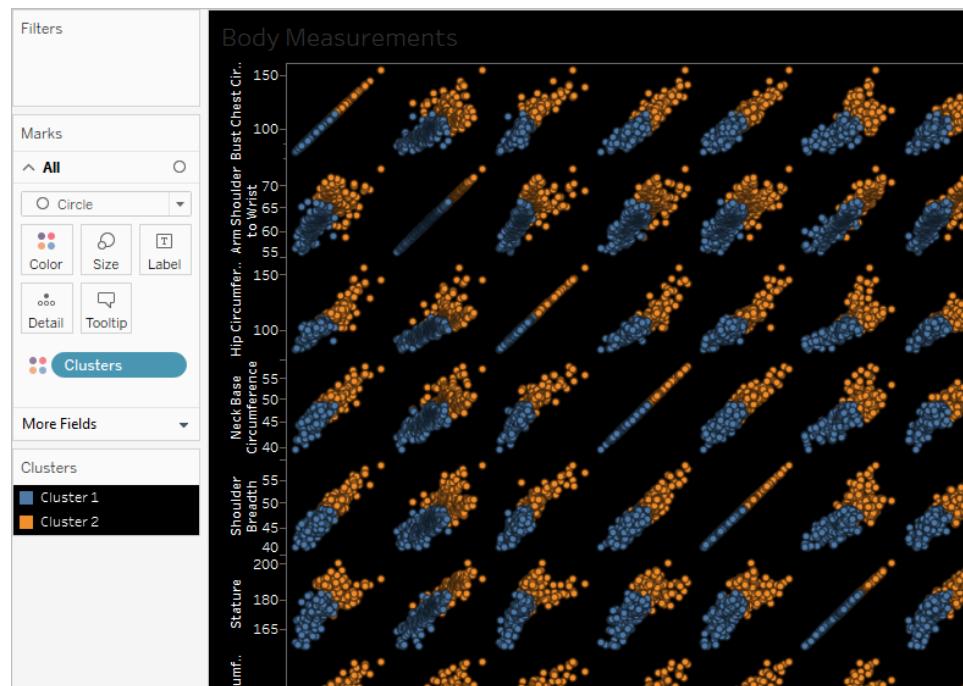
Extract

Teradata

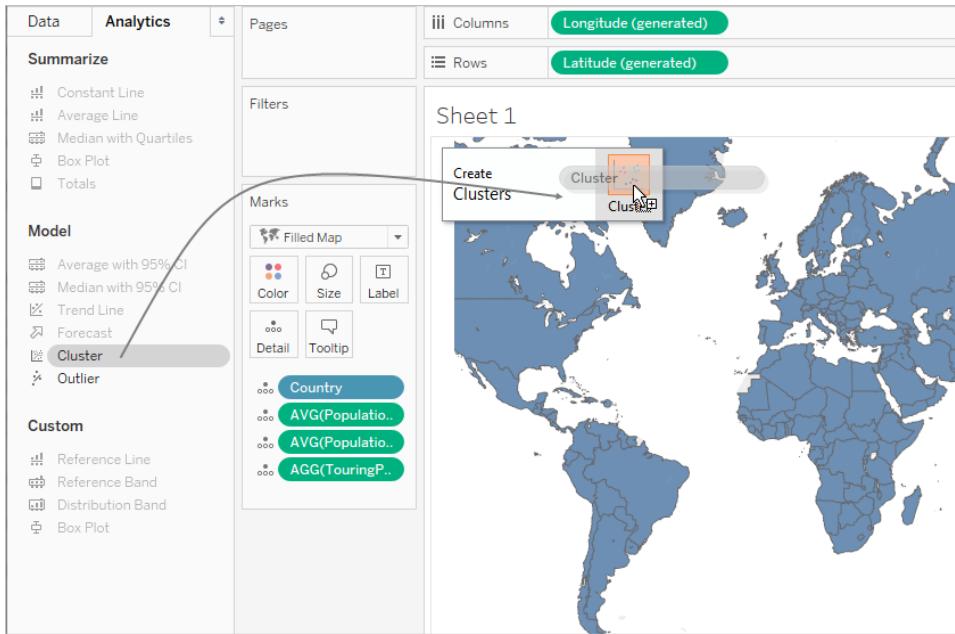
Supported.

Find Clusters in Data

Cluster analysis partitions marks in the view into clusters, where the marks within each cluster are more similar to one another than they are to marks in other clusters.



In Tableau Desktop, you create clusters by dragging **Cluster** from the **Analytics** pane and dropping it in the view.



The result is a statistical object that Tableau places on **Color**. If there is already a field on **Color**, Tableau moves that field to **Detail** and replaces it on **Color** with the clustering results.

Tableau assigns each mark in the view to one of the clusters; in some cases, it also assigns marks that do not fit well into any of clusters to a "Not Clustered" cluster.

When you add a cluster to the view, or edit an existing cluster on **Color**, Tableau displays a dialog box where you can:

- Add variables to or remove variables from clusters. Variables are the field that Tableau uses to compute clusters.
- Specify the number of clusters you want—or let Tableau determine the number of clusters automatically.
- Change the aggregation for a variable by right-clicking it and opening a context menu.

If you drag a cluster to the **Data** pane, it becomes a group dimension in which the individual members (Cluster 1, Cluster 2, etc.) contain the marks that the cluster algorithm has determined are more similar to each other than they are to other marks.

After you drag a cluster group to the **Data** pane, you can use it in other worksheets. See [Create a Tableau Group from Cluster Results](#) on page 925.

Clustering constraints

Clustering is available in Tableau Desktop, but is not available for authoring on the web (Tableau Server, Tableau Online). Clustering is also not available when any of the following conditions apply:

- When you are using a cube (multidimensional) data source.
- When there is a blended dimension in the view.
- When there are no fields that can be used as variables (inputs) for clustering in the view.
- When there are no dimensions present in an aggregated view.

When any of those conditions apply, you will not be able to drag **Clusters** from the Analytics pane to the view.

In addition, the following field types cannot be used as variables (inputs) for clustering:

- Table calculations
- Blended calculations
- Ad-hoc calculations
- Generated latitude/longitude values
- Groups
- Sets
- Bins
- Parameters
- Dates
- Measure Names/Measure Values

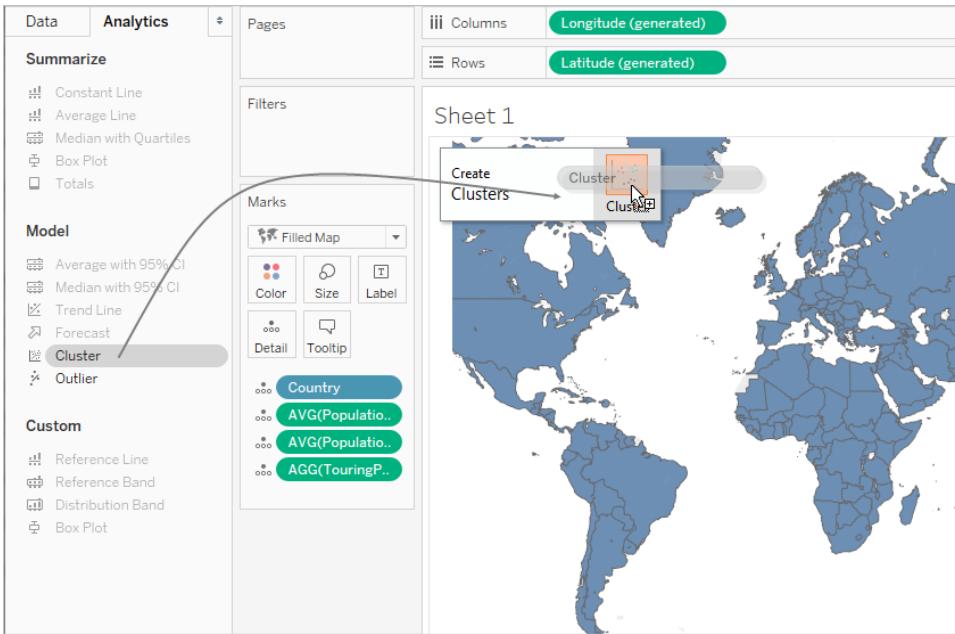
See Also

[Quick Start: Find Clusters in Data](#) on page 1719

Create Clusters

To find clusters in a view in Tableau, follow these steps.

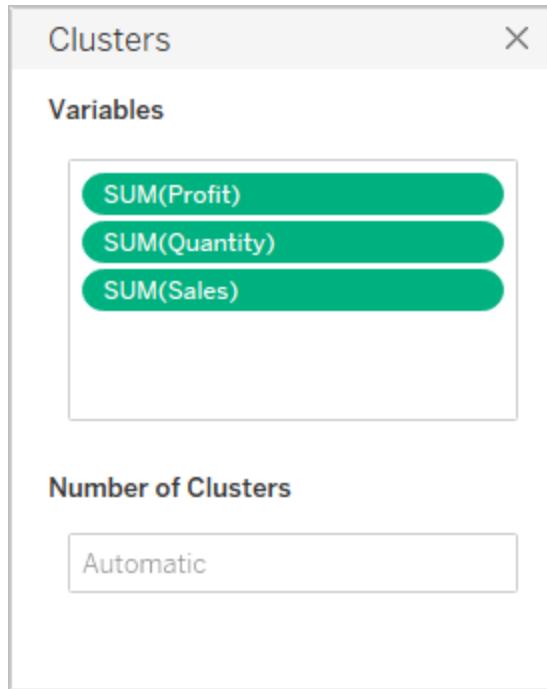
1. Create a view.
2. Drag **Cluster** from the **Analytics** pane into the view, and drop it on in the target area in the view:



You can also double-click **Cluster** to find clusters in the view.

Two things happen when you drop or double-click **Cluster**:

- Tableau adds **Clusters** on **Color**, and colors the marks in your view by cluster.
- Tableau displays a Clusters dialog box, where you can customize the cluster.



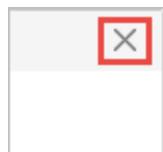
3. Customize the cluster results by doing either of the following in the Clusters dialog box:

- Drag new fields from the **Data** pane into the Variables area of the Clusters dialog box.

When you add variables, measures are aggregated using the default aggregation for the field; dimensions are aggregated using ATTR, which is the standard way that Tableau aggregates dimensions.

- Specify the number of clusters. If you do not specify a value, Tableau will go as high as 25 clusters in trying to determine the number of clusters. If you specify a value, you can choose any number between 2 and 50.

4. When you finish customizing the cluster results, click the X in the upper-right corner of the Clusters dialog box to close it:



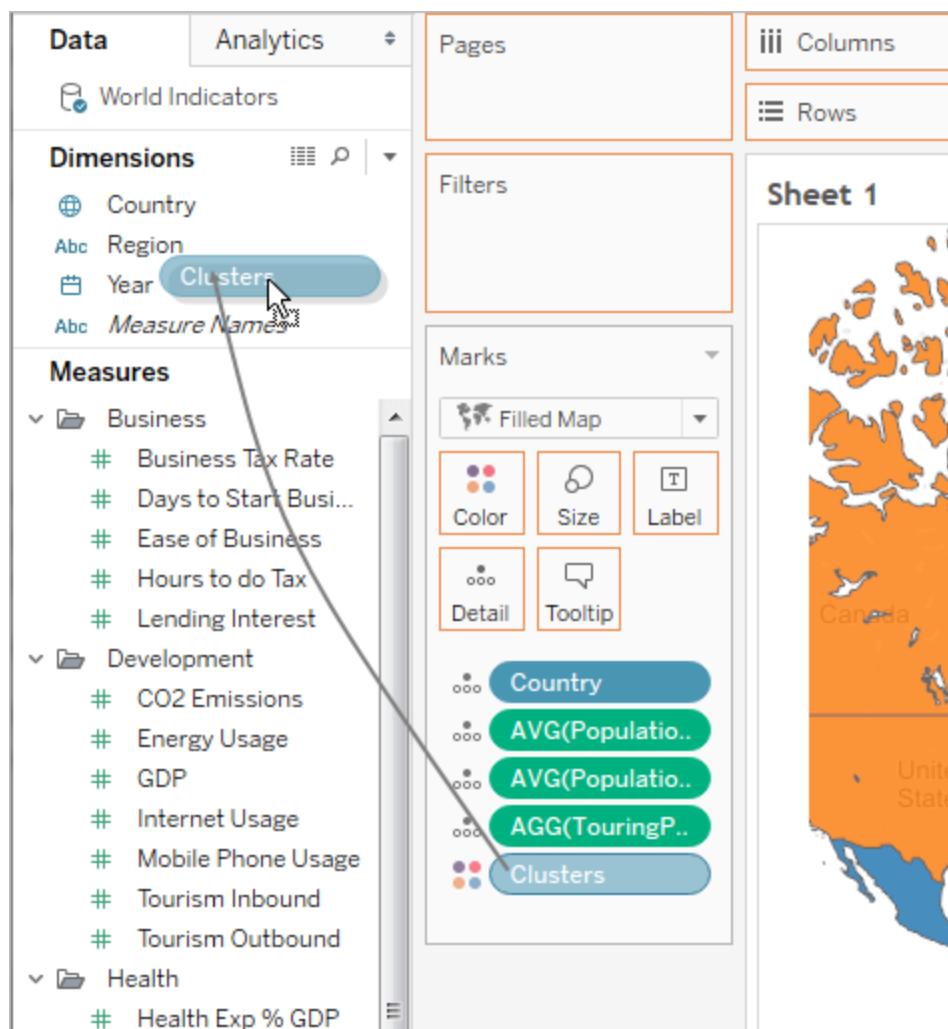
Note: You can move the cluster field from Color to another shelf in the view. However, you cannot move the cluster field from the **Filters** shelf to the **Data** pane.

To edit Clusters you have previously created, right-click (Control-click on a Mac) the **Clusters** field on **Color** and choose **Edit clusters**.

For an example showing the process of creating clusters with sample data (world economic indicators), see [Example - Create Clusters from World Economic Indicators Data](#) on page 932.

Create a Tableau Group from Cluster Results

Drag **Clusters** from the **Marks** card (or from any other shelf you had already dragged it to) to the **Data** pane to create a Tableau group:



After you create a group from clusters, the group and the original clusters are separate and distinct. Editing the clusters does not affect the group, and editing the group does not affect the cluster results. The group has the same characteristics as any other Tableau group. It is part of the data source. Unlike the original clusters, you can use the group in other worksheets in the

workbook. So if you rename the saved cluster group, that renaming is not applied to the original clustering in the view. See [Groups](#) on page 497.

Constraints on Saving Clusters

You will not be able to save Clusters to the **Data** pane under any of the following circumstances:

- When the measures in the view are disaggregated and the measures you are using as clustering variables are not the same as the measures in the view. See [Disaggregating Data](#) on page 338.
- When the Clusters you want to save are on the **Filters** shelf.
- When **Measure Names** or **Measure Values** is in the view.
- When there is a blended dimension in the view.

Statistics for Clustering

The Describe Clusters dialog box provides information about the models that Tableau computed for clustering. You can use these statistics to assess the quality of the clustering.

When the view includes clustering, you can open the Describe Clusters dialog box by right-clicking **Clusters** on the **Marks** card (Control-clicking on a Mac) and choosing **Describe Clusters**. The information in the Describe Clusters dialog box is read-only, though you can click **Copy to Clipboard** and then paste the screen contents into a writeable document.

The Describe Clusters dialog box has two tabs: a Summary tab and a Models tab.

Describe Clusters – Summary Tab

The Summary tab identifies the inputs that were used to generate the clusters and provides some statistics that characterize the clusters.

Inputs for Clustering

Variables

Identifies the fields Tableau uses to compute clusters. These are the fields listed in the Variables box in the Clusters dialog box.

Level of Detail

Identifies the fields that are contributing to the view's level of detail—that is, the fields that determine the level of aggregation. For details, see [Dimensions and the Level of Detail](#) on page 277.

Scaling

Identifies the scaling method used for pre-processing. Normalized is currently the only scaling method Tableau uses. The formula for this method, also known as min-max normalization, is $(x - \min(x)) / (\max(x) - \min(x))$.

Summary Diagnostics

Number of Clusters

The number of individual clusters in the clustering.

Number of Points

The number of marks in the view.

Between-group sum of squares

A metric quantifying the separation between clusters as a sum of squared distances between each cluster's center (average value), weighted by the number of data points assigned to the cluster, and the center of the data set. The larger the value, the better the separation between clusters.

Within-group sum of squares

A metric quantifying the cohesion of clusters as a sum of squared distances between the center of each cluster and the individual marks in the cluster. The smaller the value, the more cohesive the clusters.

Total sum of squares

Totals the between-group sum of squares and the within-group sum of squares. The ratio (between-group sum of squares)/(total sum of squares) gives the proportion of variance explained by the model. Values are between 0 and 1; larger values typically indicate a better model. However, you can increase this ratio just by increasing the number of clusters, so it could be misleading if you compare a five-cluster model with a three-cluster model using just this value.

Cluster Statistics

For each cluster in the clustering, the following information is provided.

Items

The number of marks within the cluster.

Centers

The average value within each cluster (shown for numeric items).

Most Common

The most common value within each cluster (only shown for categorical items).

Describe Clusters – Models Tab

Analysis of variance (ANOVA) is a collection of statistical models and associated procedures useful for analyzing variation within and between observations that have been partitioned into groups or clusters. In this case, analysis of variance is computed per variable, and the resulting analysis of variance table can be used to determine which variables are most effective for distinguishing clusters.

Relevant Analysis of variance statistics for Tableau Clustering include:

F-statistic

The F-statistic for one-way, or single-factor, ANOVA is the fraction of variance explained by a variable. It is the ratio of the between-group variance to the total variance.

The larger the F-statistic, the better the corresponding variable is distinguishing between clusters.

p-value

The p-value is the probability that the F-distribution of all possible values of the F-statistic takes on a value greater than the actual F-statistic for a variable. If the p-value falls below a specified significance level, then the null hypothesis (that the individual elements of the variable are random samples from a single population) can be rejected. The degrees of freedom for this F-distribution are $(k - 1, N - k)$, where k is the number of clusters and N is the number of items (rows) clustered.

The lower the p-value, the more the expected values of the elements of the corresponding variable differ among clusters.

Model Sum of Squares and Degrees of Freedom

The Model Sum of Squares is the ratio of the between-group sum of squares to the model degrees of freedom. The between group sum of squares is a measure of the variation between cluster means. If the cluster means are close to each other (and therefore close to the overall mean), this value will be small. The model has $k-1$ degrees of freedom, where k is the number of clusters.

Error Sum of Squares and Degrees of Freedom

The Error Sum of Squares is the ratio of within-group sum of squares to the error degrees of freedom. The within-group sum-of-squares measures the variation between observations within each cluster. The error has $N-k$ degrees of freedom, where N is the total number of observations (rows) clustered and k is the number of clusters.

The Error Sum of Squares can be thought of as the overall Mean Square Error, assuming that each cluster center represents the "truth" for each cluster.

How Clustering Works in Tableau

Cluster analysis partitions the marks in the view into clusters, where the marks within each cluster are more similar to one another than they are to marks in other clusters. Tableau distinguishes clusters using color.

Note: For additional insight into how clustering works in Tableau, see the blog post [Understanding Clustering in Tableau 10](#).

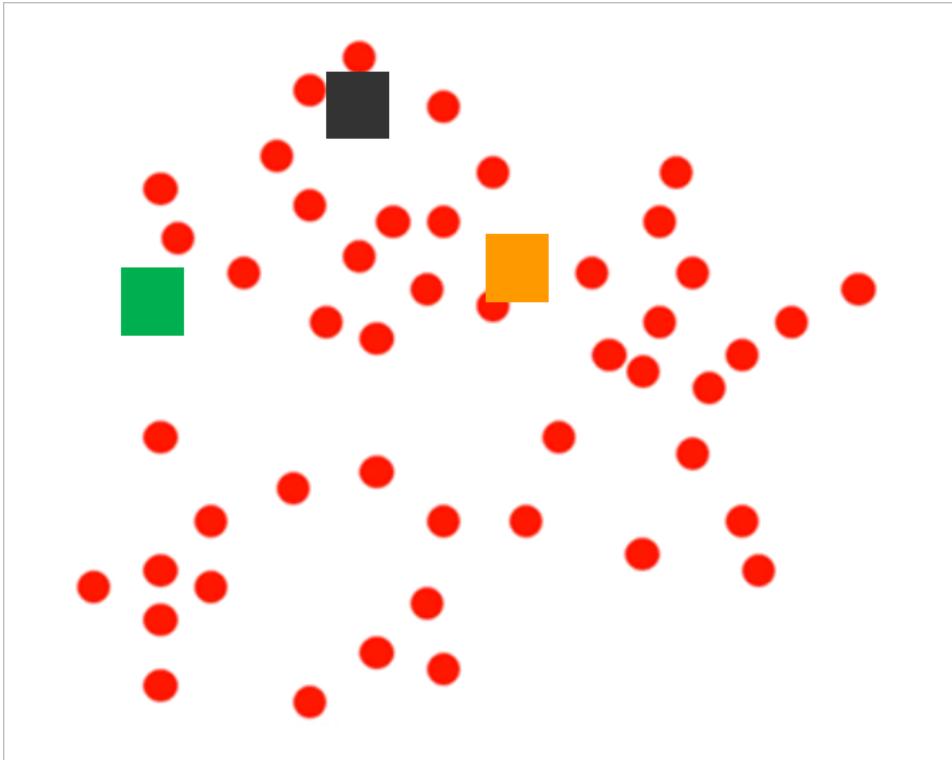
The clustering algorithm

Tableau uses the k-means algorithm for clustering. For a given number of clusters k , the algorithm partitions the data into k clusters. Each cluster has a center (centroid) that is the mean value of all the points in that cluster. K-means locates centers through an iterative procedure that minimizes distances between individual points in a cluster and the cluster center. In Tableau, you can specify a desired number of clusters, or have Tableau test different values of k and suggest an optimal number of clusters (see [Determining the optimal number of clusters on page 931](#)).

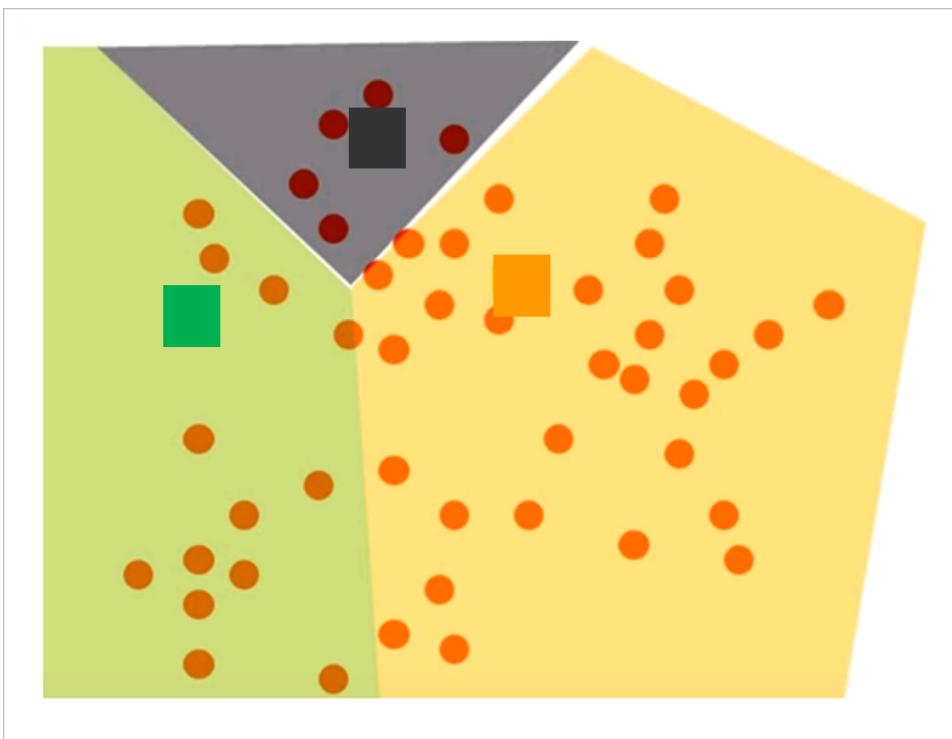
K-means requires an initial specification of cluster centers. Starting with one cluster, the method chooses a variable whose mean is used as a threshold for splitting the data in two. The centroids of these two parts are then used to initialize k-means to optimize the membership of the two clusters. Next, one of the two clusters is chosen for splitting and a variable within that cluster is chosen whose mean is used as a threshold for splitting that cluster in two. K-means is then used to partition the data into three clusters, initialized with the centroids of the two parts of the split cluster and the centroid of the remaining cluster. This process is repeated until a set number of clusters is reached.

Tableau uses Lloyd's algorithm with squared Euclidean distances to compute the k-means clustering for each k . Combined with the splitting procedure to determine the initial centers for each $k > 1$, the resulting clustering is deterministic, with the result dependent only on the number of clusters.

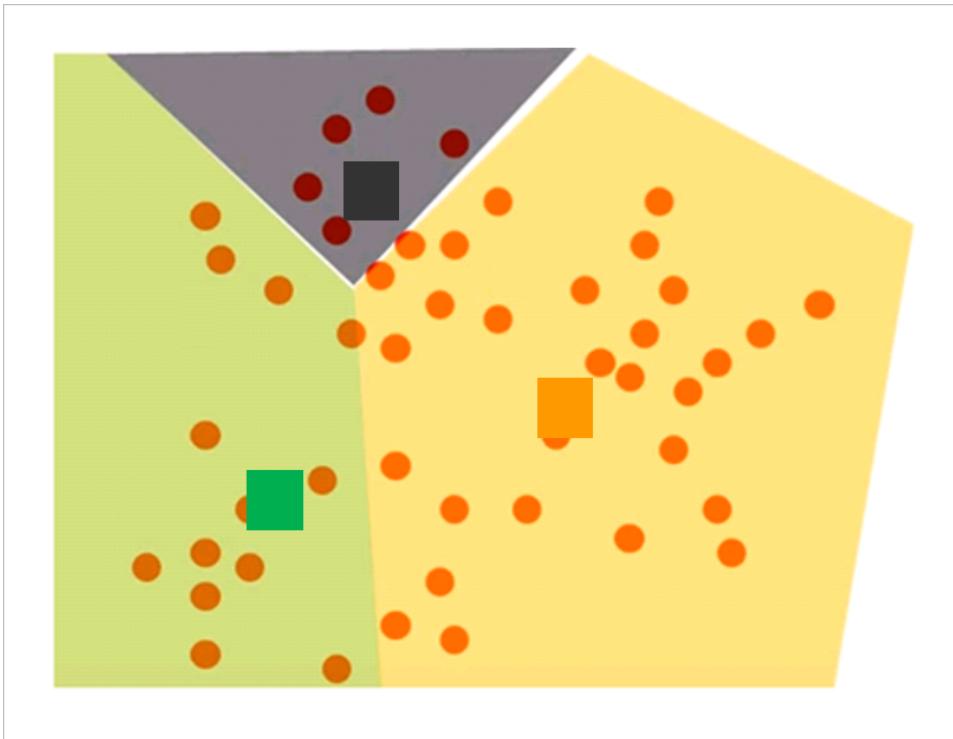
The algorithm starts by picking initial cluster centers:



It then partitions the marks by assigning each to its nearest center:



Then it refines the results by computing new centers for each partition by averaging all the points assigned to the same cluster:



It then reviews the assignment of marks to clusters and reassigns any marks that are now closer to a different center than before.

The clusters are redefined and marks are reassigned iteratively until no more changes are occurring.

Determining the optimal number of clusters

Tableau uses the Calinski-Harabasz criterion to assess cluster quality. The Calinski-Harabasz criterion is defined as

$$\frac{SS_B}{SS_W} \times \frac{(N-k)}{(k-1)}$$

where SS_B is the overall between-cluster variance, SS_W the overall within-cluster variance, k the number of clusters, and N the number of observations.

The greater the value of this ratio, the more cohesive the clusters (low within-cluster variance) and the more distinct/separate the individual clusters (high between-cluster variance).

Since the Calinski-Harabasz index is not defined for k=1, it cannot be used to detect one-cluster cases.

If a user does not specify the number of clusters, Tableau picks the number of clusters corresponding to the first local maximum of the Calinski-Harabasz index. By default, k-means will be run for up to 25 clusters if the first local maximum of the index is not reached for a smaller value of k. You can set a maximum value of 50 clusters.

Note: If a categorical variable (that is, a dimension) has more than 25 unique values, then Tableau will disregard that variable when computing clusters.

What values get assigned to the Not Clustered category?

When there are null values for a measure, Tableau assigns values for rows with null to a **Not Clustered** category. Categorical variables (that is, dimensions) that return * for ATTR (meaning that all values are not identical) are also not clustered.

Scaling

Tableau scales values automatically so that columns having a larger range of magnitudes don't dominate the results. For example, an analyst could be using inflation and GDP as input variables for clustering, but because GDP values are in trillions of dollars, this could cause the inflation values to be almost completely disregarded in the computation. Tableau uses a scaling method called *min-max normalization*, in which the values of each variable is mapped to a value between 0 and 1 by subtracting its minimum and dividing by its range.

Example - Create Clusters from World Economic Indicators Data

The Tableau clustering feature partitions marks in the view into clusters, where the marks within each cluster are more similar to one another than they are to marks in other clusters. This example shows how a researcher might use clustering to find an optimal set of marks (in this case, countries) in a data source.

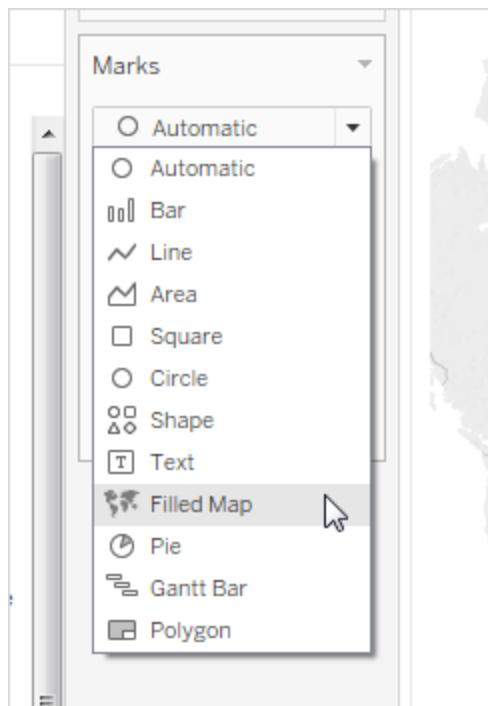
The objective

As life expectancy increases around the world, and as older people remain more active, senior tourism can be a lucrative market for companies that know how to find and appeal to potential customers. The World Indicators sample data set that comes with Tableau contains the kind of data that might help companies identify the countries where there are enough of the right kind of customers.

Finding the right countries

Here is an example of how Tableau clustering could help such a company identify the countries where a senior tourism business could succeed. Imagine you are the analyst. Here is how you might proceed.

1. Open the **World Indicators** sample data source in Tableau Desktop.
2. Double-click **Country** in the **Data** pane.
Tableau automatically creates a map view, with a mark in each country.
3. On the **Marks** card, change the mark type to **Filled Map**:



You should now see a map projection where all countries are filled with a solid color:

Sheet 1



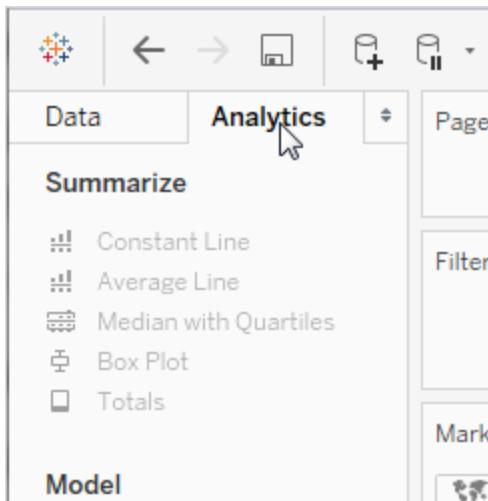
4. The next step is to identify the fields that you will use as variables for clustering. Here are the fields you choose:

Field	Reason for inclusion
Life Expectancy Female and Life Expectancy Male	Where people are living longer, there are more likely to be people who are interested in traveling later in life.
Population Urban	It is easier to market services in areas with greater population density.
Population 65+	The target population is older residents with the time and funds to travel.
TourismPerCapita	This is a measure that you must create as a named calculated field. The formula is: $\text{SUM}([\text{Tourism Outbound}]) / \text{SUM}([\text{Population Total}])$ Tourism Outbound aggregates the money (in US dollars) that residents of a country spend annually on international travel. But this total must be divided by the population of each country

	to determine the average amount each resident spends on international travel.
--	---

There is no guarantee that these are the ideal fields to choose, or that these fields will produce cluster results that are clear and unambiguous. Clustering is an iterative process—experimentation leads to discovery which leads, in turn, to more experimentation.

5. Drag these five fields from the **Data** pane to **Detail** on the **Marks** card.
6. Click to open the **Analytics** pane:



7. Drag **Cluster** from the **Analytics** pane and drop it in the view:

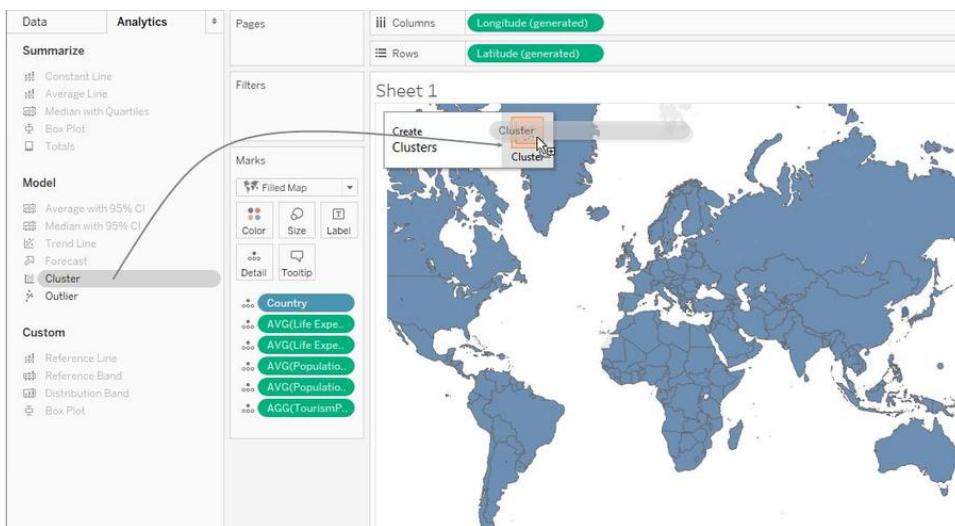
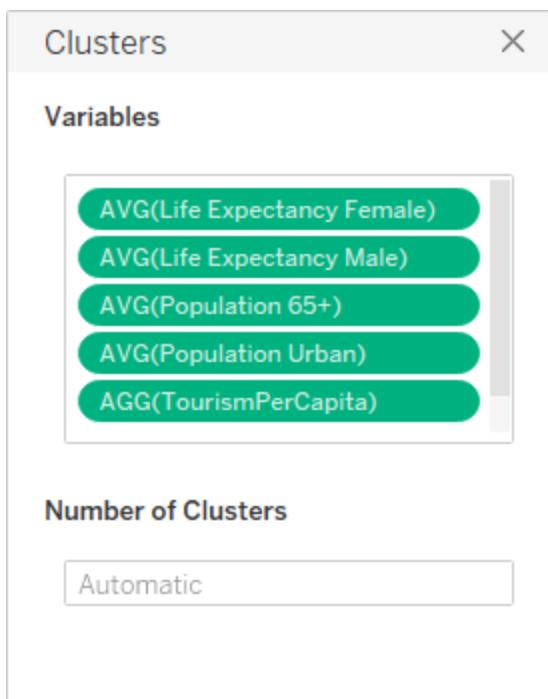
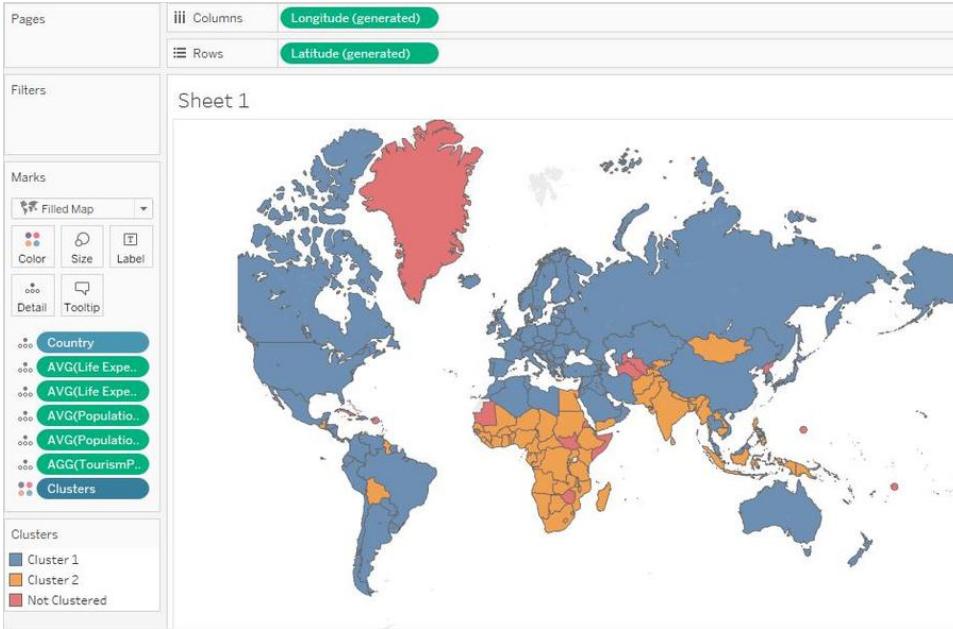


Tableau displays the Clusters dialog box and adds the measures in the view to the list of variables:



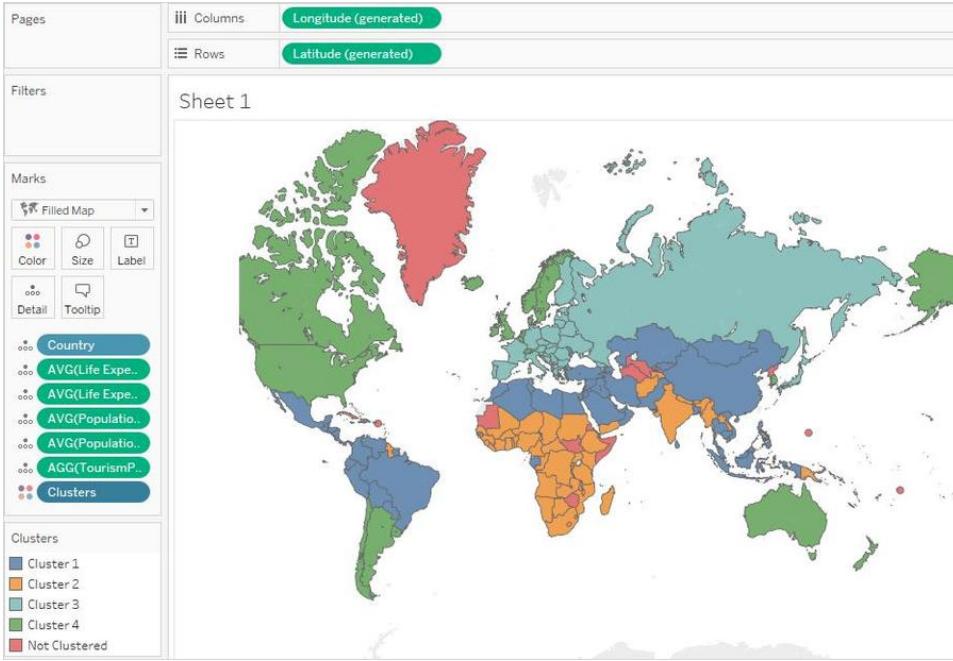
It also updates the view by adding clusters to **Color**. In this case, Tableau finds two distinct clusters, and is unable to assign certain countries (colored reddish-pink) to either cluster:



Note: See [How Clustering Works in Tableau](#) on page 929 for details on data that Tableau assigns to "Not clustered."

8. You decide that two clusters isn't enough—you don't have the resources to set up shop in half the countries in the world. So you type 4 in the **Number of Clusters** field in the Clusters dialog box.

The map becomes more interesting:



But how do these clusters relate to the variables you have chosen? Which one correlates best with the factors that support senior tourism? It's time to look at the statistics behind the clusters.

- Close the Clusters dialog box by clicking the X in its upper-right corner:



- Click the **Clusters** field on the **Marks** card and choose **Describe Clusters**.

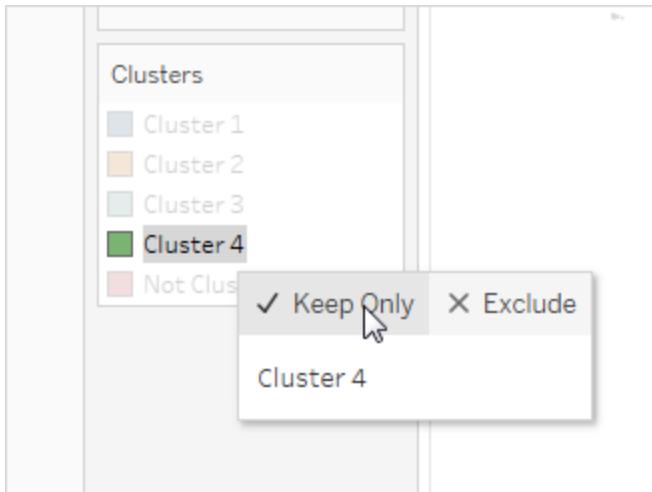
The table at the bottom of the **Models** tab in the Describe Clusters dialog box shows the average value for each variable in each cluster:

Clusters	Number of Items	Avg. Life Expectancy Female	Avg. Life Expectancy Male	Avg. Population 65+	Avg. Population Urban	TourismPerCapita
Cluster 1	69	74.216	69.003	0.054943	0.5325	197.12
Cluster 2	52	56.75	54.429	0.031889	0.33127	20.775
Cluster 3	29	79.164	71.706	0.15493	0.65532	320.92
Cluster 4	30	81.287	76.355	0.11606	0.87123	1360.4
Not Clustered	28					

Cluster 4 has the highest life expectancy (both male and female), the highest concentration of urban population, and the highest expenditure for international tourism: \$1360.40 per capita. The only variable for which Cluster 4 does not have the highest value is **Population 65+**, where Cluster 3 has the advantage: 0.15493 (just under 16%) to 0.11606 (just over 11%) in Cluster 4.

The clustering algorithm does not know whether you are looking for the maximum value for these variables, the minimum value, or something in the middle—it just looks for correlation. But you know that higher values for these variables is the signal you're looking for, and Cluster 4 is the best choice.

11. You could attempt to pick out the Cluster 4 countries from the map, but there is an easier way. Close the **Describe Clusters** dialog box and then click **Cluster 4** on the Color legend and choose **Keep Only**.



12. Choose Text Table from ShowMe.

You now see a list of the countries in Cluster 4:

Clusters	Country	
Cluster 4	Argentina	Abc
	Australia	Abc
	Bahamas, The	Abc
	Belgium	Abc
	Canada	Abc
	Chile	Abc
	Curacao	Abc
	Cyprus	Abc
	Denmark	Abc
	Hong Kong SAR, China	Abc
	Iceland	Abc
	Ireland	Abc
	Israel	Abc
	Korea, Rep.	Abc
	Kuwait	Abc
	Lebanon	Abc
	Luxembourg	Abc
	Macao SAR, China	Abc
	Malta	Abc
	Netherlands	Abc
	New Zealand	Abc
	Norway	Abc
	Puerto Rico	Abc
	Qatar	Abc
	Singapore	Abc
	Sweden	Abc
	Switzerland	Abc
	United Kingdom	Abc
	United States	Abc
	Uruguay	Abc

This list is not the end of the process. You might try clustering again with a somewhat different set of variables and maybe a different number of clusters, or you might add some countries to the list and remove others, based on other factors. For example, if your tours are mostly to tropical locales, you might remove countries like Curacao and the Bahamas from the list, because tropical tours might not appeal to residents of those countries.

Another option is to filter your data before you re-cluster, to only show countries with populations above a certain threshold, or to target countries in a particular geographical area.

Percentages

Any analysis in Tableau can be expressed in terms of percentages. For example, rather than viewing sales for every product, you might want to view each product's sales as a percentage of the total sales for all products.

You calculate percentages by selecting the **Analysis > Percentages Of** menu item. When you do this, all measures on the worksheet are displayed as a percentage based on all the table data.

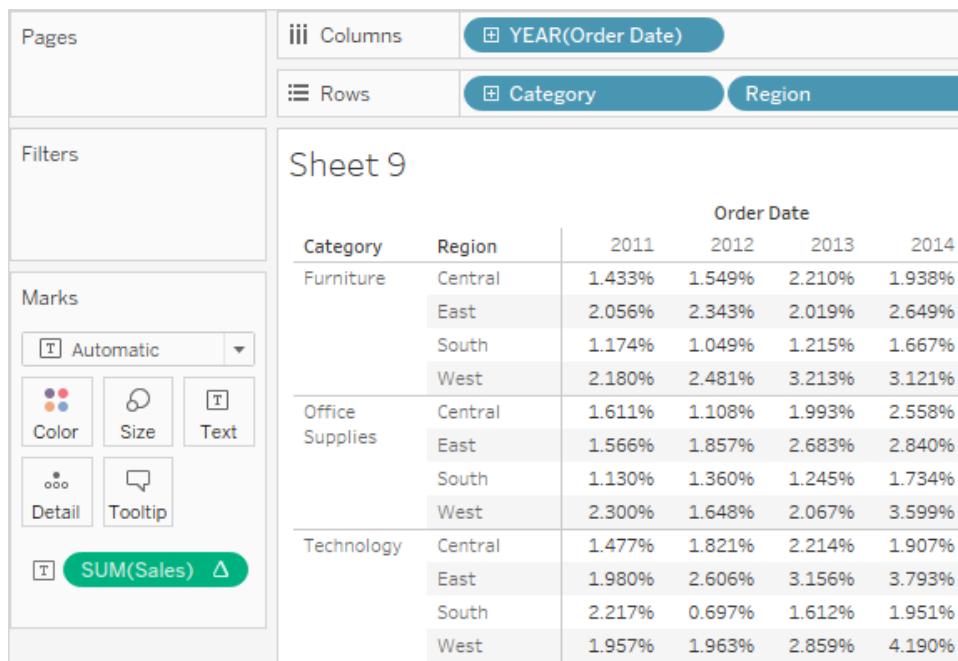
The percentage options on the Analysis menu correspond to the percentage table calculations. When you select a percentage option, you are actually adding a Percent of Total table calculation. See [Transform Values with Table Calculations](#) on page 843 for more information.

About Percentages

There are two factors that contribute to the percentage calculation:

- The aggregation – Percentages are calculated on the basis of the current aggregation for each measure. Refer to [Percentages and Aggregations](#) on the next page for more information.
- The data to which you compare all percentage calculations – Percentages are a ratio of numbers. The numerator is the value of a given mark. The denominator depends on the type of percentage you want, and is the number to which you compare all your calculations. The comparison can be based on the entire table, a row, a pane, and so on. By default, Tableau uses the entire table. Other percentage calculations are available via the **Percentage of** menu item. Refer to [Percentage Options](#) on page 944 for more information.

The figure below is an example of a text table with percentages. The percentages are calculated with the **Sales** measure aggregated as a summation, and are based on the entire table.



Percentages and Aggregations

Percentages are computed on the basis of the aggregation for each measure. Standard aggregations include summation, average, and several others. See [Aggregations](#) on page 329 for more information.

For example, if the aggregation applied to the **Sales** measure is a summation, then the default percentage calculation (percent of table) means that each number displayed is the $\text{SUM}(\text{Sales})$ for that mark divided by the $\text{SUM}(\text{Sales})$ for the entire table.

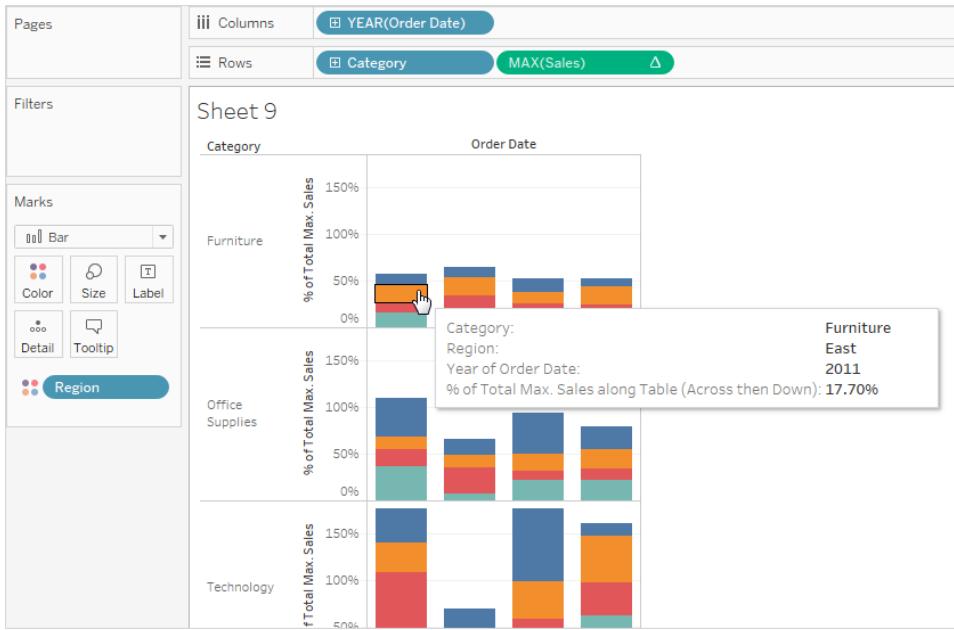
In addition to using predefined aggregations, you can use custom aggregations when calculating percentages. You define your own aggregations by creating a calculated field. Once the new field is created, you can use percentages on the field as you would any other field. See [Aggregate Calculations](#) on page 838 for more information.

Percent calculations can also be applied to disaggregated data. In this case, all values are expressed as the percentage of a summation. You cannot choose any other aggregation.

Example – Percentages and Aggregations

The view below shows a nested bar chart created using two dimensions and a measure that is aggregated as a maximum. Additionally, the data are color-encoded by a dimension and the default percentage calculation has been applied. Notice that the axis labels are modified to reflect the percent calculation.

The tooltip reveals that the maximum sales for furniture in the east in 2011 is 17.70% of the maximum for the entire table. What is the maximum for the table? If you recreate the view you'll see that the maximum occurs in the South, in the Technology category, in the year 2011. The tooltip for this bar segment would reveal a maximum sales of 100%.



The next view displays two disaggregated measures as a scatter plot. Again, the default percentage calculation has been applied as reflected by the modified axis labels.

The tooltip shows that the selected data point constitutes -0.475 percent of total profit and a 0.3552 percent of total sales. Percentage calculations are based on the entire data source.



Percentage Options

Computing a percentage involves specifying a total on which the percentage is based. In Tableau, the default percentage calculation is based on the entire table. However, you can change the default by selecting a different percentage option from the **Analysis > Percentage Of** menu.

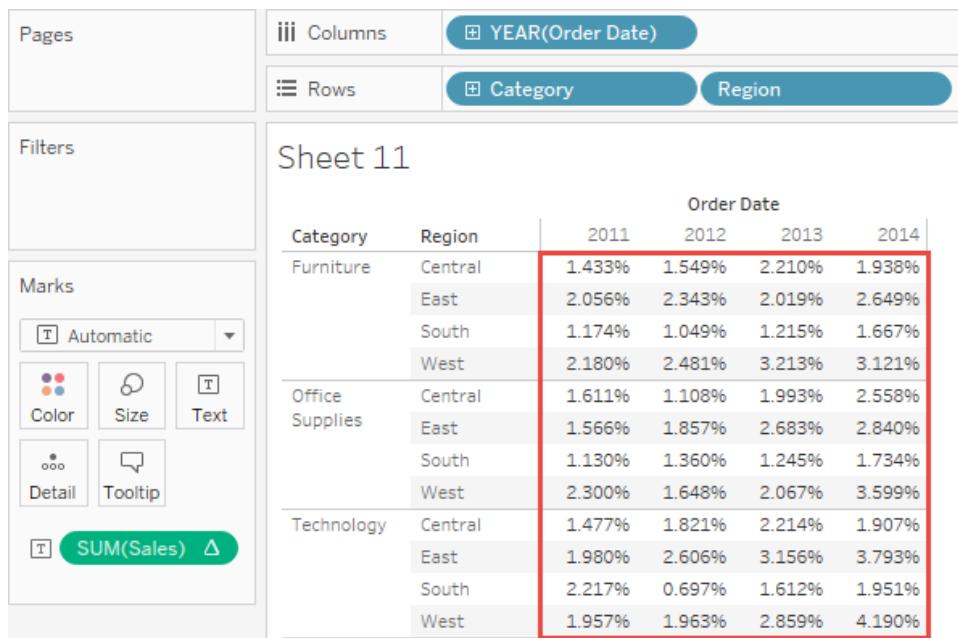
The option you choose is applied uniformly to all measures that appear on a worksheet. For instance, you cannot choose **Percent of Column** for one measure and **Percent of Row** for another.

If you are unsure what the current percentage calculation means, display the grand totals. This provides more information about each row and column. For example, if you select **Percent of Row** while displaying grand totals, you will see that the total for each row is exactly 100%. See [Grand Totals on page 979](#) for more information on grand totals.

The percent calculation options are described in the following sections. In each case, the grand totals are displayed as well.

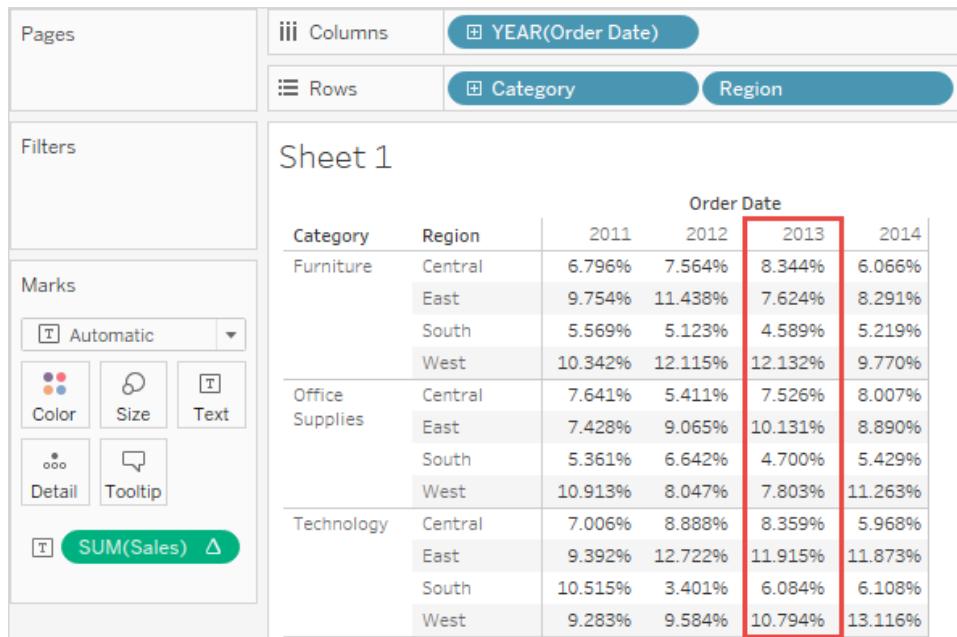
Percent of Table

When you select **Percentage Of > Table** from the **Analysis** menu, each measure on the worksheet is expressed as a percentage of the total for the entire worksheet (table). For example, Technology in the East region accounts for 3.79% of total sales in 2014. The grand totals for rows show that 2014 accounts for 31.95% of the total sales. Summing the grand totals for rows or for columns yields 100% of the total.



Percent of Column

When you select **Percentage of > Column** from the **Analysis** menu, each measure on the worksheet is expressed as a percentage of the total for the column. The values within the red box add up to 100%.



Percent of Row

When you select **Percentage of Row**, each measure on the worksheet is expressed as a percentage of the total for the row. The values within the red box add up to 100%.

Sheet 1

Order Date					
Category	Region	2011	2012	2013	2014
Furniture	Central	20.09%	21.73%	31.00%	27.18%
	East	22.68%	25.84%	22.27%	29.22%
	South	22.99%	20.55%	23.80%	32.66%
	West	19.83%	22.57%	29.22%	28.39%
Office Supplies	Central	22.15%	15.24%	27.42%	35.19%
	East	17.50%	20.76%	30.00%	31.75%
	South	20.66%	24.87%	22.76%	31.71%
	West	23.93%	17.14%	21.50%	37.43%
Technology	Central	19.91%	24.54%	29.85%	25.70%
	East	17.16%	22.59%	27.36%	32.89%
	South	34.23%	10.76%	24.89%	30.13%
	West	17.84%	17.90%	26.06%	38.20%

Percent of Pane

When you select **Percentage of > Pane** from the **Analysis** menu, each measure on the worksheet is expressed as a percentage of the total for the panes in the view. This option is equivalent to **Percent of Table** when the table consists of only a single pane.

In the following view, the red box constitutes a single pane; the values within the red box add up to 100%.

Sheet 1

Order Date					
Category	Region	2011	2012	2013	2014
Furniture	Central	4.435%	4.797%	6.843%	6.000%
	East	6.366%	7.253%	6.252%	8.201%
	South	3.635%	3.248%	3.763%	5.162%
	West	6.750%	7.683%	9.949%	9.664%
Office Supplies	Central	5.146%	3.541%	6.369%	8.173%
	East	5.002%	5.932%	8.573%	9.074%
	South	3.610%	4.346%	3.977%	5.541%
	West	7.350%	5.266%	6.603%	11.497%
Technology	Central	4.057%	5.002%	6.083%	5.239%
	East	5.439%	7.159%	8.670%	10.421%
	South	6.090%	1.914%	4.428%	5.361%
	West	5.376%	5.393%	7.855%	11.512%

Percent of Row in Pane

When you select **Percentage of > Row in Pane** from the **Analysis** menu, each measure on the worksheet is expressed as a percentage of the total for a row within a pane. This option is equivalent to as **Percent of Row** when the table is only a single pane wide.

In the following view, the red box constitutes a row within a pane; the values within the red box add up to 100%.

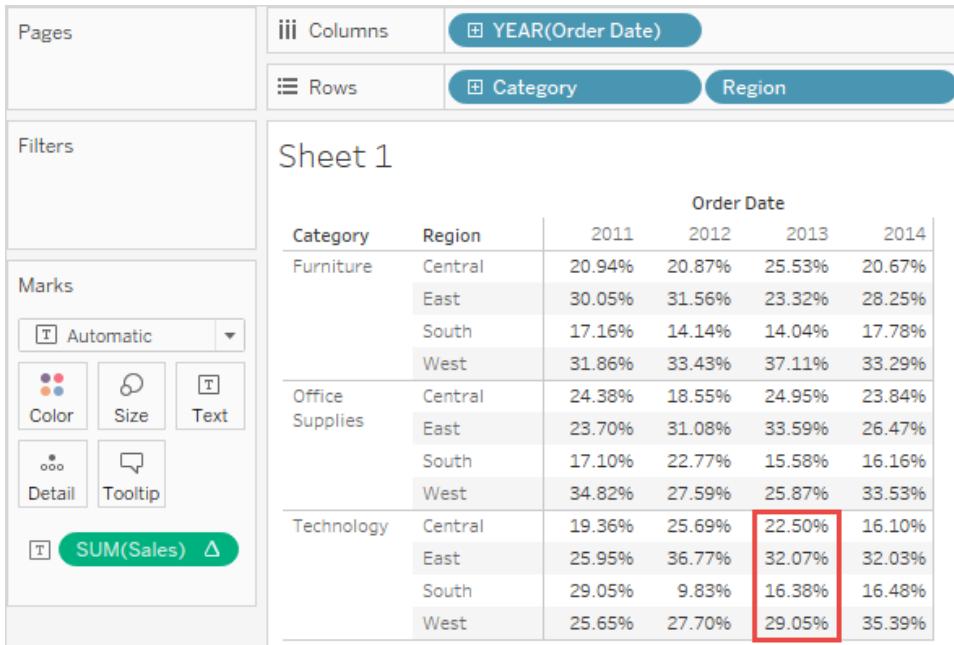


Note: If you place **Measure Names** as the inner dimension on the **Columns** shelf (that is, the dimension farthest to the right), Tableau will return 100% for each mark because you cannot total up the values for multiple measure names. For example, you can't total up the values for SUM(Sales) and SUM(Profit).

Percent of Column in Pane

When you select **Percentage of > Column in Pane** from the **Analysis** menu, each measure in the worksheet is expressed as a percentage of the total for a column within a pane. This option is equivalent to as **Percent of Column** when the table is only a single pane high.

In the following view, the red box constitutes a column within a pane; the values within the red box add up to 100%.



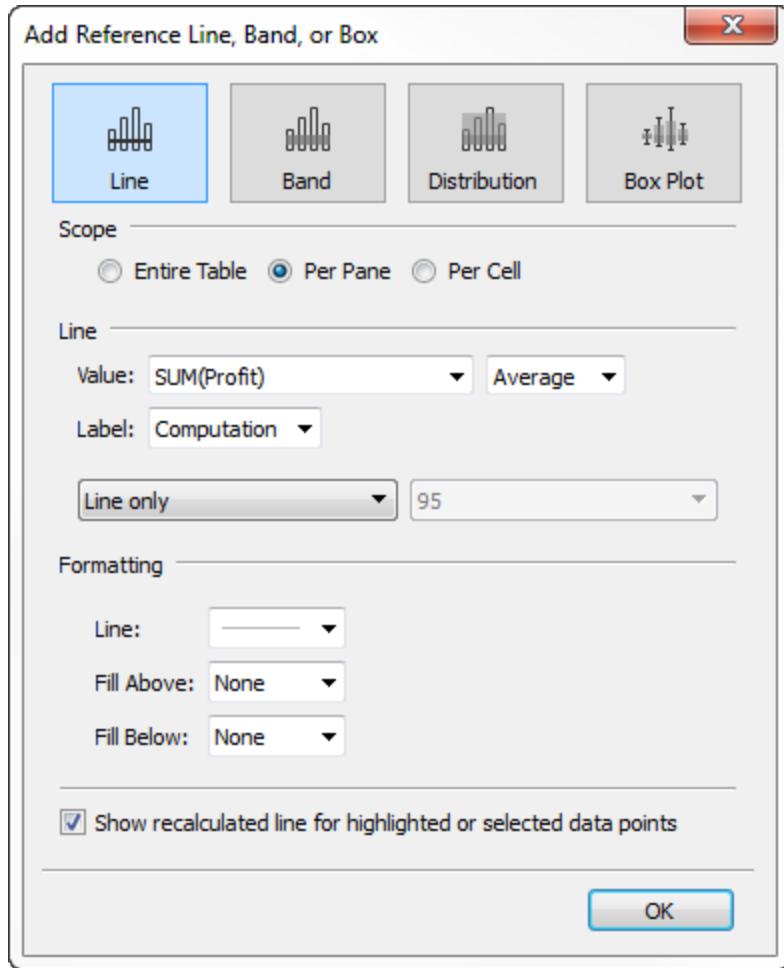
If you place **Measure Names** as the inner dimension on the **Rows** shelf (that is, the dimension farthest to the right on the shelf), Tableau will return 100% for each mark because you cannot total up the values for multiple measure names. For example, you can't total up the values for SUM(Sales) and SUM(Profit).

Reference Lines, Bands, Distributions, and Boxes

You can add a reference line, band, distribution or box plot to identify a specific value, region, or range on a continuous axis in a Tableau view. For example, if you are analyzing the monthly sales for several products, you can include a reference line at the average sales mark so you can see how each product performed against the average. Or you might use a box plot to shade a particular area along the axis, show a distribution, or show the full range of values.

Note: You can use the Analytics Pane to quickly drag a reference line into the view. See [Analytics Pane on page 135](#).

Tableau lets you add an unlimited number of reference lines, bands, distributions, and box plots to a view. Right-click (control-click on a Mac) on an axis and select **Add Reference Line** to open the Add Reference Line, Band, or Box dialog box:



Note: Reference lines, bands, distributions, and boxes are not available when the view is a map using online or offline maps.

Types of Reference Lines, Bands, and Boxes

You can add reference lines, bands, distributions or box plots to any continuous axis in the view. In Tableau Desktop, you can add or edit any of these objects, but when you are working on the web, you can add or edit reference lines or bands.

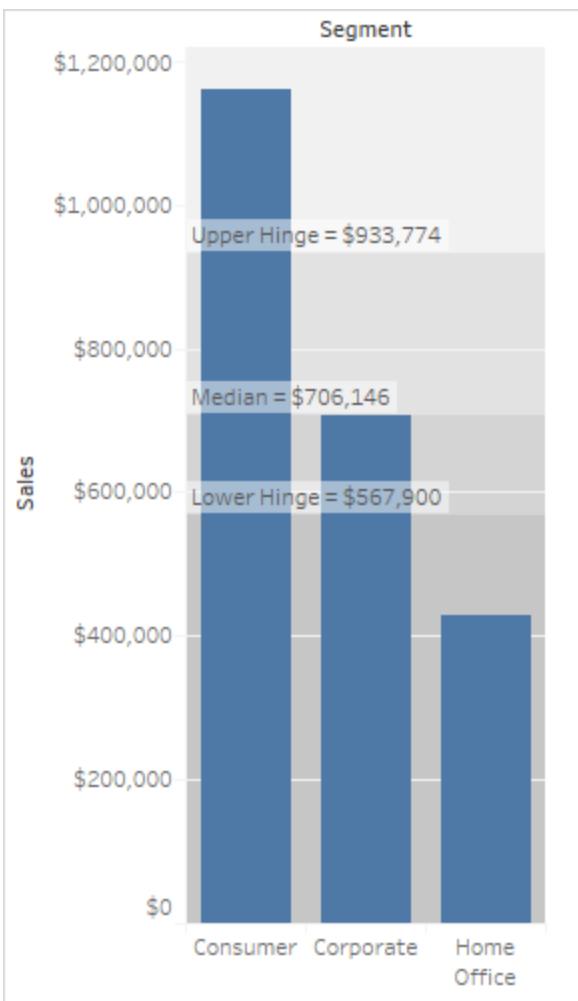
- **Line** - adds a line at a constant or computed value on the axis. Computed values can be based on a specified field. You can also include confidence intervals with a reference line. See [Adding Reference Lines](#) on page 953.



- **Band** - shades an area behind the marks in the view between two constants or computed values on the axis. See [Adding Reference Bands](#) on page 959.



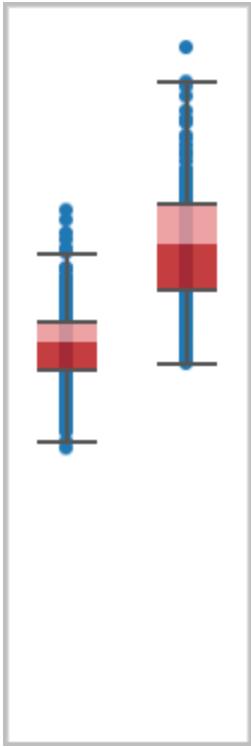
- **Distribution** - adds a gradient of shading to indicate the distribution of values along the axis. Distribution can be defined by percentages, percentiles, quantiles (as in the following image), or standard deviation.



Reference distributions can be used to create bullet charts. See [Adding Reference Distributions \(Bullet Graphs\) on page 963](#).

- **Box Plot** - adds a box plot that describes the distribution of values along the axis. Box plots show quartiles and whiskers. Tableau provides different box plot styles, and allows you to configure the location of the whiskers and other details.

See [Adding Box Plots on page 970](#).

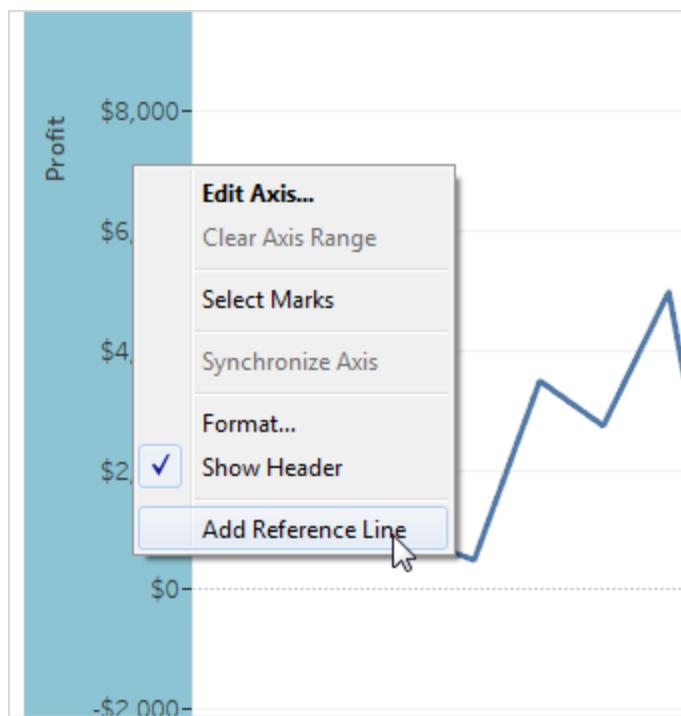


Adding Reference Lines

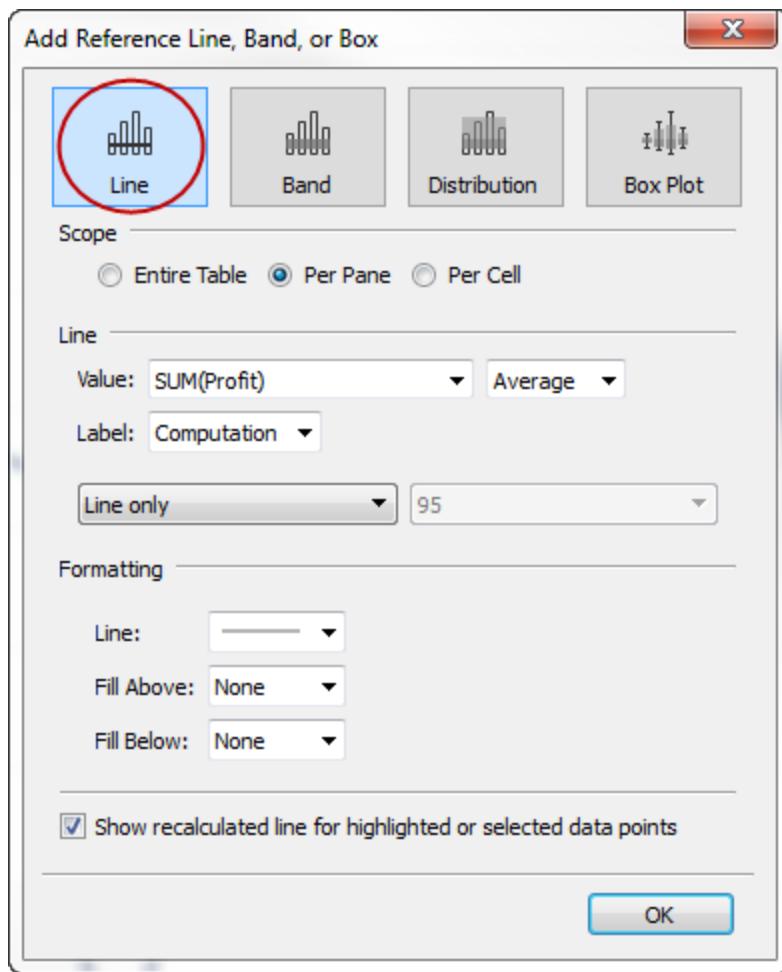
You can add a reference line to any continuous axis in the view.

To add a reference line:

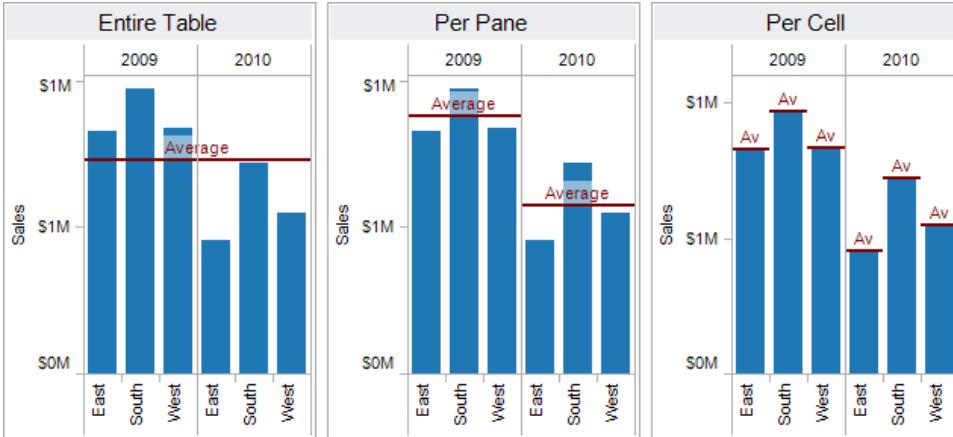
1. Right-click (control-click on Mac) on a quantitative axis and select **Add Reference Line**.



2. In the Add Reference Line, Band, or Box dialog box, select **Line**.



3. Specify the scope for the line:



Adds a reference line to the entire table across all panes.

Adds a reference line on a per pane basis. Computed reference lines are recalculated for each pane in the view.

Adds a reference line within each cell. Computed reference lines are recalculated for each cell in the view.

- In the **Value** list, select a field to use as the basis for your reference line. You can also select an existing parameter, or create a new parameter on the fly. See [Creating Parameters on page 989](#).

You cannot select a continuous field that isn't currently in the view as the basis for your reference line. If you want to use such a continuous field, do the following:

- Close the Add Reference Line, Band, or Box dialog box.
- Drag a continuous field from the **Data** pane to the Details target on the Marks card.
- Change the continuous field's aggregation if necessary.

This will not change the view, but it will allow you to use that continuous field as the basis for your reference band.

- Re-open the Add Reference Line, Band, or Box dialog box and resume at step 2, above.

- Select an aggregation. The aggregations that are displayed depend on the continuous field you select:

- Total** - places a line at the aggregate of all the values in either the cell, pane, or the entire view. This option is particularly useful when computing a weighted average rather than an average of averages. It is also useful when working with a calculation with a custom aggregation. The total is computed using the underlying data and behaves the same as selecting one of the totals option the Analysis menu.
- Sum** - places a line at the SUM of all the values in either the cell, pane, or entire

view.

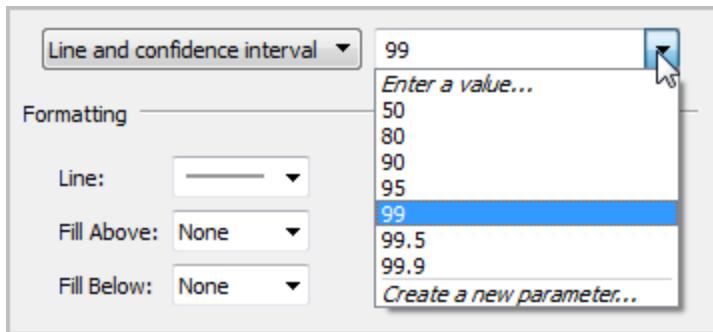
- **Constant**- places a line at the specified value on the axis.
- **Minimum** - places a line at the minimum value.
- **Maximum** - places a line at the maximum value.
- **Average** - places a line at the average value along the axis.
- **Median**- places a line at the median value.

6. Select how you want to label the line:

- **None** –select this option to not show a label for the reference line.
- **Value** – select this option to show a label corresponding to the line's value on the axis.
- **Computation** – select this option to display the name of the continuous field that is the basis for your reference line and any computation that is performed.
- **Custom** – select this option to build a custom label in the text box. You can use the menu to the right of the text box to insert values such as the computation or the value. You can also type text directly into the box, so you could create a value such as <Field Name> = <Value>.

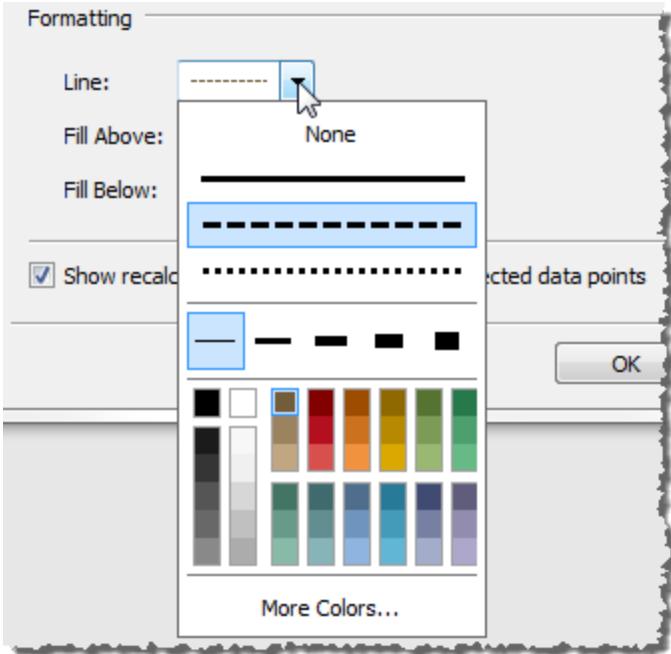
7. Specify whether to display the line with a confidence interval, just the line, or just the confidence interval.

Confidence interval distribution bands shade the region in which the population average will fall n of the time, where n is the value you select in the drop-down on the right.

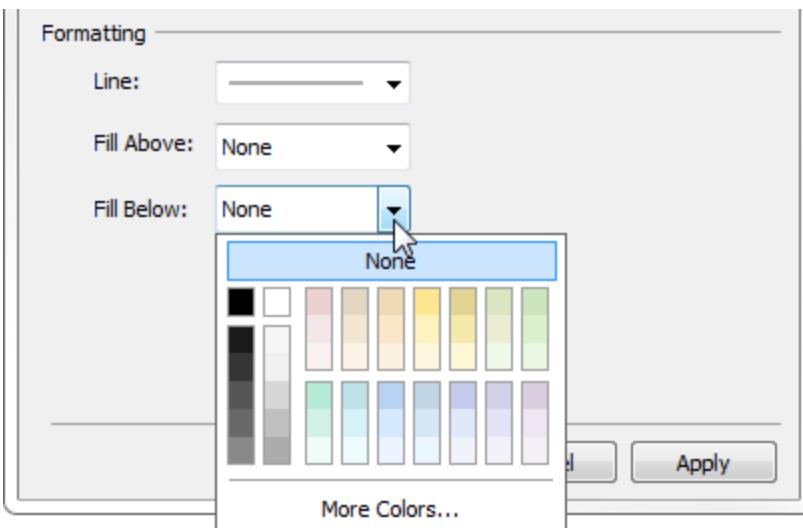


The higher the value you select, the wider the bands will be.

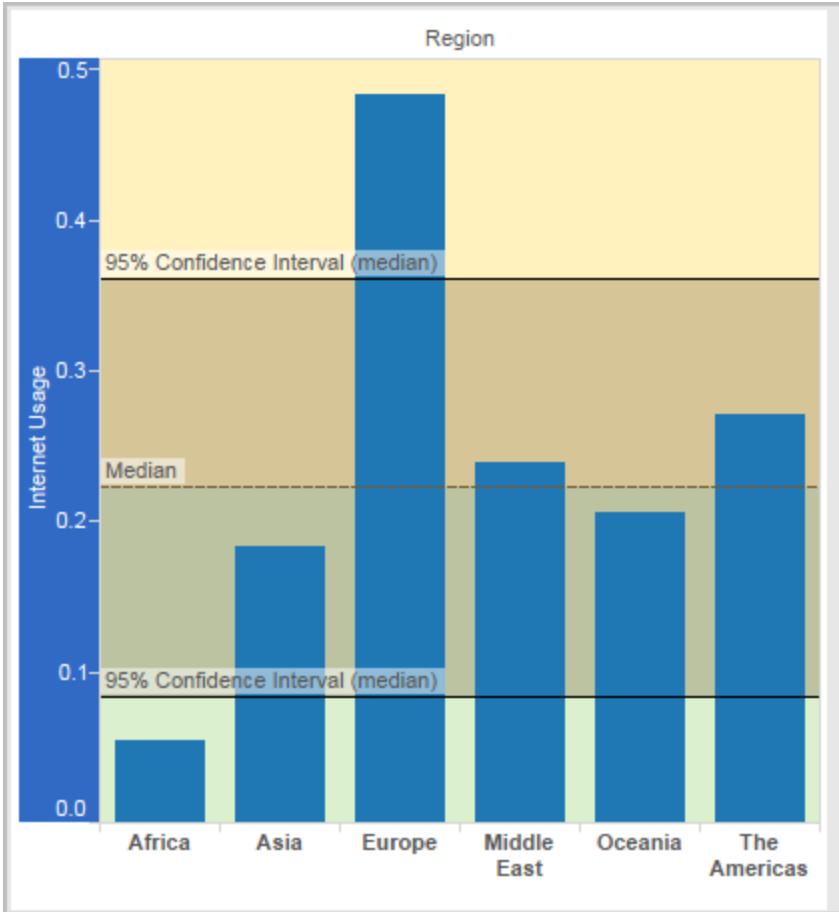
8. Optionally, specify formatting options for the line.



9. Optionally, add a fill color above and below the line.



When you are displaying a line and a confidence interval, the shading will be darker within the confidence interval, and lighter beyond it:



When you are displaying a confidence interval without a line, the fill colors are disregarded, though your settings are retained and then applied if you decide later to show a line.

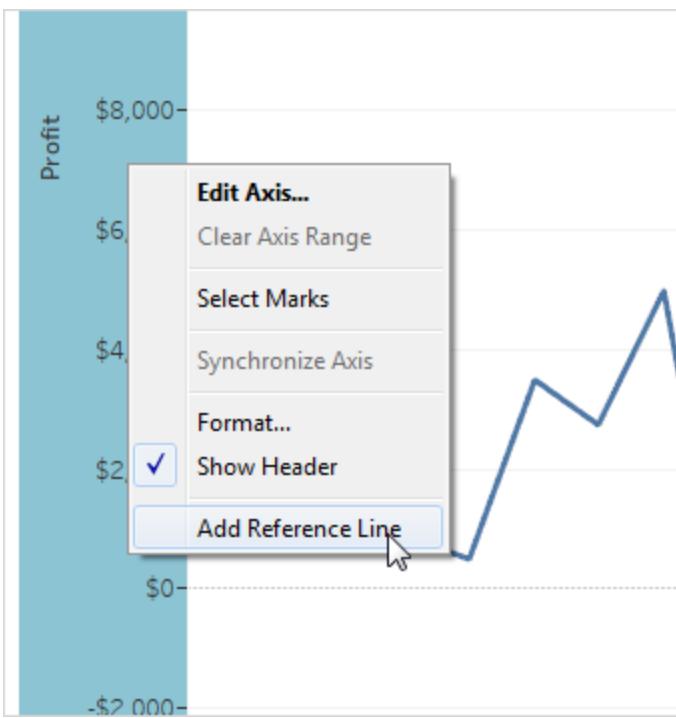
10. Specify whether to **Show recalculated line for highlighted or selected data points**.
For more information, see [Marks and Data Analysis](#) on page 562

Adding Reference Bands

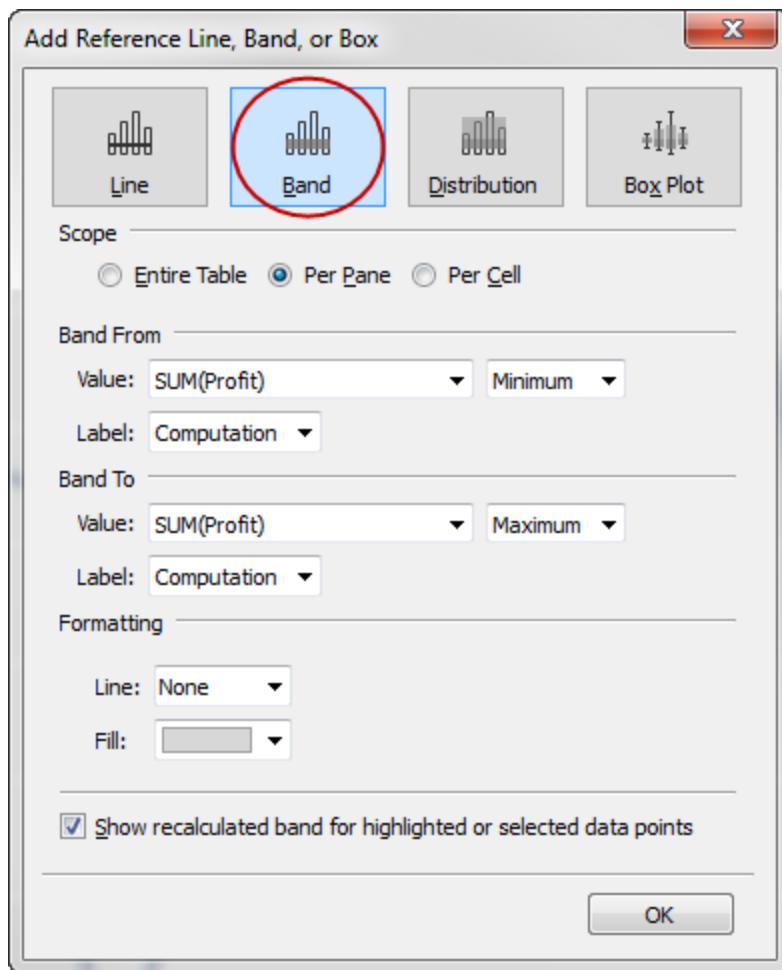
Reference bands are shaded areas behind the marks in the view between two constant or computed values on the axis. You can add reference bands to any continuous axis in the view.

To add a reference band:

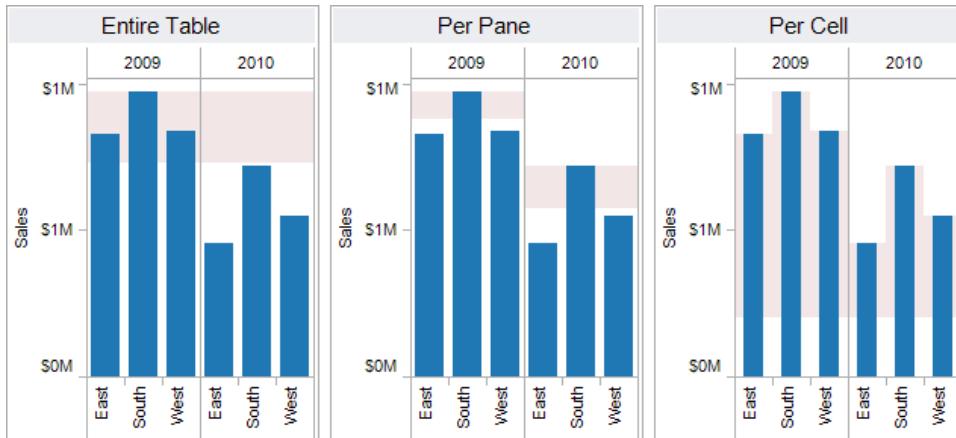
1. Right-click (Control-click on Mac) on a quantitative axis and select **Add Reference Line**.



2. In the Add Reference Line, Band, or Box dialog box, select **Band**.



3. Specify the scope for the bands:



Adds a reference band to the entire table across all panes.

Adds a reference band on a per pane basis. Computed reference bands are recalculated for each pane in the view.

Adds a reference band within each cell. Computed reference bands are recalculated for each cell in the view.

- Select two continuous fields to use as the basis for your reference band, one in the **Band From** area and one in the **Band To** area. You can also select an existing parameter, or create a new parameter on the fly. Just be sure not to select the same continuous field and aggregation in both areas.

You cannot select a continuous field that isn't currently in the view as the basis for your reference band. If you want to use such a continuous field, do the following:

- Close the Add Reference Line, Band, or Box dialog box.
- Drag a continuous field from the **Data** pane to the Details target on the Marks card.
- Change the continuous field's aggregation if necessary.

This will not change the view, but it will allow you to use that continuous field as the basis for your reference band.

- Re-open the Add Reference Line, Band, or Box dialog box and resume at step 2, above.

- Select a computation, one for the **Band From** value and one for the **Band To** value. The aggregations that are displayed depend on the continuous field you select:

- Total** - extends the band to a value that is at the aggregate of all the values in either the cell, pane, or the entire view. This option is particularly useful when computing a weighted average rather than an average of averages. It is also useful when working with a calculation with a custom aggregation. The total is computed using the underlying data and behaves the same as selecting one of the totals option the Analysis menu.
- Sum** - extends the band to a value that is at the SUM of all the values in either the

cell, pane, or entire view.

- **Constant**- extends the band to a value that is at the specified value on the axis.
- **Minimum** - extends the band to a value that is at the minimum value.
- **Maximum** - extends the band to a value that is at the maximum value.
- **Average** - extends the band to a value that is at the average value along the axis.
- **Median**- extends the band to a value that is at the median value.

6. Select how you want to label the band:

- **None** –select this option to not show a label for the reference band.
- **Value** – select this option to show a label corresponding to the band's value on the axis.
- **Computation** – select this option to display the name of the continuous field that is the basis for your reference band and any computation that is performed.
- **Custom** – select this option to build a custom label in the text box. You can use the menu to the right of the text box to insert values such as the computation or the value. You can also type text directly into the box, so you could create a value such as <Field Name> = <Value>.

7. Format the reference band. You can mark the two values with a line or select a shading color for the band.



8. Click **OK**.

9. Choose an option for **Show recalculated band for highlighted or selected data points**. For more information, see [Marks and Data Analysis on page 562](#)

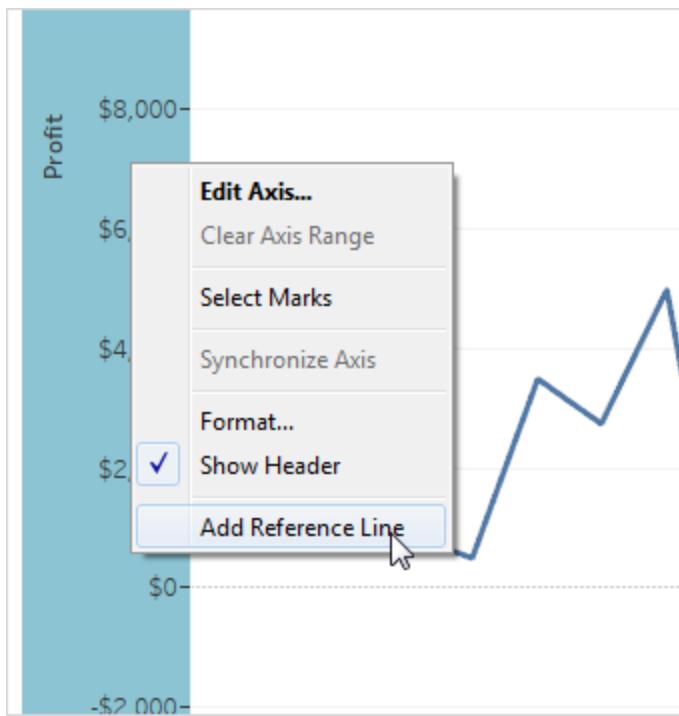
Adding Reference Distributions (Bullet Graphs)

Reference distributions are a special type of reference band. A reference distribution adds a gradient of shading to indicate the distribution of values along the axis. Distributions can be defined by confidence interval, percentages, percentiles, quantiles, or standard deviation. In addition to the shading, you can add a line to mark a constant or computed value along the axis.

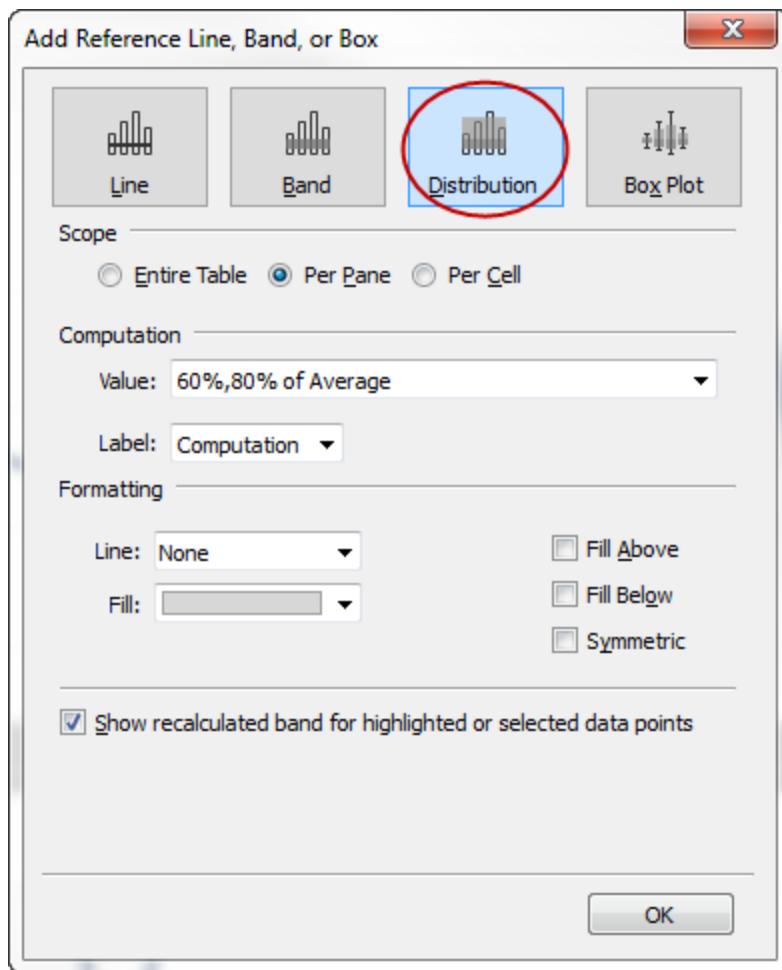
Basic Reference Distributions

To add a reference distribution:

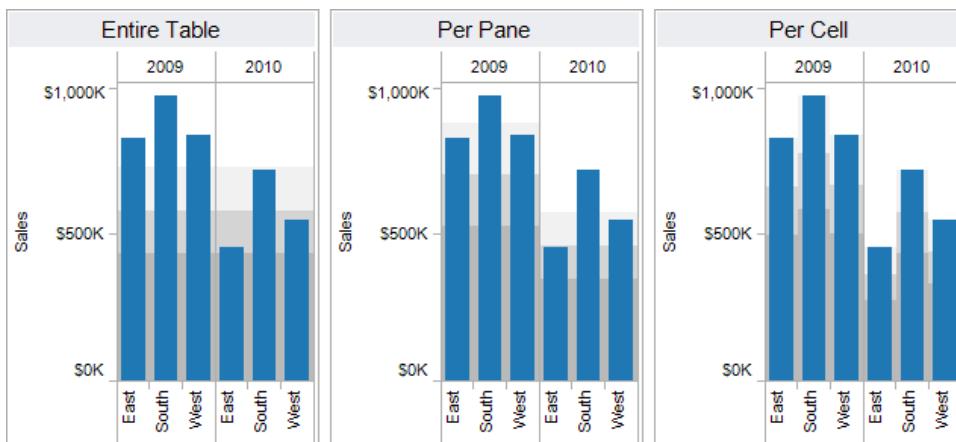
1. Right-click (Control-click on a Mac) on a quantitative axis and select **Add Reference Line**.



2. In the Add Reference Line, Band, or Box dialog box, select **Distribution**.



3. Select one of the following scopes:



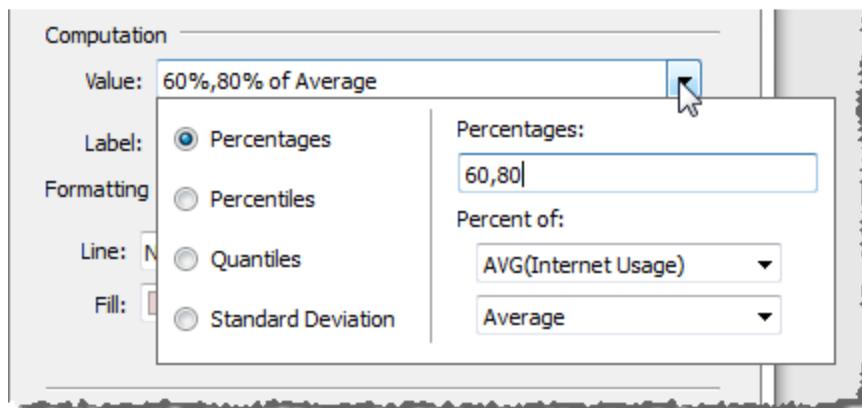
Adds a reference distribution to the entire table across all panes.

Adds a reference distribution on a per pane basis. Computed distributions are recalculated for each pane in the view.

Adds a reference distributions within each cell. Computed distributions are recalculated for each cell in the view.

4. Select the computation that will be used to create the distribution:

- **Percentages** - shades the interval between which lie specified percentages of values. Use a comma to separate multiple percentage values (for example, 60, 80).



- **Percentiles** - shades intervals at the specified percentiles. For a single percentile line, choose a single value from the **Percentiles** drop-down list. For shading bands, choose *Enter one or more values* from the Value drop-down list, and then enter two or more numerical values (for example, 60, 80 or 25, 50, 75).
- **Quantiles** - breaks the view into the specified number of tiles using shading and lines. When you select this computation, you must also specify the number of tiles (from 3 to 10, inclusive). For example, if you select 3, Tableau calculates the boundaries between the first, second and third terciles by calling the general

quantile function and asking for the 33.33 and the 66.66 quantiles. It then shades the three terciles differently.

Tableau uses estimation type 7 in the R standard to compute quantiles and percentiles.

- **Standard Deviation** - places lines and shading to indicated the specified number of standard deviations above and below the mean. When you select this option you must specify the factor, which is the number of standard deviations and whether the computation is on a sample or the population.
5. Specify formatting options. You can format the lines at the distribution boundaries (for example, style, thickness, and color) as well as the fill gradient. Select from a list of predefined gradients. Select Symmetric to use a single color instead of a gradient. You can also specify whether to add additional shading above and below the defined distribution.
 6. Specify whether to **Show recalculated band for highlighted or selected data points**. For more information, see [Marks and Data Analysis on page 562](#)

Bullet Graphs

Reference distributions can also be used to create bullet graphs. A bullet graph is a variation of a bar graph developed to replace dashboard gauges and meters. The bullet graph is generally used to compare a primary measure to one or more other measures in the context of qualitative ranges of performance such as poor, satisfactory, and good. You can create a bullet graph by adding two reference lines: a distribution to indicate the qualitative ranges of performance, and a line to indicate the target.

1. Select one or more dimensions, and two measures in the **Data** pane. The bullet graph will compare measure values. For example, budget vs. actual; actual vs. target; etc. Select multiple fields in the **Data** pane by holding down the Ctrl key as you click fields.

Data Analytics

Sample - Superstore

Dimensions

- Customer
 - Customer Name
 - Segment
- Order
 - Order Date
 - Order ID
 - Ship Date
 - Ship Mode
- Location
 - Country
 - State
 - City
 - Postal Code
- Product
 - Category
 - Sub-Category
 - Manufacturer
 - Product Name
 - Profit (bin)
 - Region
 - Measure Names

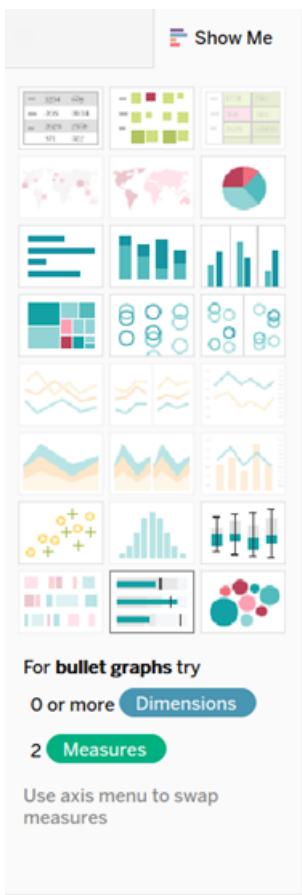
Measures

- Discount
- Profit
- Profit Ratio
- Quantity
- Sales
- Latitude (generated)
- Longitude (generated)
- Number of Records
- Measure Values

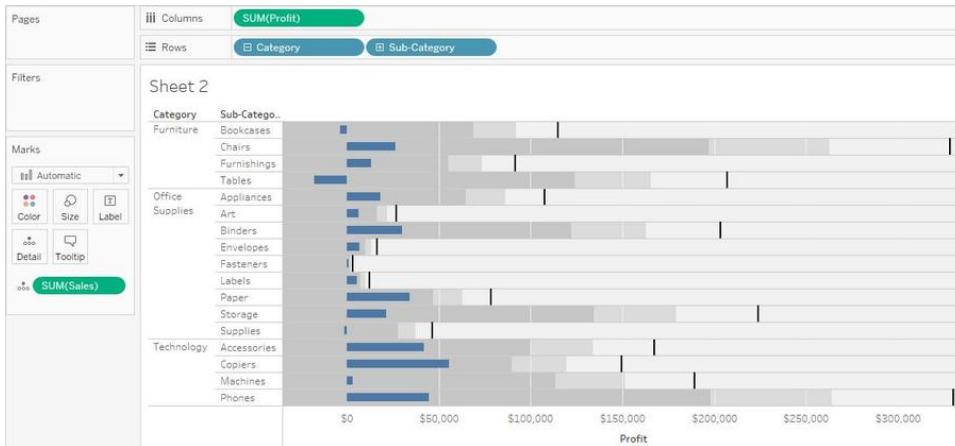
2. Click the **Show Me** button in the toolbar.

Show Me

3. Select **Bullet Graph** in the Show Me pane.



Two reference lines are added. By default, Tableau adds a reference distribution that is defined at 60% and 80% of the Average of the measure on Detail. It also adds a reference line that marks the Average of that same measure. The other measure is placed on the Rows shelf.



To quickly swap the two measures, right-click (Control-click on a Mac) on the continuous axis and choose **Swap Reference Line Fields**.

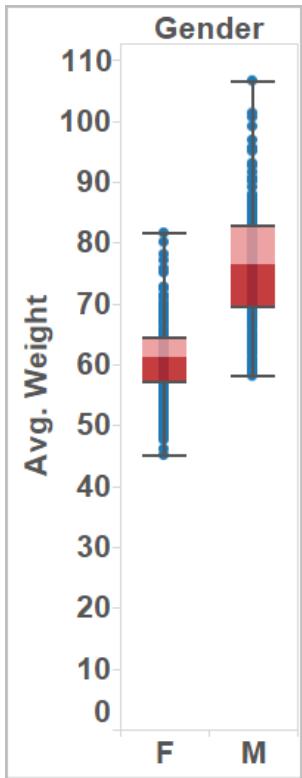
When you use ShowMe to add a bullet graph, Tableau adds a reference line and a reference distribution. You can edit either of these to change its definition. For example, you may want to add 100%, or draw a line at a constant value. Right-click (Control-click on a Mac) on the continuous axis and choose **Edit Reference Line**. You'll then see a sub-menu listing all lines, bands, and distributions currently in the view.

Adding Box Plots

Use box plots, also known as box-and-whisker plots, to show the distribution of values along an axis.

Boxes indicate the middle 50 percent of the data (that is, the middle two quartiles of the data's distribution).

You can configure lines, called *whiskers*, to display all points within 1.5 times the interquartile range (in other words, all points within 1.5 times the width of the adjoining box), or all points at the maximum extent of the data, as shown in the following image:



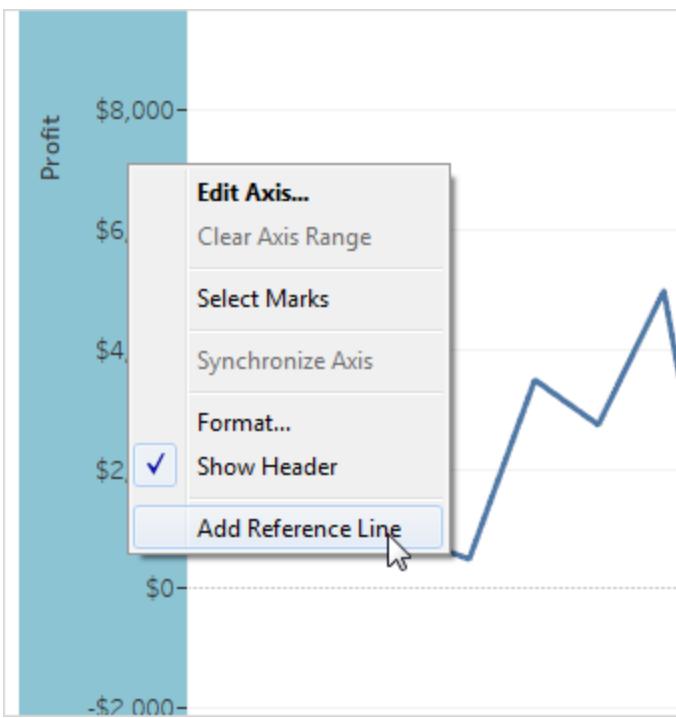
Boxplots are also available from the Show Me pane when you have at least one measure in the view:



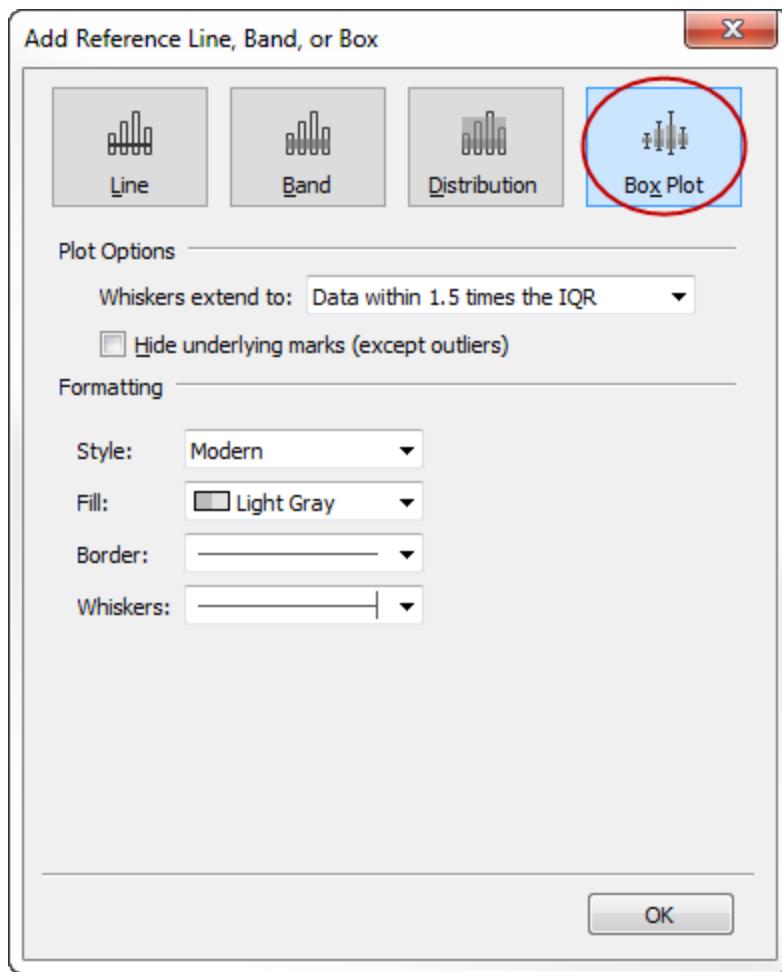
For information on Show Me, see [Show Me](#) on page 595

To add a box plot:

1. Right-click (Control-click on a Mac) on a quantitative axis and select **Add Reference Line**.



2. In the Add Reference Line, Band, or Box dialog box, select Box Plot.

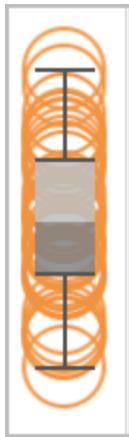


3. Under Plot Options, specify placement for the whiskers:
 - **Data within 1.5 times the IQR** - places whiskers at a location that is 1.5 times the interquartile range—that is, 1.5 times further out than the width of the adjoining box. This is also known as a schematic box plot.
 - **Maximum extent of the data** - places whiskers at the farthest data point (mark) in the distribution. This is also known as a skeletal box plot.
4. Specify whether to **Hide underlying marks (except outliers)**—that is, whether to hide all marks except those beyond the whiskers.
5. Configure the appearance of the plot by selecting a **Style**, **Fill**, **Border**, and **Whiskers**.

Show Me Vs. Add Reference Line, Band, or Box

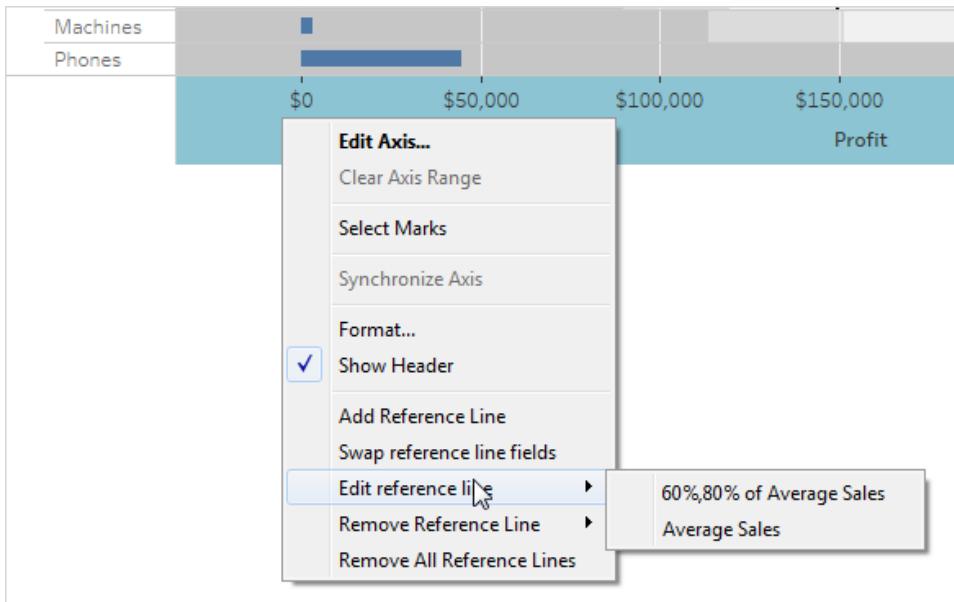
The difference between adding a box plot using Show Me and adding a box plot using Add Reference Line is that with Show Me, the box plot is your visualization, whereas with Add Reference Line, Band, or Box, you are adding a box plot to an existing visualization. For

example, you could create the following view by first selecting a circle view in Show Me, and then adding a box plot from Add Reference Line:

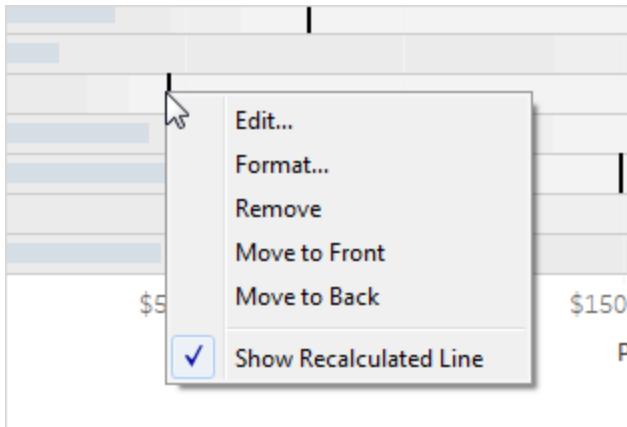


Editing Reference Lines, Bands, Distributions, or Boxes

After you've added a reference line, band, or box, you can edit the definition. To do this, right-click (Control-click on a Mac) the continuous axis and select **Edit Reference Line**. If there are multiple reference lines, bands, or boxes in the view, use the additional menu to select the one you want to edit.



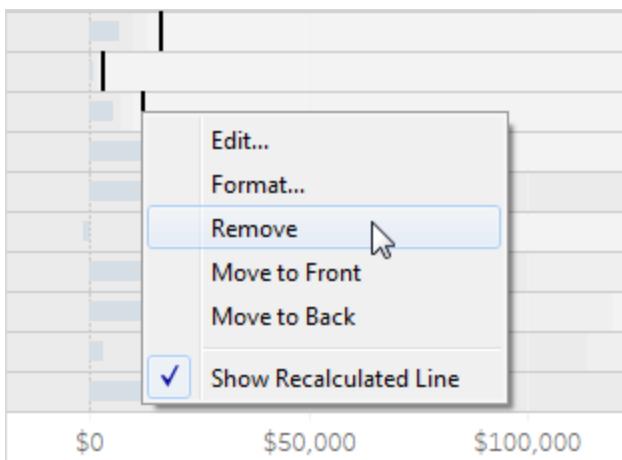
When you have multiple reference lines, bands, or boxes, you may want to change the order they are drawn in the view. To reorder a reference line, right-click (Control-click on a Mac) the line and select **Move to Front** or **Move to Back**.



Removing Reference Lines, Bands, Distributions or Boxes

You can remove an individual reference line, band, or box, or remove them all at once.

To remove an individual reference line, band, or box, right-click (Control-click on a Mac) the reference line, band, or box in the view and select **Remove**. If you are removing a reference band or distribution that doesn't include a line, right-click at the beginning or end of the shaded area. In distributions, you can also right-click between the different shades in the gradient.



To remove all reference lines, bands, or boxes at once, click on a line or on the outer edge of a band and choose **Remove**.

Totals

You can automatically compute grand totals and subtotals for the data in a view. By default Tableau uses the underlying data to compute totals. However, if you are using a multidimensional data source you can specify whether to compute the total on the server using the underlying data or locally using the data that you see in the table. In Tableau, multidimensional data sources are supported only in Windows.

To add totals to the view:

1. On the left side of the workspace, click the **Analytics** pane.
2. Drag and drop **Totals** onto one of the options in the Add Totals dialog box that appears.

The screenshot shows the Tableau interface with the Analytics pane open. In the Marks shelf, the 'Totals' button is selected. A callout from the 'Totals' button points to the 'Add Totals' dialog box, which is displayed over the data grid. The 'Totals' option in the dialog box is highlighted with a red box. The data grid shows sales data for various categories like Binders, Bookcases, Chairs, etc., with columns for Profit and Loss.

		\$-1,044	\$11,268	\$3,901	\$16,097
Binders					
Bookcases					
Chairs					
Copiers					
Envelopes					
Fasteners					
Furnishings					
Labels					
Machines					
Paper					
Phones					
Storage					
Supplies					
Tables					

You can also add grand totals and subtotals to the view using the Analysis menu. To do so, select **Analysis > Totals**, and then select to add row or column grand totals, or subtotals.

Local vs. Server Computation

If you are using a multidimensional data source, you can specify whether the computation is performed on the server using the underlying data in the data source, or locally in Tableau using the data that you see in the table.

Note: In Tableau, multidimensional data sources are supported only in Windows.

The default setting is to compute all totals on the server if you are connected to a Microsoft Analysis Services data source, and locally if you are connected to an Essbase data source using the aggregation specified in the cube. However, there are cases when it is not possible for the server to compute the expected subtotals due to filtering or perfect pivoting.

For example, let's say you have a view showing the sales of Amaretto, Columbian, and Decaf Irish Cream coffees. Then you filter the view to only show Amaretto and Columbian coffee sales. When you turn on subtotals for the **Product Type** field, so you can see the total sales for all Coffees, one of the following will happen:

- If the total can be computed using the filter, the correct total will display. In this case Tableau would compute the total sales for all Coffees.
- If the total cannot be computed using the filter, the totals cells in the view will be empty. In this case you would want to specify a local computation that only includes the values you see in Tableau.

To specify a local computation:

1. Select **Total Using** on the context menu for the measure you are using to calculate the subtotals.

If you are not displaying totals, this menu item does not appear. See [Show Grand Totals](#) on the next page.

Then select an aggregation.

The screenshot shows a Tableau interface with a context menu open over a data table. The menu path is: **SUM(Profit) ▾** → **Total Using (Automatic)** → **Automatic**. The menu items under **Total Using (Automatic)** are: **Automatic** (selected), **Sum**, **Average**, **Minimum**, **Maximum**, and **Hide**.

	Fasteners
Labels	\$12
Paper	\$23
Storage	\$23
Supplies	(\$1)
Total	\$12
Accessories	\$43
Copiers	\$53
Machines	\$33
Phones	\$43
Total	\$146

2.

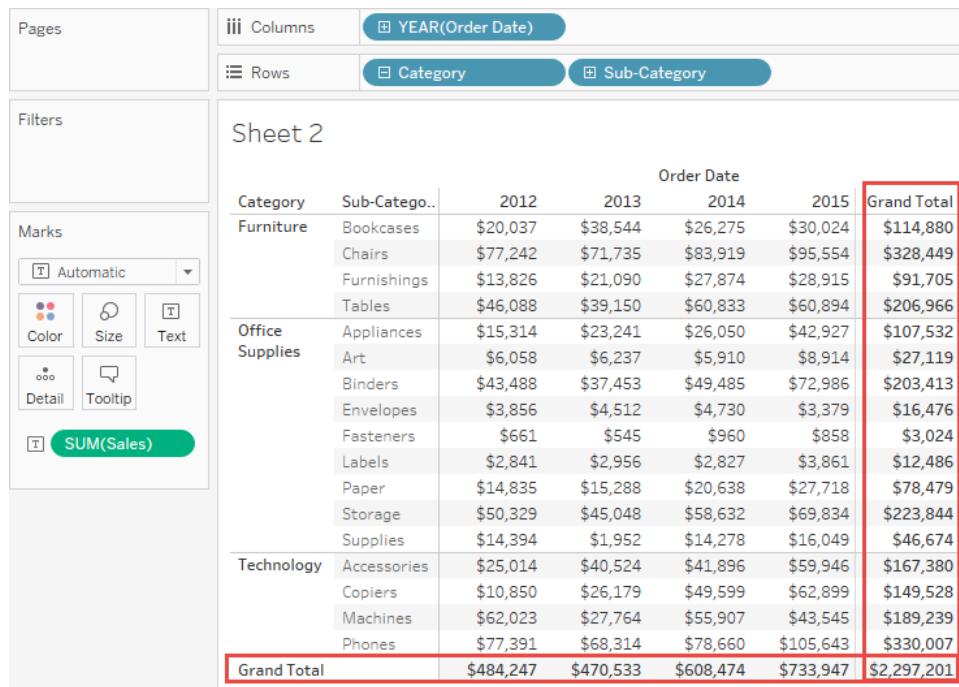
Hide tells Tableau to not show a total for the field.

An additional value, **Server**, may be available. Server computation is not always available and sometimes the totals will be blank for specific members in the view. When using server computation keep in mind the following information.

- Server computation is only available for ASO cubes.
- Server computation is not available for dynamic hierarchies. This means that if the members in the view are part of a dimension or hierarchy that is tagged as dynamic, they cannot be included in the set of values you are using to compute the totals and will show up as blank in the view.
- If you are computing totals for a calculated field whose formula makes assumptions about other calculated members at different levels in the hierarchy, the totals will display as blank in the view.

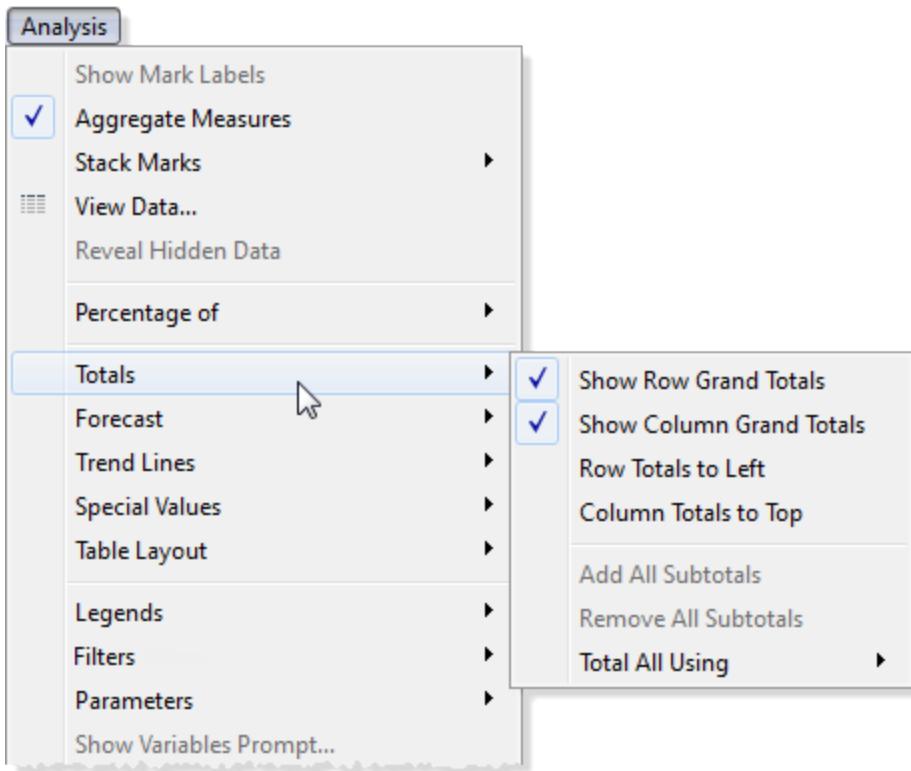
Grand Totals

Any view in Tableau can include grand totals. For example, in a view showing the total sales for each department and category for four years, you can turn on grand totals to also see the totals across all products and all years.



Show Grand Totals

You can calculate grand totals by selecting one of the **Grand Totals** options on the **Analysis > Totals** menu. The grand totals are added as an additional row or column to your table.

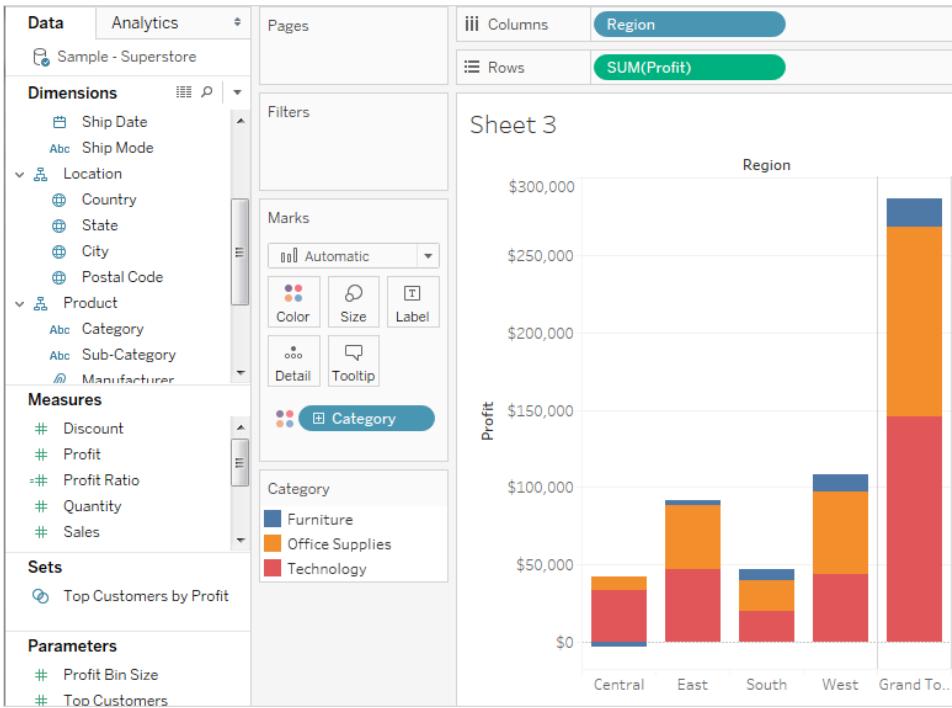


You can also drag total in from the **Analytics** pane. See [Analytics Pane](#) on page 135.

The following rules dictate whether you can turn on grand totals:

- The view must have at least one header – Headers are displayed whenever you place a dimension on the **Columns** shelf or the **Rows** shelf. If column headers are displayed, you can calculate grand totals for columns. If row headers are displayed, you can calculate grand totals for rows.
- Measures must be aggregated – The aggregation determines the values displayed for the totals. See [Grand Totals and Aggregations](#) on page 985 for more information.
- Grand Totals cannot be applied to continuous dimensions.

You can also display totals for graphical views of data. In the figure below, only column totals are calculated because the table contains only column headers.



Note: By default, totals are computed on the server if you are connected to a Microsoft Analysis Services data source, and locally if you are connected to an Essbase data source, using the aggregation specified in the cube. Refer to [Local vs. Server Computation on page 977](#) to learn more.

Options for Calculating Grand Totals

When you first turn on grand totals (see [Show Grand Totals on page 979](#)), the totals are computed using disaggregated data in the underlying data source. Consider the following view:

Sheet 4

Region

Ship Mode	Category	Central	East	South	West	Grand T..
First Class	Furniture	\$329	\$306	\$390	\$348	\$339
	Office Supplies	\$111	\$120	\$120	\$98	\$111
	Technology	\$300	\$524	\$292	\$571	\$463
Same Day	Furniture	\$316	\$367	\$225	\$338	\$329
	Office Supplies	\$79	\$103	\$82	\$89	\$89
	Technology	\$320	\$689	\$767	\$620	\$614
Second Class	Furniture	\$332	\$347	\$402	\$390	\$366
	Office Supplies	\$127	\$140	\$151	\$142	\$139
	Technology	\$381	\$312	\$604	\$345	\$389
Standard Class	Furniture	\$348	\$356	\$336	\$351	\$349
	Office Supplies	\$119	\$115	\$123	\$116	\$118
	Technology	\$449	\$522	\$500	\$381	\$457

The average that you see at the right of the first row under Grand Total is \$339. But if you compute the average for the four values you see in the row (\$329, \$306, \$390, \$348), the result comes to \$343.25, not \$339. The discrepancy is due to the fact that Tableau is averaging the data in the underlying data source. In this case there are more than four numbers to average, perhaps many more. The result is derived by averaging all values that have the properties Ship Mode = First Class, Category = Furniture, without regard to region.

To see totals that correspond to the numbers you see in your view requires two averaging operations in Tableau: first, values for the individual marks (or cells) in the view must be derived—for example, by averaging all values that have the properties Ship Mode = First Class, Category = Furniture, and Region = Central. Then the results for each region must be derived, by averaging the results for the individual marks. Fortunately, you do not need to perform two operations. To display a result of this type, from the **Analysis** menu choose **Totals > Total All Using > Average**. Now the average is performed on the values you see, and not on the disaggregated data in the data source:

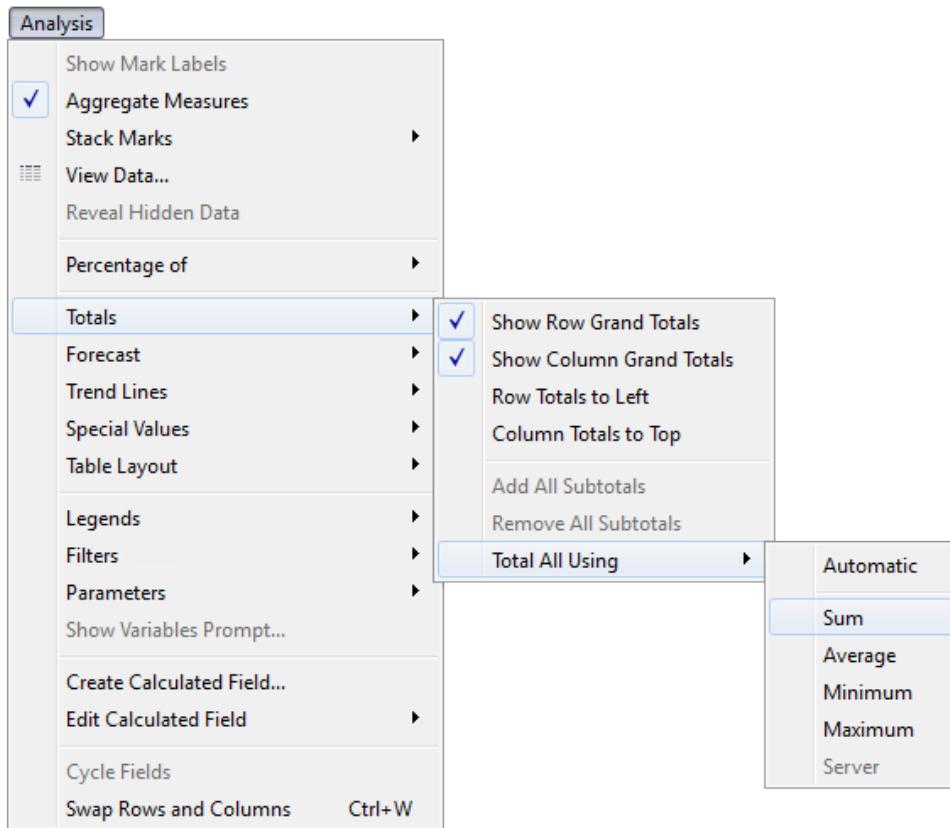
Pages		Region						
		Ship Mode Category						
Filters		Sheet 4						
Marks		Ship Mode	Category	Central	East	South	West	Grand T..
		First Class	Furniture	\$329	\$306	\$390	\$348	\$343
			Office Supplies	\$111	\$120	\$120	\$98	\$112
			Technology	\$300	\$524	\$292	\$571	\$422
		Same Day	Furniture	\$316	\$367	\$225	\$338	\$311
			Office Supplies	\$79	\$103	\$82	\$89	\$88
			Technology	\$320	\$689	\$767	\$620	\$599
		Second Class	Furniture	\$332	\$347	\$402	\$390	\$367
			Office Supplies	\$127	\$140	\$151	\$142	\$140
			Technology	\$381	\$312	\$604	\$345	\$410
		Standard Class	Furniture	\$348	\$356	\$336	\$351	\$348
			Office Supplies	\$119	\$115	\$123	\$116	\$118
			Technology	\$449	\$522	\$500	\$381	\$463

This kind of total is sometimes referred to as a two-pass total, because the average you see in the grand total column is aggregated twice--once to derive the column or row value, and then again across column or rows to derive the grand total.

Configuring Grand Totals

When grand totals are turned on in the view (either row grand totals or column grand totals), you can specify how totals should be computed.

To configure grand totals, from the **Analysis** menu choose **Totals > Total All Using** to display a submenu:



When you choose **Current Aggregation** from the submenu, totals are computed as they always were in earlier versions of Tableau: with column and row totals based on the configured aggregation for the field. In this case, computations are based on the underlying data, which is disaggregated, and not on the data in the view. See [Options for Calculating Grand Totals on page 981](#). And for details on how Tableau computes totals using the current aggregation, see [Grand Totals and Aggregations on the next page](#).

You can quickly set the aggregation for a grand totals by clicking the column or row header and choosing an aggregation from the drop-down list on the tooltip:

Total	Appliances	Art	Binders
\$60,7	+ ↕ ↕	Automatic	Remove
\$31,6	53 items selected	Automatic	14,897
\$17,5		Sum	
Total		Average	\$3,099
\$15,1		Minimum	\$413
\$9,677	\$4,019	Maximum	\$6,321
\$6,904	\$1,197	Server	\$235
\$6,343	\$1,454		
\$2,395			
\$0,150	\$0,000		\$4,897

When you choose any of the other values (**Sum**, **Average**, **Minimum**, or **Maximum**), all totals are computed using the selected aggregation. The computations are performed on the aggregated data you see in the view.

You can also configure an aggregation for a specific field in the view. To do this, right-click (Control-click on a Mac) a field in the view and choose **Total Using**. With the first value, **Automatic**, Tableau uses the currently configured aggregation for the field. If you choose any of the other values (which are other types of aggregation), that aggregation is used for totals.

Grand Totals and Aggregations

When you turn on grand totals, the initial values are computing using the current aggregation for the fields in the view. In this case, totals are based on the underlying data rather than the data in the view.

For example, if you are totaling the **SUM(Profit)** for several products, the grand total will be the sum of the sums of profit. For aggregations such as SUM, you can easily verify the grand total because a summation of a group of sums is still a summation. However, be aware that your results may be unexpected when using other aggregations, especially custom aggregations.

For details, See [Configuring Grand Totals on page 983](#). You can verify any calculation such as an aggregation or a grand total by viewing the underlying disaggregated data.

The following table summarizes the standard aggregations and the grand totals that are calculated by default when, from the Analysis menu, **Totals > Total All Using** is set to the default value **Automatic**.

Aggregation	Calculation Description
Sum	Shows the sum of the values shown in the row or column.
Average	Shows the average of the values shown in the row or column.
Median	Shows the median for the values shown in the row or column.
Count; Count Distinct	Shows how many values or distinct values are displayed in the rows and columns in the view.
Minimum	Shows the minimum value shown in the row or column.
Maximum	Shows the maximum value shown in the row or column.
Percentile	Shows the average percentile for all values shown in the row or column.
Standard Deviation	The grand total using standard deviation is the standard deviation of the values shown in the row or column.

Aggregation	Calculation Description
Variance	The grand total using variance is not the variance of the rows and columns in which they reside but rather of the underlying data behind the row or column.

Subtotals

Any data view in Tableau can include subtotals. For example, you may have a view containing the total sales for two product types broken down by specific products. In addition to seeing the sales for each product you may want to see the total sales for each product type.

You can add subtotals for all fields or for selected fields. To add subtotals for all fields, choose **Totals > Add All Subtotals** from the **Analysis** menu. You then have the option of turning subtotals off for one or more individual fields. You do this by toggling subtotals off, as per the instructions below for turning subtotals off for individual fields.

To calculate subtotals for an individual field, right-click (Control-click on a Mac) the field in the view and choose **Subtotals** from the context menu.

The screenshot shows the Tableau Data View interface. On the left, there's a grid of data with columns labeled 'Category', 'Container', and 'Department'. The data includes categories like 'Appliances', 'Binders and Binder Accessories', 'Bookcases', 'Chairs & Chairmats', and 'Computer Peripherals', with various sub-categories and their corresponding containers and departments. On the right, a context menu is open over the 'Category' field. The menu items are: Filter..., Show Filter, Sort..., Format..., Show Header (with a checked checkmark), Include in Tooltip (with a checked checkmark), Edit Aliases..., Dimension, Attribute, Measure (with a dropdown arrow), Subtotals (which is highlighted with a blue selection bar), and Remove. At the bottom of the menu, there are two numerical values: '\$199,978' and '\$143,184'.

A check mark then appears next to Subtotals on the context menu.

You can quickly set the aggregation for subtotals by clicking the column or row header and choosing an aggregation from the drop-down list on the tooltip:

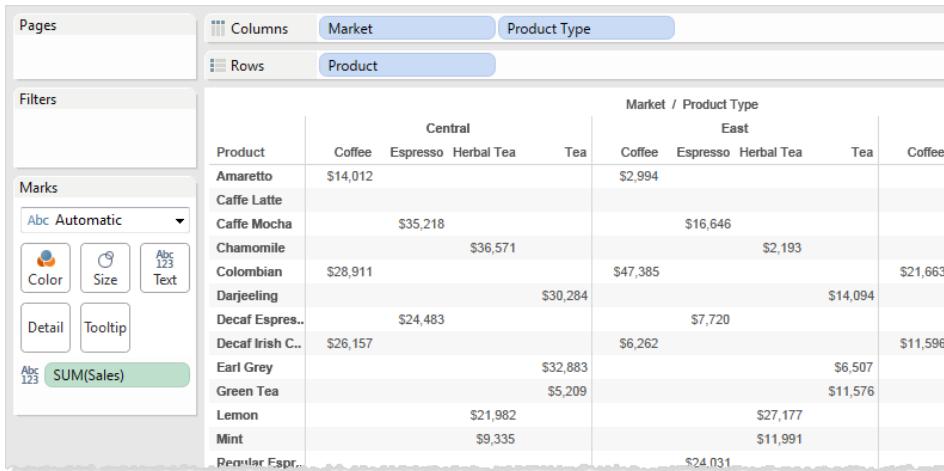
A screenshot of a Tableau interface showing a tooltip for aggregation settings. The tooltip is titled 'Automatic' and includes options: Automatic, Sum, Average, Minimum, Maximum, and Server. The 'Automatic' option is selected. The background shows a data view with columns for 'Total', 'Appliances', 'Art', and 'Binders'. The 'Total' column has a value of '\$60,7'. A dropdown menu is open over the 'Total' column header, showing 'Automatic' as the current selection.

When you turn on subtotals for a specific field, the totals will change based on where that field is in the view. Consider the following example. The view below shows the sales for different product types sold across four different markets. Each product type is broken down by specific products. In addition, subtotals are turned on so that the view shows the total sales for each product type.

A screenshot of a Tableau dashboard titled 'Pages'. The dashboard includes a 'Filters' section with a 'Marks' card containing 'Color', 'Size', 'Text', 'Detail', and 'Tooltip' buttons, and a 'SUM(Sales)' measure. The main view is a table titled 'Market' with columns: Product Type, Product, Central, East, South, and West. The table data is as follows:

Product Type	Product	Market			
		Central	East	South	West
Coffee	Amaretto	\$14,012	\$2,994	\$9,263	
	Colombian	\$28,911	\$47,385	\$21,663	\$30,352
	Decaf Irish Cream	\$26,157	\$6,262	\$11,596	\$18,233
	Total	\$69,080	\$56,641	\$33,259	\$57,848
Espresso	Caffe Latte		\$15,443	\$20,456	
	Caffe Mocha	\$35,218	\$16,646	\$14,166	\$18,874
	Decaf Espresso	\$24,483	\$7,720	\$15,381	\$30,578
	Regular Espresso		\$24,031		
	Total	\$59,701	\$48,397	\$44,990	\$69,908
Herbal Tea	Chamomile	\$36,571	\$2,193	\$11,183	\$25,631
	Lemon	\$21,982	\$27,177	\$14,494	\$32,273
	Mint	\$9,335	\$11,991		\$14,384
	Total	\$67,888	\$41,361	\$25,677	\$72,288
Tea	Darjeeling	\$30,284	\$14,094		\$28,773
	Earl Grey	\$32,883	\$6,507		\$27,382
	Green Tea	\$5,209	\$11,576		\$16,065
	Total	\$68,376	\$32,177		\$72,220

Now let's move the Product Type field from the **Rows** shelf to the **Columns** shelf. The view still shows the sales for four different product types; but now, the product types are broken down by market. Because subtotals were turned on for the Product Type, the subtotals are the sum of the sales completed in each market.



Note: By default subtotals are computed on the server if you are connected to an SSAS data source and locally if you are connected to an Essbase data source using the aggregation specified in the cube. That means if you are analyzing the average sales for each product, calculating the subtotals for each product type would result in the average sales of all products within that product type. This is not the case if you perform the computation locally instead of remotely.

Move Totals

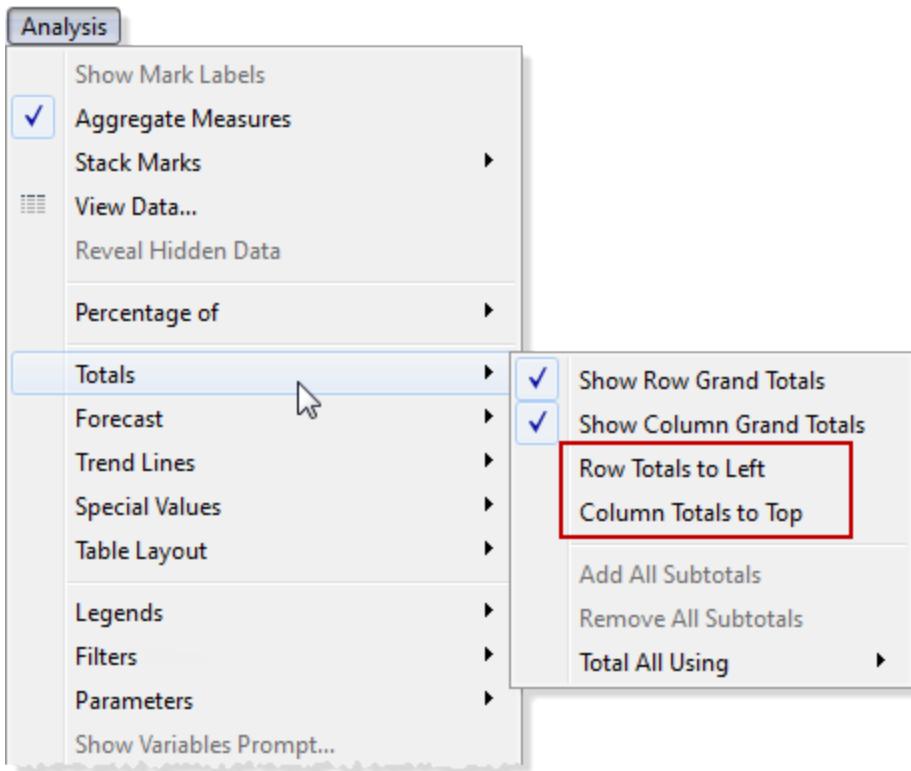
By default, row grand totals and subtotals appear on the right of the view, and column grand totals and subtotals appear at the bottom of the view. You can also select to display totals on the left or top of the view.

To move row totals to the left of the view:

- Select **Analysis > Totals**, and then select **Row Totals to Left**.

To move column totals to the top of the view:

- Select **Analysis > Totals**, and then select **Column Totals to Top**.



Parameters

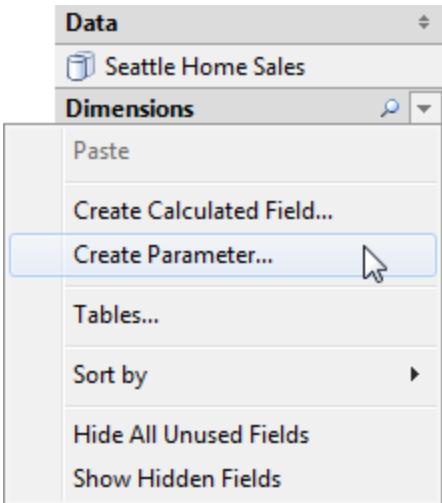
Parameters are dynamic values that can replace constant values in calculations, filters, and reference lines. For example, you may create a calculated field that returns true if Sales is greater than \$500,000 and otherwise return false. You can replace the constant value of “500000” in the formula with a parameter. Then using the parameter control you can dynamically change the threshold in your calculation. Alternatively, you may have a filter to show the top 10 products by profit. You can replace the fixed value “10” in the filter to by a dynamic parameter so you can quickly look at the top 15, 20, and 30 products.

Creating Parameters

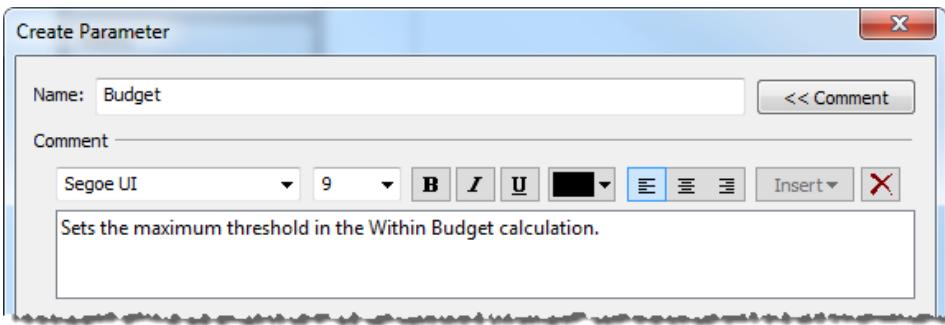
You can create a new parameter based on a selected field in the Data pane, or you can create a new parameter from any location where you can use a parameter (for example, as you add a reference line or create a filter). Follow the instructions below to create a new parameter from the Data pane.

1. In the **Data** pane, right-click (Control-click on a Mac) a field to base the parameter on and select **Create > Parameter**.

Alternatively, you can use the drop-down arrow in the upper right corner and select **Create Parameter**.

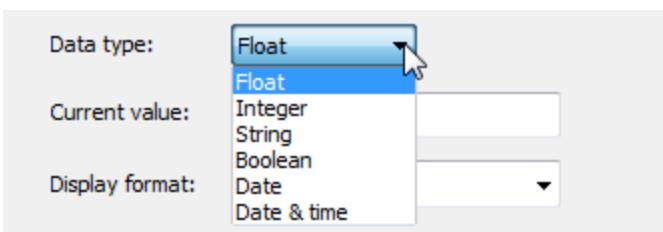


In the Create Parameter dialog box, give the field a **Name** and optionally write a **Comment** to describe the parameter.



2.

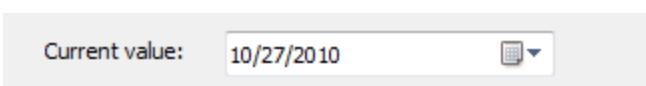
Specify the data type for the values it will accept:



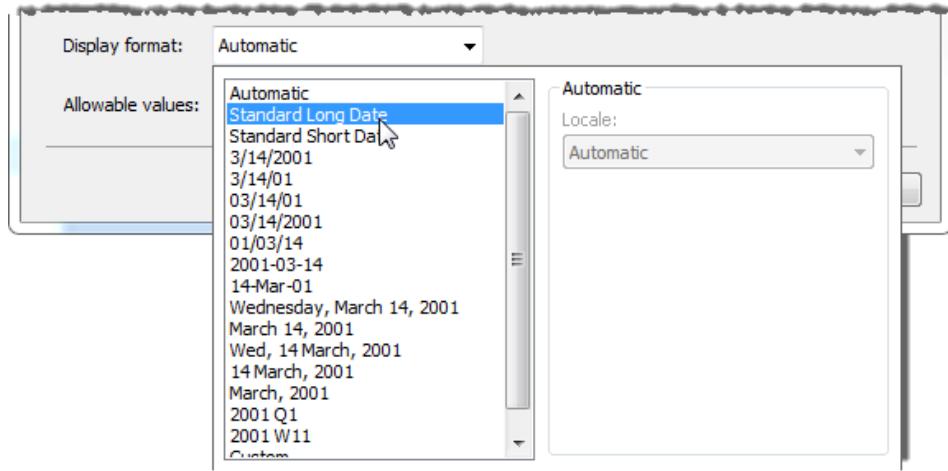
3.

Specify a current value. This is the default value for the parameter.

4.



Specify the display format to use in the parameter control.



5.

6. Specify how the parameter will accept values. You can select from the following options:

- **All** - the parameter control is a simple type in field.
- **List** - the parameter control provides a list of possible values for you to select from.
- **Range** - the parameter control lets you select values within a specified range.

The availability of these options is determined by the data type. For example, a string parameter can only accept all values or a list. It does not support a range.

If you select List, you must specify the list of values. Click in the left column to type a value. Each value can also have a display alias. You can copy and paste a list of values by clicking **Paste from Clipboard**. Alternatively you can add the members of a field as the list of values by selecting **Add from Field**.

Allowable values: All List Range

List of values

Value	Display As
1	Apples
2	Bananas
3	Oranges
4	Pears
Add	

Add from Parameter ►

Add from Field ►

Paste from Clipboard

Clear All

If you select **Range** you must specify a minimum, maximum and step size. For example, you can define a date range between January 1, 2010 and December 31, 2010, with the step size set to 1 month to create a parameter control that lets you select each month in 2010.

Range of values

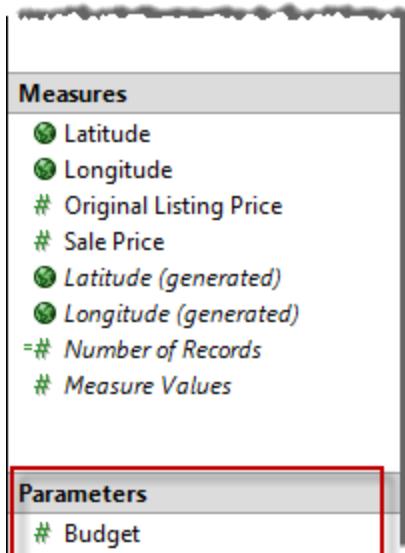
Minimum: 1/ 1/2010

Maximum: 12/31/2010

Step size: 1 Months

7. When finished, click **OK**.

The parameter is listed in the Parameters section at the bottom of the **Data** pane.



It is also available everywhere else you can use a parameter—for example, on the Top tab in the Filter dialog box, or in the Reference Line dialog box. Parameters are global across the workbook and can be used in any worksheet.

Editing Parameters

You can edit parameters from the Data pane or the parameter control. Follow the instructions below to edit a parameter:

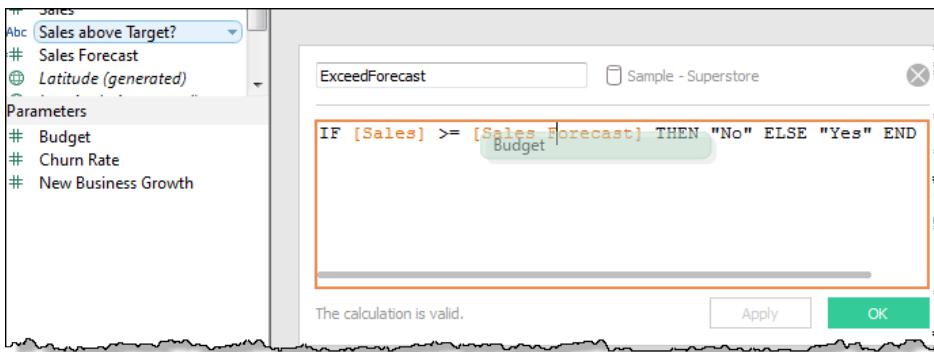
1. Do one of the following:
 - Right-click (Control-click on a Mac) the parameter in the **Data** pane and select **Edit**.
 - Select **Edit Parameter** on the parameter control card menu.
 2. In the Edit Parameter dialog box, make the modifications as necessary.
 3. When finished, click **OK**. The parameter is updated along with any calculations that use it.

To delete a parameter, right-click it in the Data pane and select **Delete**. Any calculated fields that use the deleted parameter will become invalid.

Using Parameters in Calculations

Parameters give you a way to dynamically modify values in a calculation. Rather than manually editing the calculation (and all dependent calculations), you can use a parameter. Then when you want to change the value, you open the parameter control, change the value, and all of the calculations that use that parameter are updated.

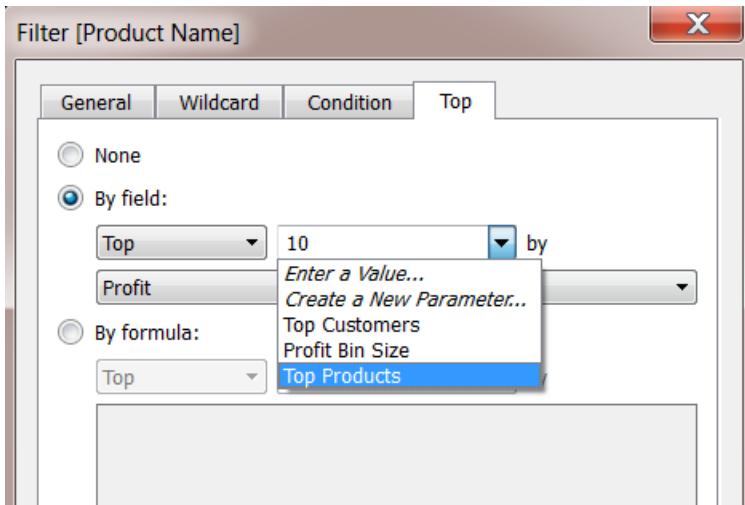
To use a parameter in a calculation, drag the parameter from the Data pane and drop it in the calculation editor, either at a new location in the formula or to replace a part of the current formula:



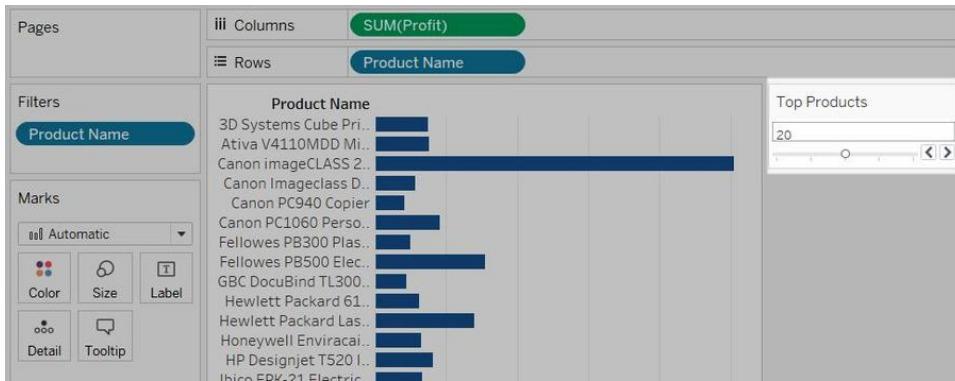
Using Parameters in Filters

Parameters give you a way to dynamically modify values in a TopN filters. Rather than manually setting the number of values you want to show in the filter, you can use a parameter. Then when you want to change the value, you open the parameter control and the filter updates. For example, when creating a filter to show the Top 10 products based on total profit, you may want to use a parameter instead of the fixed “10” value. That way, you can quickly update the filter to show the top 10, 20, or 30 products.

A list of parameters is available in the drop-down lists on the **Top** tab of the Filter dialog box. Select the parameter you want to use in the filter.



To show the parameter control, right-click the parameter in the **Data** pane and select **Show Parameter Control**. Use the parameter control to modify the filter to show the top 10 products, 15 products, 20 products, and so on.



Use Parameters to Make Views More Interactive

Parameters are useful when you want to add interactivity and flexibility to a report, or to experiment with what-if scenarios. Suppose you are unsure which fields to include in your view or which layout would work best for your viewers. You can incorporate parameters into your view to let viewers choose how they want to look at the data.

When you work with parameters, consider the following two things that are important in making them useful:

- They need to be used in calculations.
- The parameter control needs to be displayed so that viewers can interact with it.

Before you begin, decide which fields you want to make interactive. For example, you could allow users to view the categories within a dimension by color, or to view sales data over a period of time that they choose, and so on. The example described here sets up a table for which users can select the dimensions to display in the columns and rows.

Tip: To learn how to create parameters for what-if and other common scenarios, see the on-demand training video [Parameters](#) on the Tableau website. Also see [Use Parameters to Add Multiple Views to Your Viz](#) and [Using a parameter to change the view in a dashboard](#).

Create the parameters and calculated fields

These steps use the Superstore sample to create a new parameter while you build the calculated field that will take advantage of it.

Step 1

Select **Analysis > Create Calculated Field**.

Step 2

In the Calculated Field dialog box, for **Name**, type **Column 1 Category**.

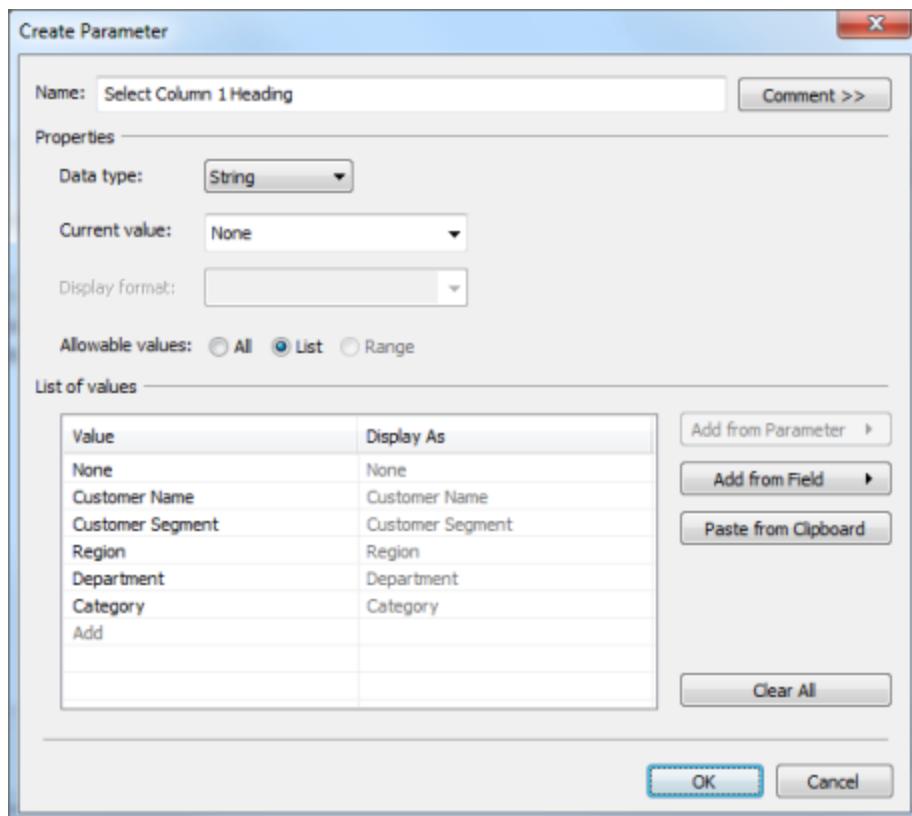
Step 3

Next to **Parameters**, click **Create**, and in the **Create Parameter** dialog box, complete the following steps.

1. Name the parameter appropriately so that viewers can tell what changing it will do. This example uses **Select Column 1 Heading**.
2. For **Data type**, select **String**.
3. For **Allowable Values**, select **List**, type **None** as the first value in the list, and then press **Enter**.
4. Complete the list by typing the names of the additional dimension fields that you want to expose through the parameter.

Note: This example uses the customer name, customer segment, region, department, and category fields. These are all dimensions of the same data type (string). If you wanted to include a measure such as profit in this list, one option would be to convert the measure to a string value. You would do this when you build the calculated field, using the STR() function. This article covers only the single data type scenario.

The **Display As** aliases default to the field name, and for this exercise you can leave them as they are.



5. Click **OK** to return to the Calculated Field dialog box.

Step 4

Repeat the previous step to create the following additional parameters:

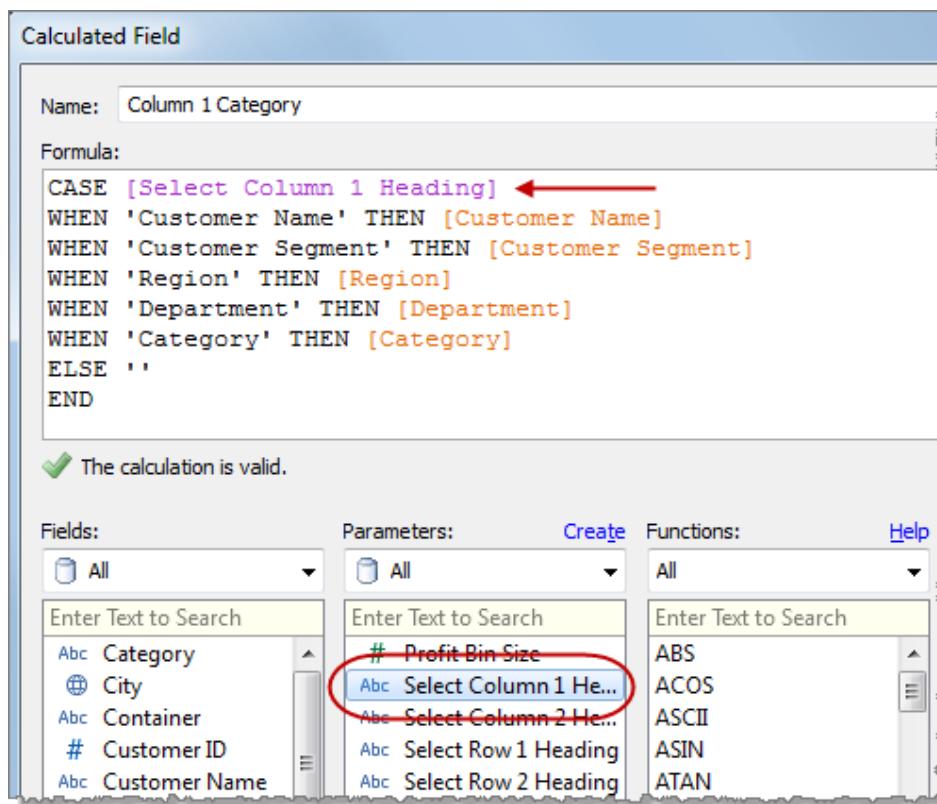
- Select Column 2 Heading
- Select Row 1 Heading
- Select Row 2 Heading

Tip: Instead of typing each value in the list, click **Add from Parameter** to add them from **Select Column 1 Heading**.

Step 5

In the Calculated Field dialog box, for **Formula**, build the following calculation:

```
CASE [Select Column 1 Heading]
WHEN 'Customer Name' THEN [Customer Name]
WHEN 'Customer Segment' THEN [Customer Segment]
WHEN 'Region' THEN [Region]
WHEN 'Department' THEN [Department]
WHEN 'Category' THEN [Category]
ELSE ''
END
```



Confirm that the status message indicates that the formula is valid, and then click **OK**.

Note: ELSE accounts for the **None** value that you included in the parameter, and it returns an empty string.

Step 6

Create three more calculated fields, one for each of the additional parameters you created:

Parameter name	Calculated field name
Select Column 2 Heading	Column 2 Category
Select Row 1 Heading	Row 1 Category
Select Row 2 Heading	Row 2 Category

The basic formula for each calculated field is the same as in the previous step, except that you reference a different parameter in each CASE statement.

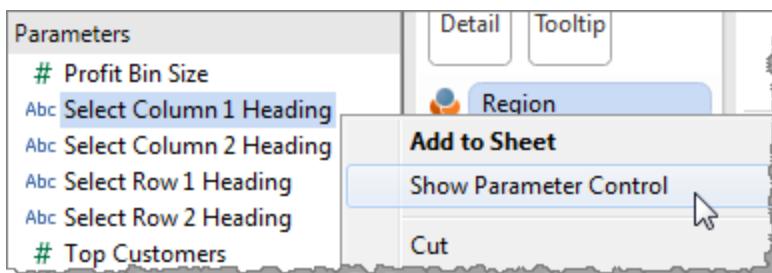
[Give viewers a way to interact with views](#)

Now you expose the parameter control so users can select the categories they want to display.

Step 1

For each parameter you created, do the following:

In the Parameters pane, right-click the parameter and select **Show Parameter Control**.



Step 2

From the Dimensions pane, drag the calculated fields you created to the Columns and Rows shelves.

Step 3

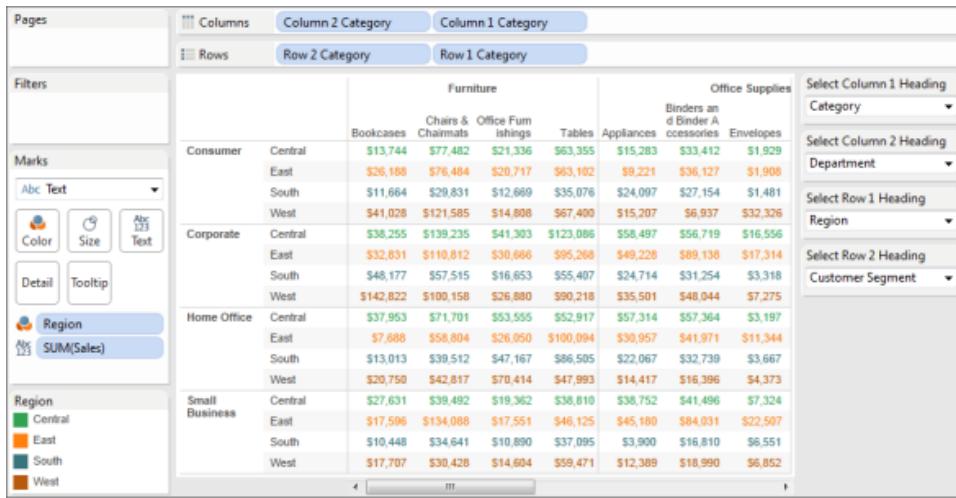
From the Measures pane, drag a measure to the view. In this example, **Sales** is placed on **Label** on the Marks card.

Step 4

Test your parameters by selecting fields in the parameter controls.

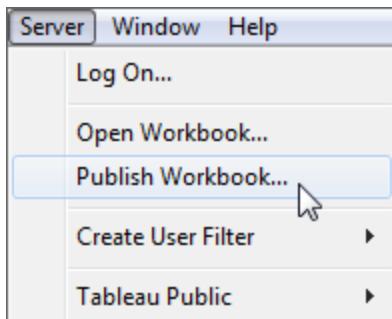
Tips:

- Sort the dynamic dimension fields alphabetically.
- Hide field labels for rows and columns.



Step 5

Reset all parameters to **None** and publish the workbook to Tableau Server.



Viewers can set up their own reports, save their parameter settings, and share views with others.

Additional information

For information about building views dynamically, see [Swap Measures Using Parameters](#).

For information about the different areas in the Tableau interface in which you can create and incorporate parameters, see [Parameters](#) and its related topics in the Tableau Desktop Help.

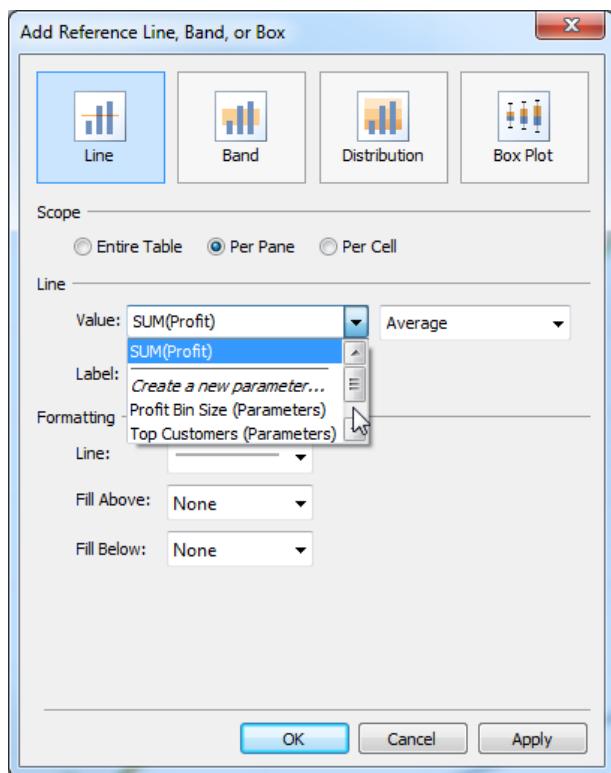
Disclaimer: This topic includes information about parameters from a third-party blog, [The Information Lab: Data School](#). Please note that while we make every effort to keep references to third-party content accurate, the information we provide here might

change without notice as content on [The Information Lab: Data School](#) changes. For the most up-to-date information, please consult [The Information Lab: Data School](#) blog.

Using Parameters in Reference Lines

Parameters give you a way to dynamically modify a reference line, band, or box. For example instead of showing a reference line at a fixed location on the axis, you can reference a parameter. Then you can use the parameter control to move the reference line.

A list of parameters is available in the Value drop-down list in the Add Reference Line, Band, or Box dialog box. Select the parameter you want to use.

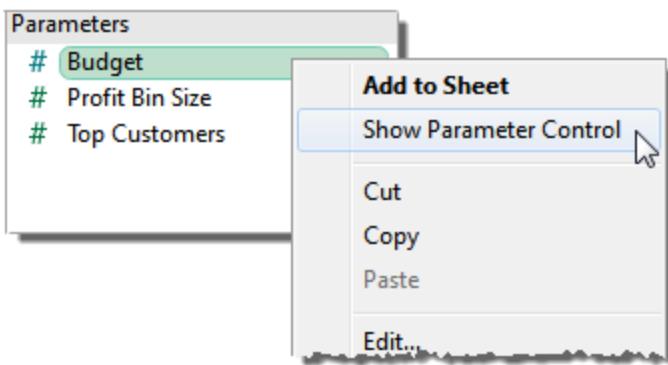


The reference line is drawn at the Current Value specified by the parameter. To open the parameter control, right-click (Control-click on a Mac) the parameter in the **Data** pane and then select **Show Parameter Control**. Use the parameter control to change where the reference line is drawn.

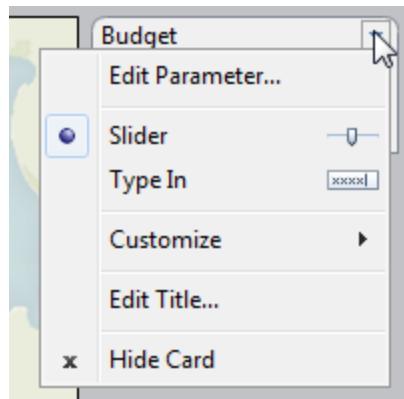
Parameter Controls

The parameter control is a worksheet card that lets you modify the parameter value. Parameter controls are very similar to filter cards in that they contain controls that modify the view. You can open parameter controls on worksheets and dashboards and they are included when you save to the web or publish to Tableau Server.

To open the parameter control, right-click (Control-click) the parameter in the **Data** pane and select **Show Parameter Control**.



Like other cards, parameter controls have a menu that you can open using the drop-down arrow in the upper right corner of the card. Use this menu to customize the display of the control. For example, you can show a list of values as radio buttons, a compact list, a slider, or a type in field. The options available on this menu depend on the data type of the parameter as well as whether it accepts all, a list, or a range of values



Example – Add a Parameter to a Map View

This example uses the World Indicators sample data source to demonstrate the following:

- How to build a map view that shows the birth rate for each country in the world.
- How to create a calculated field that distinguishes countries with a low birth rate from those with a high birth rate.
- How to create and display a parameter so that users can set the threshold for low vs. high birth rate.

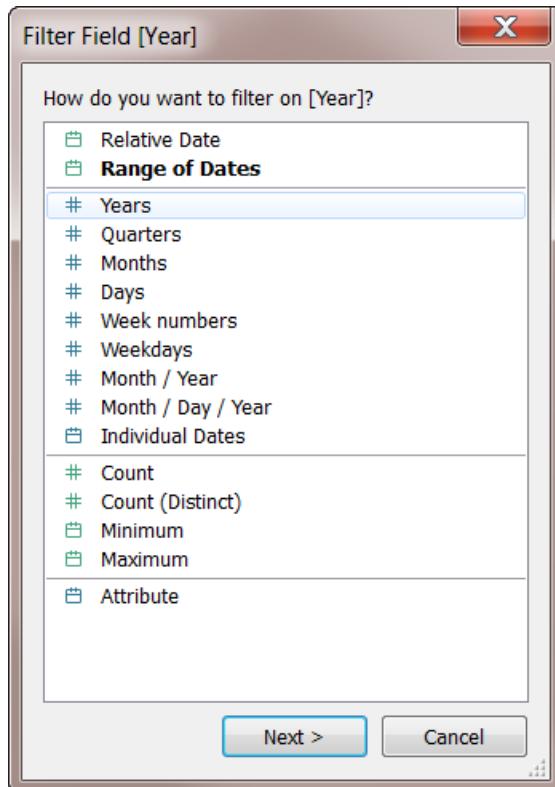
Build a map view

1. In the **Data** pane, double-click **Latitude** and then **Longitude**.

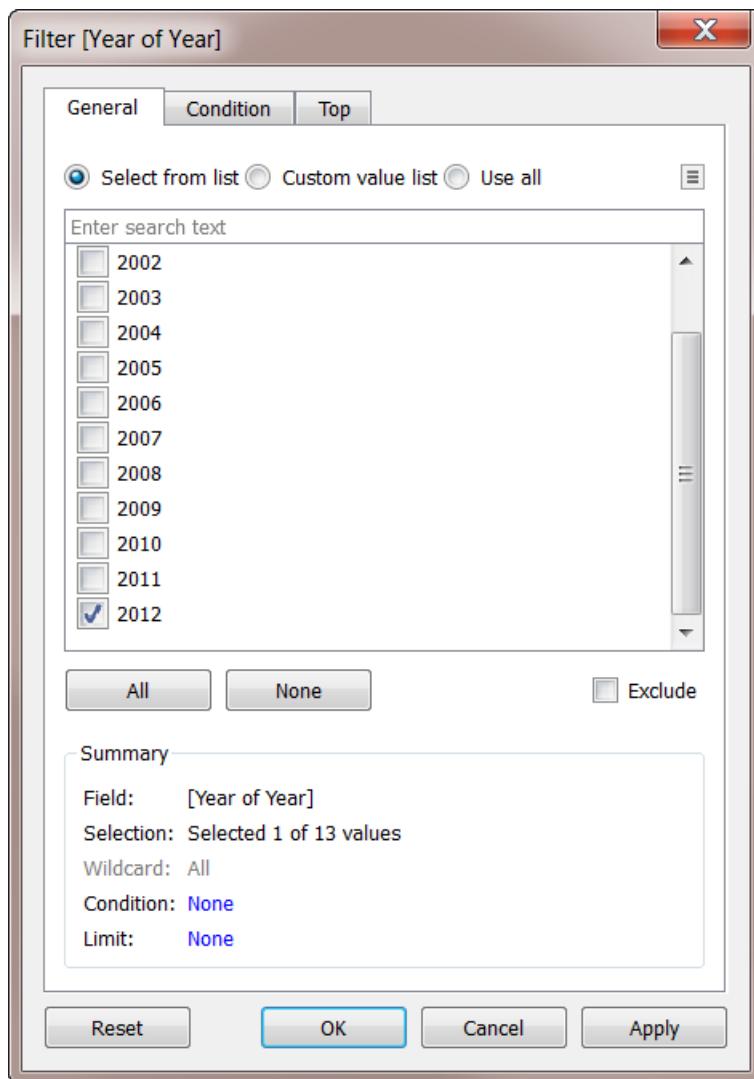
Tableau puts **Longitude** on **Columns**, **Latitude** on **Rows**, and displays a map of the world.

2. Drag the **Year** dimension to **Filters**.

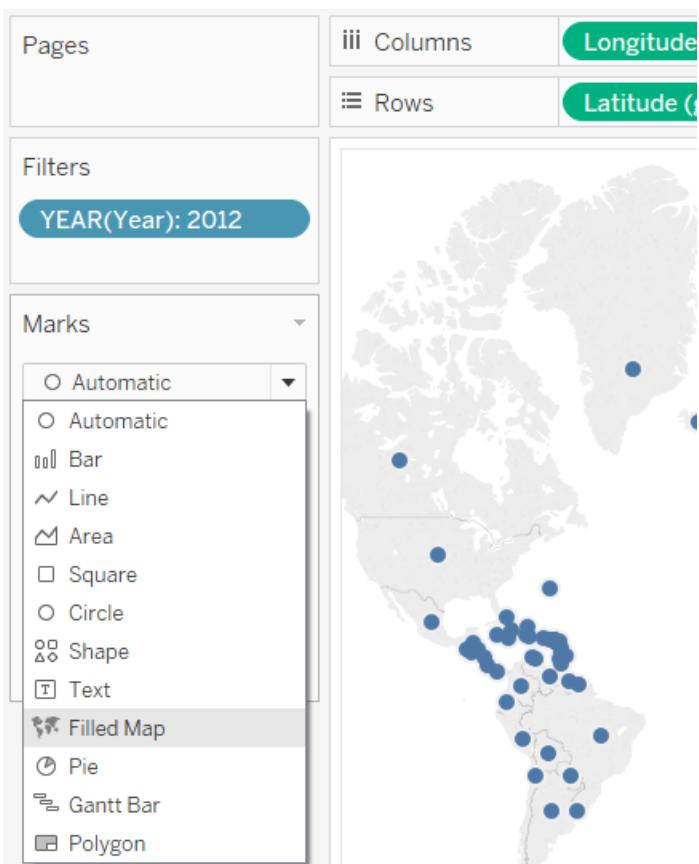
3. In the Filter Field [Year] dialog box, choose **Years** and then click **Next**:



4. In the Filter [Year of Year] dialog box, select **2012** and then click **OK**:

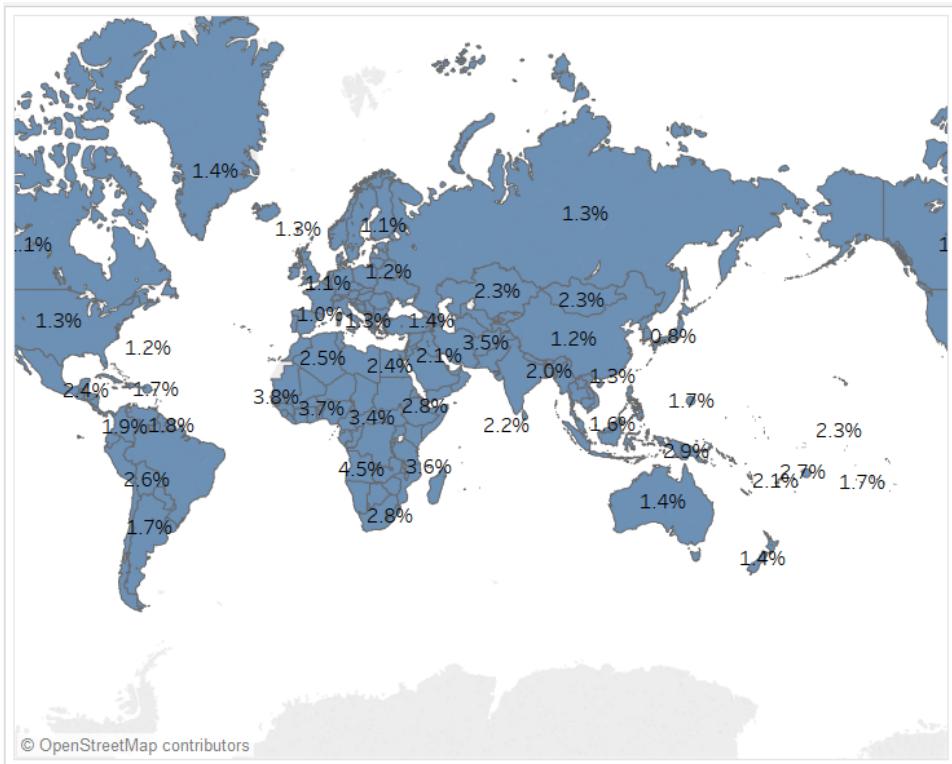


5. Drag the **Country** dimension to **Detail**.
6. Set the Marks type to Filled Map:



7. Drag the **Birth Rate** measure to **Label**.

You now have a map that shows birth rates for countries around the world:



You can zoom the map or hover to see a tooltip for any country.

Create a calculated field to set a threshold

Next, you'll distinguish a low birth rate from a high birth rate.

1. From the top menu, select **Analysis > Create Calculated Field**.
2. Name the field **High Birth Rate** and type or paste this calculation in the formula field:

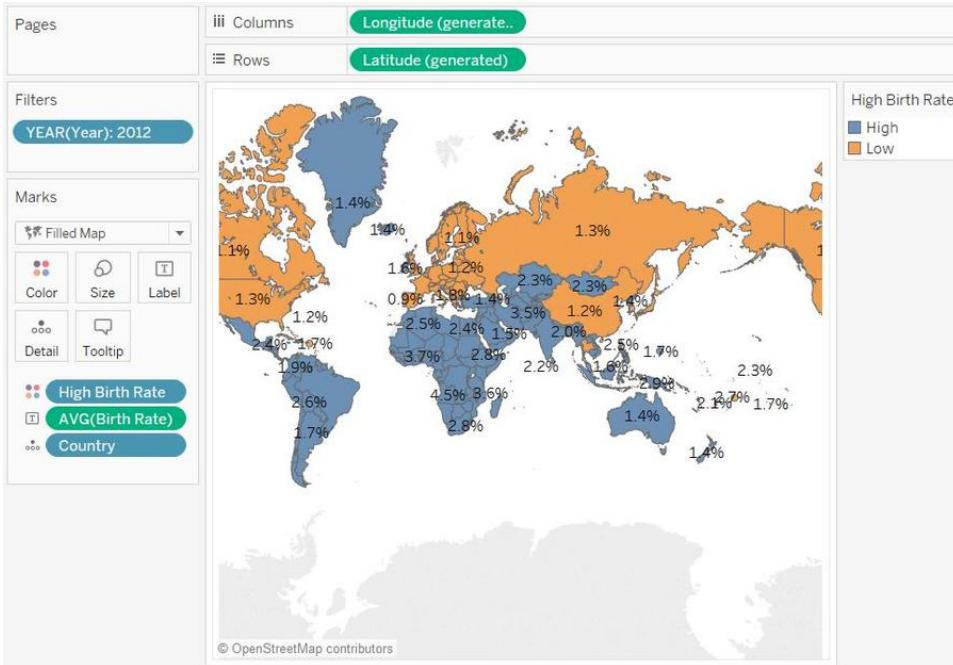
```
IF ([Birth Rate]) >= 0.014 THEN "High" ELSE "Low" END
```

The value 0.014 is equivalent to 1.4%. The range of actual values that we can see on the map range from below 1% up to nearly 5%.

When you click **OK** to apply and save this calculation, Tableau categorizes it as a dimension.

3. Drag **High Birth Rate** to **Color**.

The map now shows low birth rate countries in one color and high birth rate countries in another:



But the definition of high birth rate as anything equal to or greater than 1.4% is arbitrary—that value was chosen because it divided the world's countries about evenly.

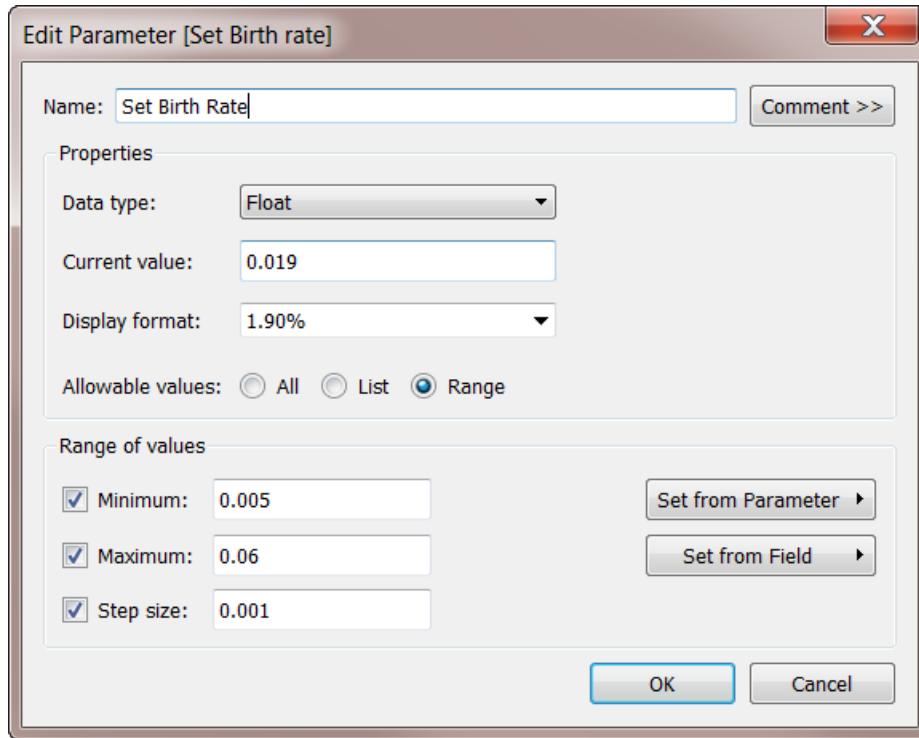
Instead, you can let users define that threshold, or give them a control that they can use to see how changing the threshold changes the map. To do this, you create a parameter.

[Create a parameter](#)

1. Right-click (control-click on a Mac) in the **Data** pane and select **Create > Parameter**.

Note: If you clicked on a field on the Data pane, some fields might be filled. You can just change the fields as needed.

2. In the Create Parameter dialog box, name the new parameter **Set Birth Rate** and configure it as shown:



For information on the fields in the Create Parameter dialog box, see [Creating Parameters](#) on page 989.

Because the **Data Type** is **Float**, the parameter control, when you display it in the next procedure, will be in the form of a slider. This is because floating point values are continuous—there are an infinite number of possible values.

The **Current value** sets the default for the parameter: 0.019 is 1.9%. The **Range of values** section sets the minimum and the maximum values and the step size—that is, the least amount by which the value can change.

3. Click **OK**.

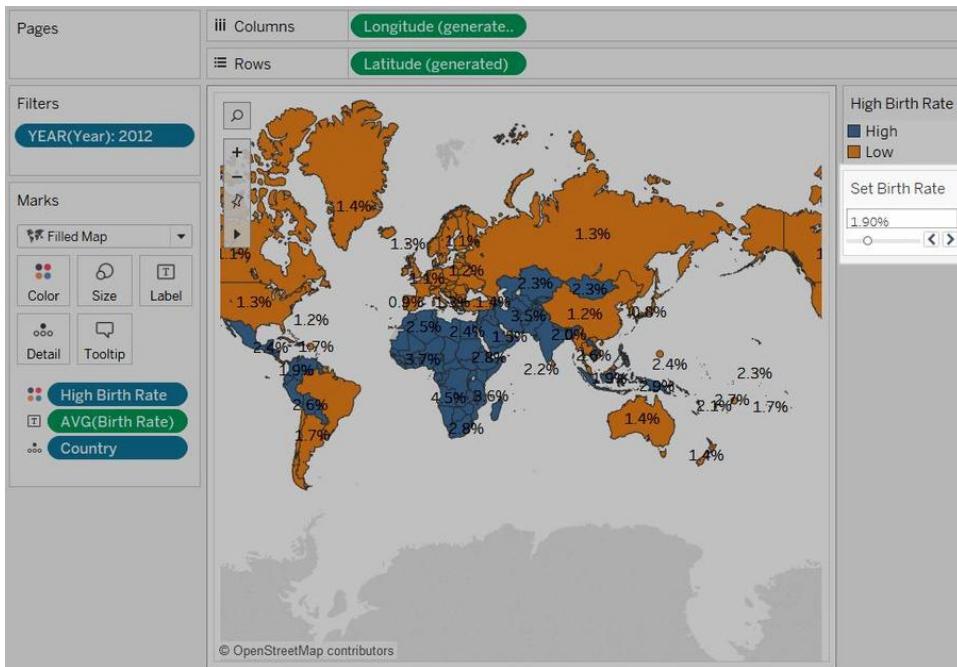
Create and display the parameter control

Now you must connect the parameter to the **High Birth Rate** field.

1. Right-click **High Birth Rate** in the **Data** pane and select **Edit**.
2. Replace the hard-coded 0.014 value in the field definition with the parameter name:

```
IF ([Birth Rate]) >= [Set Birth Rate] THEN "High" ELSE "Low"
END
```
- Then click OK.
3. Right-click the **Set Birth Rate** parameter in the **Data** pane and select **Show Parameter Control**.

By default, the parameter control is shown on the right. Now you and users of your view can raise or lower this value incrementally to see how changing the definition of "high birth rate" affects the map.



Example – Swap Measures Using Parameters

You may want to create a view that shows how the values of two measures compare against each other. But what if you also wanted to be able to choose which measures were being compared—or better yet, add a control to the view that would let any user select the measures to be compared. You can create such a view using parameters and calculated fields. The calculated fields replace the measures in the view and can be set interactively by the user with parameter controls.

Overview

The process requires you to do the following:

1. Create the parameters
2. Create calculated fields to change the measures in the view
3. Set up the view

This article includes an additional section that you can use as an alternative to step 2:

- Create calculated fields to change measures and specify aggregations

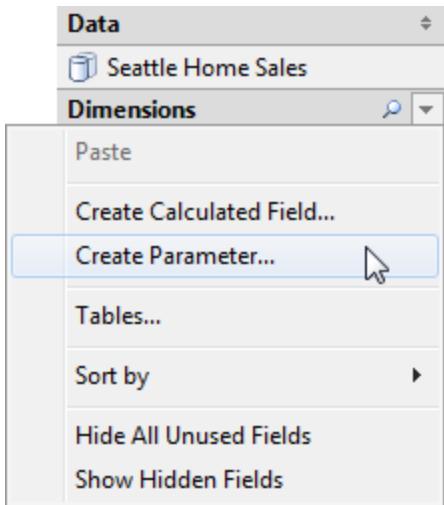
The following sections break these procedures down into specific instructions.

The scenario uses the **Sample - Superstore** data source provided with Tableau Desktop.

Create the parameters

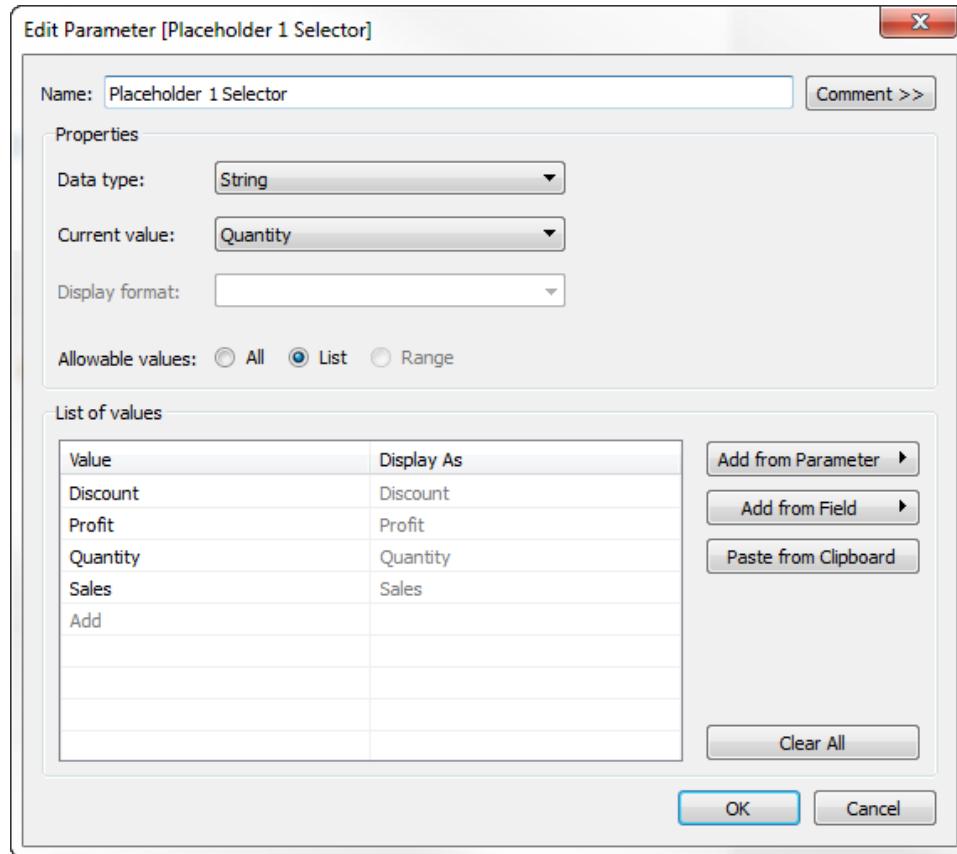
Follow these steps.

1. Click the drop-down arrow to the right of the word Dimensions in the Data pane and select **Create Parameter**.



2. In the Create Parameter dialog box, do the following:
 - a. Name the parameter **Placeholder 1 Selector**.
 - b. Set the **Data type** to **String**.
 - c. Skip down to the **Allowable values** field, and choose **List**.
 - d. Type individual measure names in the **List of values** area: **Discount**, **Profit**, **Quantity**, and **Sales**.

The Edit Parameter dialog box should now look like this:



- e. Click **OK** to close the Edit Parameter dialog box.
 - 3. Create a second parameter, **Placeholder 2 Selector**, with the exact same configuration.
- There are different ways you can do this. The easiest way is to click **Placeholder 1 Selector** in the **Data** pane, choose **Duplicate**, and then change the name of the duplicated parameter to **Placeholder 2 Selector**.

Create calculated fields to change the measures in the view

Follow these steps.

1. Choose **Analysis > Create Calculated Field** to open the calculation editor. Name the calculation **Placeholder 1** and type or paste the following in the formula area:

```
CASE [Placeholder 1 Selector]
WHEN "Discount" THEN [Discount]
WHEN "Profit" THEN [Profit]
WHEN "Quantity" THEN [Quantity]
```

```
WHEN "Sales" THEN [Sales]  
END
```

Note: The calculated field must reference the **Value** entry for a given row, and not the **Display As** value.

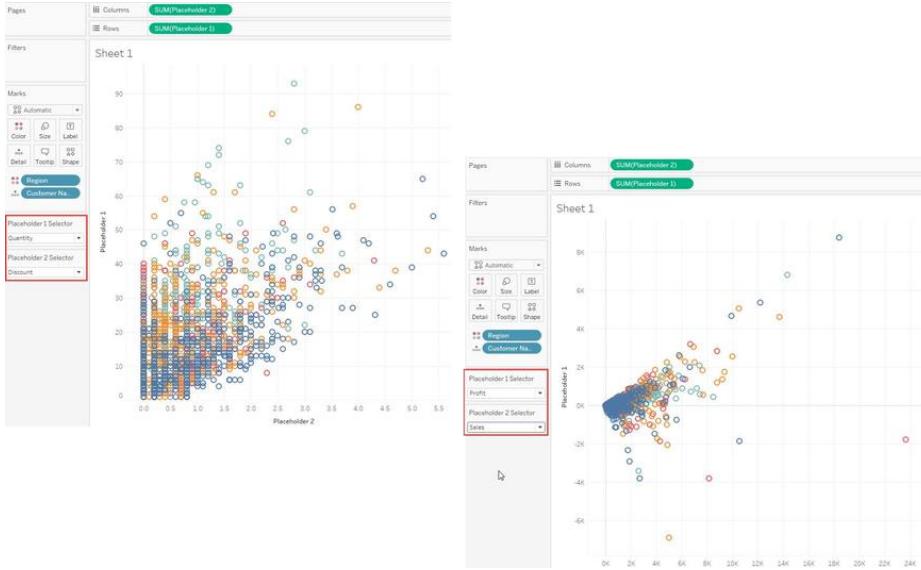
2. Click **OK** to exit the calculation editor.
3. Create a second calculated field, **Placeholder 2**, with the same definition. Again, the easiest way to do this is to click **Placeholder 1** in the **Data** pane, choose **Duplicate**, and then change the name of the duplicated field to **Placeholder 2**.

Set up the view

Follow these steps.

1. Drag **Placeholder 2** to **Columns** and **Placeholder 1** to **Rows**.
Because you dragged measures to both shelves, the default view is a scatter plot. For more on why Tableau does this, see [Example – Scatter Plots and Aggregation on page 339](#).
2. Drag **Customer Name** to **Detail** and **Region** to **Color**.
3. In the Parameters area of the Data pane, click each **Placeholder 1 Selector** and choose **Show Parameter Control**. Then do the same for **Placeholder 2 Selector**.
4. Tableau displays the parameter controls beyond the right side of the view by default. Drag them over to the left side to make them easier for your users to see.

Your view is now complete. The parameter controls let users select the measures to be used on the X and Y axes. For example, the view below on the left shows **Quantity** vs. **Discount**, while in the view the right the parameter controls have been used to show **Profit** vs. **Sales**.



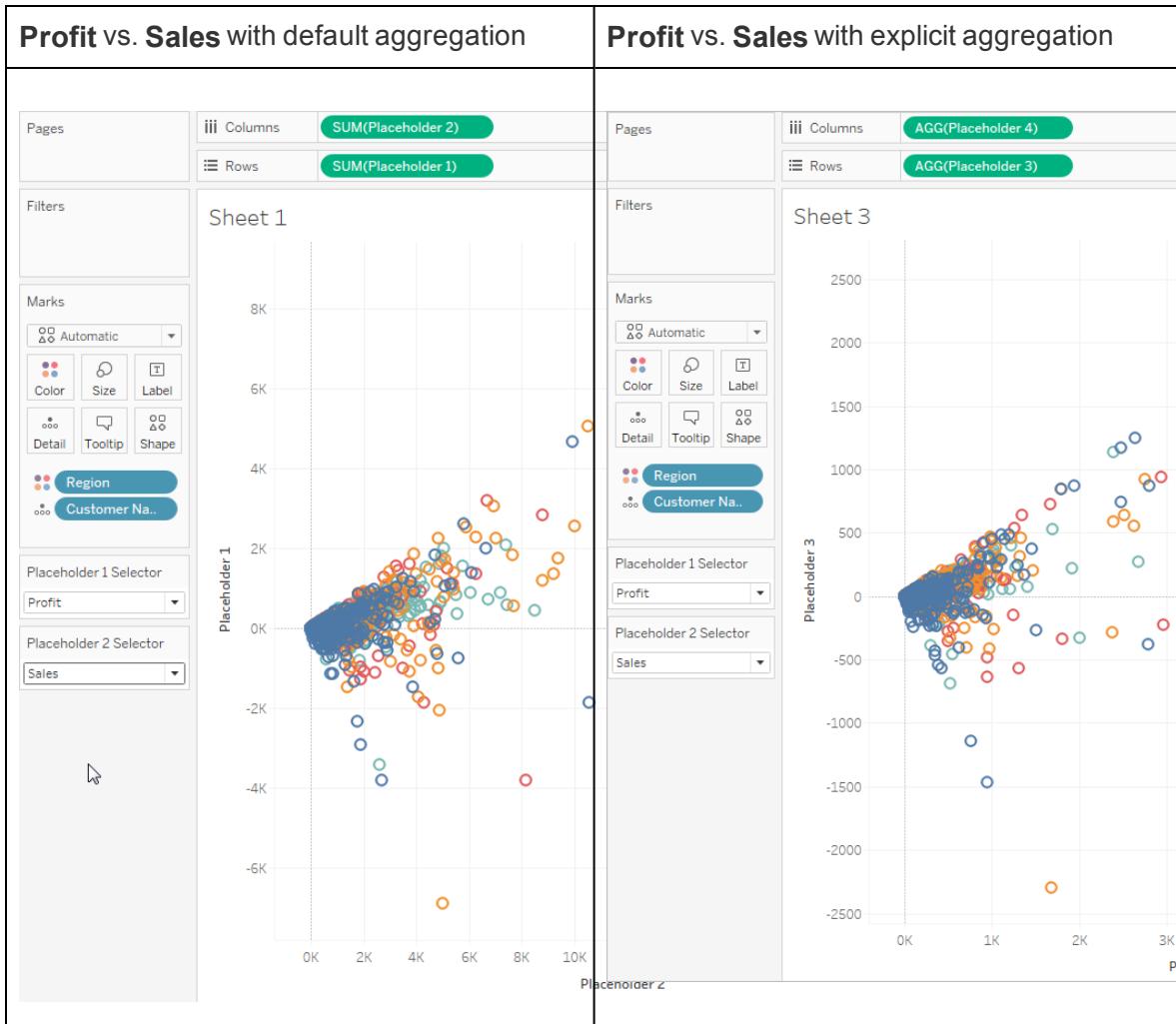
Create calculated fields to change measures and specify aggregations

As an alternative to the [Create calculated fields to change the measures in the view](#) section, above, consider creating calculated fields that specify aggregations for individual measures. As written above, the calculated fields do not specify aggregations. Notice in the image above that Tableau automatically assigns an aggregation (SUM) to the **Placeholder 1** and **Placeholder 2** fields. But you know your data, and you may want to dictate which aggregation Tableau uses for your measures. So instead of the calculated field definition above, consider a definition like the following:

```
CASE [Placeholder 1 Selector]
WHEN "Discount" THEN SUM([Discount])
WHEN "Profit" THEN AVG([Profit])
WHEN "Quantity" THEN SUM([Quantity])
WHEN "Sales" THEN AVG([Sales])
END
```

It's up to you to decide whether to explicitly aggregate measures in your field definitions. The only thing you cannot do is mix-and-match: that is, you cannot define aggregations for some measures, but not for others.

Here's how the **Profit** vs. **Sales** scatter plot changes when you specify AVG as the aggregation for these fields, as opposed to not specifying an aggregation and letting Tableau default to SUM:



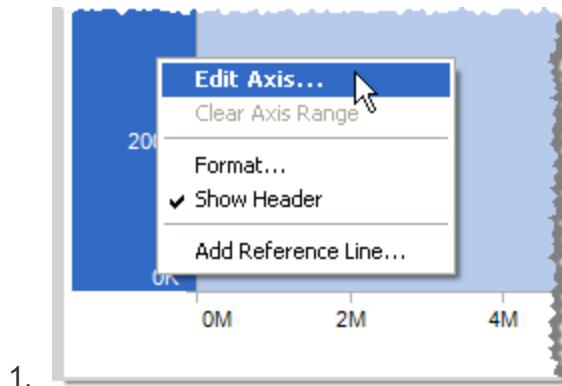
Similar, but different.

Log Axes

Sometimes you will have a measure that uses a logarithmic scale as opposed to linear. For example, some well known logarithmic scales include the Richter magnitude scale to measure the strength of earthquakes, pH to measure acidity, and the stellar magnitude scale, which measures the brightness of stars. You can edit the axis scale for any measure to be logarithmic using the Edit Axis dialog box. By default the tick marks are drawn at powers of ten, however, you can specify any base that is greater than 1.

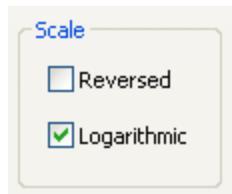
To change the scale of an axis:

Right-click (Control-click on a Mac) the axis in the view and select **Edit Axis**.



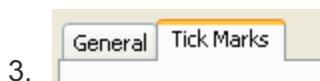
1.

In the Edit Axis dialog box, select **Logarithmic** for the axis scale.



2.

Select the **Tick Marks** tab.



3.

4. Select one of the following Major Tick mark options:

- **Automatic** - the major tick marks are drawn at powers of 10.
- **Fixed** - the major tick marks are drawn at a specified exponent. Type a number into the Powers of text box.
- **None** - major tick marks are not shown.

5. When finished, click **OK**.

You can also reverse the axis by selecting **Reverse** in the Scale area on the General tab of the Edit Axis dialog box.

If your data contains negative values Tableau cannot plot them on a logarithmic scale. All values with a negative value will be displayed at 1 on the axis. You can then filter these records to exclude them from the view.

Analytics How To's

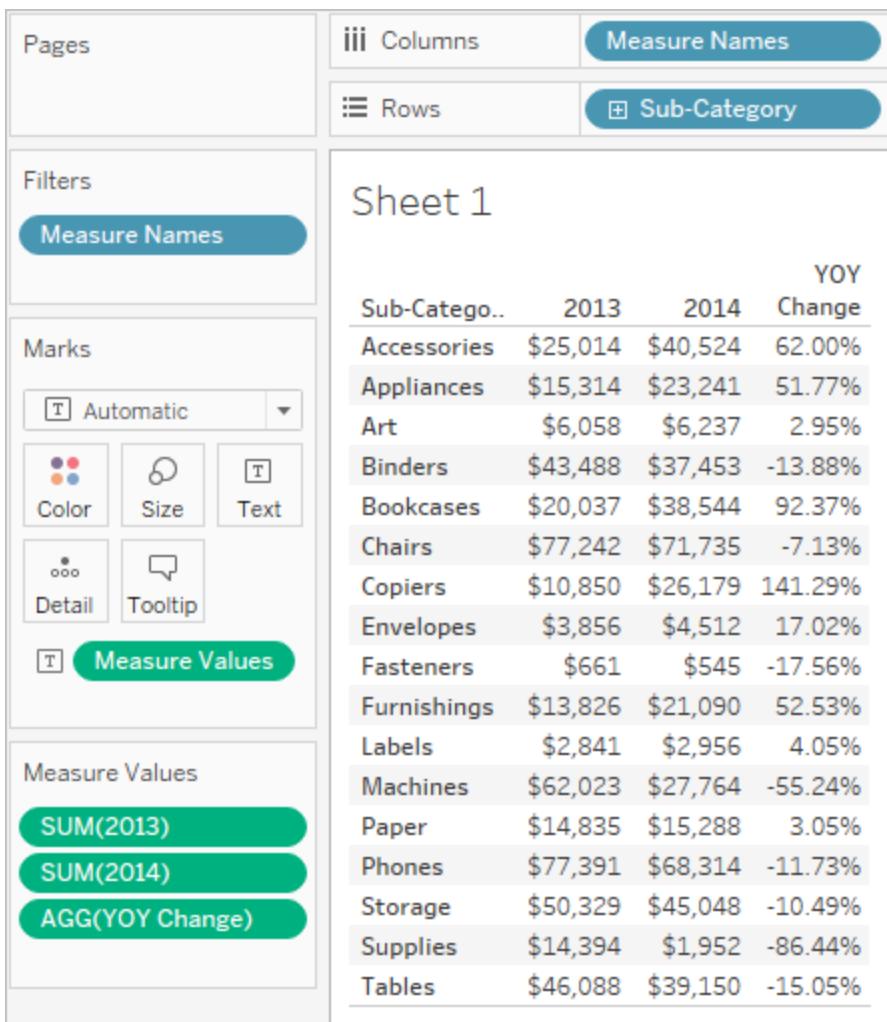
The topics in this section illustrate how to create views that include industry-standard analytical benchmarks and visualization types.

Add a Calculated Column to a View

Sometimes, columns in your text table do not allow you to display the results of certain calculations as you might expect. In cases like this, you can create a calculated column that uses a single formula that automatically adjusts the value for each row in the table. For example, suppose you want to create a view that displays the sales for each year in several columns and the year-over-year (YOY) percentage change in the final column. There are different ways to get the result you want, but a fairly straightforward approach is to create custom calculations use them in the view.

Objective

To create a view like the following, that shows sales results for two years in the first two columns, and then the year-over-year change, as a percentage, in the third column.



How to Build It

The scenario uses the **Sample - Superstore** data source provided with Tableau Desktop to show how to build the view shown above.

Create the necessary calculated fields

1. Connect to the **Sample - Superstore** data source.
2. Choose **Analysis > Create Calculated Field** to open the calculation editor. Name the calculation **2013** and type or paste the following in the formula area:

```
IF YEAR([Order Date]) = 2013 THEN [Sales] ELSE 0 END
```
3. Create a second calculated field and name it **2014**; the formula is the same except with 2014 instead of 2013:

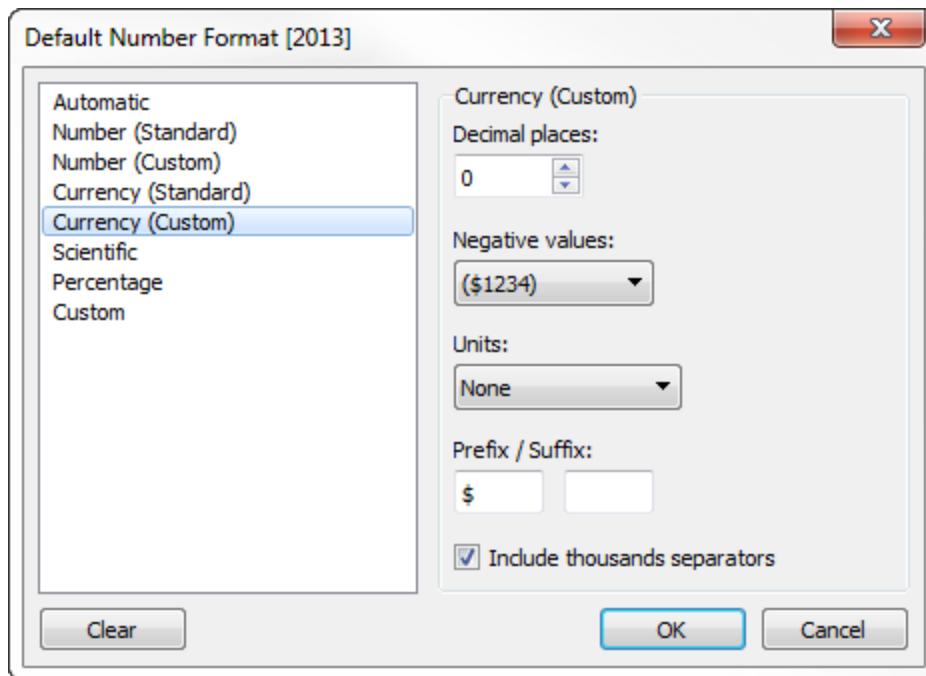
```
IF YEAR([Order Date]) = 2014 THEN [Sales] ELSE 0 END
```
4. Create a third calculated field, **YOY Pct. Change**, to calculate the change from 2013 to

2014:

$(\text{SUM}([\text{2014}]) - \text{SUM}([\text{2013}])) / \text{SUM}([\text{2013}])$

Format the calculated fields

1. Click **2013** in the Measures area of the **Data** pane and choose **Default Properties > Number Format**.
2. In the Default Number Format dialog box, set the format to **Currency (Custom)** and the **Decimal places** to **0**.



3. Format the **2014** measure the exact same way.
4. Format the **YOY Pct. Change** field as **Percentage**, with **2** decimal places.

Build the view

1. Drag **2013** to Text on the Marks card.
2. Double-click **2014** and then **YOY Pct. Change**.
3. Drag **Measure Names** from **Rows to Columns**.
4. Drag **Sub-Category** to **Rows**.

Your view should now look like the one at the beginning of this topic.

Calculate Z-scores

In statistics, the z-score (or standard score) of an observation is the number of standard deviations that it is above or below the population mean.

To calculate a z-score you must know the population mean and the population standard deviation. In cases where it is impossible to measure every observation of a population, you can estimate the standard deviation using a random sample.

Create a z-score visualization to answer questions like the following:

- What percentage of values fall below a specific value?
- What values can be considered exceptional? For example, in an IQ test, what scores represent the top five percent?
- What is the relative score of one distribution versus another? For example, Michael is taller than the average male and Emily is taller than the average female, but who is relatively taller within their gender?

As a general rule, z-scores lower than -1.96 or higher than 1.96 are considered unusual and interesting. That is, they are statistically significant outliers.

This article demonstrates how to calculate a z-score in Tableau.

1. Connect to the **Sample - Superstore** data source provided with Tableau Desktop.
2. Create a calculated field to calculate average sales.

Choose **Analysis > Create Calculated Field** to open the calculation editor. Name the calculation **Average Sales** and type or paste the following in the formula area:

```
WINDOW_AVG(SUM([Sales]))
```

3. Create another calculated field to calculate the standard deviation. Name the calculation **STDEVP Sales** and type or paste the following in the formula area:

```
WINDOW_STDEVP(SUM([Sales]))
```

4. Create one more calculated field, this one to calculate the z-score. Name the calculation **Z-score** and type or paste the following in the formula area:

```
(SUM([Sales]) - [Average Sales]) / [STDEVP Sales]
```

5. Drag **Z-Score** from the **Data** pane to **Columns** and **State to Rows**.

Notice that the **Z-score** field on Columns has a table calculation icon on the right side (that is, a small triangle):



The **STDEVP Sales** function is based on the **WINDOW_STDEVP** function, which is a table calculation function. The **Z-Score** function, in turn, is a table calculation function because it includes **STDEVP Sales** in its definition. When you use a calculated field that includes a table calculation function in a view, it's the same as adding a table calculation to a field manually. You can edit the field as a table calculation. In fact, that's what you do next.

6. Click the **Z-score** field on Columns and choose **Compute Using > State**.

This causes the z-scores to be computed on a per-state basis.

7. Click the **Sort Descending** icon on the toolbar:

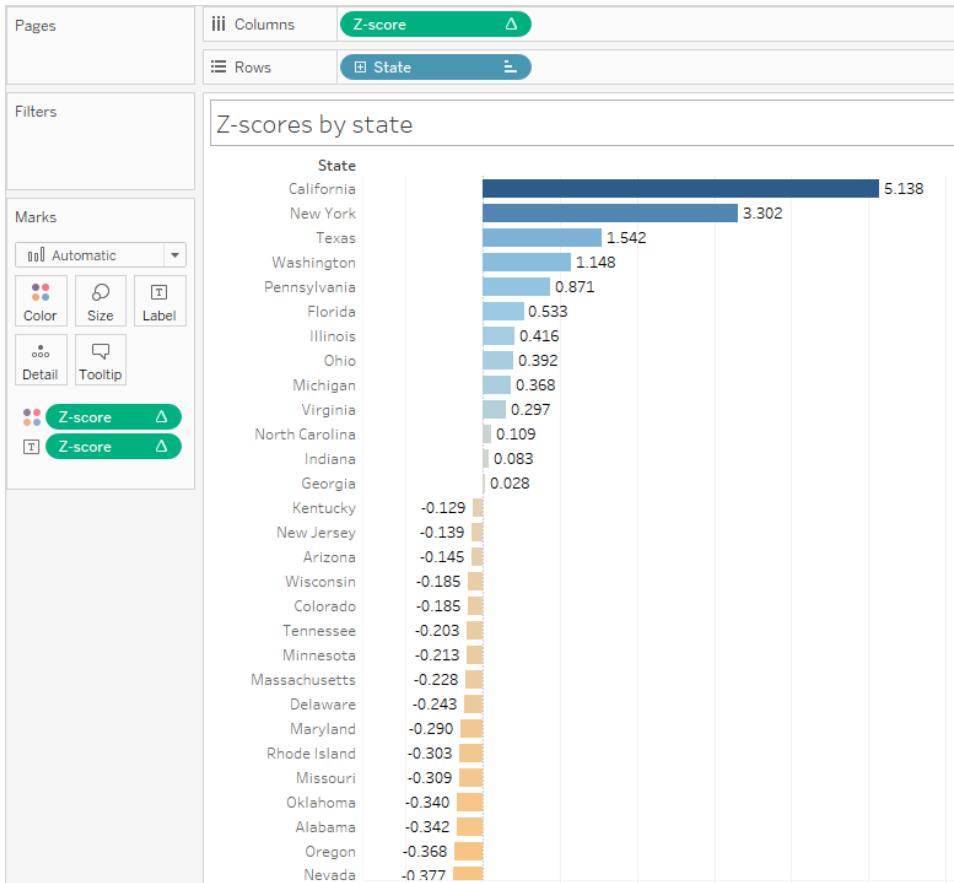


8. Hold down the Ctrl key and drag the **Z-score** field from **Columns** to **Color**.

Ctrl + Drag copies a field as currently configured to an additional location.

9. Ctrl + Drag **Z-score** from Columns once again. This time drop it on **Label**.

You now have a distribution of z-scores broken out by state. California and New York both have z-scores greater than 1.96. You could conclude from this that California and New York have significantly higher average sales than other states.



Visualize Key Progress Indicators

This article shows how to create a view that shows Key Progress Indicators (KPIs). A Key Performance Indicator is a measurable value that shows how effectively a company is achieving key business objectives. At a high level, the procedure requires you to do the following:

1. Create a view that includes the field or fields (measures) you want to assess.
2. Create a calculated field that establishes the threshold that demarcates success from failure.
3. Update the view to use KPI-specific shape marks to show which values are above the threshold and which are below.

Note: When connected to Microsoft Analysis Services, any KPI calculations defined in the cube are not available in Tableau. But as the procedure below demonstrates, you can write your own KPI calculations directly inside Tableau, and then use Tableau parameters to create highly flexible what-if KPI analysis.

The scenario uses the **Sample - Superstore** data source provided with Tableau Desktop to show how to build a KPI view that displays a green check mark for any sales figure over \$25,000, and a red X for any sales figure under \$25,000.

Create a view that includes the field you want to assess

In this case, that field is **Sales**.

1. Connect to the **Sample - Superstore** data source.
2. From the Dimensions area of the Data pane, drag **Sub-Category** to **Rows** and **Region** to **Columns**.
3. From the Measures area of the Data pane, drag **Sales** to **Text** on the Marks card.

Create a calculated field that establishes the threshold that demarcates success from failure

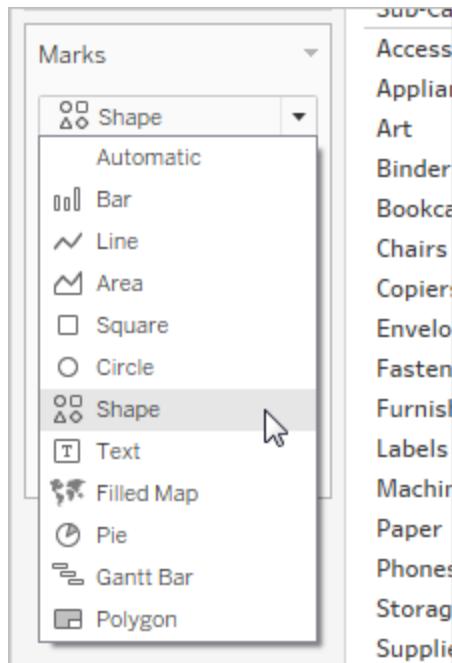
1. Choose **Analysis > Create Calculated Field** to open the calculation editor. Name the calculation KPI and type or paste the following in the formula area

```
IF SUM ([Sales]) > 25000 THEN "Above Benchmark" ELSE "Below Benchmark" END
```

2. Click **OK**.

Update the view to use KPI-specific shape marks

1. On the Marks card, select Shape from the drop-down list of views:



2. Drag the **KPI** field from the **Measures** area of the **Data** pane to **Shape** on the Marks

card.

3. Click **Shape** on the Marks card to open the **Edit Shape** dialog box.

For more about shapes, see [Shape Mark on page 181](#).

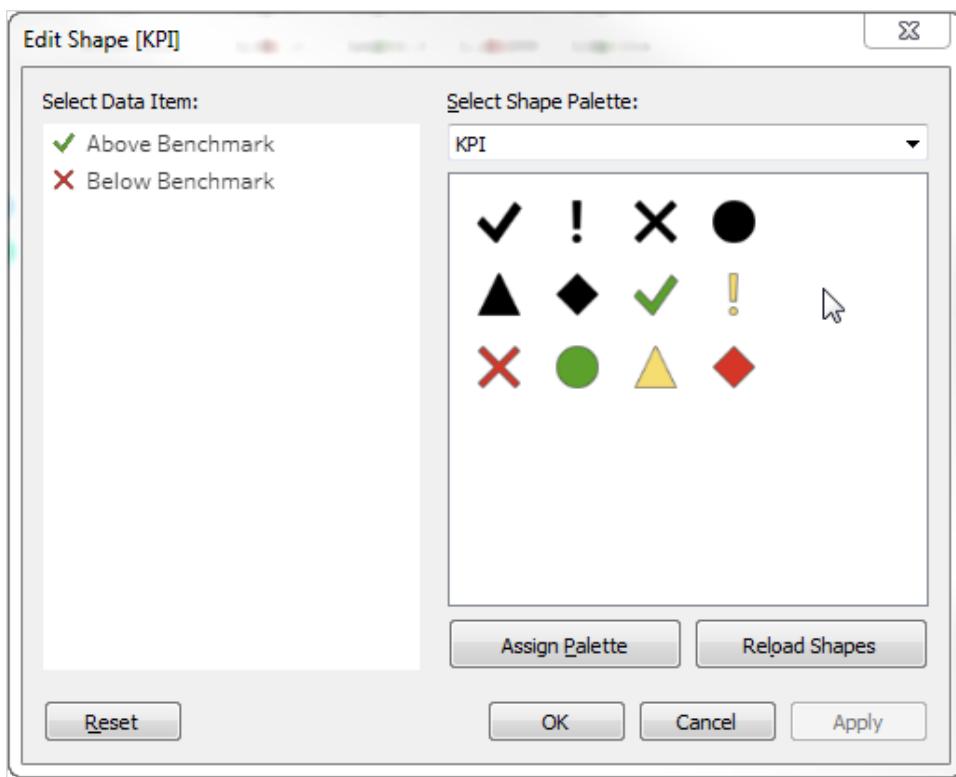
4. From the **Select Shape Palette** drop down list, choose **KPI**.

Now you are ready to associate specific values for the KPI field with specific shapes.

5. Click **Above Benchmark** under **Select Data Item** and then click the green check mark in the palette.

6. Click **Below Benchmark** under **Select Data Item** and then click the red X in the palette.

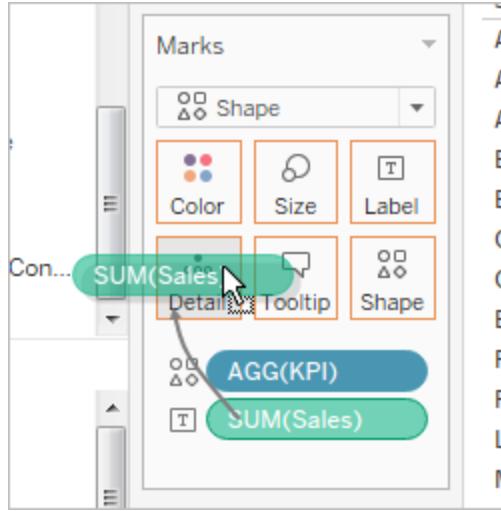
The Edit Shape dialog box should now look like this:



7. Click **OK** to close the Edit Shape dialog box.

The shapes in the view show the correct indicators. Now you just need to hide the sales numbers.

8. Drag **SUM(Sales)** on the Marks card to **Detail**.



You now have a completed view that show how individual products (sub-categories) are performing across all four regions:

Pages

iii Columns Region

Rows Sub-Category

Filters

Sheet 8

Sub-Catego..	Central	East	South	West
Accessories	✓	✓	✓	✓
Appliances	✗	✓	✗	✓
Art	✗	✗	✗	✗
Binders	✓	✓	✓	✓
Bookcases	✗	✓	✗	✓
Chairs	✓	✓	✓	✓
Copiers	✓	✓	✗	✓
Envelopes	✗	✗	✗	✗
Fasteners	✗	✗	✗	✗
Furnishings	✗	✓	✗	✓
Labels	✗	✗	✗	✗
Machines	✓	✓	✓	✓
Paper	✗	✗	✗	✓
Phones	✓	✓	✓	✓
Storage	✓	✓	✓	✓
Supplies	✗	✗	✗	✗
Tables	✓	✓	✓	✓

The view may not be terribly exciting on its own, but it would make a nice addition to a dashboard that showed other performance metrics.

Create a Pareto Chart

A Pareto chart is a type of chart that contains both bars and a line graph, where individual values are represented in descending order by bars, and the ascending cumulative total is represented by the line. It is named for Vilfredo Pareto, an Italian engineer, sociologist, economist, political scientist, and philosopher, who formulated what has become known as the Pareto principle. Pareto made the observation that 80% of land was typically owned by 20% of the population. Pareto extended his principle by observing that 20% of the peapods in his garden contained 80% of the peas. Eventually, the principle was further extrapolated by others to propose that for many events, roughly 80% of the effects come from 20% of the causes. In business, for example, 80% of profits not infrequently derive from 20% of the available products.

In Tableau, you can apply a table calculation to sales data to create a chart that shows the percentage of total sales that come from the top products, and thus identify the key segments of your customer base that are most important for your business's success.

The procedure uses the **Sample - Superstore** data source provided with Tableau Desktop.

Preparing for the analysis

Before starting your analysis, decide what questions you want answered. These questions determine the category (dimension) and number (measure) on which to base the analysis. In the example to follow, the question is which products (captured by the **Sub-Category** dimension) account for the most total sales.

At a high level, the process requires you to do the following:

1. Create a bar chart that shows **Sales** by **Sub-Category**, in descending order.
2. Add a line chart that also shows **Sales** by **Sub-Category**.
3. Add a table calculation to the line chart to show sales by Sub-Category as a **Running Total**, and as a **Percent of Total**.

The scenario uses the **Sample - Superstore** data source provided with Tableau Desktop

Create a bar chart that shows Sales by Sub-Category in descending order

1. Connect to the **Sample - Superstore** data source.
2. From the Dimensions area of the **Data** pane, drag **Sub-Category** to **Columns**.
3. From the Measures area of the **Data** pane, drag **Sales** to **Rows**.
4. Click **Sub-Category** on **Columns** and choose **Sort**.

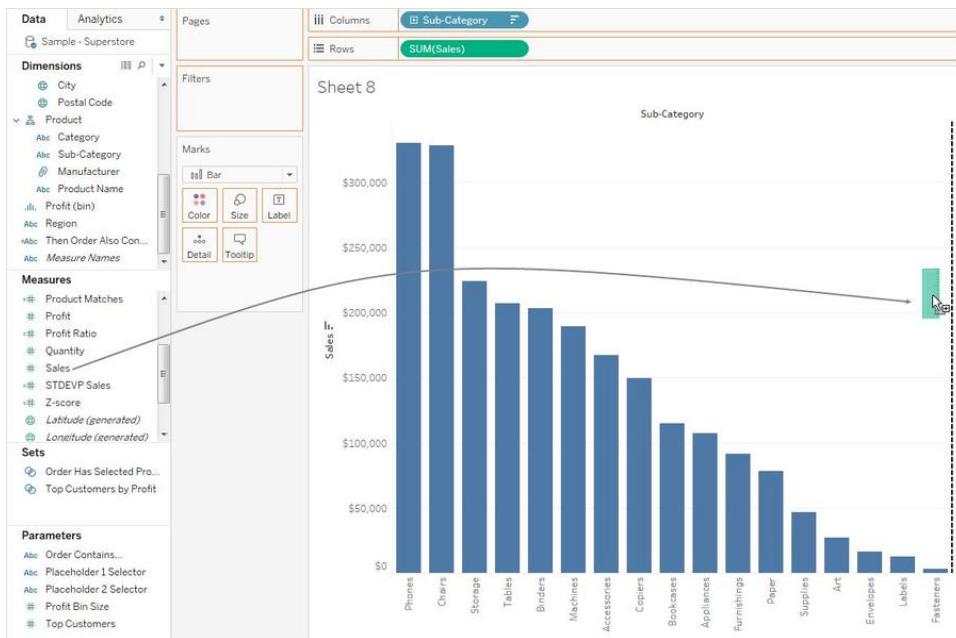
In the Sort dialog box, do the following:

- a. Under **Sort order**, choose **Descending**.
- b. Under **Sort by**, choose **Field**.
- c. Leave all other values unchanged, including **Sales** as the selected field and **Sum** as the selected aggregation.
- d. Click **OK** to exit the Sort dialog box.

Products are now sorted from highest sales to lowest.

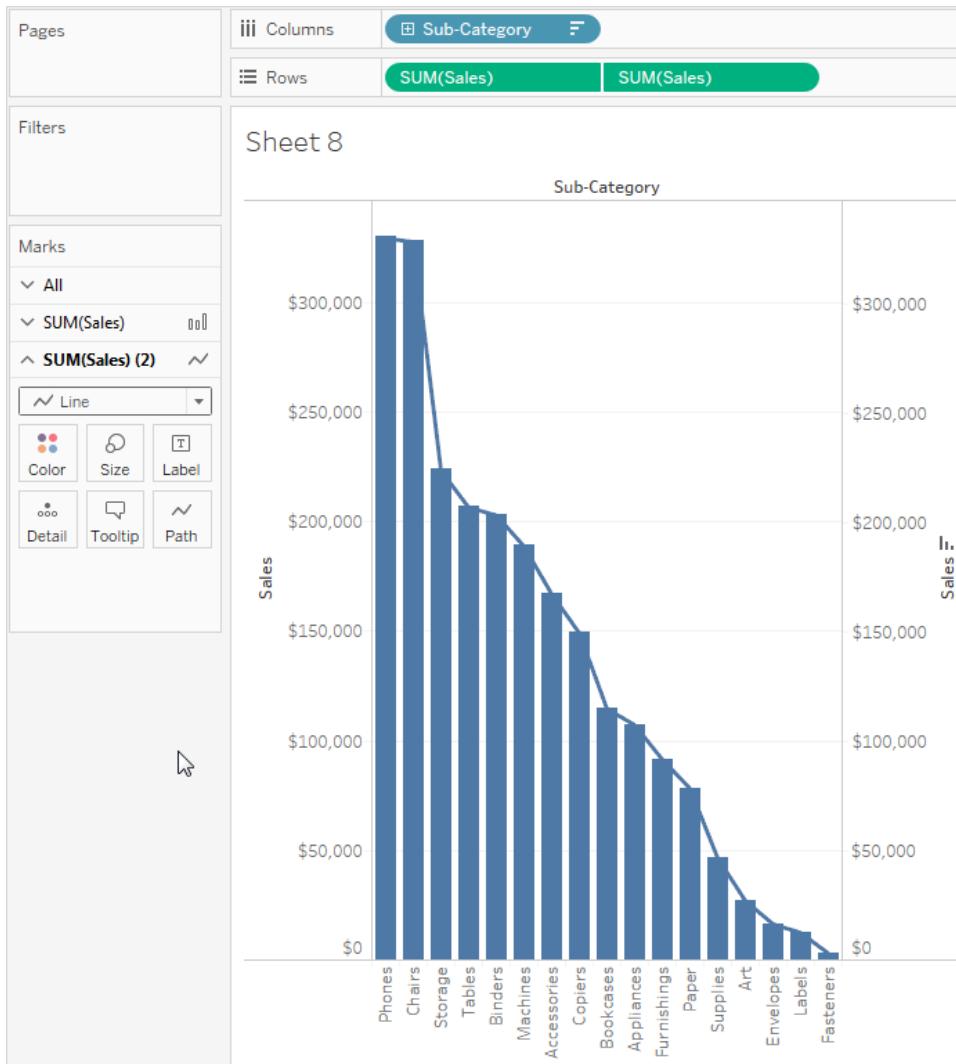
Add a line chart that also shows Sales by Sub-Category

1. From the Measures area of the Data pane, drag **Sales** to the far right of the view, until a dotted line appears:



2. Drop Sales, to create a dual-axis view. It's a bit hard to see that there are two instances of the Sales bars at this point, because they are configured identically.
3. Select **SUM(Sales) (2)** on the Marks card, and change the mark type to **Line**.

This is what the view should look like at this point:



Add a table calculation to the line chart to show sales by Sub-Category as a running total, and as a percent of total

1. Click the second copy of **SUM(Sales)** on **Rows** and choose **Add Table Calculation**.
2. Add a primary table calculation to **SUM(Sales)** to present sales as a running total.

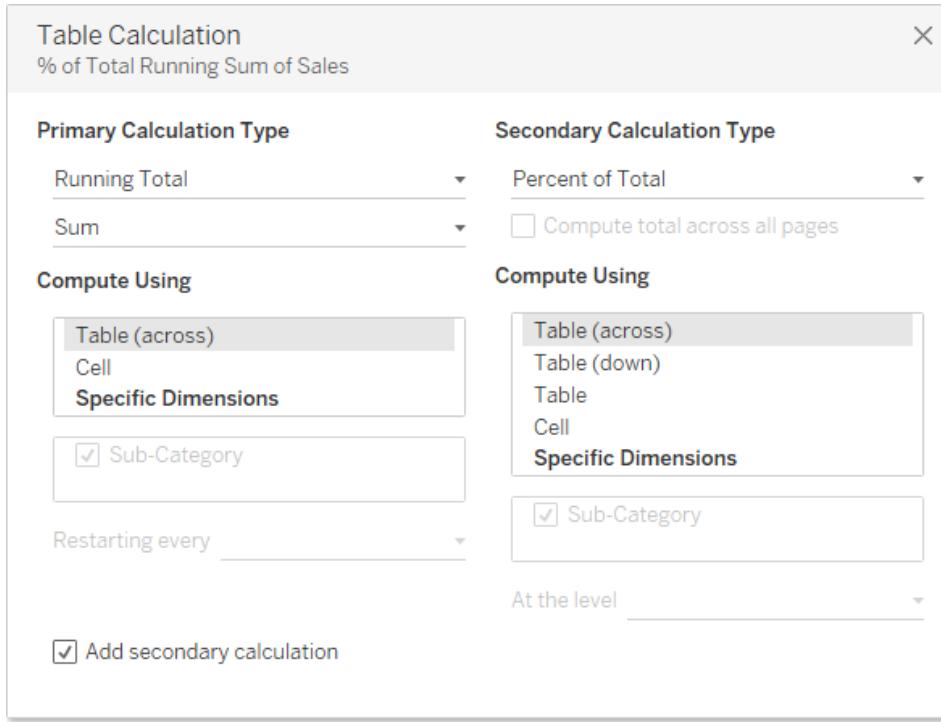
Choose **Running Total** as the **Calculation Type**.

Do not close the Table Calculation dialog box.

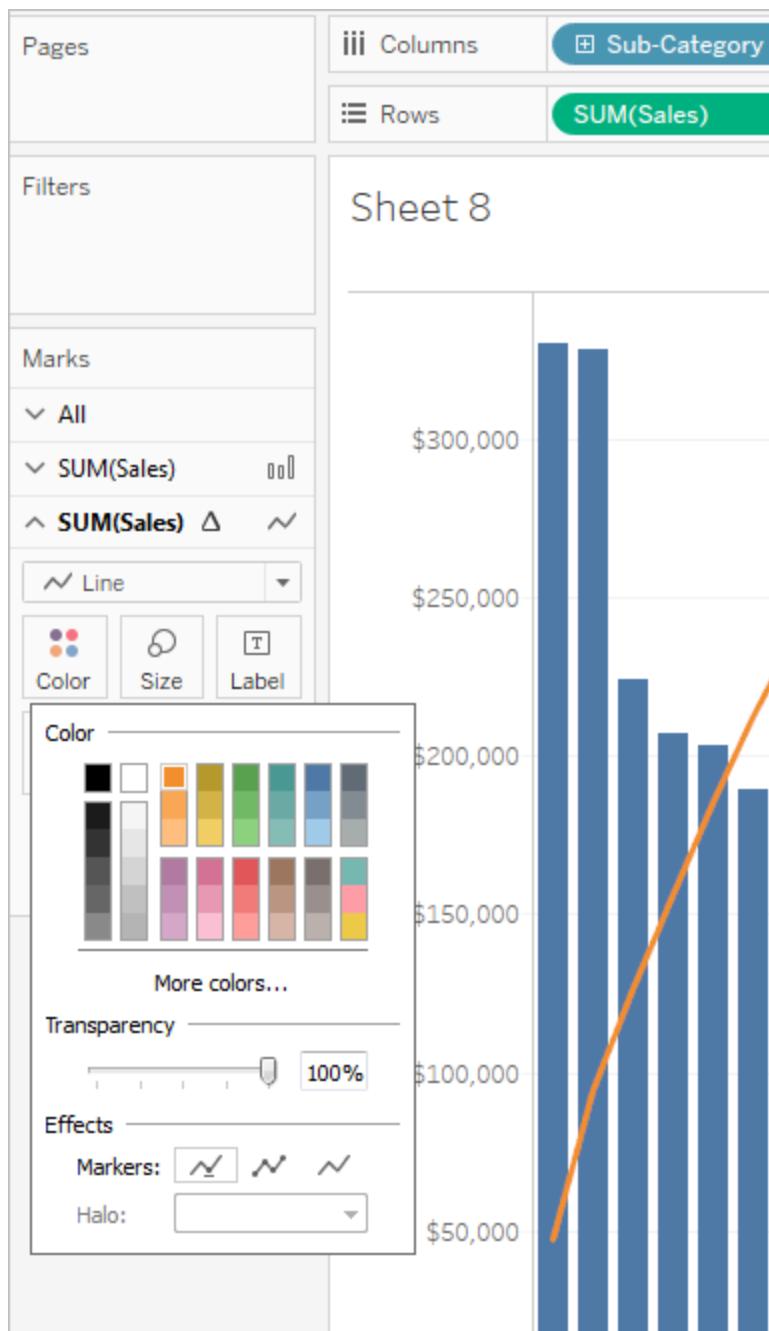
3. Add a secondary table calculation to present the data as a percent of total.

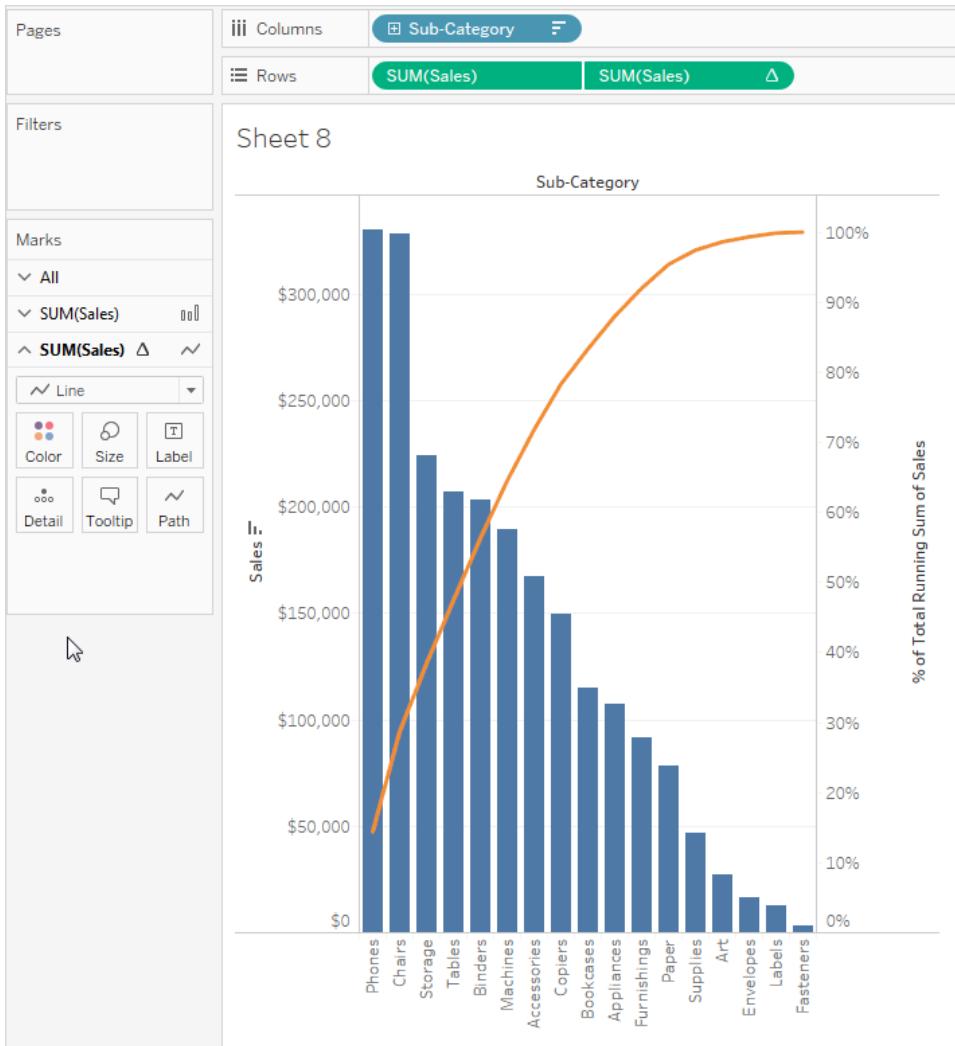
Click **Add Secondary Calculation** and choose **Percent of Total** as the **Secondary Calculation Type**.

This is what the Table Calculation dialog box should look like at this point:



4. Click the X in the upper-right corner of the Table Calculations dialog box to close it.
5. Click Color in the Marks card to change the color of the line.





Additional information

For additional tips on how you would compare the percentage of sales with the percentage of products, or draw reference lines that help make the 80-20 principles more apparent, see the Tableau On-Demand Training video titled [Pareto](#).

Create a Population Pyramid

A population pyramid, also known as an age structure diagram, shows the distribution of various age groups in a population.

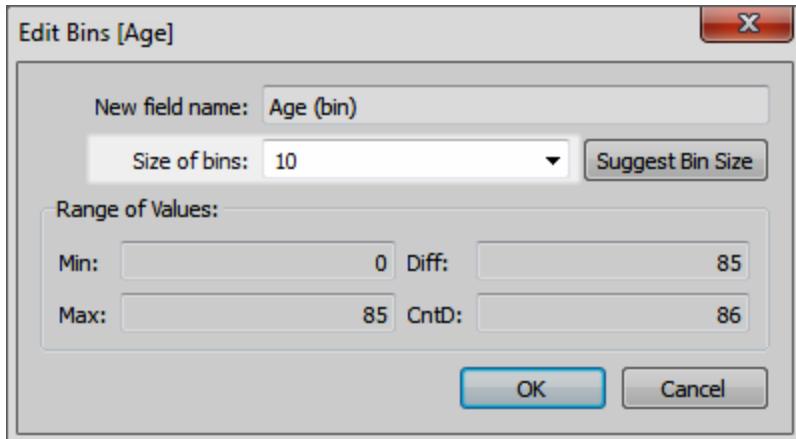
A common distribution often used with this type of visualization is female and male populations by age. To create a population pyramid using Tableau, first separate the population (measure) into two groups, females and males, and then create "bins" for the age cohorts you want to represent in the population pyramid.

For example, suppose you are working with a table from the United States Census Bureau that contains population, gender, and age data.

To create a bin and divide a measure into two groups.

1. Download and open the following workbook from Tableau Public.
 1. Go to
<https://public.tableau.com/profile/tableau.docs.team#!/vizhome/CreatePopulationPyramid/Sheet1>.
 2. Click **Download Workbook** in the upper-right corner and then open the workbook.
2. Select **Worksheet > Clear > Sheet**.
3. In the Data pane, right-click the **Age** field and select **Create > Bins**.
4. In the Create Bins dialog box, enter a bin size based on the age groups you're interested in focusing on, and then click **OK**.

In this example, the bin size is 10. This means that the age cohorts are defined by increments of 10 years.



5. Drag the bin you just created to the Rows shelf.

Columns	
Rows	Age (bin)
0	Abc
10	Abc
20	Abc
30	Abc
40	Abc
50	Abc
60	Abc
70	Abc
80	Abc

6. Select **Analysis > Create a Calculated Field**, and then do the following:

- Enter a name for the calculation. For this example, enter **Male Population**.
- Enter a formula similar to the following to isolate the male constituent from population:

```
IF [Gender] = 1 THEN [ESTBASE2010] END
```

In this case, the census data has defined the Gender value for male as "1." The field "ESTBASE2010" contains estimated population values.

7. Similar to the step 5, select **Analysis > Create a Calculated Field**, and then do the following:

- Enter a name for the calculation. For this example, enter **Female Population**.
- Enter a formula similar to the following to isolate the female constituent from population:

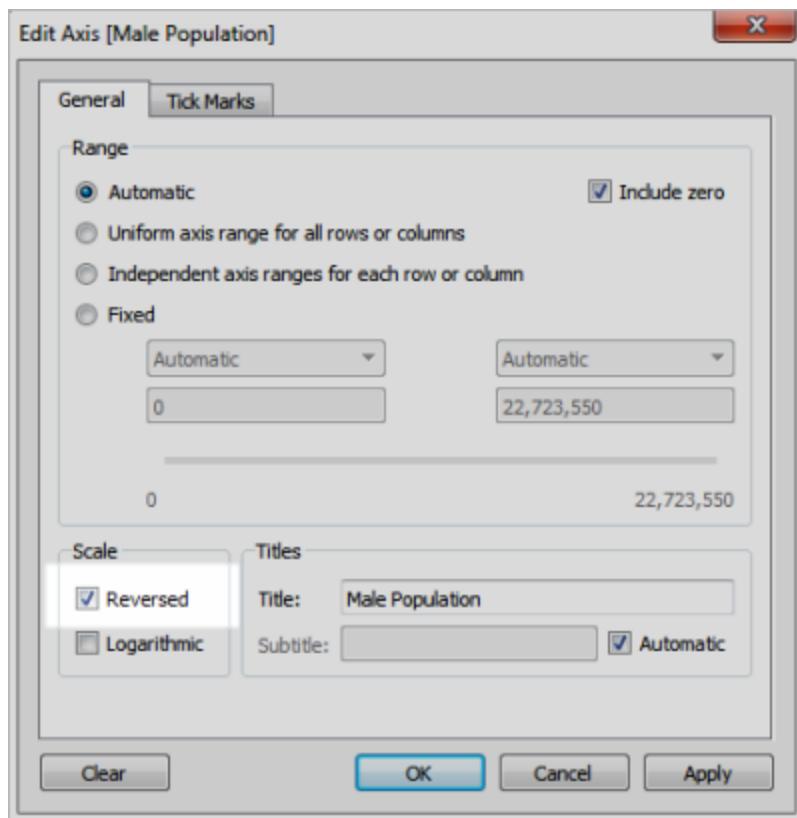
```
IF [Gender] = 2 THEN [ESTBASE2010] END
```

In this case, the census data has defined the Gender value for female as "2." The field "ESTBASE2010" contains estimated population values.

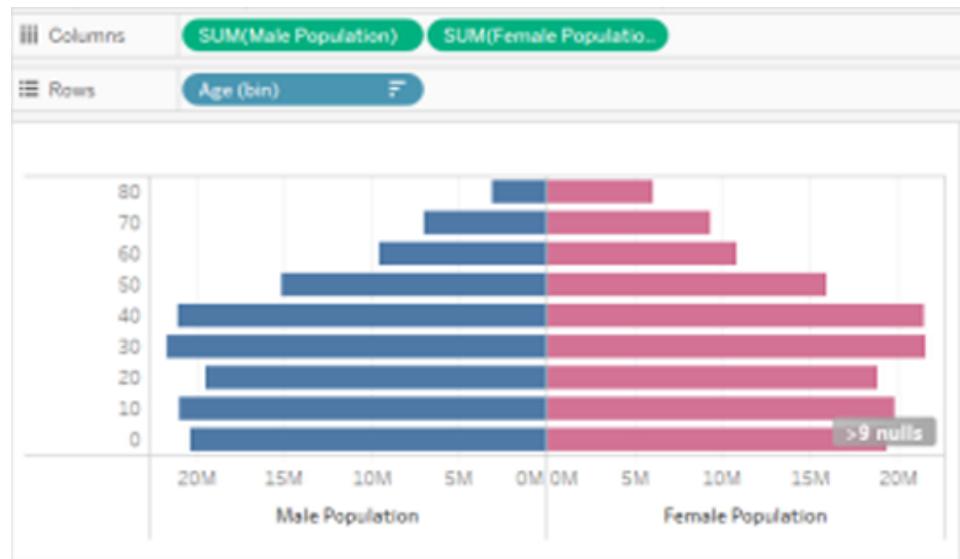
8. Drag the calculated fields you created to the Columns shelf and the **Gender** field to the Color shelf.

Note: To change the color assignments, click the Color shelf and then click **Edit Colors**.

9. Right-click the axis for Male Population, select **Edit Axis**, and then select the check box for **Reversed** to reverse the order the values display on the axis, and then click **OK**.



After changing the sort order of the bin, the population pyramid looks like the following:



Create a Co-Occurrence Visualization Using a Parameter and a Set

You can create co-occurrence visualizations in Tableau. With a co-occurrence visualization, users can pick one field value and then see which other fields values it co-occurs with, and how often. One practical application of such analysis is a market basket analysis, which you can use to discover and understand customer purchasing behavior. You can use a market basket analysis to answer questions like the following:

- How many people bought both Product A and Product B?
- What other products do people who purchased Product A generally buy?
- Which other courses do students who enrolled in Course A frequently enroll in?

Follow the steps in this article to create a co-occurrence visualization (in this case, a market basket analysis view) using a parameter, calculated fields, and a set. At a high level, the steps involved are:

1. Create a parameter that you will use to dynamically modify the view based on the item you select.
2. Create calculated fields that you will use to return which items are also ordered when a particular item is ordered.
3. Create a set to determine whether an order has the item that was selected in the parameter control.
4. Build the view to display which items are also contained in an order with the selected item.

The scenario uses the **Sample - Superstore** data source provided with Tableau Desktop. The details are covered in the following sections.

Create a Parameter

Follow these steps to create a parameter and then display a parameter control that users can use to pick a **Sub-Category** value.

1. Connect to the **Sample - Superstore** data source.
2. Right-click (control-click on a Mac) in the **Data** pane and select **Create > Parameter**.
3. In the Create Parameter dialog box, do the following:
 - a. Name the parameter **Order Contains**.
 - b. For the **Data Type**, choose **String**.
 - c. For **Allowable Values**, choose **List**.
 - d. In the **List of values** section click **Add from Field > Sub-Category**.
 - e. Click **OK**.

4. Right-click the **Order Contains** parameter in the Data pane and choose **Show Parameter Control**.

Step 2: Create Calculated Fields

Follow these steps to create the calculated fields that you will use to show which items are also ordered when the item selected by the user (via the parameter control) is ordered.

1. Create a calculated field to identify products that the order also contains (in addition to the one that the user selects).

Choose **Analysis > Create Calculated Field** to open the calculation editor. Name the calculated field **Then Order Also Contains** and type or paste the following in the formula area:

```
IF [Sub-Category] <> [Order Contains] THEN [Sub-Category] END
```

You may have to replace < ; > with <> after you paste.

2. Create another calculated field to identify matching products.

Name the field **Product Matches** and type or paste the following in the formula area:

```
IF [Sub-Category] = [Order Contains] THEN 1 END
```

Create a Set

1. Now create a set to determine whether an order has the item that was selected in the parameter control.
2. Hover the cursor over the **Order ID** dimension in the Data pane, click the down arrow at the right end of the field, and choose **Create > Set**.

The screenshot shows the Tableau Data Source pane on the left, displaying various dimensions and measures. The 'Order' dimension is expanded, and 'Order ID' is selected. A context menu is open over 'Order ID', with the 'Create' option highlighted. A sub-menu has opened under 'Create', with 'Set...' highlighted. The main menu also includes options like 'Duplicate', 'Rename', 'Hide', 'Aliases...', 'Transform', 'Convert to Measure', 'Change Data Type', 'Geographic Role', 'Default Properties', 'Group by', 'Folders', 'Hierarchy', 'Replace References...', and 'Describe...'. The right side of the interface shows 'Sheet 1' with a grid and a placeholder text 'Drop field here'.

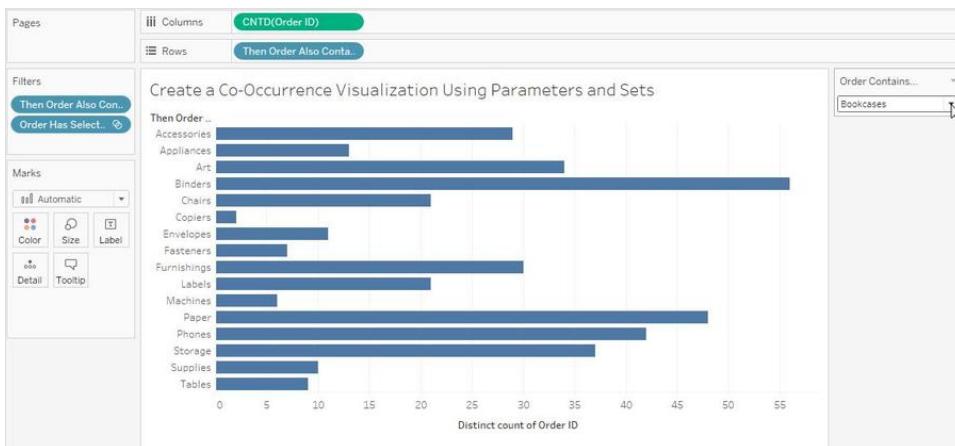
3. In the Create Set dialog box, type **Order Has Selected Product** in the **Name** text box.
4. Go to the Condition tab, select **By field**, and in the drop-down lists make the following selections and entries to build the condition:
 - a. In the first drop-down list, select **Product Matches**.
 - b. In the second drop-down list, select **Sum**.
 - c. In the next drop-down list, select **>=**.
 - d. In the last text box, type **1**.
 - e. Click **OK**.
5. Click **OK**.

Build the View

Finally, build the view to display which items are also contained in an order with the selected item.

1. Drag **Then Order Also Contains** to Columns.
 2. Drag **Order ID** to Rows.
- In the warning dialog box, click **Add all members**.
3. Click the **Order ID** field on Rows and choose **Measure > Count (Distinct)** to change the aggregation.
 4. Right-click (control-click on a Mac) the Null bar on the x-axis and choose **Exclude**.
 5. Drag the **Order Has Selected Product** set to the **Filters** shelf.
 6. Press **Ctrl+W** to swap the fields on Rows and Columns.

You (or your users) can now use the **Order Contains** parameter control to select an item in an order, and then see a bar chart showing which other items are also included in orders with the selected item.



Visualize Benford's Law

Benford's Law is a mathematical law that states that the leading, or left-most, digit in many real-life data sources is distributed in a very specific manner. Specifically, the number 1 occurs as the leading digit about 30% of the time, and as numbers get larger they occur less frequently, with the number 9 occurring less than 5% of the time. When fraudsters are fabricating data, they may not know to create fake data that conforms to Benford's law, and in some cases this makes it possible to detect fake data or at least to create doubt about its veracity.

This article describes how to apply Benford's Law to Sales data, using the **Sample - Superstore** data source provided with Tableau Desktop.

The process requires you to do the following:

1. Create calculated fields to use in your view
2. Set up the view

The following sections break these procedures down into specific instructions.

[Create calculated fields to use in your view](#)

Follow these steps

1. Choose **Analysis > Create Calculated Field** to open the calculation editor. Name the calculation **Leftmost Integer** and type or paste the following in the formula area:

```
LEFT(STR([Sales]),1)
```

2. Create a second calculated field and name it **Benfords Law**. Type or paste the following in the formula area:

```
LOG(INT([Leftmost Integer])+1)-LOG(INT([Leftmost Integer]))
```

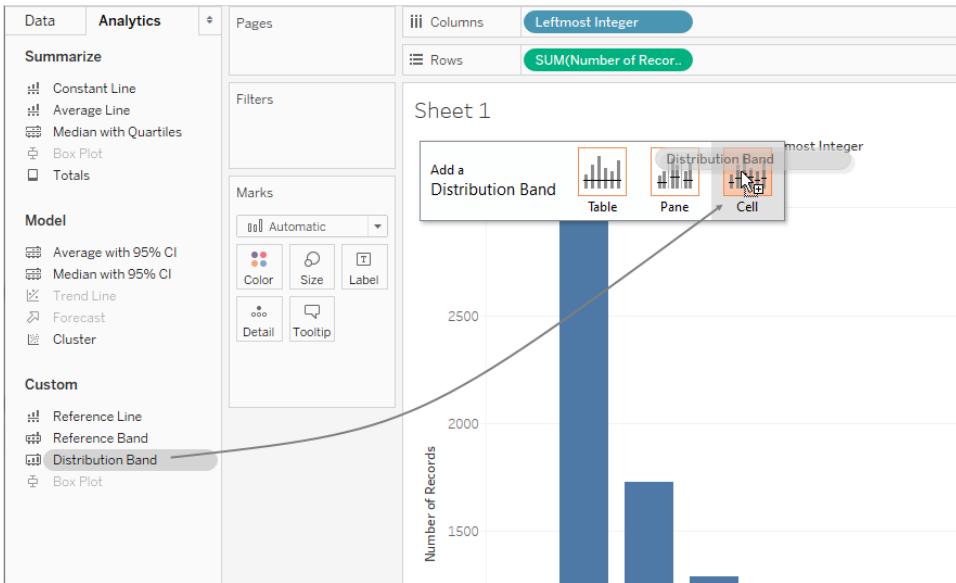
[Set up the view](#)

Follow these steps.

1. Drag **Leftmost Integer** from the **Dimensions** area of the **Data** pane to **Columns**, and then drag **Number of Records** from the **Measures** area of the **Data** pane to **Rows**.
2. Click **SUM(Number of Records)** on Rows and choose **Quick Table Calculation > Percent of Total**.

Your view now shows the distribution of first digits, and the size of the bars (decreasing from left to right) suggests that the data in this case conforms to Benford's law. But we can do more to frame the data by adding reference distributions.

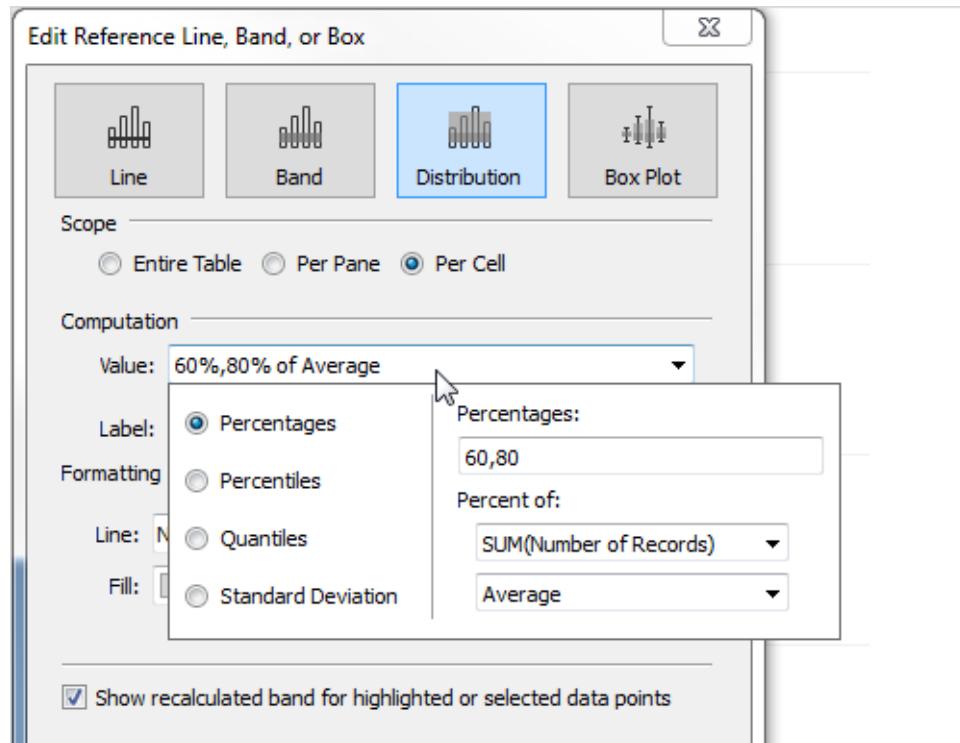
3. Drag **Benfords Law** from the Measures area of the Data pane to Detail on the Marks card, and then click **Benfords Law** on the Marks card and choose **Measure > Minimum**.
4. Switch from the **Data** pane to the **Analytics** pane and drag **Distribution Band** into the view. Drop it on **Cell**.



Note: Distribution Bands are supported on web platforms starting with Tableau 10.2.

5. In the Edit Reference Line, Band, or Box dialog box, do the following:

- Click in the Value field to view an additional set of options:



- In the Percentages area, type 80,100,120.

This specifies that you want bands spanning from 80 to 100 percent, and from 100 to 120 percent. Next you will specify what value the percentages are referencing.

- In the **Percent of** field, choose **MIN(Benfords Law)**.

The Value field should now read 80%,100%,120% of Average Min. Benfords Law.

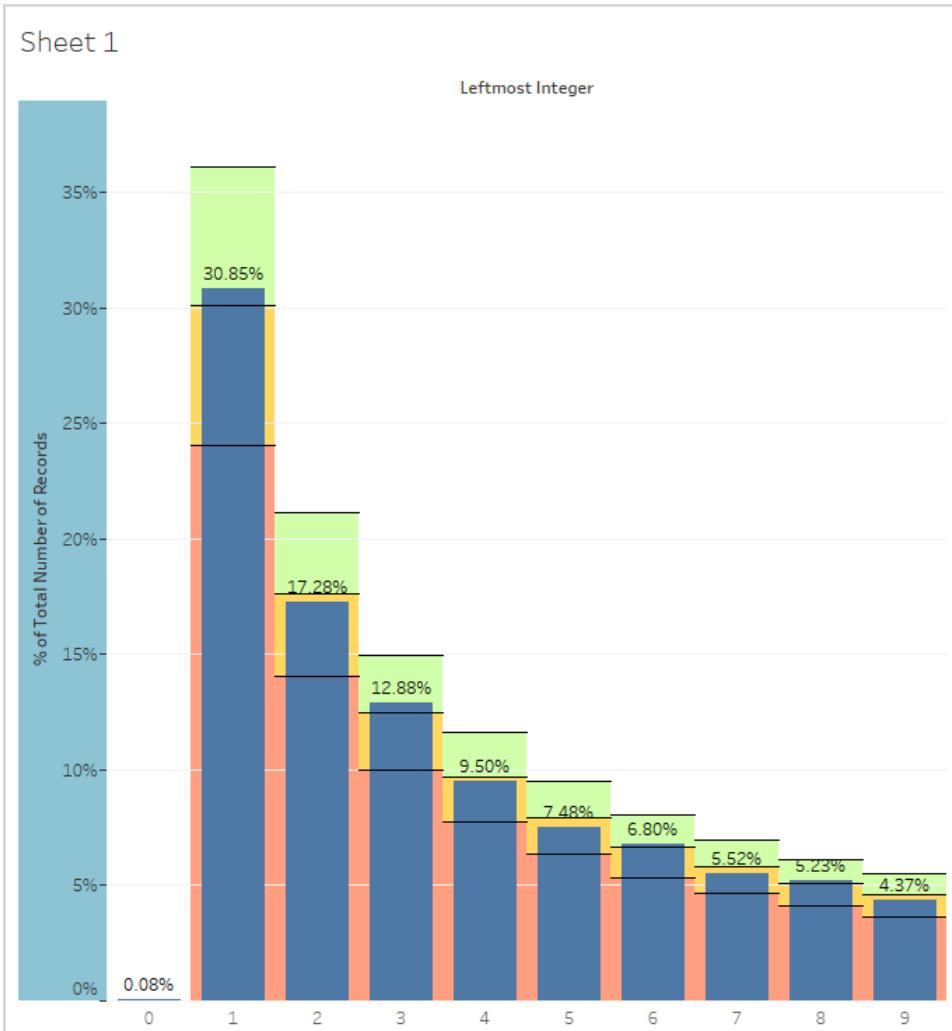
The remaining steps configure the appearance of the reference bands.

- Set **Label** to **None**.
- Set **Line** to the thinnest available line.
- Choose **Fill Below**.
- From **Fill**, select **Stoplight**.
- Click **OK** to exit the Edit Reference Line, Band, or Box dialog box.

- Click the toolbar button to display mark labels:



The finished view should look like this:



Even though Superstore is demo data, it's realistic as far as conforming to Benford's law. The blue bars that indicate actual percentages of initial digits align very well with the 100% value (that is, the line that separates the green zone from the yellow zone in the distribution bands) that shows expected Benford values in the view.

Present Your work

The ideal visualization combines science with art. With formatting, dashboards, and stories you can make your data discoveries clearer, more persuasive, and beautiful.

Format Your Work

Formatting is an important part of both your analysis and presentation. You can format almost everything you see on a worksheet including the fonts, shading, alignment, borders, and graph

lines. For example, in a text table you may want to add banded shading to help you visually separate consecutive groups of rows or columns. In a scatter view with reference lines you may want to change the line thickness and color. All of these settings can be changed using the Format window.

Most often you will want to specify format settings for the entire worksheet, all rows, or all columns. However, Tableau also allows you to format individual parts of the view as well. For example, you can format specific fields, resize the cells and the table, and edit individual axes.

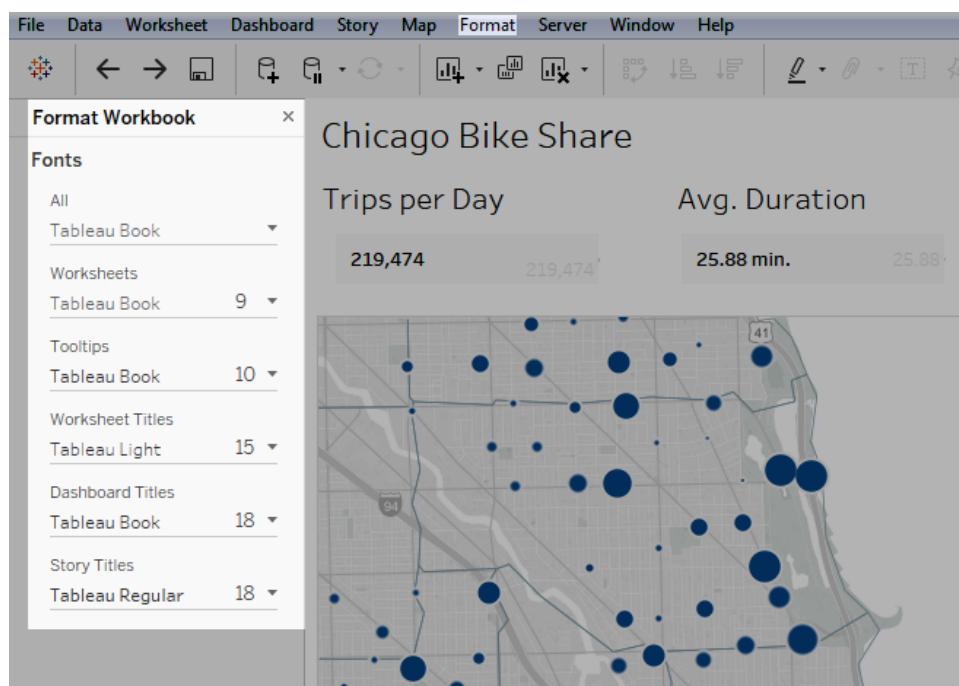
Visual Best Practices

The fonts, colors, shading, alignment, borders, and grid lines in your visualization are important parts of both your analysis and the story you're telling. Tableau products are designed so that you can create great-looking visualizations that use visual best practices by default, freeing you from the need to think about things like fonts and colors—unless you want to.

If you do want to customize, you can control the look of almost everything you see on a worksheet. This topic outlines visual best practices and tips to keep in mind while you customize, from ideal workflow to how to get the most out of tooltips.

Format from largest to smallest

As you change the look and feel of your work, use a "biggest to smallest" workflow. Start by formatting fonts and titles at the workbook level, then move on to the worksheet level. Save formatting the individual parts of a view for last. A workbook is the largest possible "container" for formatting changes and making changes at that level first will save you time.



See [Format at the Workbook Level](#) on page 1059, [Format at the Worksheet Level](#) on page 1062, and [Format Parts of a View](#) on page 1068 for details.

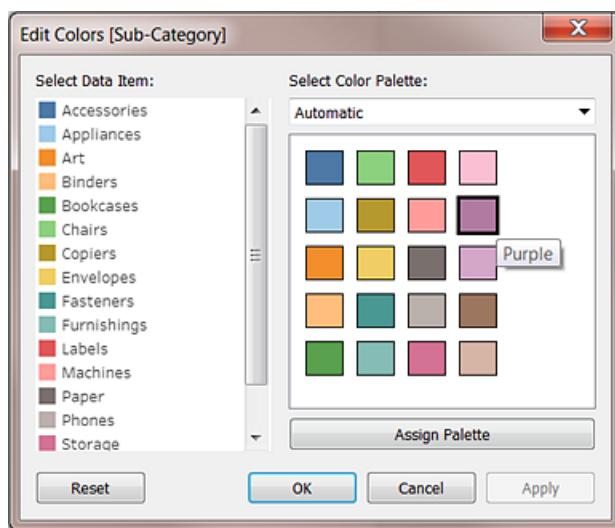
Change color with purpose

If you want to change the colors used by your visualization, Tableau makes it easy: just click the **Color** card, then **Edit Colors**. For example, maybe you want to emphasize a key finding. You can use neutral colors with a single, bright color to highlight what you want your viewers to pay attention to. If you want to match your company's brand, you can create a custom palette. See [Create Custom Color Palettes](#) on page 1100 for steps.

When you change colors, keep the following best practices in mind..

Discrete fields and categorical palettes

Discrete fields are ones where the field's values are unique. One example of this is customer names. Tableau automatically associates discrete fields with *categorical* palettes, which have colors that are designed to be distinct from one another yet also work well together in the visualization as a whole.

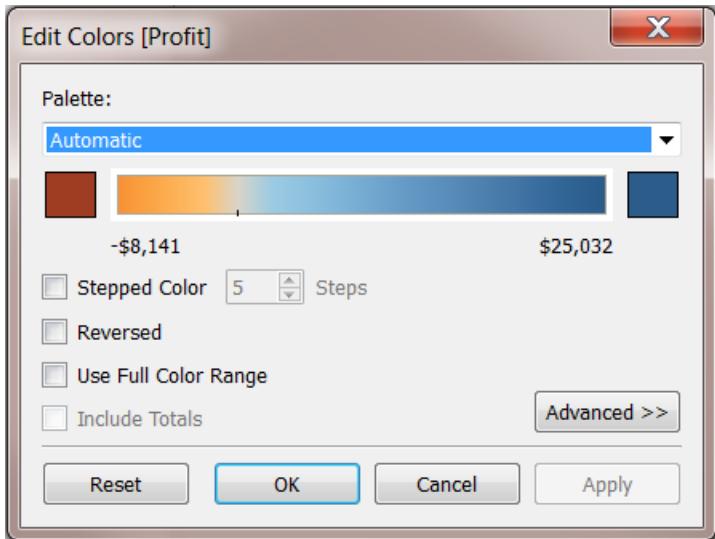


Continuous fields and quantitative palettes

With continuous fields, the field's values are part of a whole (such as sales over time). Tableau automatically associates these fields with *quantitative* palettes, where colors are along a continuum or range. If the field's values are positive, a single-color range is best:



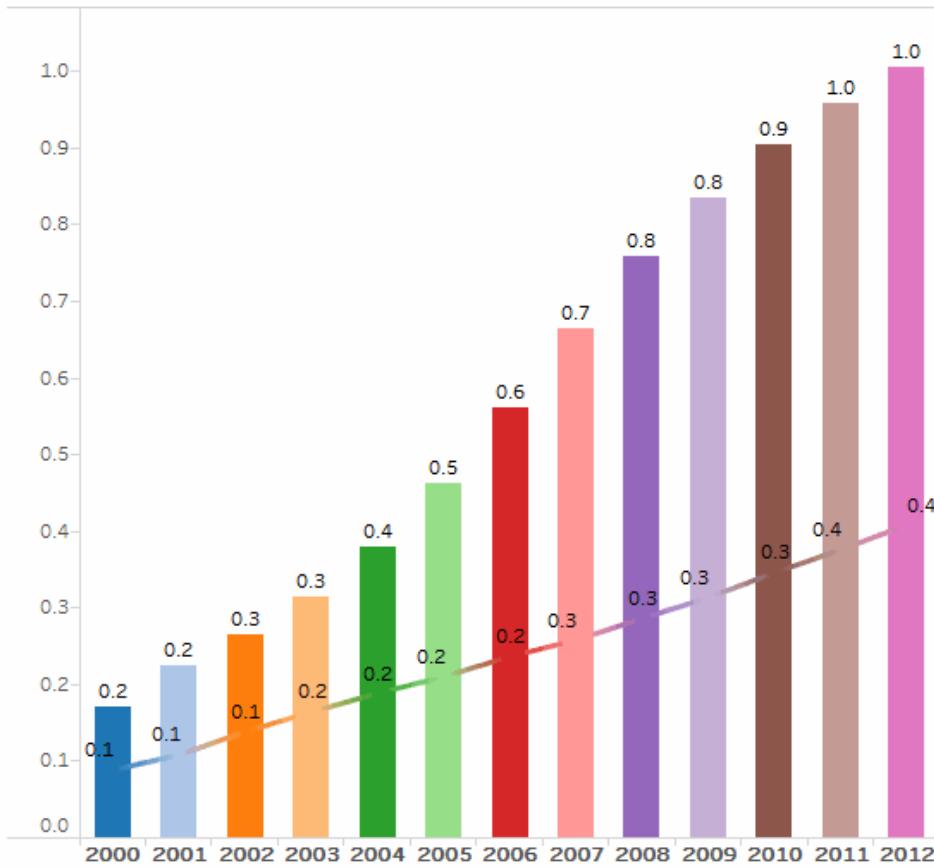
If there are both positive and negative values, choose a two-color range:



Limit colors

Color used correctly can enhance analysis. Too many colors can create visual overload for your users and impede analysis. In the view below, the stacked bars represent cell phone subscriptions and the trend line is internet usage over time. In the first image, Years is on the **Colors** shelf. Notice how hard it is to pick out the trend line among all the other colors and how easy it is to spot the trend line in the second image, where there are only two colors.

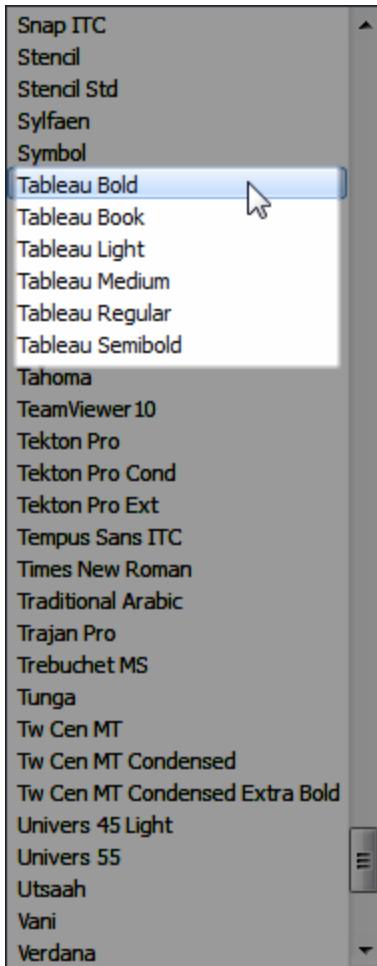
Too many values on colors



Click the image to replay it.

Use fonts that optimize online readability

The fonts in the Tableau typeface were designed to be paired with data visualizations and are optimized for legibility at small sizes. Tableau visualizations use them by default.



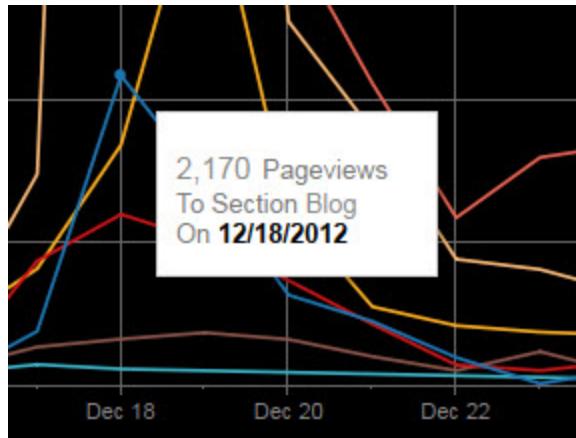
If you need to use a different font, choose one that optimizes readability online, such as the following:

- Arial
- Trebuchet MS
- Verdana
- Times New Roman
- Lucida sans

Reinforce your story with tooltips

A person looking at your dashboard will intuitively use his or her mouse to explore marks, and this makes tooltips appear. Tooltips appear by default for most views and they are a great way to reinforce your data story. You can customize your tooltips by clicking **Worksheet > Tooltip**.

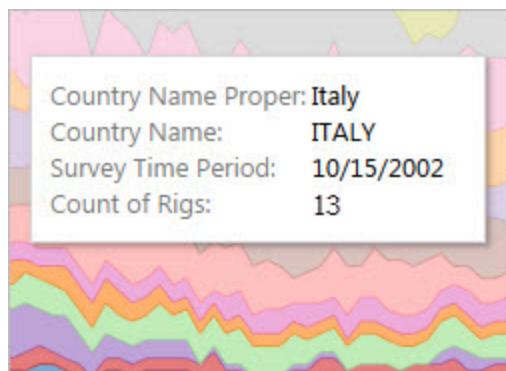
For example, here's a tooltip for a view on pageviews for a web site:



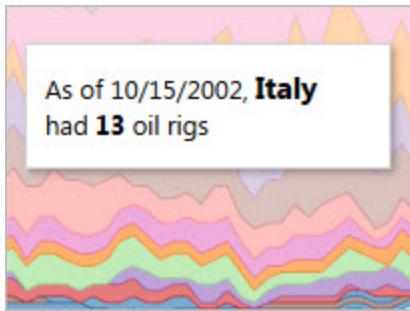
Here's an example of how the above tooltip could be rewritten and formatted to highlight what's important to the people looking at this view:



You can also rewrite your tooltips to tell a mini-story. For example, here's a tooltip about the number of oil rigs in Italy.



Below, the tooltip is rewritten as a phrase and key elements have been put in bold to draw the viewer's attention.



Consider your axes

By default, an axis range in Tableau will automatically adjust, based on the data in your visualization. Filtering and other actions can change the axis range. If you are trying to create a specific comparison between two views, changing axes ranges can make analysis difficult for your users. If this is the case, you can set your axes to a specific, fixed range. For steps on how to do this, see [Try it: Change the Axis Range on page 1050](#).

If you're working with data that has a very large range, a fixed axis may not be practical. If this is the case, add grid lines to your visualization. Grid lines can help your users stay oriented when the axis automatically adjusts. See [Format lines on page 1066](#) for more information.

Edit Axes

When you add a measure to the Columns or Rows shelf, you add an axis to the view. An axis shows data points that lie within a range of values.

For each axis you can specify the range, scale, and tick mark properties. Axis formatting options are available in the **Edit Axis** dialog box.

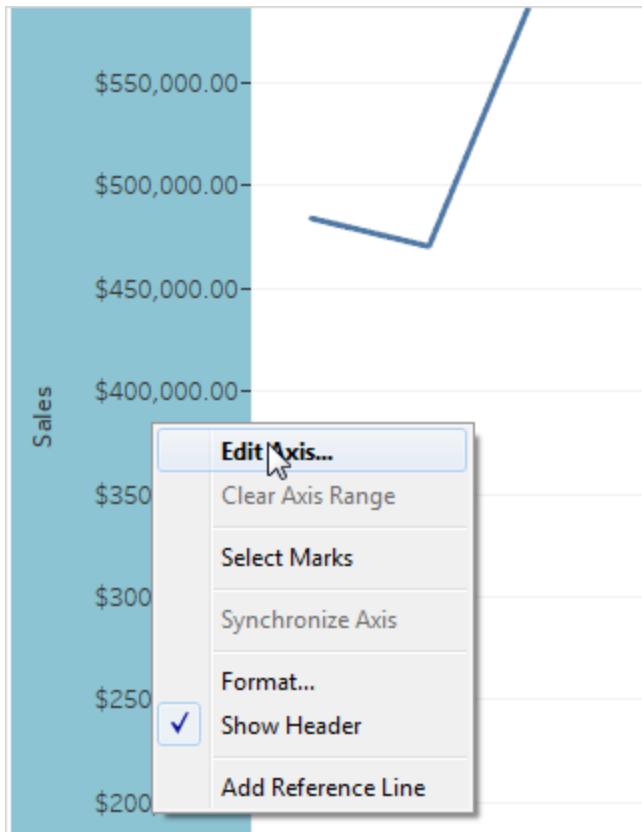
When you select an axis, the marks associated with the axis are not selected so that you can edit and format the axis without modifying the marks. To select the marks, right-click the axis and select **Select Marks**. You can also select each mark individually. For more information, see [Select Marks on page 566](#).

Change an axis range

You can limit the axis range to focus the view to where the data points lie. For example, you may have a view showing sales over four years. The automatic axis may range from 0 to \$750,000 but your sales never went below \$470,000. You could adjust the Axis Range so that it starts at \$470,000 thus focusing on where the data points actually lie.

To change an axis range:

1. Right-click (control-click on Mac) on the axis that you want to edit and select **Edit Axis**.



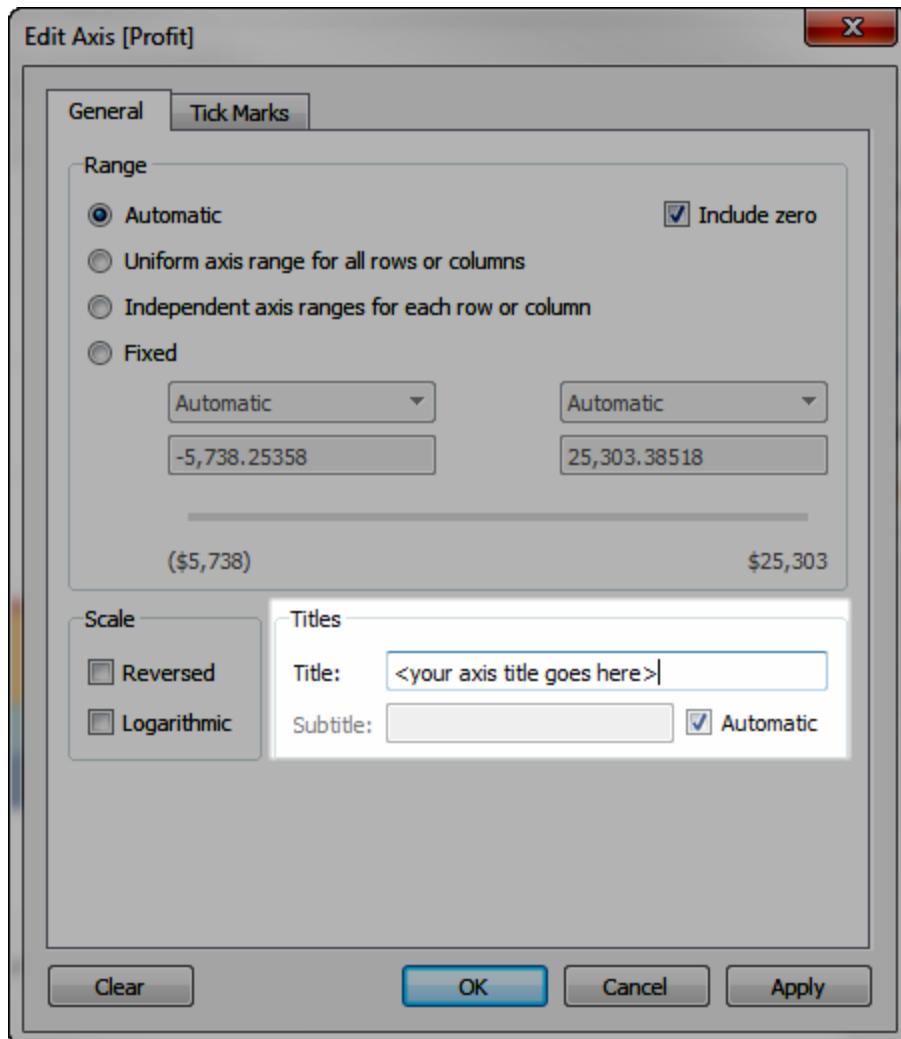
2. In the **Edit Axis** dialog box, select one of the following options:
 - **Automatic**. Automatically bases the axis range on the data used in the view.
 - **Uniform axis range for all rows or columns**. Sets the axis range uniformly to the maximum data range for all panes in the view.
 - **Independent axis ranges for each row or column**. Makes the axis range independent. The axis range varies for each pane in the view, depending on the range of data in each pane.
 - **Fixed**. Specifies to start and/or end the axis at a specific value. You can fix both ends of the axis or only one end. When you fix both ends of the axis, the axis range is determined by the values you specify. When you fix only one end of the axis, you must set the other end of the axis as automatic, uniform, or independent.
3. You can also specify whether to include zero. When you clear the **Include zero** check box, the axis range adjusts to show only the range of values in the data.
4. Click **OK**.

Change the appearance of an axis

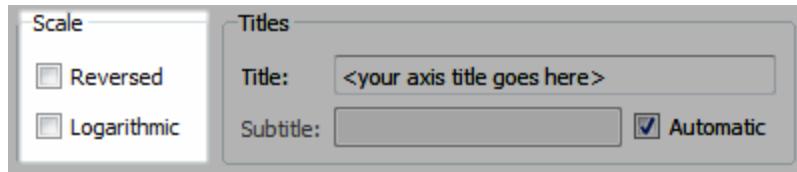
Every axis has a title that is automatically generated based on the fields in the view. You can specify a custom axis title and add a subtitle using the Edit Axis dialog box. You can also specify the scale of the axis, such as whether to use a logarithmic scale or whether to reverse the axis.

To change the appearance of an axis:

1. Right-click (control-click on Mac) on the axis that you want to edit and select **Edit Axis**.
2. In the Edit Axis dialog box, type a new title in the **Title** text box.



3. To add a custom subtitle, clear the **Automatic** check box.
4. To change the scale of the axis, select one of the following options:
 - **Reversed**- select this option to reverse the order of values on the axis.
 - **Logarithmic** - select this option to use a logarithmic scale on the axis.



5. Click **OK**.

Format tick marks

You can specify how often the tick marks are displayed along the axis. Tableau allows you to modify both the Major and Minor tick marks. Major tick marks are accompanied by unit labels while Minor tick marks simply represent smaller increments between the major marks. You can choose to use automatic or fixed tick marks or have none at all.

To format tick marks:

1. Right-click (control-click on Mac) the axis you want to edit and select **Edit Axis**.
2. In the Edit Axis dialog box, select the **Tick Marks** tab.
3. For both **Major** and **Minor tick marks**, select from one of the following options:
 - **Automatic** - select this option to automatically show tick marks based on the data in the view.
 - **Fixed** - select this option to specify how often the tick mark should display and the starting value.
 - **None** - select this option to hide the tick marks completely.
4. Click **OK**.

Try it: Change the Axis Range

In this example you will build three views using the same data. However, each view will use a different axis range format. These views use the **Sample-Superstore** data source to display the aggregated total sales for three product categories over the course of four years.

Build a view with a uniform axis range

A uniform axis range means that the same range is applied to each row or column in your view. The range is automatically generated based on the underlying data values.

1. Place the **Order Date** dimension on the **Columns** shelf and the **Category** dimension on the **Rows** shelf.
2. Place the **Sales** measure on the **Rows** shelf.

The measure is automatically aggregated as a summation and an axis is added to the view. By default the view uses a uniform axis range. Notice that the axis range is the

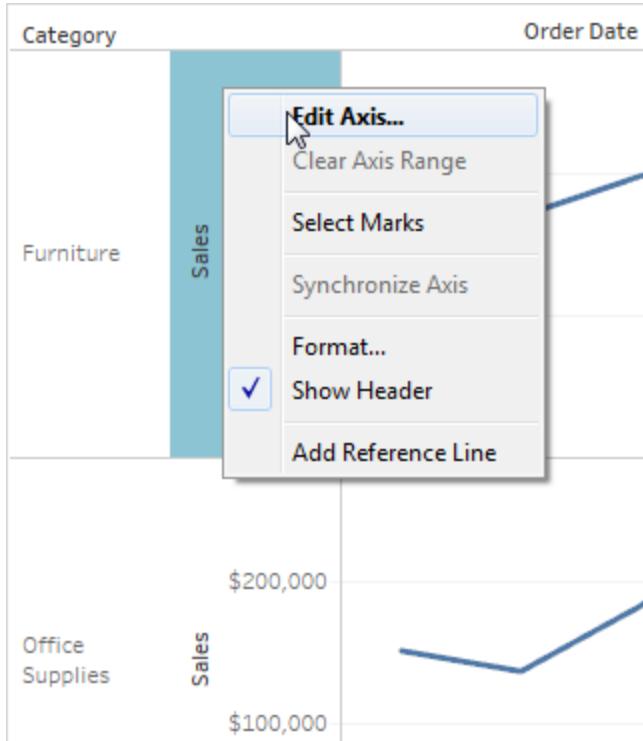
same, from zero to 250,000 for each product category.



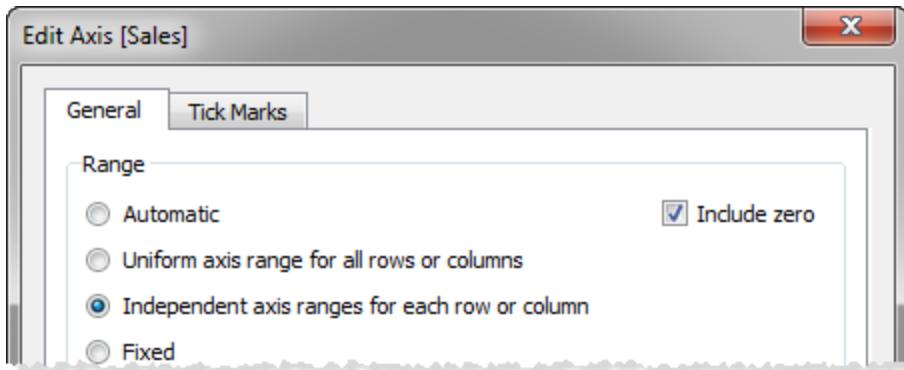
Build a view using independent axis ranges

When you use an independent axis range, each row or column will have its own axis range based on the underlying data values.

1. Right-click (control-click on Mac) the SUM(Sales) axis in the view and select **Edit Axis**.



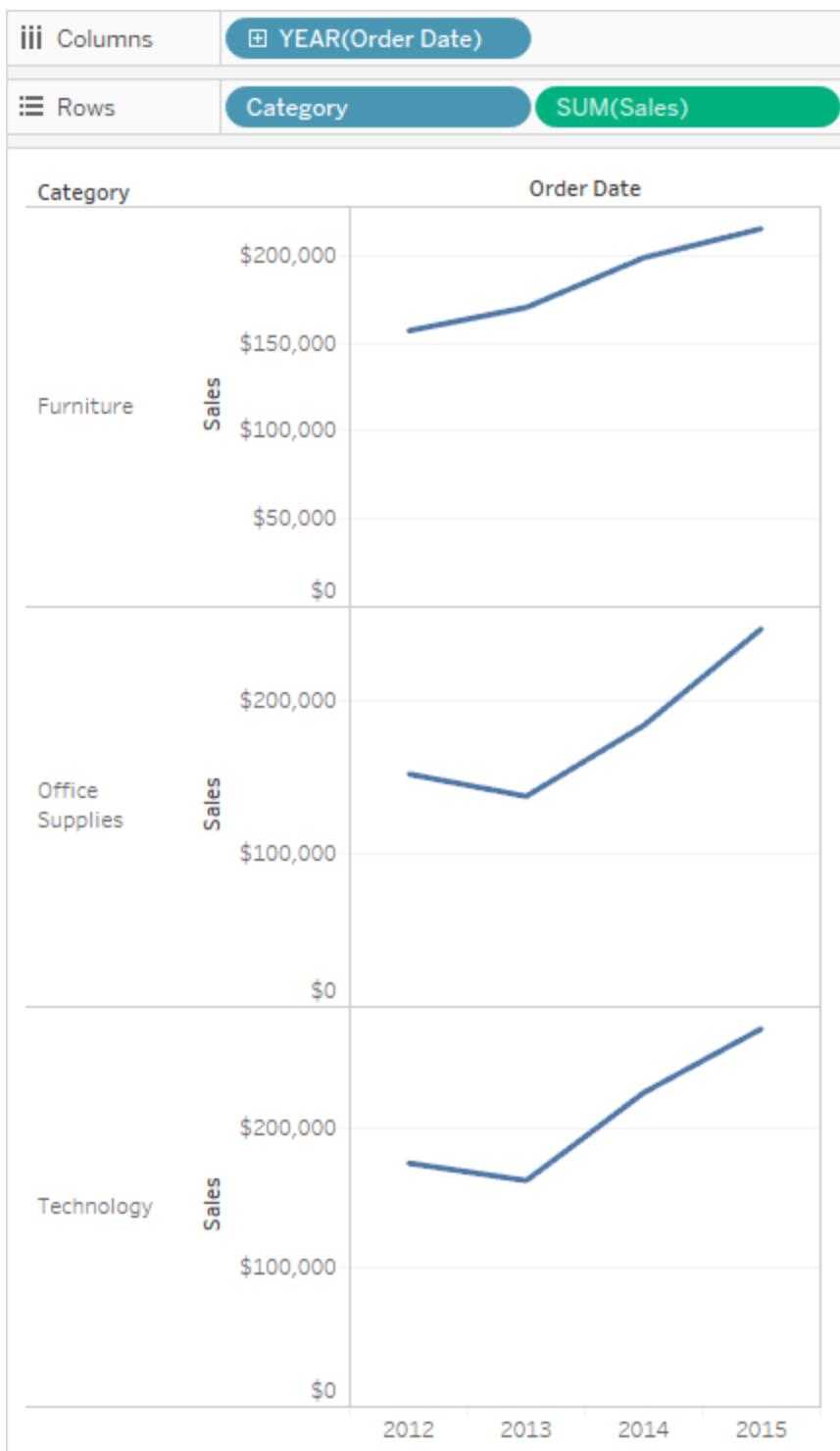
2. In the Edit Axis dialog box, select **Independent axis ranges for each row or column**.



3. Click **OK**.

The axis range for each product category are now independent from each other. The Technology and Office Supplies categories still range from zero to 250,000, but the

Furniture category ranges from zero to a little over 200,000.



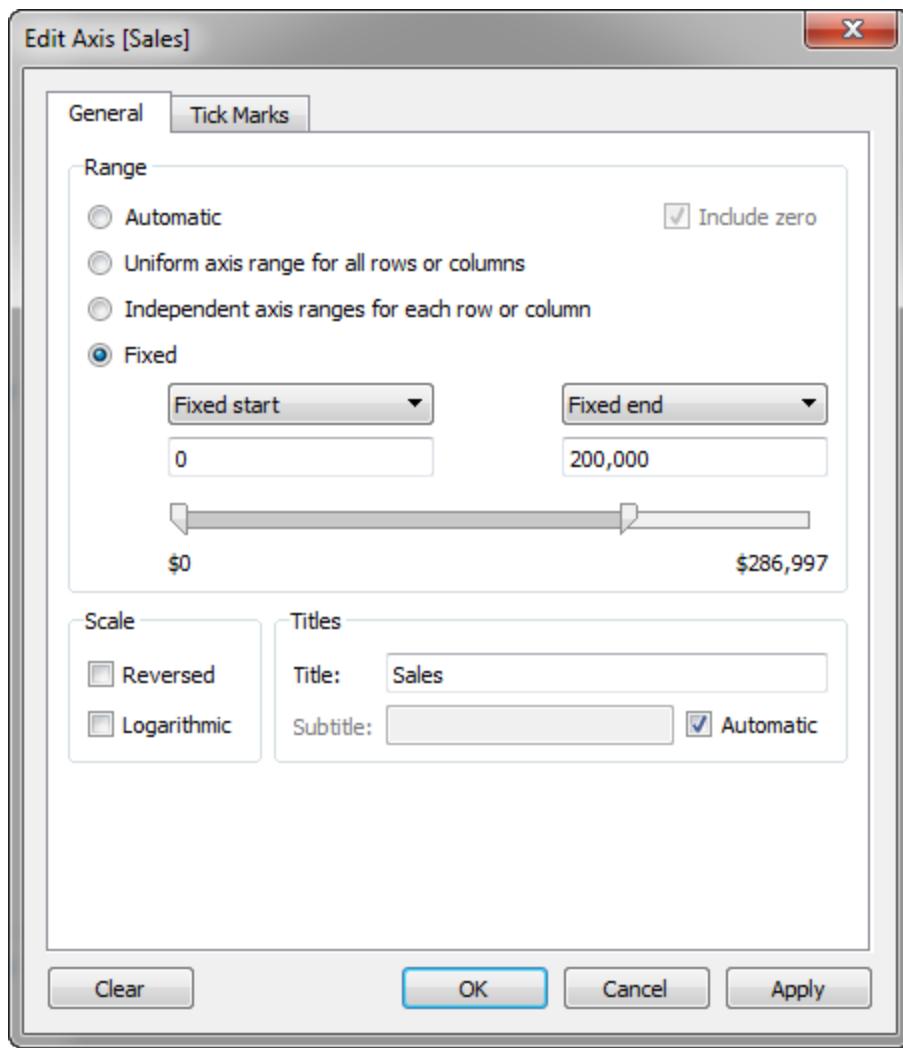
Build a view using a fixed axis range

You can fix one or both ends of an axis in order to customize how much of the data you can see in the view.

The first example shows how to fix both ends of the axis. The second shows how to fix only one end of the axis, and then define the axis range for all panes in the view.

To fix both ends of the axis:

1. Right-click (control-click on Mac) the **SUM(Sales)** axis in the view and select **Edit Axis**.
2. In the Edit Axis dialog box, select **Fixed**, and then define **Start** and **End** values either by typing into the text boxes or by dragging the sliders toward each other. For this example, type **0** for **Start** and **200,000** for **End**.



3. Click **OK**.

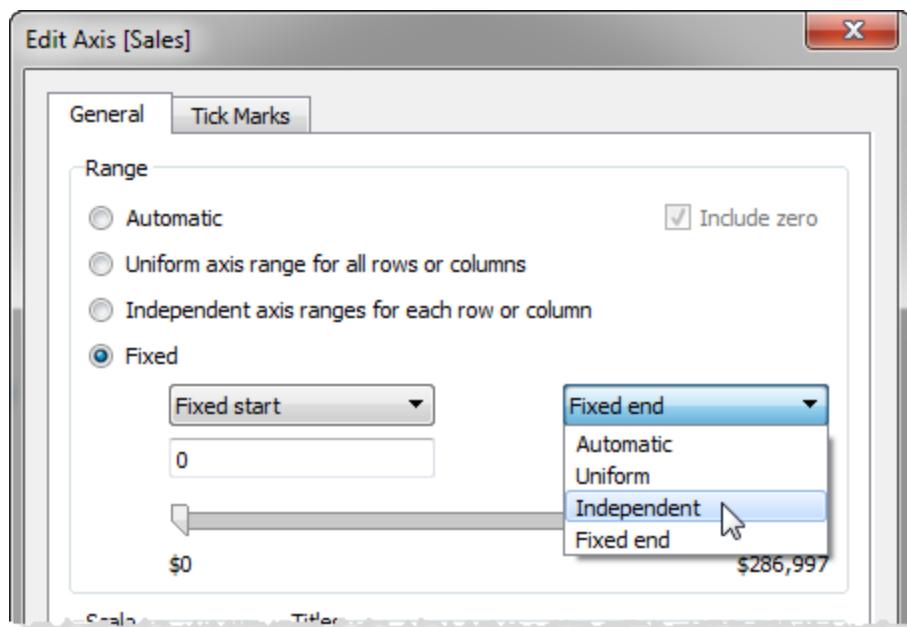
The categories now display only up to \$200,000 in sales. The axes are marked with a pin symbol  , which indicates that you have limited the axis range and that some data might not be displayed.



To fix one end of the axis:

1. Right-click (control-click on Mac) the **SUM(Sales)** axis in the view and select **Edit Axis**.
2. In the Edit Axis dialog box , select **Fixed**, click the **Fixed End** drop-down menu, and then

select **Independent**.



3. Click **OK**.

Notice that the categories now have slightly different axis ranges. For example, Office Supplies has an axis range from 0 to 250,000 dollars, while Furniture only has a range from 0 to 200,000 dollars.



For more information about the difference between automatic, uniform, and independent axis ranges, see [Edit Axes](#) on page 1047.

Format at the Workbook Level

You can quickly change how fonts and titles look in every view in a workbook by specifying format settings at the workbook level instead of at the worksheet level. For example, you might want to use a specific font, size, and color for every worksheets' title, but leave your dashboard titles the way they are.

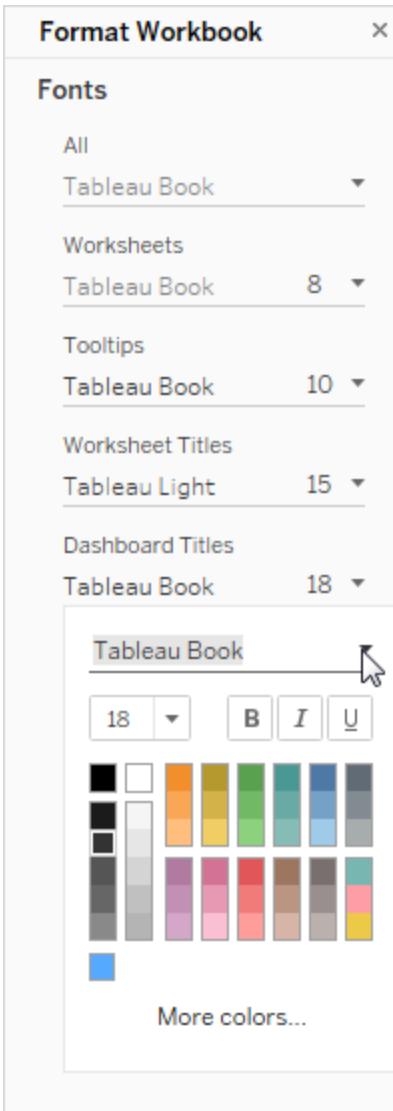
You can also change the theme used by your workbook. Themes control items like the default font, colors, and line thickness. When you create a new workbook, it automatically uses the Default theme, which incorporates visual best practices.

Change all fonts in your workbook

You can change all fonts in your workbook or you can change fonts for only certain areas, such as just worksheet titles.

To change fonts in a workbook:

1. On the **Format** menu, select **Workbook**.
2. The **Format Workbook** pane replaces the **Data** pane on the left and provides a series of drop-down lists where you can change all font settings in a workbook, as well as the font settings for titles of worksheets, stories, and dashboards.



If you have made font changes at the worksheet level, such as on a filter card or a worksheet title, changing the font at the workbook level will overwrite those changes.

Upgrade or change your workbook theme

If your workbook is set to a theme other than Default, it will retain that theme when you upgrade from one version of Tableau to another, but you can easily change the theme to Default.

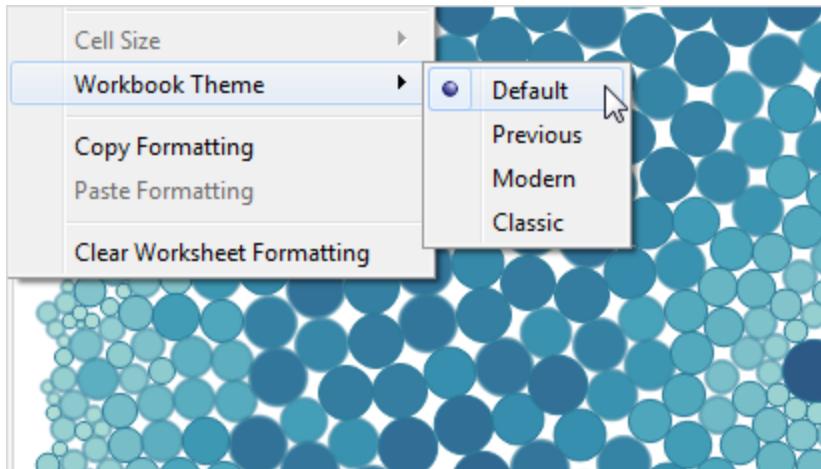
Conversely, you can change your workbook's theme from Default to a different, earlier theme.

If you upgraded from version 9.3 to version 10.0 and your theme was set to Default in version 9.3, it will be set to Previous in version 10.0. After you switch to the Default theme, check your

workbook to confirm that items are sized appropriately. You might need to make some manual adjustments.

To change your workbook theme:

1. On the **Format** menu, select **Workbook Theme**.
2. Choose a theme.



Each theme is associated with a different version of Tableau Desktop.

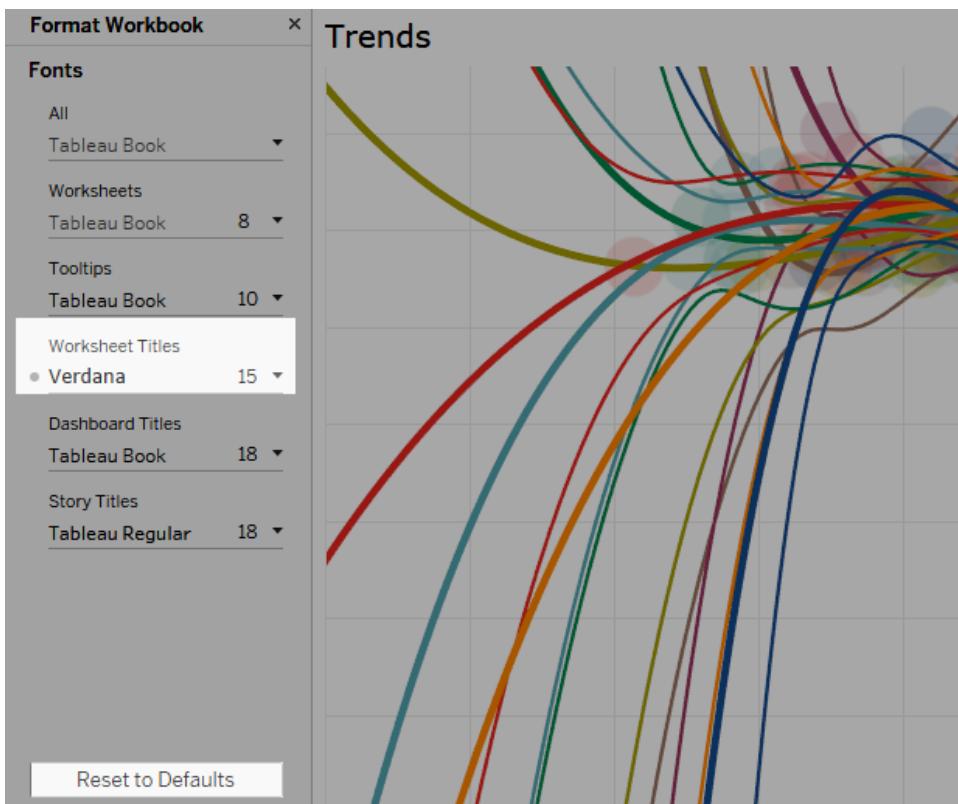
This workbook theme...	Is associated with these Tableau versions...
Default	Version 10.x
Previous	Versions 8.0.x – 9.3.x
Modern	Versions 3.5 – 7.0.x
Classic	Versions 1.0 – 3.2

Reset a workbook to its default settings

When you make changes to your workbook's font settings, a gray dot appears next to the setting in the **Format Workbook** pane. You can quickly switch back to default settings using the **Reset to Defaults** button.

To return your workbook to its default settings:

1. On the **Format** menu, select **Workbook**.
2. In the **Format Workbook** pane, click **Reset to Defaults**.



Format at the Worksheet Level

You can format settings for fonts, alignment, shading, borders, lines and tooltips at the worksheet level. When you make formatting changes at this level, they apply only to the view you're working on.

For example, you might want to remove all the borders in a text table, or add shading to every other column in a view.

To access these settings, select the **Format** menu, then choose the part of the view, such as **Font**, or **Border**, that you want to format.

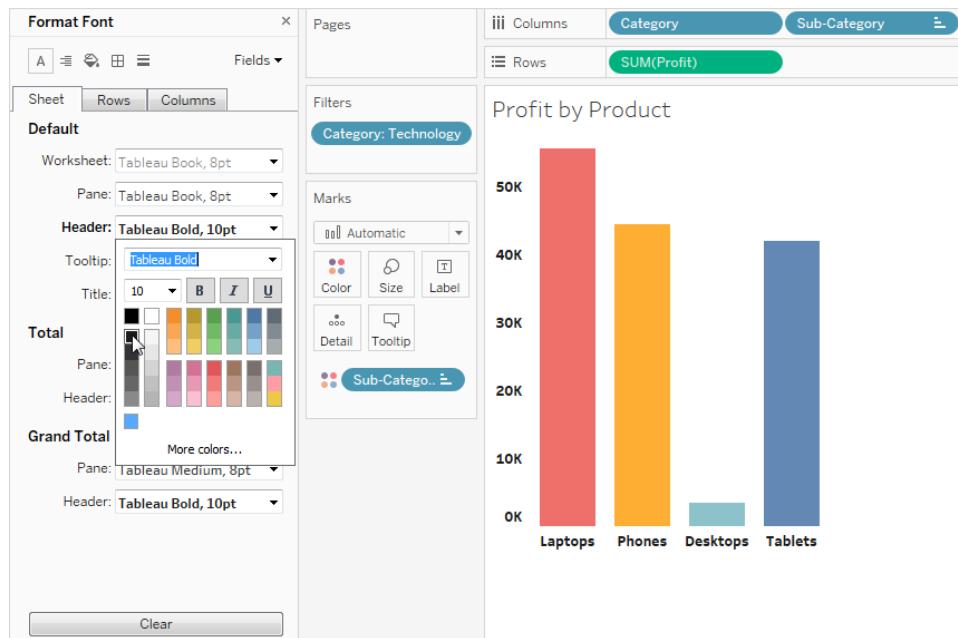
You can also copy and paste formatting at the worksheet level.

To make large-scale formatting changes to every view in your workbook, the best place to start is at the workbook level. See [Format at the Workbook Level on page 1059](#) for details on what you can change.

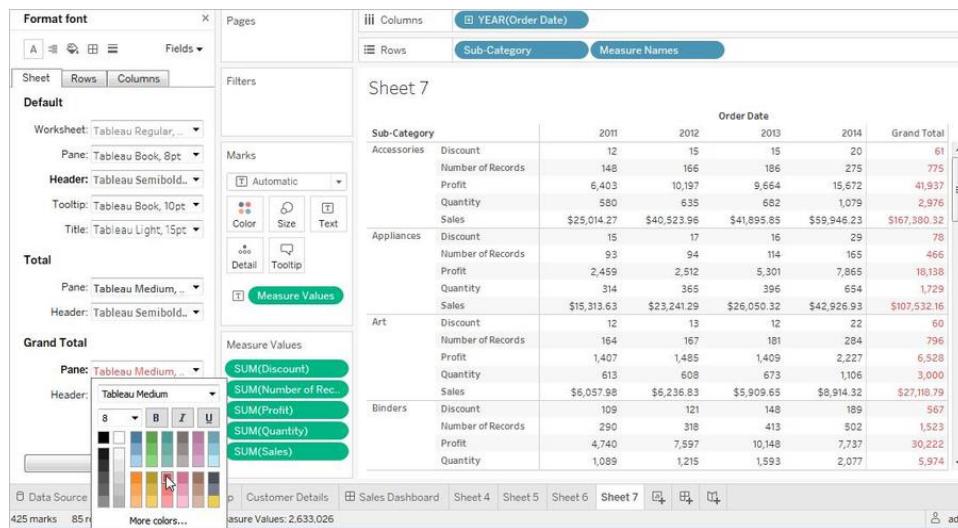
Format fonts

For a view, you can specify the font, style, size, and color for either the pane text or header text,

or both. For example, in the view below, the header text is set to use the Tableau Bold font.



If you have totals or grand totals in the view, you can specify special font settings to make these values stand out from the rest of the data. This is particularly useful when you are working with a text table. The view below shows a text table in which the grand totals are formatted to be dark red.

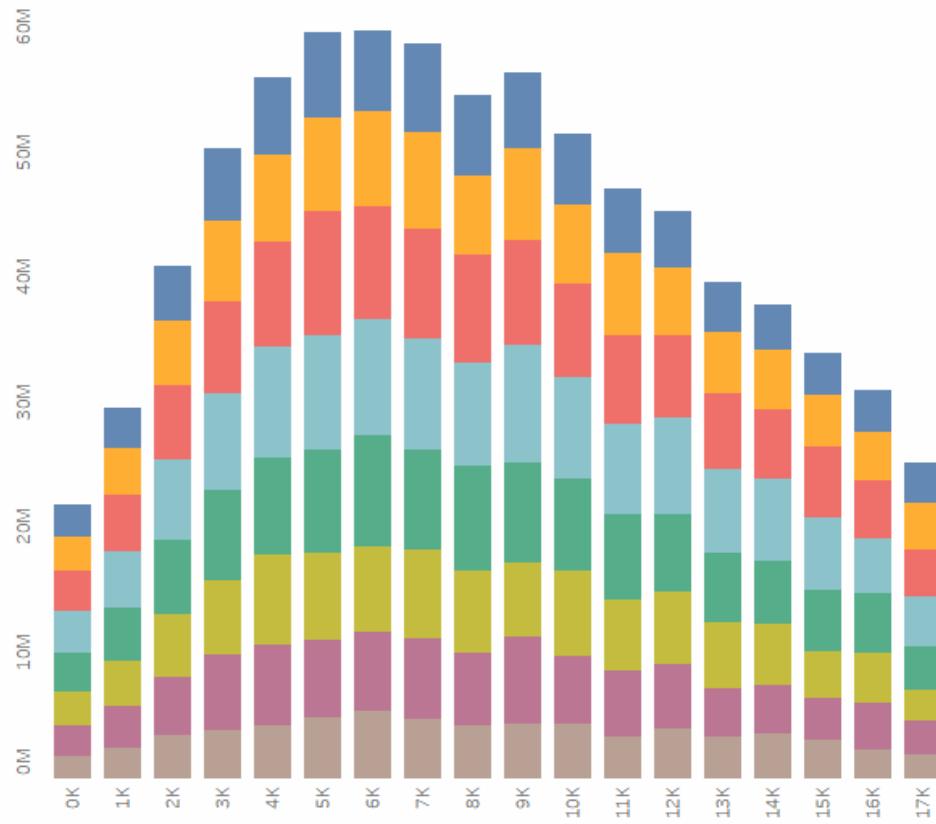


Format text alignment

Tableau uses visual best practices to determine how text is aligned in a view, but you can also customize text. For example, you can change the direction of header text so that it is horizontal

(normal) instead of vertical (up).

Header Text is Vertical



Click the image to replay it.

For each text area you can specify the following alignment options:

Horizontal - Controls whether text aligns to the left, right, or center.

Vertical Alignment - Controls whether text aligns at the top, middle, or bottom.

Direction - Rotates text so that it runs horizontally (normal), top-to-bottom (up), or bottom-to-top (down).

Wrap - Controls whether long headers and pane text wrap to the next line or are abbreviated.

Note: If cells are not large enough to show more than one row of text, turning on wrapping will have no visible effect. If this happens, you can hover the cursor over a cell until a double-sided arrow appears, and then click and drag down to expand the cell size.

Format shading

Shading settings control the background color of the pane and headers for totals, grand totals,

as well as for the worksheet areas outside those areas.

You can also use shading to change or add banding. Banding is when the background color alternates from row to row or column to column. Banding is useful for text tables because the alternating shading helps your eye distinguish between consecutive rows or columns.

Table with Banding

Clinton St & Washington Blvd	1,033	2,873
Canal St & Jackson Blvd	942	2,355
Clinton St & Madison St	994	1,946
Franklin St & Jackson Blvd	1,649	1,829
Daley Center Plaza	1,147	1,772
Michigan Ave & Lake St	2,919	1,522
Dearborn St & Adams St	1,456	1,464
Clark St & Randolph St	956	1,447
Clinton St & Lake St	701	1,370
Dearborn St & Monroe St	838	1,353
State St & Kinzie St	1,957	1,304
Kingsbury St & Erie St	877	1,049

[Click the image to replay it.](#)

For banding on rows or columns, you can use the following options:

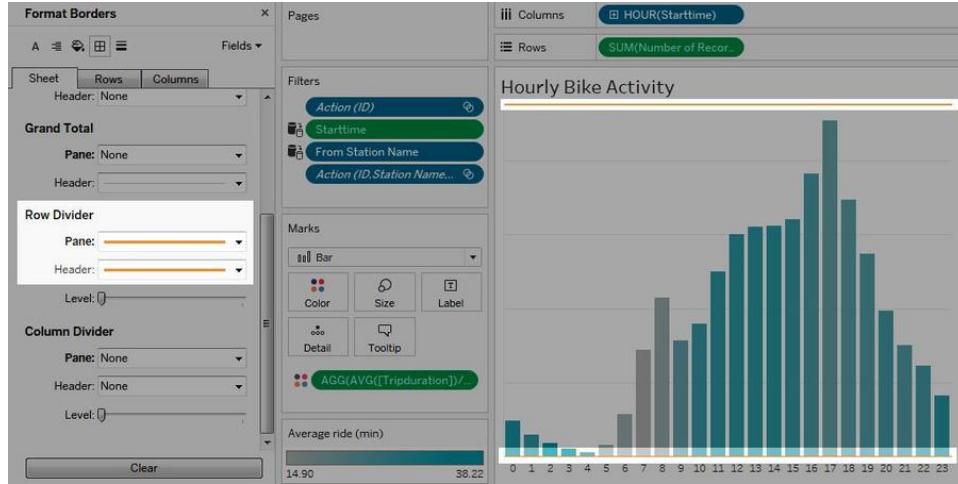
Pane and Header- The color the bands use.

Band Size - The thickness of the bands.

Level - If you have nested tables with multiple dimensions on the rows and columns shelves, this option allows you to add banding at a particular level.

Format borders

Borders are the lines that surround the table, pane, cells, and headers in a view. You can specify the border style, width, and color for the cell, pane, and header areas. Additionally, you can format the row and column dividers. For example, in this view the **Row Divider** borders are formatted to use an orange color:



Row and column dividers serve to visually break up a view and are most commonly used in nested text tables. You can modify the style, width, color, and level of the borders that divide each row or each column using the row and column divider drop-downs. The level refers to the header level you want to divide by.

Format lines

You can control the appearance of the lines that are part of the view, such as grid lines and zero lines, as well as lines that help you inspect data, such as trend lines, reference lines, and drop lines.

For example, you can set trend lines to use a red color and an increased thickness:



Format highlighters

The highlighter on your worksheet can be formatted to use a different font, style, color, background color, font size, and border. Formatting highlighters allows you to better integrate them into your dashboard or worksheet style. You can also edit the title that displays on each highlighter that shows in the view.

For more information about using highlighters, see [Highlight Data Points in Context](#) on page 1125.

To format highlighters:

1. Select **Format > Highlighters** or click the drop-down arrow on the highlighter and select **Format Highlighters**.
2. In the **Format** pane, specify settings for the title and body of the highlighters.

Copy and paste worksheet formatting

After you format a worksheet, you can copy its formatting settings and paste them into other worksheets. The settings that you can copy are anything you can set in the **Format** pane, with the exception of reference lines and annotations. Adjustments like manual sizing and level of zoom are not copied.

To copy and paste formatting between worksheets:

1. Select the worksheet from which you want to copy formatting.
2. Right-click (control-click on Mac) the worksheet tab and select **Copy Formatting**.
3. Select the worksheet you want to paste the formatting into.
4. Right-click (control-click on Mac) the worksheet tab and select **Paste Formatting**.

Format Parts of a View

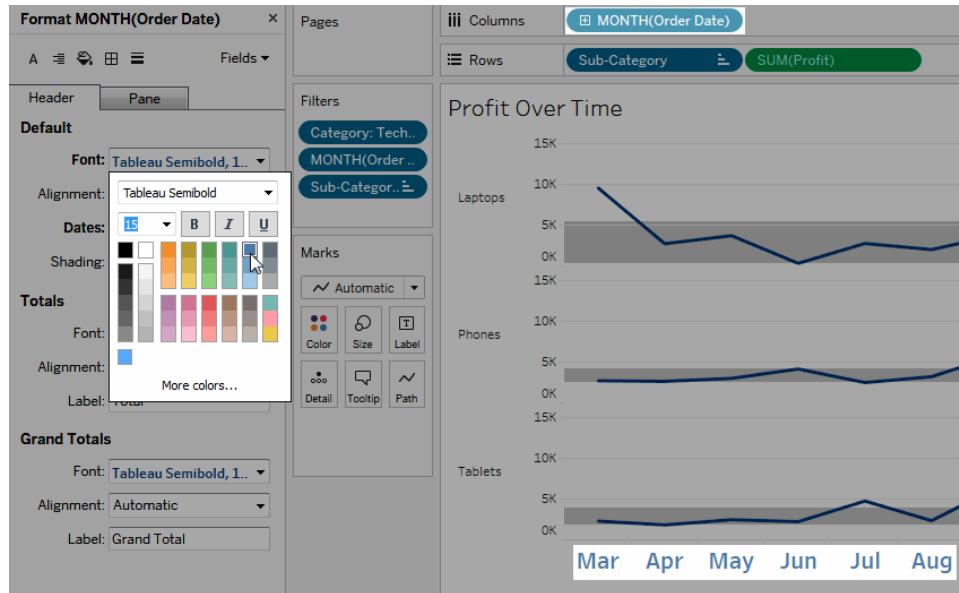
After you apply formatting settings on a large scale such as the entire workbook or worksheet level, you might want to format only certain parts of a view, such as a single title. To specify individual format settings, right-click (control-click on Mac) a specific part of the view and select **Format**.

Format Fields and Field Labels

You can format a specific field in a view, as well as individual field labels. When formatting is done this way, your formatting changes are scoped to only where the field appears in the view, or only an individual label.

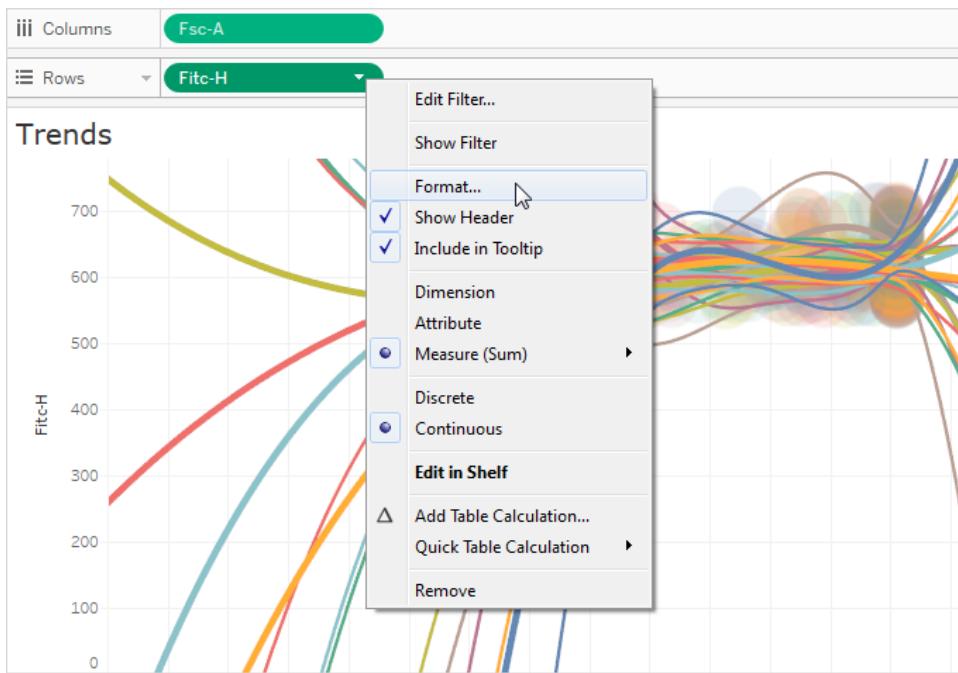
Format a field

In the view below, the **Month(Order Date)** field has been formatted so that the headers use the Tableau Semibold font, in blue. Notice that the header values along the Profit axis are not affected.



To format a specific field:

1. Right-click (control-click on Mac) the field and select **Format**.



The **Format** pane opens to settings for the selected field.

2. Make your changes in the **Format** pane.

For discrete fields, such as Region or Customer Name, you can specify font and alignment properties for both header and pane areas. For continuous fields, such as Profit or Sales, you can format font properties for the pane and axis as well as number and tick mark colors. For more information about other axis options, see [Edit Axes on page 1047](#). The view is updated as you make changes so you can quickly see the colors and formats that work with your view.

Format a field label

Field labels are row and column headings that indicate the data fields used to create a view. By default, Tableau shows field labels, but you can choose to hide them. When field labels are showing they appear in three different parts of the view: rows, columns, and the corner. The view below shows an example of each of these types of field labels.

The screenshot shows a Power BI report interface. At the top, there are two buttons: 'Columns' with 'Region' selected, and 'Rows' with 'Category' selected. Below this is a table titled 'Region / Order Date'. The table has a header row 'Category' and 'Year' ('2012', '2013', '2014', '2015'). The data rows show sales figures for Furniture, Office Supplies, and Technology. The 'Region' dimension is collapsed, indicated by a small icon next to the 'Region' button.

Category	Region / Order Date			
	East			
	2012	2013	2014	2015
Furniture	47,233	53,817	46,387	60,854
Office Supplies	35,969	42,655	61,645	65,247
Technology	45,479	59,859	72,497	87,138

You can format the font, shading, alignment, and separators for each of these types of field labels.

To format a specific field label:

1. Right-click (control-click on Mac) the field label in the view and select **Format**.
2. In the **Format** pane, specify the settings of the font, shading, and alignment field labels.

Note: When you have multiple dimensions on the rows or columns shelves, the field labels appear adjacent to each other in the table. Each field label is separated from the others with a forward slash symbol. Specify a different separator in the **Format** pane.

Format Numbers and Null Values

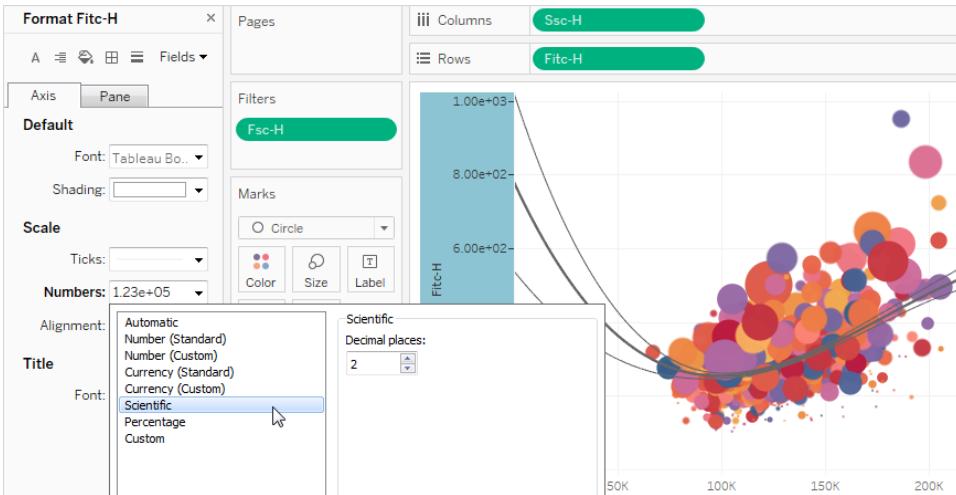
When you format a measure, you can specify the number format for both the axis and the pane text. You can select from a set of standard formats, such as number, currency, scientific, and percentage; or you can define a custom number format using Microsoft Excel style format codes.

When a measure contains null values, the nulls are usually plotted as zero. You can use formatting, however, to handle the null values in a different way, such as hiding them.

Specify a number format

1. Right-click (control-click on Mac) a measure or axis in the view and select **Format**.
2. In the **Format** pane, click the **Numbers** drop-down menu.
3. Select a number format.

Some formats require additional settings. For example, if you select **Scientific**, you must also specify the number of decimal places.



Here are the number formats and associated options available in Tableau.

Number Format	Format Options
Automatic: format is automatically selected based on either the format specified by the data source or the data contained in the field.	None.
Number (Standard): format is based on locale selected.	Locale: number format changes based on the geographical location selected.
Number (Custom): format is customized to your choice.	Decimal Places: the number of decimal places to display. Negative Values: how negative values are displayed. Units: the number is displayed using the specified units. For example, if the number is 20,000 and the units are thousands, the number will be displayed as 20K. Prefix/Suffix: characters that precede and follow each displayed number. Include thousands separators: whether the number shows separators every thousand (example: 100,000 vs. 100000).
Currency (Standard): format	Locale: currency format based on the

Number Format	Format Options
and currency symbol is based on locale selected.	geographical location selected.
Currency (Custom) : format and currency symbol is customized to your choice.	<p>Decimal Places: the number of decimal places to display.</p> <p>Negative Values: how negative values are displayed.</p> <p>Units: the number is displayed using the specified units. For example, if the number is 20,000 and the units are thousands, the number is displayed as 20K.</p> <p>Prefix/Suffix: characters that precede and follow each displayed number.</p> <p>Include thousands separators: whether the number shows separators every thousand (example: 100,000 vs. 100000).</p>
Scientific : numbers are displayed in scientific notation.	Decimal: the number of decimal places to display.
Percentage : numbers are displayed as a percentage with the percent symbol. The value of 1 is interpreted as 100% and 0 as 0%	Decimal: the number of decimal places to display.
Custom : format is based entirely on what is specified in the format options.	Custom: type in the format you want to use. This format can be specified by an Excel style number code. See Number format codes in the Microsoft Knowledge Base for details.

Set the default number format for a field

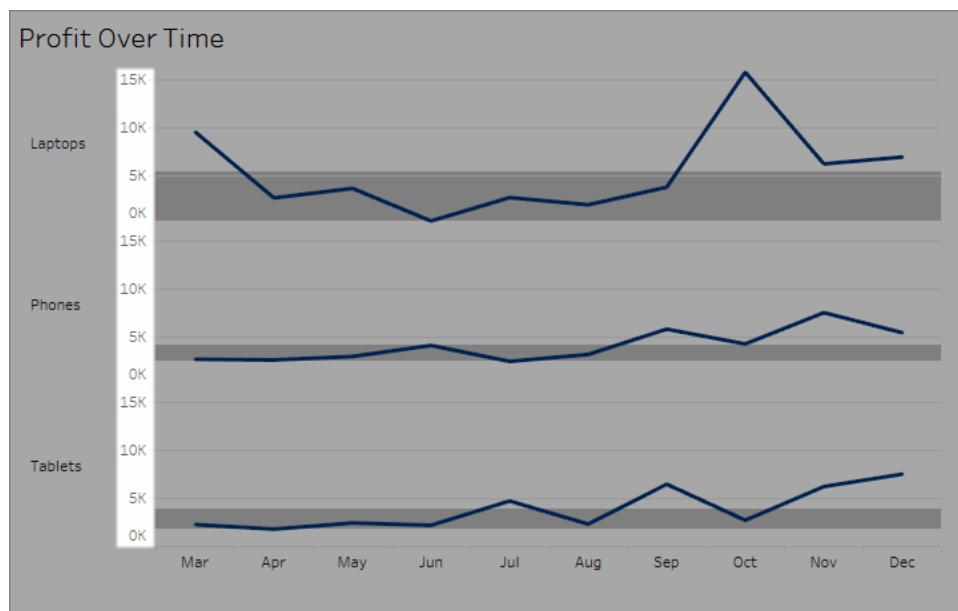
Right-click (control-click on Mac) the field in the **Data** pane and select **Default Properties > Number Format**.

In the subsequent dialog box, specify a number format to be used whenever the field is added to the view. The default number format is saved with the workbook. It is also exported when you export the connection information.

Note:Formatting numbers using the **Format** pane overrides any number formatting applied elsewhere.

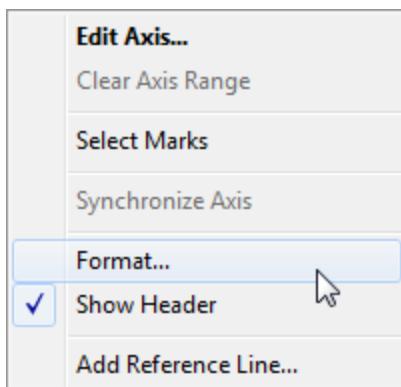
Format a measure as currency

The view in the following image shows profit over time. Notice that the profit figures on the vertical axis are not formatted as currency.



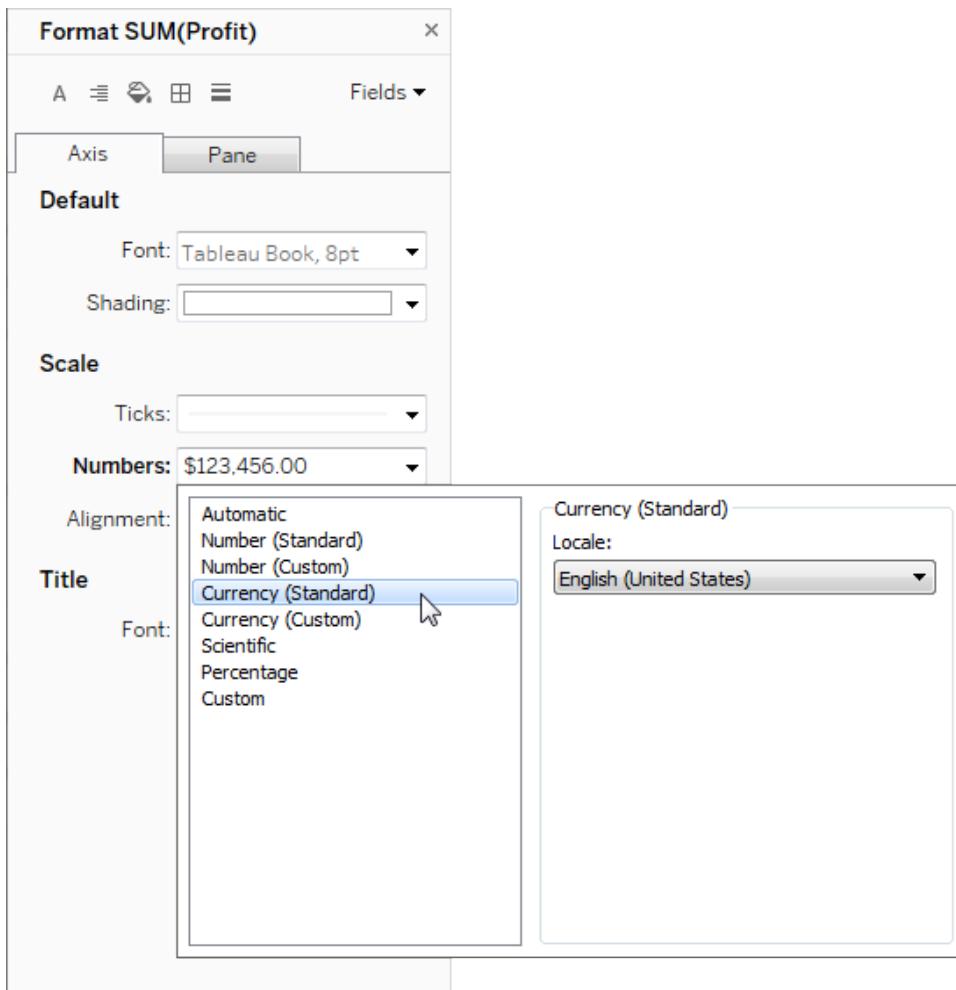
To format the numbers as currency:

1. Right-click the **Profit** axis and choose **Format**.



2. On the **Axis** tab in the **Format** pane, under **Scale**, select the **Numbers** drop-down list, and then select one of the following:

Currency (Standard) to add a dollar sign and two decimal places to the figures.



Currency (Custom) to specify the number of decimal places, how to show negative values, the units, whether to include a prefix or suffix, and whether to include a separator character.

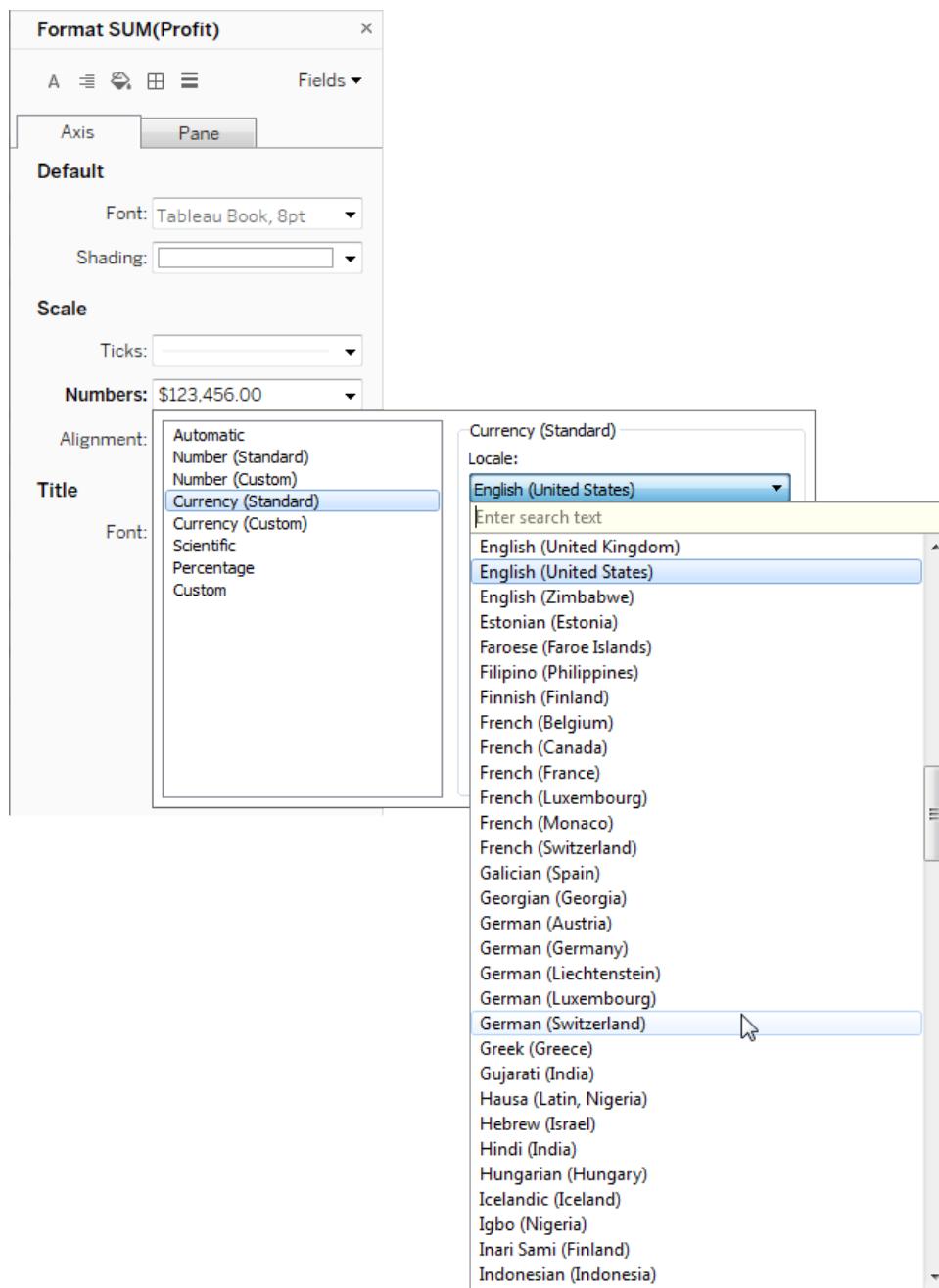
Use locale to specify number formats

By default, Tableau uses your computer's locale and language settings to format numbers. But you can explicitly set a different locale in the **Format** pane.

The following steps show how to set Swiss German currency, using the same view as in the previous section.

1. Right-click the **Profit** axis and select **Format**.
2. On the **Axis** tab, under **Scale**, select the **Numbers** drop-down list and then select **Currency (Standard)**.

3. In the **Locale** drop-down list, items appear in a **Language (Country)** format. For this example, select **German (Switzerland)**.



The view updates to show the sales figures in Swiss Francs, formatted for the German language.

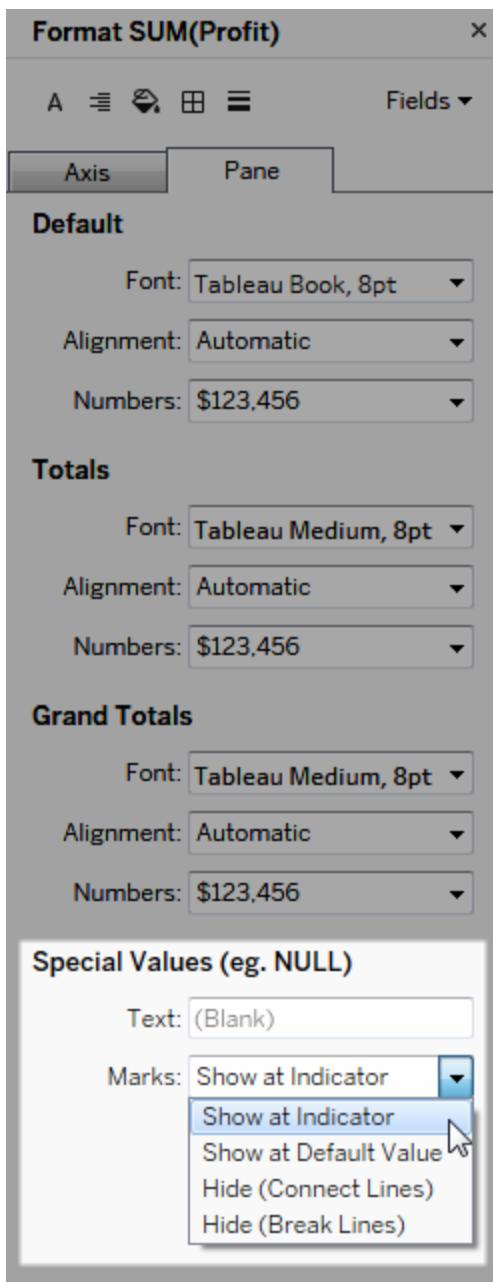
Tip: You can change the default currency setting so that every time you drag the **Profit** measure to a view it uses the settings you want. In the **Measures** pane, right-click **Profit** (or other monetary measure), and select **Default Properties > Number Format**. Then format the field as shown above.

Format null values

When a measure contains null values, they are usually plotted in a view as zero. However, sometimes that changes the view and you'd rather just suppress null values altogether. You can format each measure to handle null values in a unique way.

To format null values for a specific field:

1. Right-click the field in the view that has the null values (Control-click on a Mac) and choose **Format**.
2. Go to the **Pane** tab.
3. In the **Special Values** area, specify whether to show the null values using an indicator in the lower right corner of the view, plot them at a default value (e.g., zero for number fields), hide the values but connect lines, or hide and break lines to indicate that null values exist.



4. If you specify text in the **Text** field, it will appear in the view for null values when mark labels are turned on. See [Mark Labels on page 548](#).

Note: The Special Values area is not available for Date values.

Format Titles, Captions, Tooltips and Legends

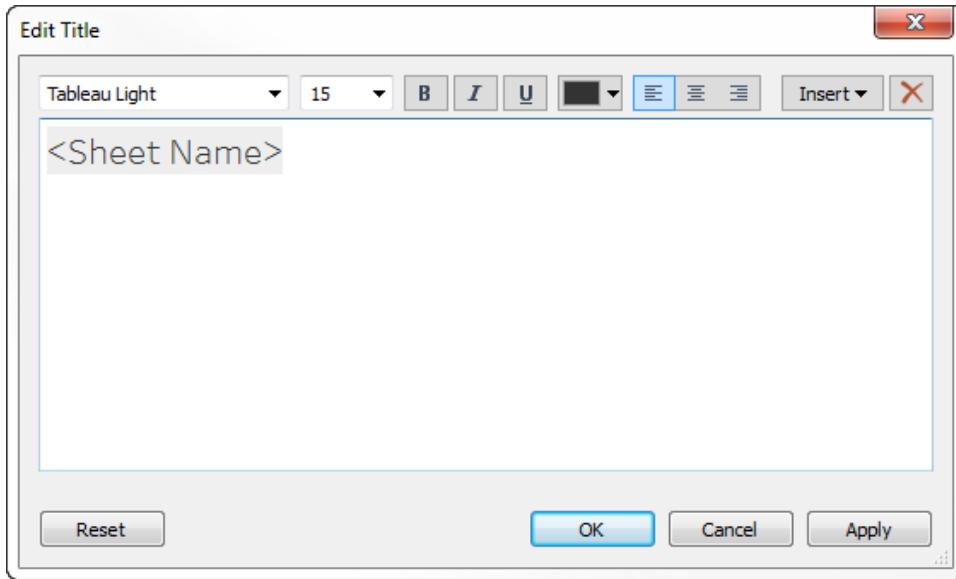
You can show titles, captions, and tooltips on any sheet. After you add a title or caption you can edit and format the text as well as modify the shading and border. If a title or caption is not

showing, from the top menu, go to the menu for the current sheet type (Worksheet, Dashboard, or Story) and select **Show Title** or **Show Caption**.

A legend card appears in the worksheet when you encode marks by dropping them on the **Color** or **Size** cards. You can format the legend font, shading, border, and alignment. You can also edit the titles that appear on each legend.

Format and edit titles, captions, and legend titles

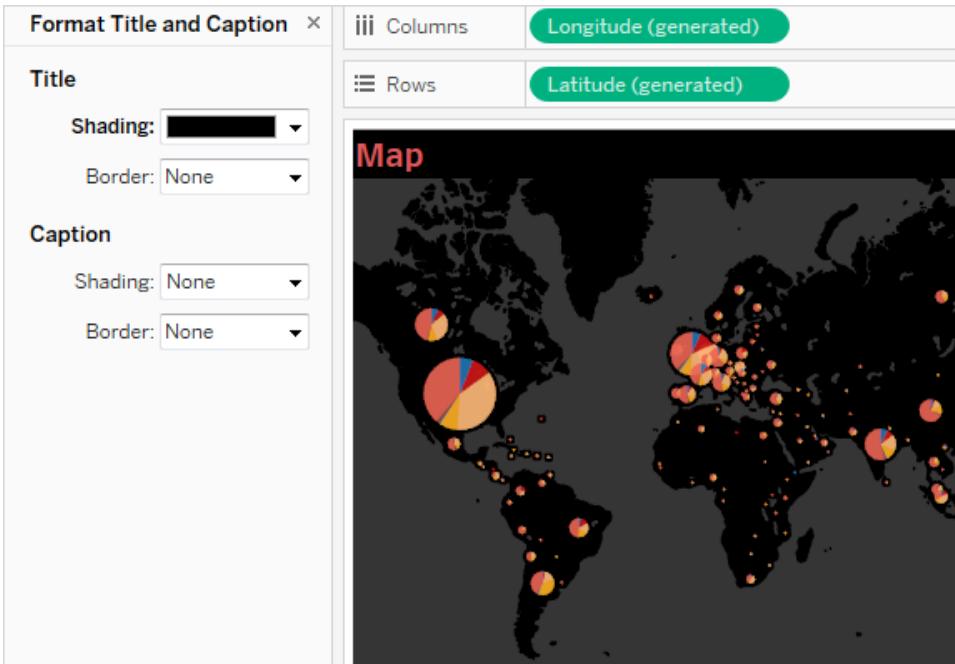
1. Do one of the following:
 - Right-click (control-click on Mac) the item you want to change and select **Edit <item>**, for example, **Edit Title**.
 - On worksheets, hover on the title, click the drop-down arrow on the right-hand side and select **Edit Title** or **Edit Caption** from the context menu.
2. In the Edit dialog box, modify the text and format the font, size, style, color, and alignment. Use the **Insert** menu to add dynamic text such as sheet properties and field values. Click **OK**.



To reset the title back to the default, click **Reset**.

Format the borders and shading of a title, caption, or legend

1. Right-click (control-click on Mac) the title, caption, or legend and select **Format <item>**—for example, **Format Title**.
2. In the **Format** pane, use the drop-down lists to change the default shading and border.

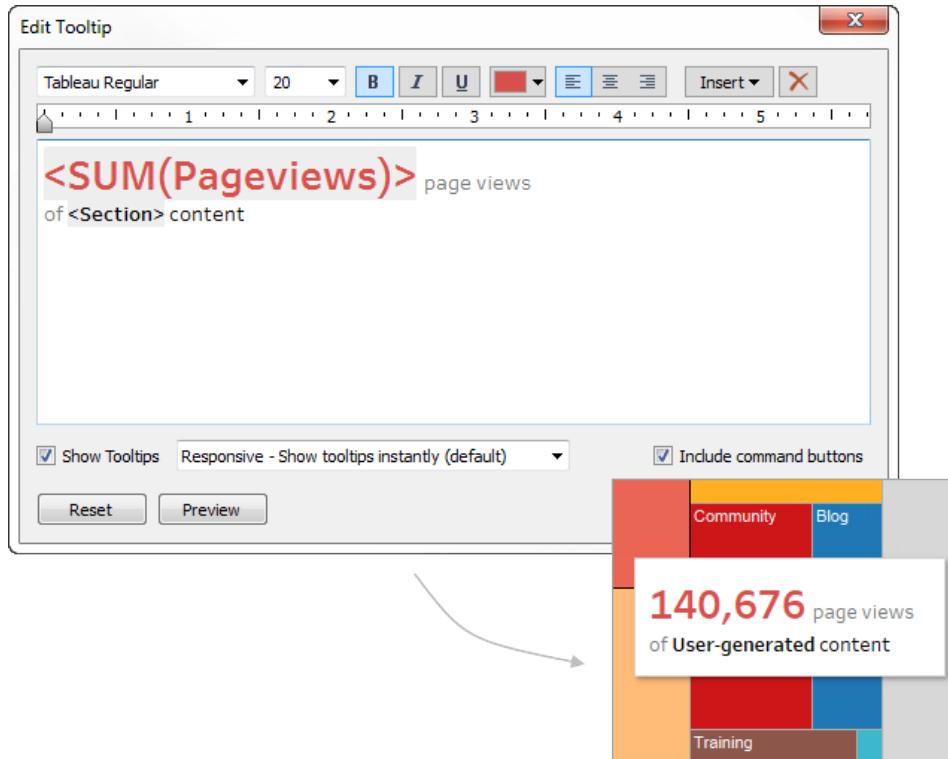


Format and edit tooltips

Tooltips are details that appear when you rest the pointer over one or more marks in the view. Tooltips also offer convenient tools to quickly filter or remove a selection or view underlying data. You can edit the tooltip to include both static and dynamic text. You can also modify which fields are included in the automatic tooltip.

To edit a tooltip:

1. Select **Worksheet > Tooltip**. Tooltips are specified for each sheet and can be formatted using the formatting tools along the top of the Edit Tooltip dialog box. Use the **Insert** menu at the top of the dialog box to add dynamic text such as field values, sheet properties, and more.



2. Use the formatting tools along the top of the Edit Tooltip dialog box.
3. (Optional): Use the **Insert** drop-down list at the top of the dialog box to add dynamic text such as field values, sheet properties, and more.

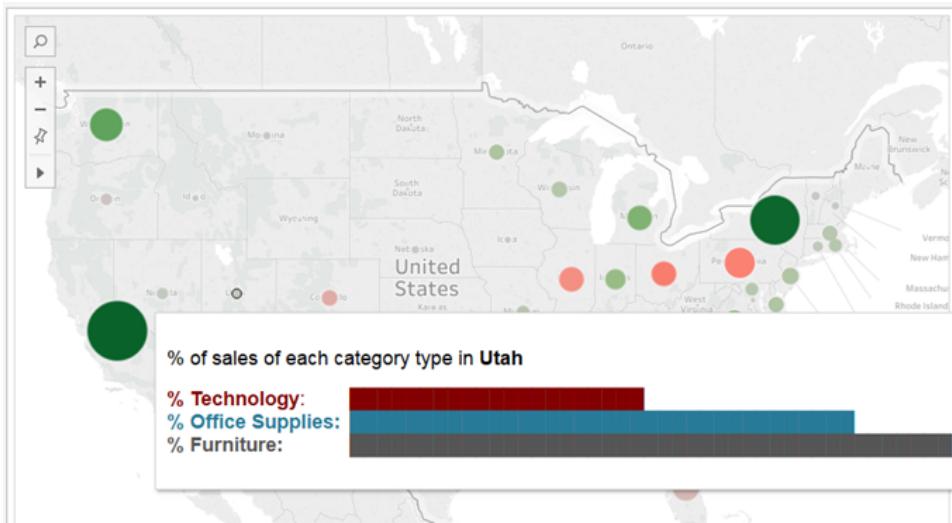
The **All Fields** option on the **Insert** menu adds all field names and values that are used in the view. Inserting the All Fields parameter updates the tooltip as you change the view. You can also use the All Fields option to exclude fields.

4. (Optional): Select the **Include command buttons** option to show filtering and view data options in the tooltip. For example, including command buttons will add **Keep Only**, **Exclude**, and **View Data** buttons to the bottom of the tooltip. These command buttons are available both in Tableau Desktop and when the view is published to the web or viewed on a mobile device.

[Try it: Show Bar Charts in Tooltips](#)

In this example you will use calculated fields to show distribution data in the form of bar charts in tooltips. For example, if you have a map view that shows sales data for each state in the United States, as you hover over each state mark in the view you might also want to show the sales for each category of your product for the state.

The result looks something like this:



To see completed examples, open the [Bar charts in tooltips](#) story in Tableau Public, and then click **Download Workbooks** in the top right corner to view the workbook details.

In this article

[Create a view below](#)

[Show bar charts in tooltips](#) on the next page

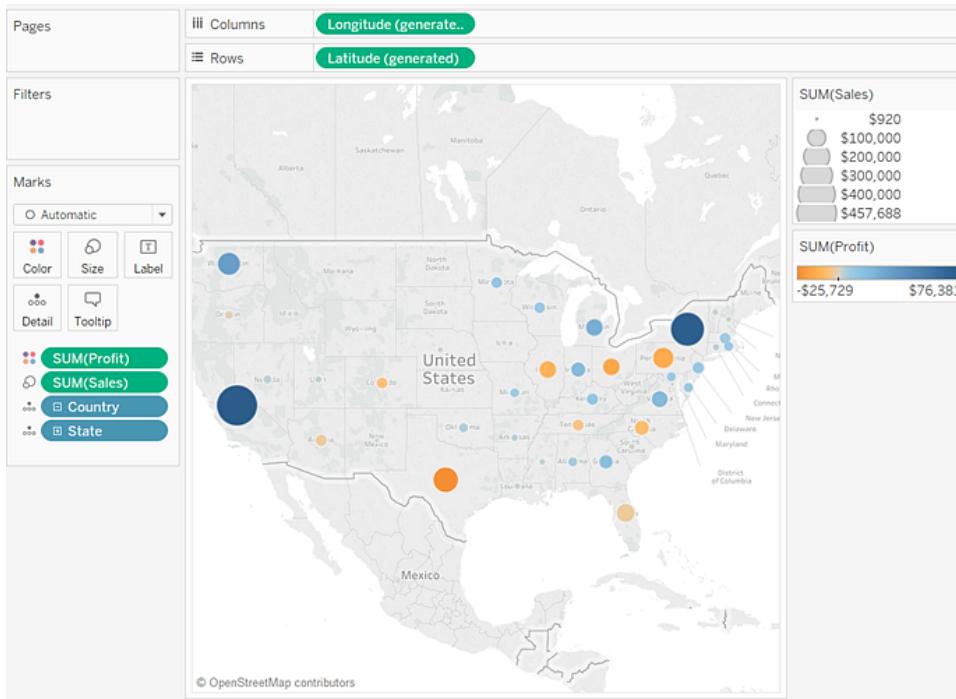
[Show bar charts with context in tooltips](#) on page 1086

Create a view

Create a view using the **Sample-Superstore** data source included with Tableau to work through this example.

1. In the Data pane, double-click the **State** dimension to create a map view.
2. On the Marks card, drag the **Sales** measure to **Size** and the **Profit** measure to **Color**.
3. On the Marks card, right-click **Size** and drag the slider to the right to make the marks a bit bigger.

Your view should look something like this:



Show bar charts in tooltips

To show bar charts in tooltips, you need to create the following calculated fields to identify the measures you want to include and to define the structure of the bars that you'll show:

- **Create cohort calculated fields:** This separates a measure so that it is specific to one member of a dimension.
- **Create calculated string fields:** This defines the bar chart and shows the length and appearance of the bar for each cohort.
- (optional) **Create calculated string fields field to show more context:** This enables you to show the remaining percent for each bar in the tooltip. For information about how to create these calculated fields, see [Show bar charts with context in tooltips on page 1086](#).

After you create the calculated fields you can add them to the tooltip in your view and then format the tooltip as needed.

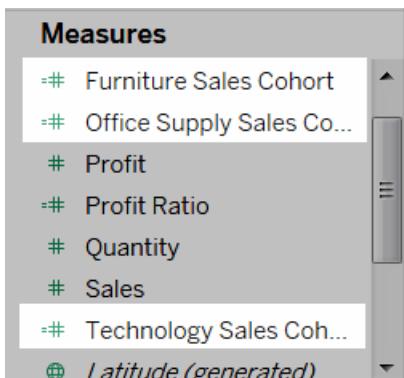
Complete the following steps to create your calculated fields.

Step 1: Create cohort calculations

Cohort calculations enable you to separate a measure for a specific dimension. In this example, you want to know the total sales for each category for each state. You can create a cohort calculation to separate **Sales**, for the dimension **Category**. There are three members in the **Category** dimension: **Furniture**, **Office Supplies**, and **Technology**. To separate sales for each of those categories, you need to create three cohort calculations for each dimension member.

1. Click **Analysis > Create Calculated Field**.
2. Name the calculated field **Furniture Sales Cohort**, then enter the following formula:
`IF [Category] = "Furniture" THEN [Sales] END`
3. Click **OK**.
4. Repeat these steps to create cohort calculated fields for the **Office Supplies** and **Technology** categories. Name them **Office Supplies Sales Cohort** and **Technology Sales Cohort**.

The calculated fields are added to **Measures** in the Data pane.



For more information about how to create calculated fields, see [Create or Edit a Calculated Field](#) on page 822.

Step 2: Create calculated string fields

In this example you use three calculated string fields to define the appearance and bar length for each cohort that you'll show in the tooltip. In the previous step, you created three cohort calculations, one for each category of sales. To display a bar for each of those categories, you define a calculated string field for each cohort calculation using the following calculation:

```
LEFT ("character string", ROUND(SUM([cohort calculated field]) / SUM([Measure])) * 100, 0))
```

This calculation has two values that you change for each field:

- "character string": This part specifies how the bar looks and uses a character that you specify to draw the bar. For example use "|" characters to create hashed bars, or use "█" characters to create solid bars. The number of characters that you need to enter here depends on how you are performing the calculation. In this example, we are calculating a percent of sales out of 100%. Since the total amount is some percentage of 100, you would need to enter 100 characters to define the total length of the bar.
 - ROUND(SUM([cohort calculated field])/SUM([Measure])*100, 0)): This part calculates the percent of total sales for the cohort, which defines the length of the bar.

The results look like the following examples:

Example: Hashed bars

A map of the United States where each state has a colored circle representing the percentage of sales in that category. A callout box titled "% of sales of each category" shows the legend:

- % Technology: (Red) Hashed bar
- % Office Supplies: (Blue) Hashed bar
- % Furniture: (Dark Blue) Hashed bar

The map shows Colorado with a red circle, which corresponds to the Technology category.

Example: Solid bars

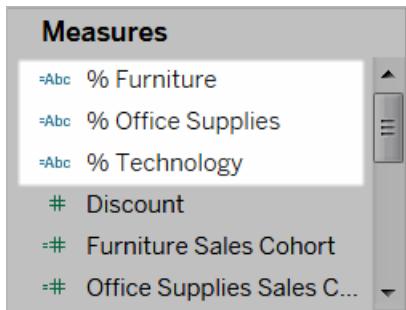
A map of the United States focusing on the Western US. A callout box titled "% of sales by category in Colorado" shows the following solid bar chart:

Category	Percentage
% Office Supplies	~45%
% Furniture	~35%
% Technology	~20%

1. Click **Analysis > Create Calculated Field**.
 2. Name the calculated field **% Furniture**, then enter the following formula:

```
LEFT  
( " [ ] ",  
    ROUND(SUM([Furniture Sales Cohort])/SUM([Sales])  
    *100, 0) )
```
 3. Click **OK**.
 4. Repeat these steps to create calculated string fields for the **Office Supplies** cohort and **Technology** cohort calculated fields. Name them **% Office Supplies** and **%Technology**.

The calculated fields are added to **Measures** in the Data pane.

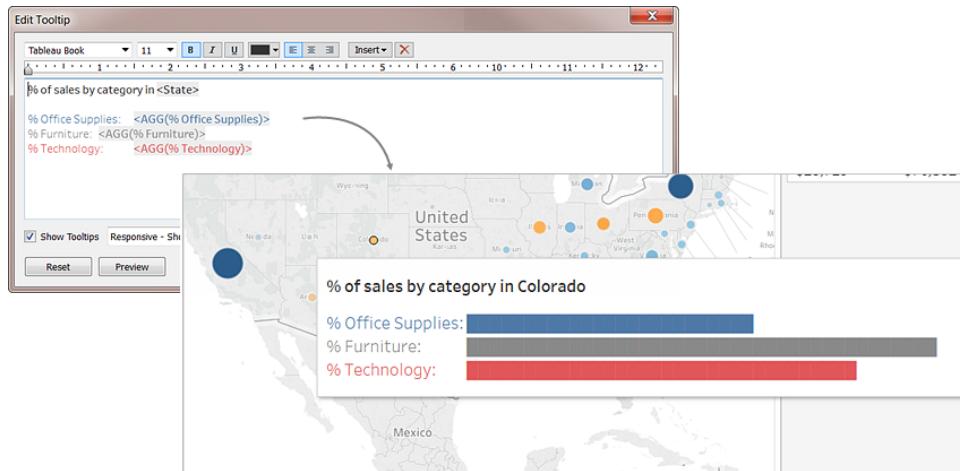


Step 3: Add the bar charts to tooltips

After you create the calculated fields you can add them to tooltips in your view to show them on hover for a mark.

1. In the view that you created earlier, drag the following calculated fields measures to **Details** on the Marks card:
 - % Furniture
 - % Office Supplies
 - % Technology
2. On the Marks card, click **Tooltip**.
Tableau automatically adds the calculated fields for you.
3. Format the data shown in the tooltip to make it easier to read. Make the following changes:
 - At the top of the dialog box, add the title "% of sales by category in" and then click **Insert** and select **State** from the list.
 - Delete the text below the title.
 - Move % Office Supplies: <AGG(% Office Supplies)> above % Furniture: <AGG(% Furniture)>
 - Add color to each of the calculated fields. For example, highlight the text for % Office Supplies: <AGG(% Office Supplies)> and select the color blue from the color palette drop-down. Select different colors for the other two measures.
 - Highlight all of the text and make it slightly bigger. For example, select "11" for the font size.
4. Click **OK** to save your changes.

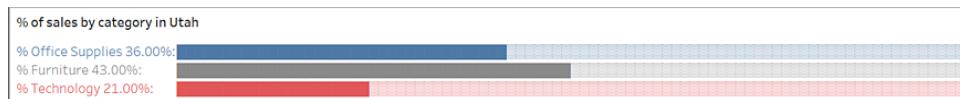
The updated tooltip dialog box looks like this:



For more information about how to format tooltips, see [Format Titles, Captions, Tooltips and Legends on page 1077](#).

Show bar charts with context in tooltips

You can further enhance your bar charts by showing the current and remaining percent of sales. To do this, create additional calculated fields. The following example shows this type of enhancement. The bar charts use different colors to distinguish total sales and the remaining sales percentage.



Step 1: Create calculated string fields

To create a calculated string field to show the remaining percentage for a bar, use the following calculation:

```
(RIGHT("character string", 100-(ROUND(SUM([Your cohort  
calculation])/SUM([Measures])*100))))
```

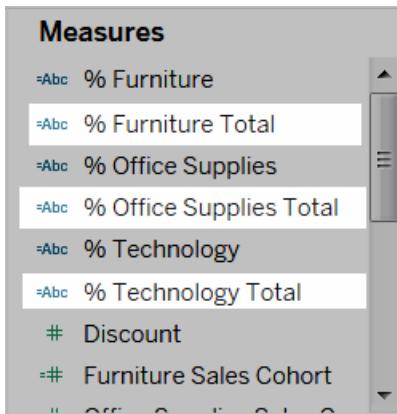
Like the calculated string fields that you created earlier, the first part of the calculation defines how the bar looks and the second half of the calculation determines the remaining percentage and the bar length.

Create three calculated string fields for each of the cohort calculations that you created earlier. Use 100 "█" characters for contrast.

1. Select **Analysis > Create Calculated Field**.
2. Name the calculated field **% Furniture Total**, then enter the following formula:

```
(RIGHT
("
", 100-(ROUND(SUM
([Furniture Sales Cohort])/SUM([Sales])*100))))
```
3. Click **OK**.
4. Repeat the steps above to create calculated string fields for the **Office Supplies** cohort and **Technology** cohort calculated fields. Name the fields **% Office Supplies Total** and **% Technology Total**.

The calculated fields are added to **Measures** in the Data pane.



Step 2: Create calculated fields to show numerical percent

To show the numerical percentage for the sales in each state, you can add a calculated field that uses the cohorts that you created previously to calculate the percentage.

1. Click **Analysis > Create Calculated Field**.
2. Name the calculated field **Furniture Percent #**, then enter the following formula:
 $\text{ROUND}(\text{SUM}([\text{Furniture Sales Cohort}])/\text{SUM}([\text{Sales}]) * 100, 0)$
3. Click **OK**.
4. Repeat the steps above to create calculated fields for the **Office Supplies** cohort and **Technology** cohort calculated fields. Name the fields **Office Supplies Percent #** and **Technology Percent #**.

The calculated fields are added to **Measures** in the Data pane.

The screenshot shows the Tableau Data pane with the title 'Measures'. A vertical scroll bar is on the right. Ten items are listed: Discount, Furniture Percent #, Furniture Sales Cohort, Office Supplies Percent #, Office Supplies Sales Cohort, Profit, Profit Ratio, Quantity, Sales, Technology Percent #, and Technology Sales Cohort. The third, fifth, and ninth items are highlighted with a white background and black text.

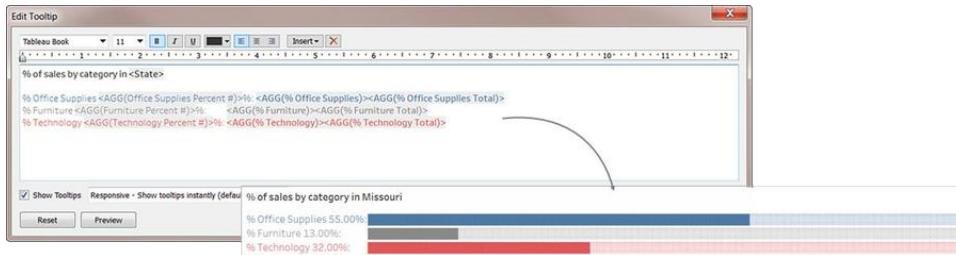
Step 3: Add the calculated fields to tooltips to show the bars

To show the new bars and sales percentage in tooltips, you add them to **Details** in the Marks card. Because you have already formatted the tooltips, you will need to manually add the new calculated fields to the tooltip so that they show the way you want them to when hovering on a mark.

1. Drag the following calculated field measures to **Details** on the Marks card:
 - Furniture Percent #
 - Office Supplies Percent #
 - Technology Percent #
 - % Furniture Total
 - % Office Supplies Total
 - % Technology Total
2. On the Marks card, click **Tooltip**.
3. In the Edit Tooltip dialog box, click **Insert** to add the calculated fields to the tooltip. Do the following to format the tooltip.
 - Place your cursor between **% Office Supplies** and the colon, and insert `<AGG (Office Supplies Percent #)>%`.
 - Place your cursor at the end of `<AGG(% Office Supplies)>` and insert `<AGG(% Office Supplies Total)>`
 - Repeat the above steps to add the measures for Furniture and Technology.

- Click **OK** to save your changes.

Your updated tooltip dialog looks like this:



Try it: Format Tooltips for Conditional Dimensions

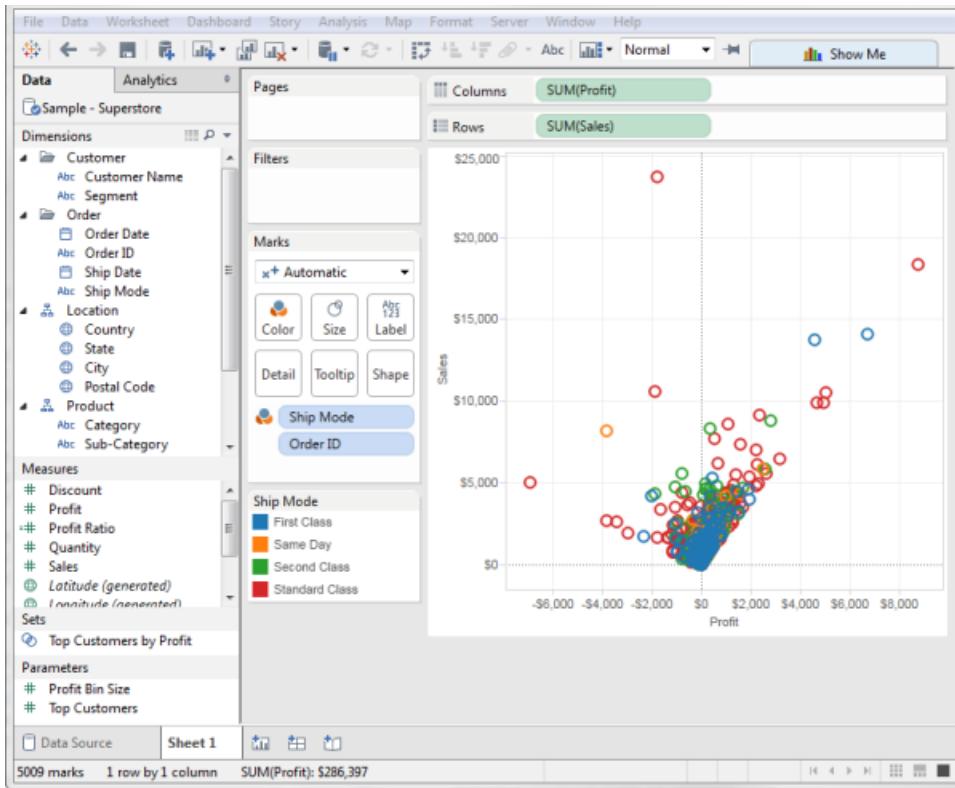
You can customize tooltips by formatting each tooltip to reflect a particular field in a dimension. This is useful for highlighting differences within a specific dimension without altering the view.

For example, you might format the tooltip for the First Class field in the Ship Mode dimension differently than the tooltip for the Standard Class field.

To do this, you break a dimension into multiple dimensions that you can then format individually.

Create the view

- Connect to your data source. This example uses the **Sample - Superstore** data source.
- Drag the **Profit** measure to the **Columns** shelf.
- Drag the **Sales** measure to the **Rows** shelf.
- Drag the **Ship Mode** dimension to **Color** on the **Marks** card.
- Drag the **Order ID** dimension to **Detail** on the **Marks** card.



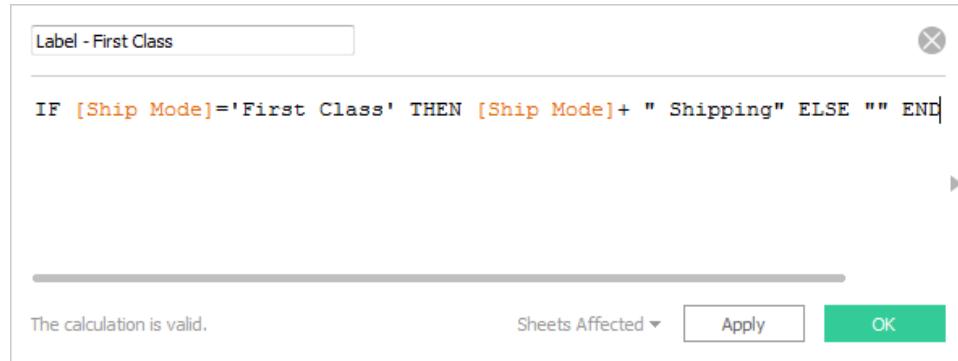
Create a calculation to separate a dimension by fields

You can use an "If-Then" statement to create a formula that takes only a specific member from a dimension. In this example, you create a calculated field for each Ship Mode.

1. Select **Analysis > Create Calculated Field**.
2. In the dialog box, enter the following:
 1. In the name text box, enter **Label - First Class**.
 2. In the formula text box, enter the following and then click **OK**.


```
IF [Ship Mode] = 'First Class' THEN [Ship Mode] + "
```

Shipping" ELSE "" END



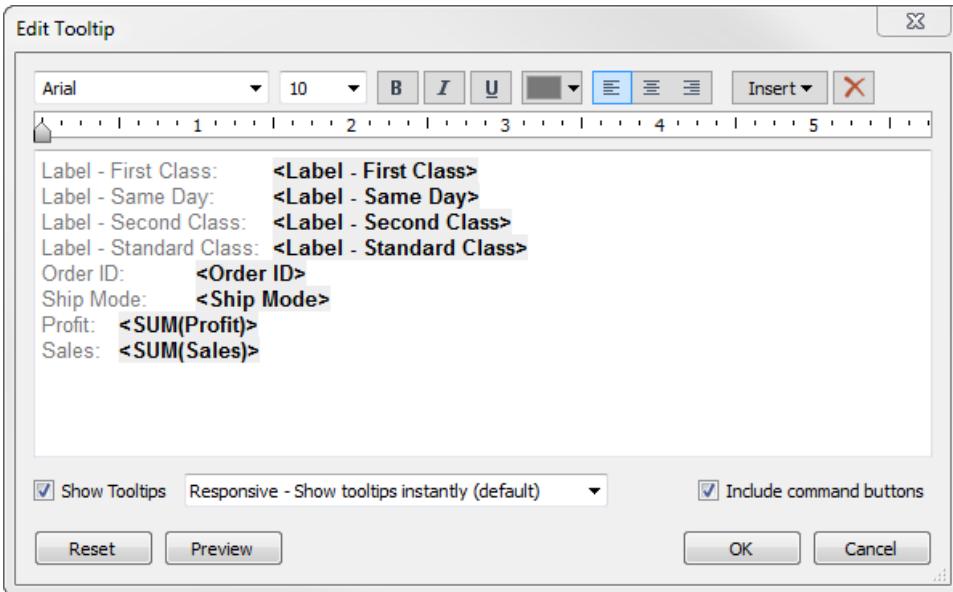
3. Create additional calculated fields for Same Day, Second Class, and Standard Class.
 1. Right-click **Label – First Class** and click **Duplicate**.
 2. Right-click the duplicated calculated field and click **Edit**.
 3. Change the word that specifies the **Ship Mode** in the label and the formula. For example, replace First Class with Same Day.

Each ship mode now appears in the **Dimensions** pane as a new dimension.

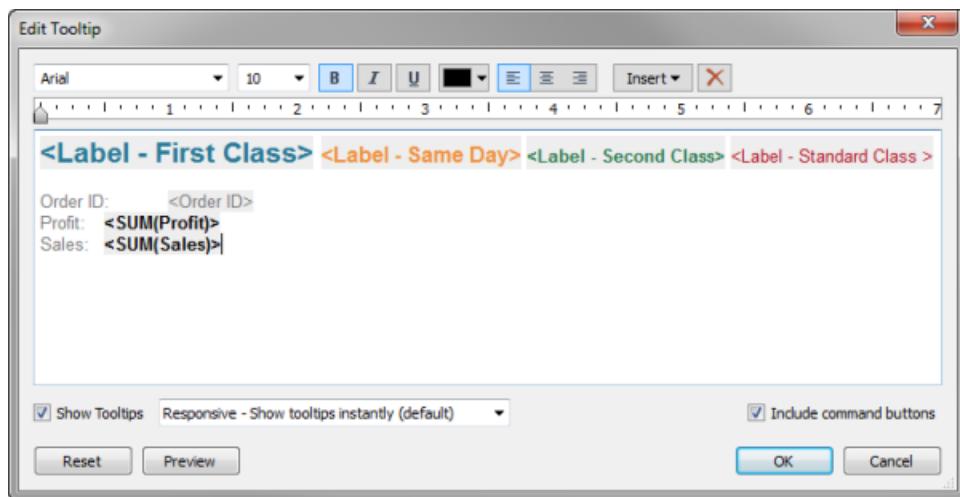
Customize tooltips

Follow these steps to format each newly created dimension in the Edit Tooltip dialog box.

1. Hold down the CTRL key, and in the **Dimensions** pane, click each of the new dimensions, and then drag them to **Detail** on the **Marks** card.
2. Click **Tooltip** on the **Marks** card to open the Edit Tooltip dialog box.



3. In the Edit Tooltip dialog box, follow these steps to create an entry that looks like the graphic below:
 1. Align your new dimensions in a row next to each other, and delete unneeded text.
 2. Delete the **Ship Mode** entry. Add any dimensions that you want to appear in the tooltip.
 3. Format each field so that the text color matches the **Ship Mode** field color that appears in the view. Adjust font and size so that the labels appear as you want them to.



4. Click **OK**.

Your tooltips now show only one field at a time because of the calculated fields you created, and each field is formatted differently. Note the difference in formatting between the first class and the standard class tooltip:

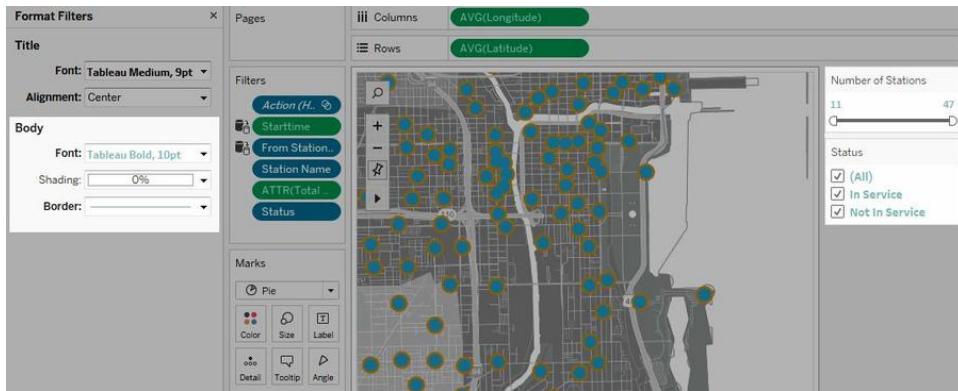


Format Filter and Parameter Cards

You can display filter and parameter cards on your worksheet, and you can format them both to use a different font, style, color, background color, font size, and border. Formatting them allows you to better integrate them into your dashboard or worksheet style.

Format a filter card

Filter cards contain controls that let users interact with your view. You can change filter cards to use custom formatting. For example, the body text in the filters below is formatted to use the Tableau Bold font, in aqua.



To format a filter card:

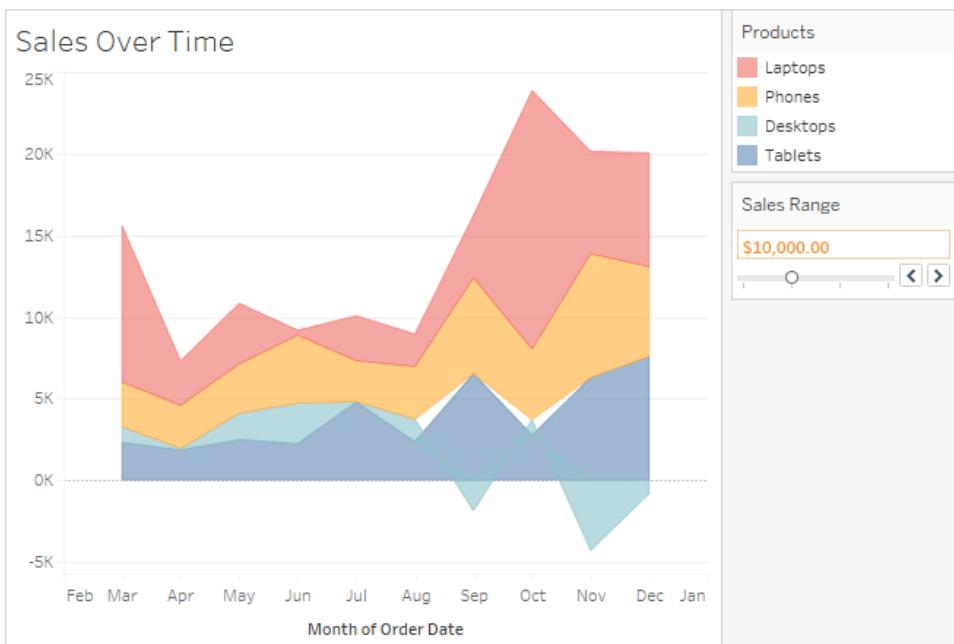
1. Select **Format > Filters**.
2. In the **Format** pane, specify the following options:
 - **Title:** You can modify the **Font** and **Alignment** of the filter titles. The title formatting only appears when the filters are used on dashboards or when the view

is published to the web.

- **Body:** You can modify the body **Font**, **Shading**, and **Border** of the filters. These settings appear on both worksheets, dashboards, and when the view is published to the web.

Format a parameter control card

Parameter controls are similar to filter cards in that they contain controls that let users modify the view. If you create a parameter control, you can customize how it looks. For example, in the view below, the Sales Range parameter is formatted so that the sales amount appears in orange.



To format a parameter control card:

1. Select **Format > Parameters**.
2. In the **Format** pane on the left side of the workbook, specify the following options:
 - **Title:** You can modify the **Font** and **Alignment** of the parameters titles. The title formatting only appears when the parameter controls are shown on dashboards or when the view is published to the web.
 - **Body:** You can modify the body **Font**, **Shading**, and **Border** around the parameters. These settings appear on worksheets, dashboards, and when the view is published to the web.

Resize Tables and Cells

You can change the size of the rows, columns, and cells that compose a table. The best way to resize your table depends on the view type and the table components you want to resize.

Use commands to resize rows and columns

By selecting **Format > Cell Size** and then the **Taller**, **Shorter**, **Wider**, or **Narrower** menu items you can resize row and columns.

For example, suppose you want to increase the width of the columns and the height of the rows for the view shown below. You can use the **Taller** and **Wider** menu items or the **Shortcuts and Commands for Resizing** on page 1098: Ctrl + up arrow and Ctrl + right arrow (Command-up arrow and Command-right arrow on a Mac). The views below use both these commands to make the view more readable.



		Order Date				Order Date				Order Date			
Segment	Region	2011				2012				2013			
		Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
Consumer	Central	\$6,116	\$8,324	\$30,149	\$22,727	\$6,759	\$14,559	\$13,726	\$14,619	\$6,116	\$8,324	\$30,149	\$22,727
	East	\$1,683	\$14,985	\$21,	\$21,	\$1,683	\$14,985	\$21,136	\$38,700	\$9,	\$9,	\$9,	\$9,
	South	\$6,570	\$9,416	\$9,	\$9,	\$6,570	\$9,416	\$9,847	\$6,477	\$14,	\$14,	\$14,	\$14,
	West	\$7,596	\$15,126	\$34,	\$34,	\$7,596	\$15,126	\$34,972	\$32,272	\$10,	\$10,	\$10,	\$10,
Corporate	Central	\$235	\$3,856	\$7,	\$7,	Segment	Region	Q1	Q2	Q3	Q4	Q1	Q2
	East	\$4,351	\$4,079	\$11,	\$11,	Consumer	Central	\$6,116	\$8,324	\$30,149	\$22,727	\$6,116	\$8,324
	South	\$7,008	\$12,443	\$5,	\$5,	East	East	\$1,683	\$14,985	\$21,136	\$38,700	\$9,	\$9,
	West	\$2,397	\$6,866	\$8,	\$8,	South	South	\$6,570	\$9,416	\$9,847	\$6,477	\$14,	\$14,
Home Office	Central	\$2,249	\$5,227	\$6,	\$6,	West	West	\$7,596	\$15,126	\$34,972	\$32,272	\$10,	\$10,
	East	\$545	\$2,000	\$5,	\$5,	Corporate	Central	\$235	\$3,856	\$7,446	\$8,364	\$3,	\$3,
	South	\$30,684	\$665	\$5,	\$5,	East	East	\$4,351	\$4,079	\$11,558	\$17,652	\$7,	\$7,
	West	\$5,013	\$3,551	\$6,	\$6,	South	South	\$7,008	\$12,443	\$5,747	\$9,466	\$5,	\$5,
						West	West	\$2,397	\$6,866	\$8,241	\$18,704	\$10,	\$10,
						Home Office	Central	\$2,249	\$5,227	\$6,576	\$2,568	\$1,	\$1,
						East	East	\$545	\$2,000	\$750	\$11,242	\$5,	\$5,
						South	South	\$30,684	\$665	\$467	\$5,035	\$1,	\$1,
						West	West	\$5,013	\$3,551	\$6,744	\$6,402	\$1,	\$1,

Note: For a given field, all members will have the same width and the same height. You cannot resize individual field members.

Manually resize rows and columns

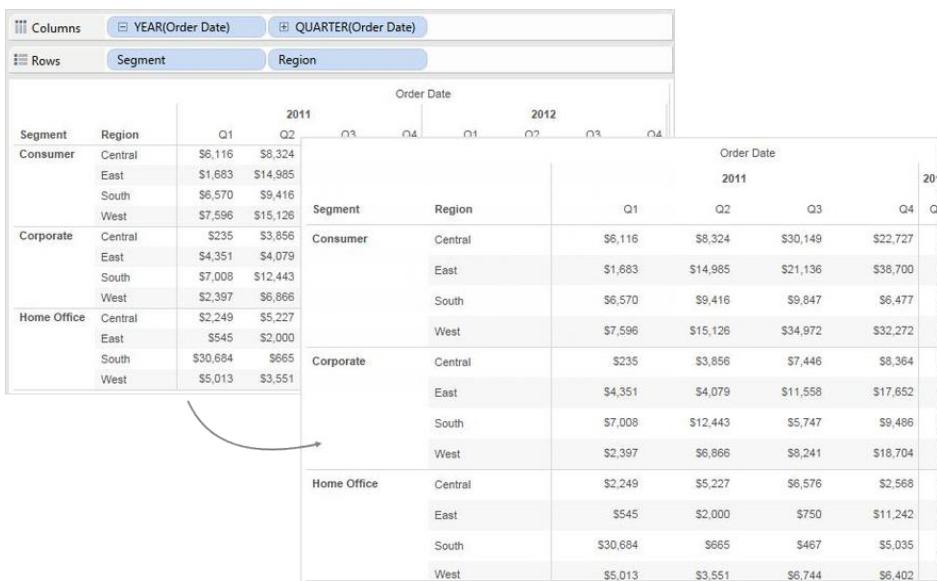
To manually resize the widths or heights of row and column headers or axes:

1. Place your cursor over the vertical or horizontal border of a header or axis.
2. When you see the resize cursor  , click and drag the border left and right or up and down.



Resize an entire table

You can increase or decrease the size of the entire table by selecting **Bigger** or **Smaller** on the **Format > Cell Size** menu. For example, to increase the width of the columns and the height of the rows for the view shown below, you can select **Format > Cell Size > Bigger**. This option increases both the width and height of the panes in a visually appropriate way. Notice that the size of the row headers increases horizontally when you resize the table.



Resize cells

Any table you can create in Tableau has the cell as its basic component. For a text table, the cell is what you would expect. It is the intersection of a row and a column, and is where the text is displayed. For more information, see [Cells on page 167](#).

However, depending on the view you construct, identifying the cell is not always possible or useful, such as in the case of a scatter plot.

Manipulating cells to enhance your data view is useful when dimensions are the inner fields on both the **Rows** and **Columns** shelves. In this case, there are two shortcuts you can select on the **Format > Cell Size** menu:

- **Square Cell** – Adjusts the view so the cell has a 1:1 aspect ratio. This results in a square cell, which is useful for heat maps.
- **Text Cell** – Adjusts the view so the cell has a 3:1 aspect ratio. This is useful for text tables.

For example, you can see the text table below with **Square Cell** and **Text Cell** applied. **Text Cell** enforces a cell aspect ratio of 3:1 and results in a compact table that is easy to read.

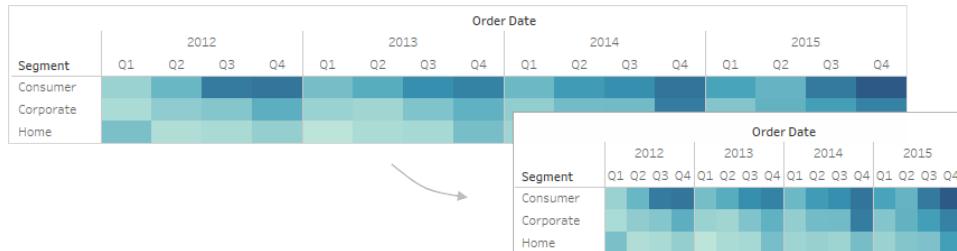
Square Cell

Segment	2011	2012
Consumer	\$266,097	\$266,536
Corporate	\$128,435	\$128,757
Home Office	\$89,716	\$75,239

Text Cell

Segment	2011	2012	2013	2014
Consumer	\$266,097	\$266,536	\$296,296	\$332,473
Corporate	\$128,435	\$128,757	\$206,943	\$242,011
Home Office	\$89,716	\$75,239	\$105,235	\$159,463

The heat map shown below is modified by selecting **Square Cell** on the **Format > Cell Size** menu. This enforces a cell aspect ratio of 1:1 and results in a compact table that is easy to analyze. You can also use the **Size** slider on the **Marks** card to adjust the size of each mark.



After changing the cell size, you can use **Ctrl+B** and **Ctrl+Shift+B** to decrease or increase the table size while maintaining the cell aspect ratio. On a Mac, the keystrokes are Command-B and Shift-Command-B.

Shortcuts and Commands for Resizing

Every view you build in Tableau is different, and different types of views require different sizing techniques. Using the keyboard shortcuts to resize rows and columns as well as the entire table makes building views more efficient. The tables below define some keyboard shortcuts and menu commands for common sizing actions.

Windows Shortcuts

Command	Keyboard Shortcut	Menu Command
Taller	Ctrl+Up	Format > Cell Size > Taller
Shorter	Ctrl+Down	Format > Cell Size > Shorter
Wider	Ctrl+Right	Format > Cell Size > Wider
Narrower	Ctrl+Left	Format > Cell Size > Narrower
Bigger	Ctrl+Shift+B	Format > Cell Size > Bigger
Smaller	Ctrl+B	Format > Cell Size > Smaller

Mac Shortcuts

Command	Keyboard Shortcut	Menu Command
Taller	Command-Up arrow	Format > Cell Size > Taller
Shorter	Command-Down arrow	Format >

Command	Keyboard Shortcut	Menu Command
		Cell Size > Shorter
Wider	Command-Right arrow	Format > Cell Size > Wider
Narrower	Command-Left arrow	Format > Cell Size> Narrower
Bigger	Shift-Command-B	Format > Cell Size > Bigger
Smaller	Command-B	Format > Cell Size > Smaller

Define Table Structure

In addition to the standard formatting, there are some other settings that define the table structure. You can modify these settings by selecting **Analysis > Table Layout > Advanced** to open the Table Options dialog box. There you can specify the aspect ratio, the default number format, row and column attributes, and the default label orientation for labels along the bottom of the view. While these settings apply to the whole view, some can be overridden using the **Format** pane.

Set the aspect ratio

The aspect ratio refers to the ratio of the pane width to the pane height. You can choose to constrain the aspect ratio to a specified amount or not constrain it at all. An unconstrained axis range can be useful because it means that the axes don't have to be the same length.

Anytime you manually resize a row or column, you are unconstraining the aspect ratio. The aspect ratio setting only applies to views containing continuous axes on both the row and column shelves. Nominative axes are not affected by the aspect ratio settings.

Set the default number format

You can define the number of decimal places to appear by default for numbers in the view. If you select **Automatic**, Tableau automatically decides the number of decimal places based on the data in the field. If you select **Manual**, you can decide to show up to 16 decimal places.

Set row attributes

Select from the following Row attributes:

- Maximum levels of row labels: determines the number of fields that can be added to the Rows shelf before the headers are combined on the same level.
- Maximum levels of horizontal row labels: determines the number of fields that can be placed on the Rows shelf before headers are automatically oriented vertically rather than horizontally.

Set column attributes

Select from the following column attributes:

- Maximum levels of column labels: determines the number of fields that can be placed on the Columns shelf before Tableau begins to combine the labels.
- Show innermost level at bottom of view when there is a vertical axis: displays the innermost level of column headers at the bottom of the view (as opposed to the top) when a vertical axis is added to the view.
- Default orientation of labels at bottom of view: determines whether labels at the bottom of the view are oriented horizontally or vertically by default. You can toggle between the horizontal and vertical options by pressing Ctrl + L on your keyboard (Command-L on a Mac).

Customize Colors and Fonts

Tableau products are designed so that you can create great-looking visualizations that use best practices by default, freeing you from needing to think about things like fonts and colors—unless you want to. If you want to change the defaults, Tableau Desktop provides formatting options that allow you to change the look of almost everything you see in a view. See [Format at the Workbook Level on page 1059](#), [Format at the Worksheet Level on page 1062](#), and [Format Parts of a View on page 1068](#) for details.

If the fonts and color palettes available in Tableau Desktop aren't what you need, you can add your own fonts and create your own color palettes. Use the links below for more information.

Create Custom Color Palettes

Tableau Desktop comes with color palettes that have been carefully designed to work well together and effectively apply color to data in many situations, such as on maps, heat maps, bar charts, etc. You can also create and use your own custom color palettes by modifying the `Preferences.tps` file that comes with Tableau Desktop. For example, you can create a custom categorical palette that matches your company's brand.

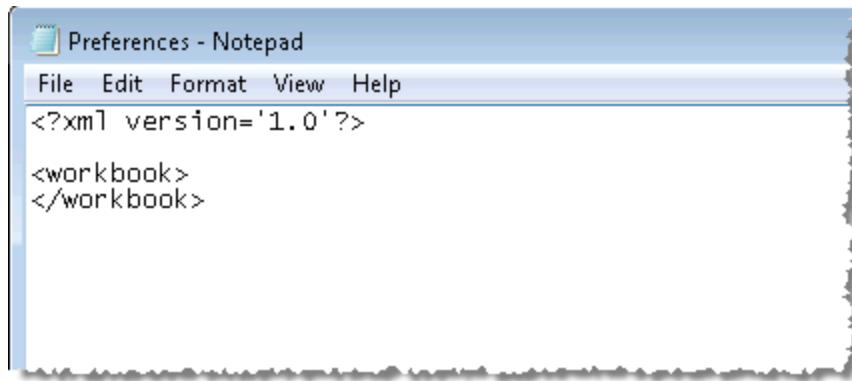
You can add as many custom palettes as you like, each with as many colors as you want. When you modify `Preferences.tps` to add colors, use the standard HTML format for the new colors (hexadecimal value `#RRGGBB` or Red Green Blue format). When you save the workbook, the color information is embedded in `Preferences.tps`, but it won't be included as reusable color encoding. This means that any colors that are in use are shown for anybody

opening the workbook; however, if they don't have the modified `Preferences.tps` file, they can't use the color information for new color encoding.

Tableau doesn't test or support custom color palettes, so be sure to back up your workbooks before you continue. Also, there is no guarantee that custom color palettes you create will work with future Tableau Desktop upgrades.

Edit your preferences file

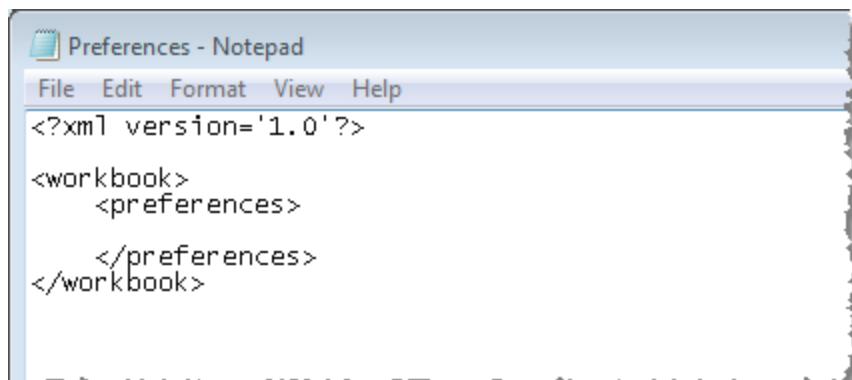
The `Preferences.tps` file is located in your My Tableau Repository. The file is a basic XML file that you can open in a text editor to modify. An unedited preferences file looks like this:



```
<?xml version='1.0'?>
<workbook>
</workbook>
```

To edit your preferences file:

1. Go to the My Tableau Repository folder in your Documents directory, and open the `Preferences.tps` file.
2. Between the opening and closing `workbook` tags, insert opening and closing `preferences` tags.



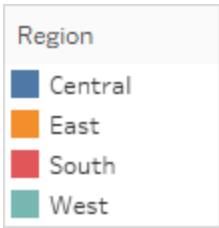
```
<?xml version='1.0'?>
<workbook>
  <preferences>
  </preferences>
</workbook>
```

3. Follow one of the next three procedures to create a custom color palette.

Create a custom categorical color palette

A categorical color palette contains several distinct colors that can be assigned to discrete

dimension members. For example, when you put a discrete dimension such as **Region** on the **Color**, the categorical color legend is used.



The following is an example of what to add between the `preferences` tags to add a categorical color palette. Note that the `type` attribute is specified as `regular`, which identifies this palette as a categorical palette.

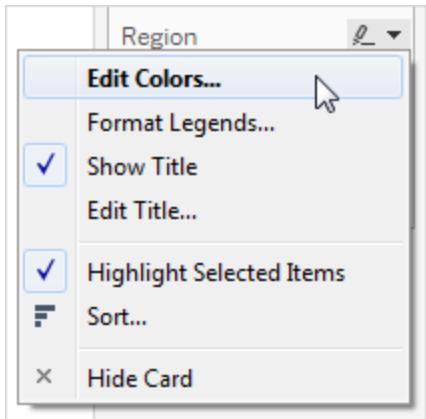
To create a custom categorical color palette:

1. In the **Preferences.tps** file, between the "preferences" tags, paste the following:

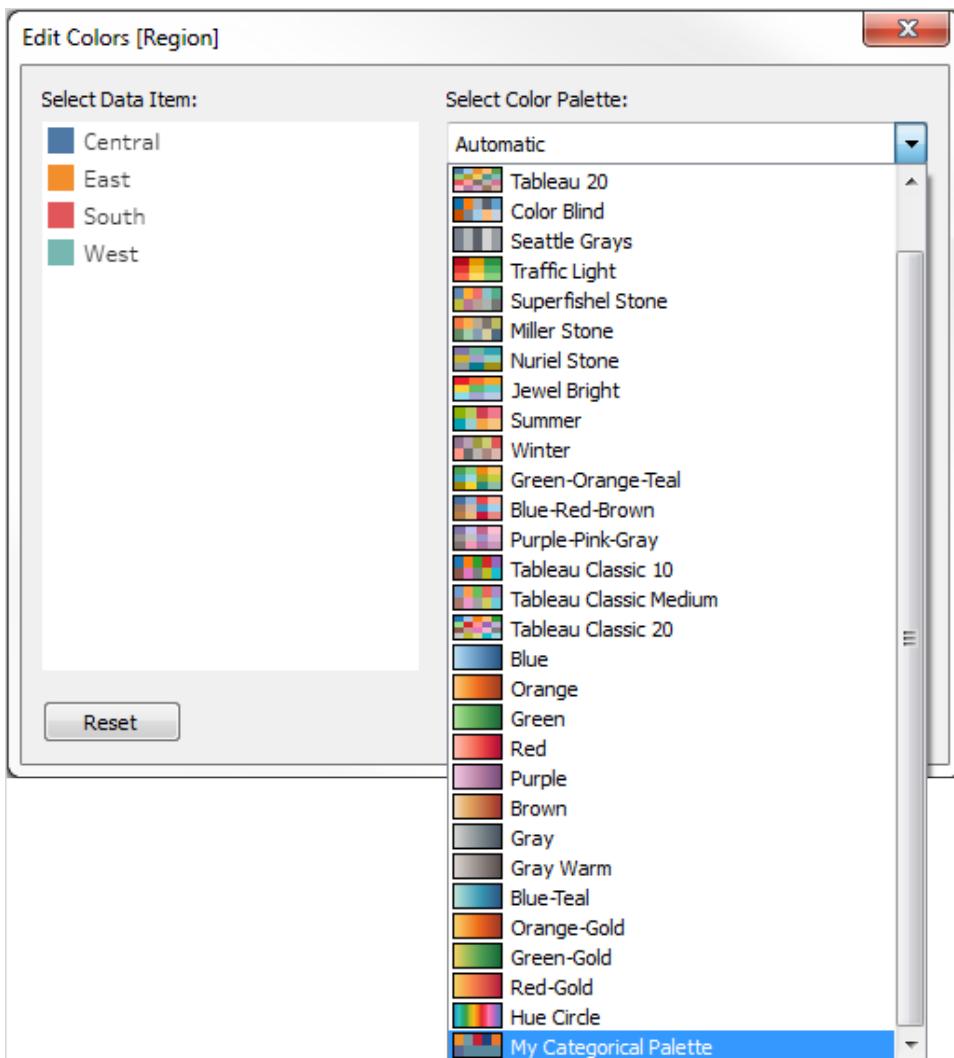
```
<color-palette name="My Categorical Palette" type="regular" >

<color>#eb912b</color>
<color>#7099a5</color>
<color>#c71f34</color>
<color>#1d437d</color>
<color>#e8762b</color>
<color>#5b6591</color>
<color>#59879b</color>
</color-palette>
```

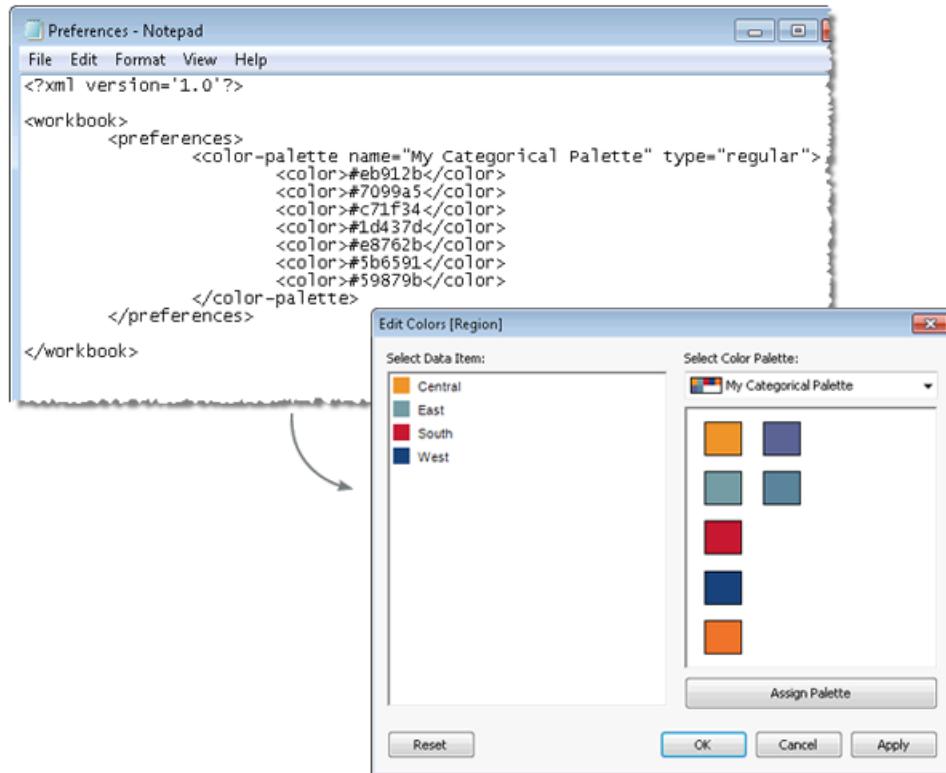
2. Save the **Preferences.tps** file and then restart Tableau Desktop.
3. Open the **Superstore - Sample** data source.
4. From the **Dimensions** pane, drag a discrete dimension, such as **Region**, to **Color**.
5. Click the color legend menu arrow and select **Edit Colors**.



6. In the Edit Colors dialog box, from the palette drop-down list, select your new custom palette.

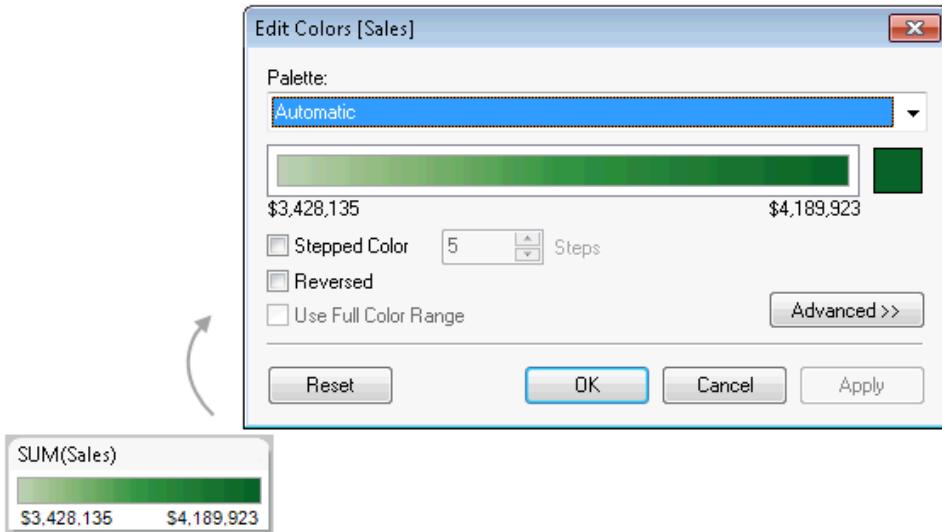


7. Click the **Assign Palette** button to assign the custom colors to each respective field.
8. Click **OK**.



Create a custom sequential color palette

Another type of palette is the sequential color palette. Typically, this type of palette shows a single color, varying in intensity. This type of color palette is used for continuous fields, typically for measures.



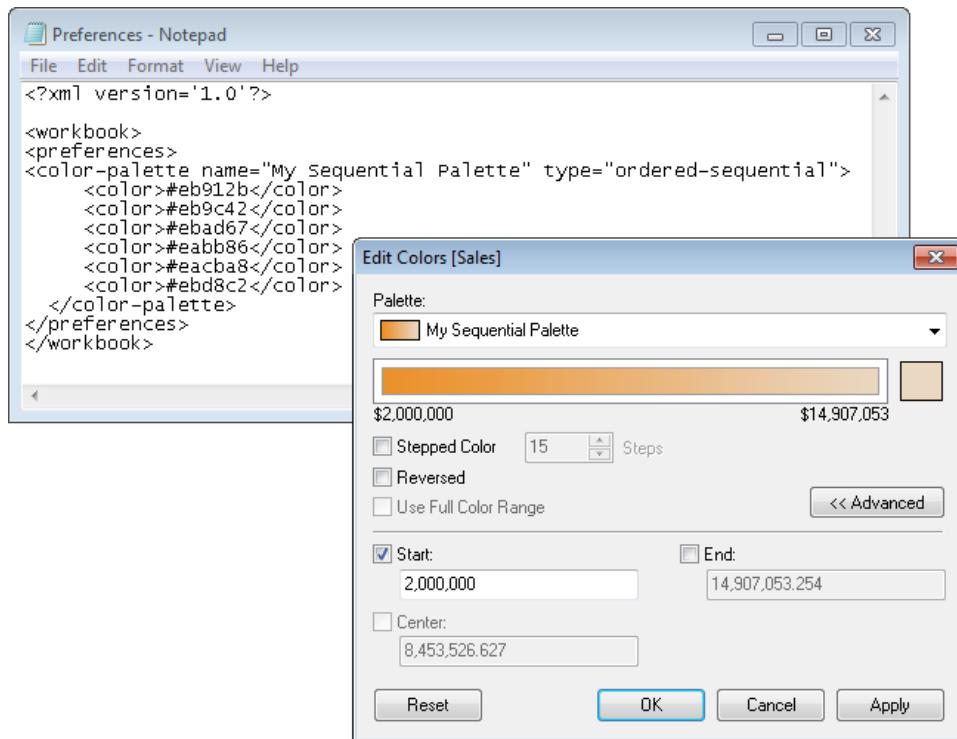
The following is an example of what to add between the preferences tags to add a sequential color palette. Note that the `type` attribute is specified as `ordered-sequential`, which identifies this palette as a sequential palette. Also, for sequential palettes you must specify each variant of the color in the sequential color range.

To create a custom sequential color palette:

1. In the `Preferences.tps` file, between the `preferences` tags, paste the following:

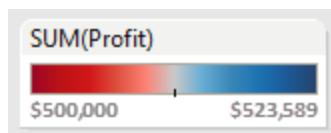
```
<color-palette name="My Sequential Palette" type="ordered-sequential" >
<color>#eb912b</color>
<color>#eb9c42</color>
<color>#ebad67</color>
<color>#eabb86</color>
<color>#eacba8</color>
<color>#ebd8c2</color>
</color-palette>
```
2. Save the `Preferences.tps` file and then restart Tableau Desktop.
3. Open the **Superstore- Sample** data source.
4. From the **Measures** pane, drag the measure (e.g., **Sales**) to **Color**.
5. Click the color legend menu arrow, and select **Edit Colors**.
6. In the Edit Colors dialog box, from the palette drop-down list, select your custom palette.
7. If you want each color gradation to be defined within a box, select the **Stepped Color** check box, and in the **Steps** text box, type the number of color steps you want to display in the bar.
8. Click the **Advanced** button.

9. Select the **Start** check box, and in the text box, type the low end number you want for the continuum.
10. Click the **Apply** button to see the result, and make adjustments as needed. The default for sequential color is to make the high end of the continuum pale and the low end intense; select the **Reversed** check box to make the high end intense and the low end, pale (this is the default when you keep the Automatic palette selection).



Create a custom diverging color palette

The third type of color palette is a diverging color palette. A diverging palette shows two ranges of values using color intensity to show the magnitude of the number and the actual color to show which range the number is from. Diverging palettes are most commonly used to show the difference between positive and negative numbers.



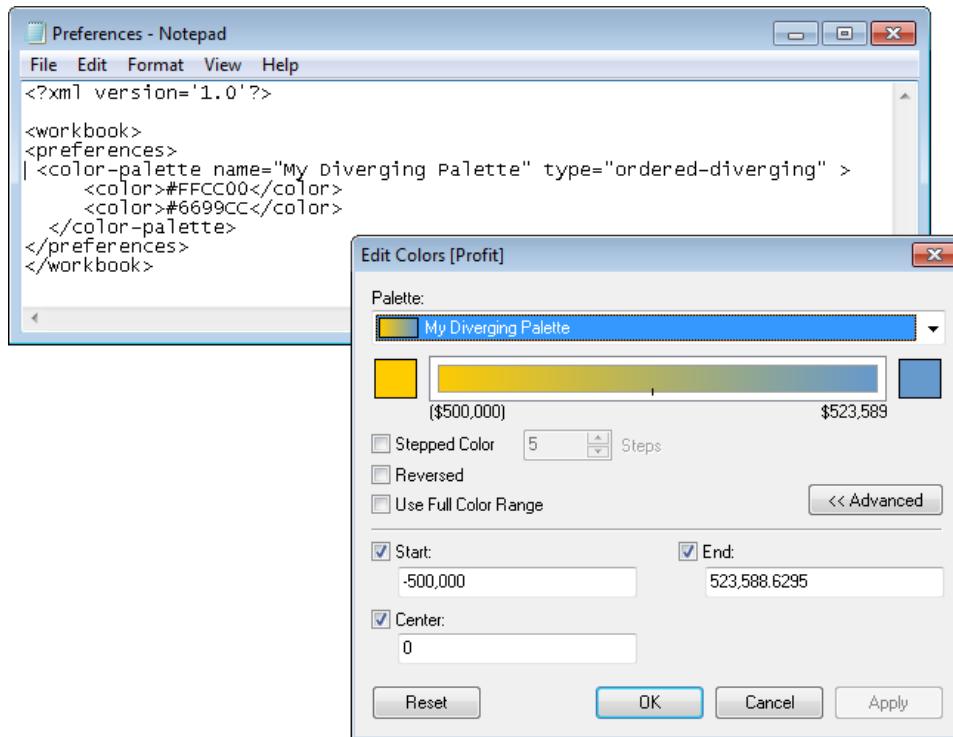
The following is an example of what to add between the `preferences` tags to add a diverging color palette. Note that the `type` attribute is specified as `ordered-diverging`, which identifies this palette as a diverging palette.

To create a custom diverging color palette:

1. In the **Preferences.tps** file, between the preferences tags, paste the following:

```
<color-palette name="My Diverging Palette" type="ordered-diverging">
<color>#eb912b</color>
<color>#59879b</color>
</color-palette>
```

2. Save the **Preferences.tps** file and then restart Tableau Desktop.
3. Open the **Superstore - Sample** data source.
4. Click the **Assign Palette** button. The colors in the palette are used in the order they appear in the Preferences file.



If you add a sequential or diverging palette, remember to change the "type" attribute from "regular" to one of the following:

- ordered-sequential
- ordered-diverging

(Optional): Assign a default custom palette to dimensions and measures and publish as a data source

After you save the workbook, the custom color palette information is embedded in the workbook (for Excel and text file-based workbooks, in the .twbx) and therefore only available for

that workbook. This means that colors that are in use are shown for anybody opening the that particular workbook. If they don't have the modified preferences file, they can't use the color information for any new color encoding.

To allow new color encoding using the custom color palette or to standardize a custom color palette for the Tableau workbooks in your organization, you can create the custom color palette using one of the options above, and then publish it as a Tableau Server data source.

1. On the same computer from which you modified the **Preferences.tps** file, open Tableau Desktop.
2. Open the **Superstore - Sample** data source.
3. Right-click a field in the **Data** pane, and select **Default Properties > Color**.
4. In the Edit Colors dialog box, associate the field values with the custom color palette, and then click **OK** when finished.
5. From the **Data** menu, select the data source, select **Publish to Server**, and then complete process to publish the data source.

After publishing the data source to Tableau Server, connect any new workbooks to this data source to use the custom color palette.

Use Custom Fonts

Tableau provides a variety of fonts for you to use in your visualizations, including the fonts in its default typeface: Tableau. However you might want to use a font other than what's provided. For example, you may want to use a font that's unique to your company's brand.

To use a custom font in Tableau, the first step is to install it on the computers that are running Tableau. This includes Tableau Desktop and, if you're publishing to Tableau Server, any computers running Tableau Server. To ensure that Tableau Server can render the font correctly, make sure the font is "web safe" in all browsers. If a font specified in Tableau Desktop isn't also installed on Tableau Server, Tableau Server uses a default font.

For more information about installing fonts, see the following links:

- [How to install or remove a font in Windows](#) (Microsoft Knowledge Base)
- [Mac Basics: Font Book](#) (Apple Support)

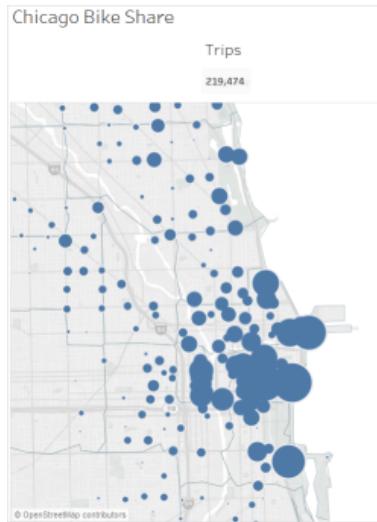
After you install the font, it appears in the Formatting pane in Tableau Desktop, where you can use it to change fonts at the workbook, worksheet, or individual view part levels. See [Format at the Workbook Level](#) on page 1059, [Format at the Worksheet Level](#) on page 1062, and [Format Parts of a View](#) on page 1068 for details.

Rebrand a Dashboard

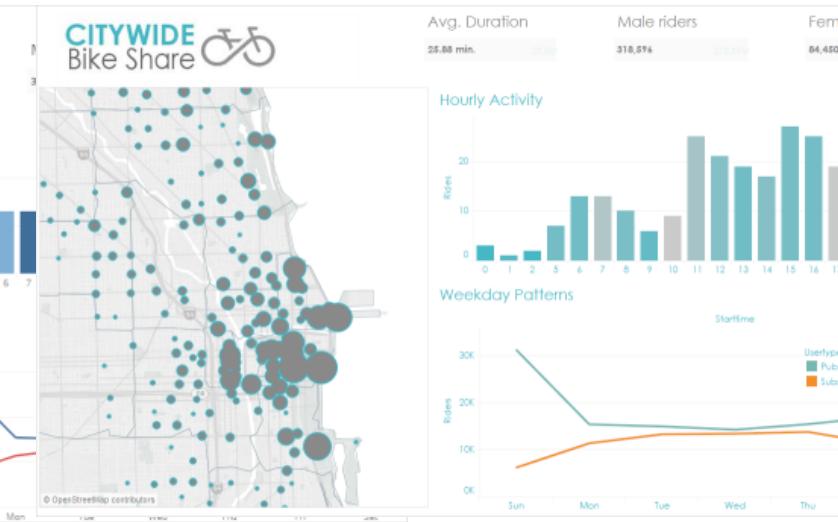
Tableau Desktop builds visual best practices into its default formatting settings so that your visualizations look great from the start. But you might want to change formatting settings for a dashboard—for example, to align them with your organization's brand.

This topic steps you through a dashboard rebranding process for a fictitious organization called Citywide Bike Share. Its brand includes turquoise, orange, and gray for its colors and Century Gothic for its font. The overall look is light and uncluttered and, of course, it's all about bikes!

Before



After

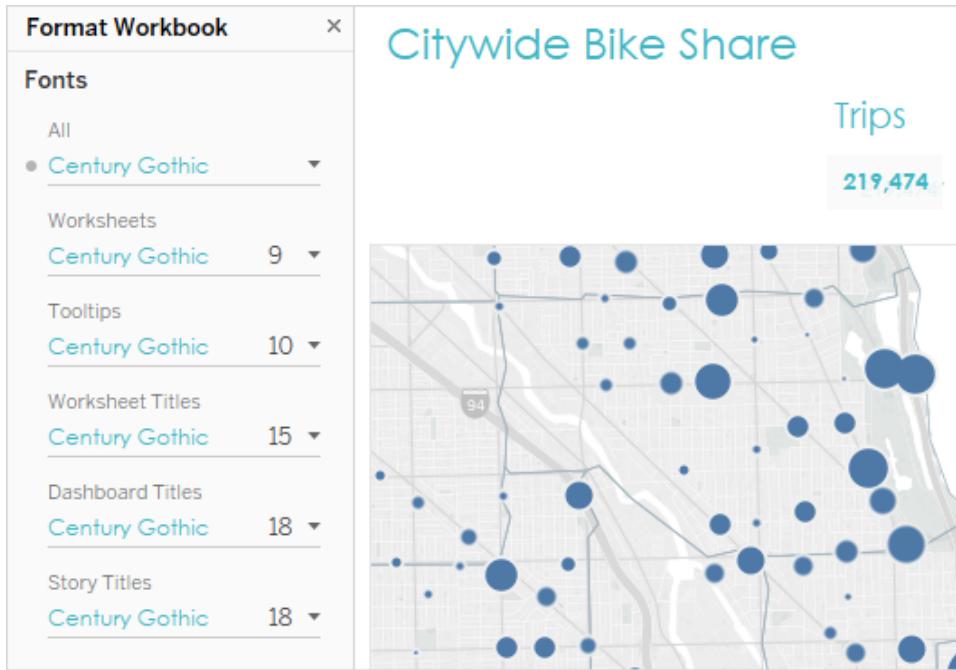


Change the fonts and font color of your titles

Fonts and colors are usually an important part of an organization's brand. The quickest way to make a fast, large-scale change to all the titles in your dashboard is to change them at the workbook level. A workbook is the largest possible container for formatting settings.

1. Click **Format > Workbook**.
2. Make your changes under **Fonts**, in the **Format Workbook** pane.

Here we've changed the font to Century Gothic and the font color to turquoise. We've chosen to change every title in the workbook, but you could also change the look of individual titles.



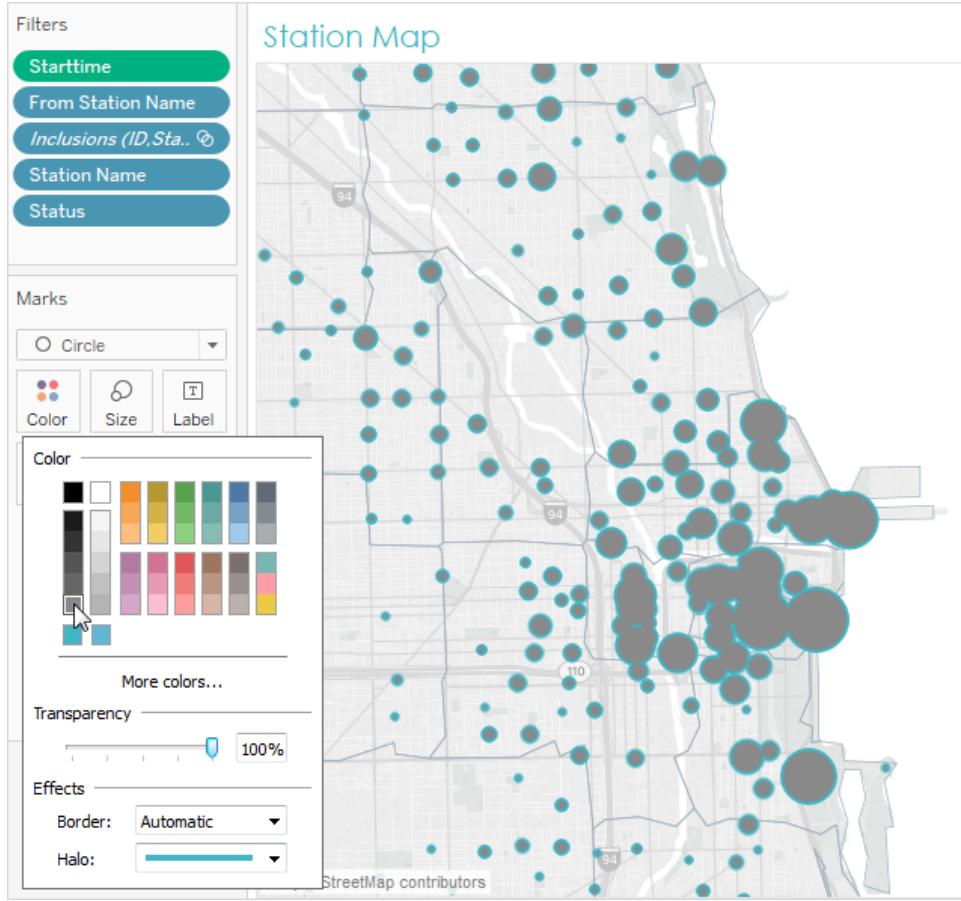
If the font you want to use doesn't exist in Tableau by default, you can add it. See [Use Custom Fonts](#) on page 1108 for details.

Change the marks colors

Marks represent data in a view. In this rebranding scenario, the marks show bike share locations. Changing marks colors is another way to communicate your organization's brand to your audience. For certain types of marks, you can also add and format halos, which is a colored ring around a mark.

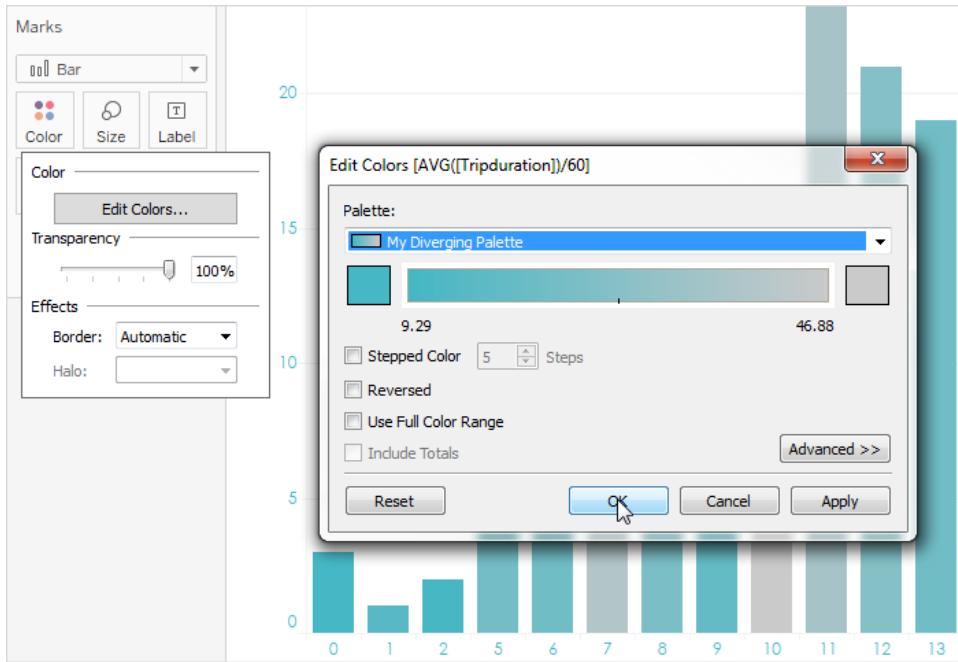
1. Go a view on your dashboard and click **Go to Sheet** from the view's shortcut menu.
2. In the view, click **Color** on the **Marks** card.
3. Make your new color selection.

Here we've changed the marks in the Chicago Ride Share map to use gray with a turquoise halo:



4. Repeat the above steps for every view in your dashboard.

If a color you want to use isn't available, you can create your own custom palette. See [Create Custom Color Palettes](#) on page 1100 for details. For example, here we've changed the colors in the bar chart by created a custom diverging palette.



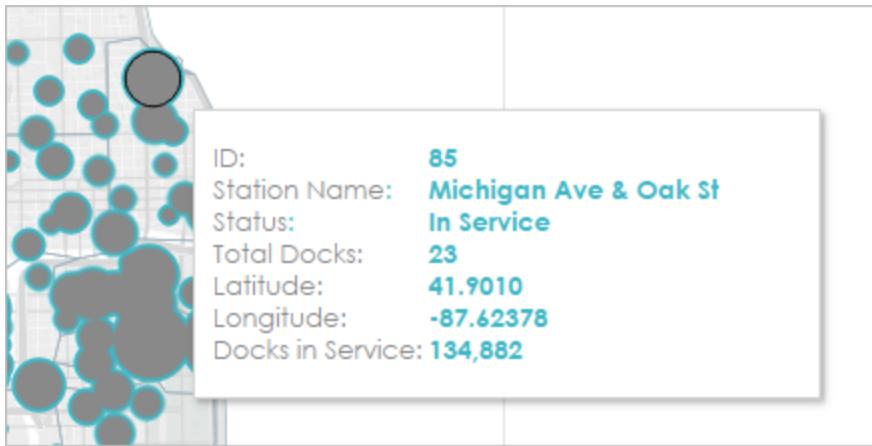
Now, with just a few changes to fonts and marks colors, the dashboard branding is already taking shape.



Customize your tooltips

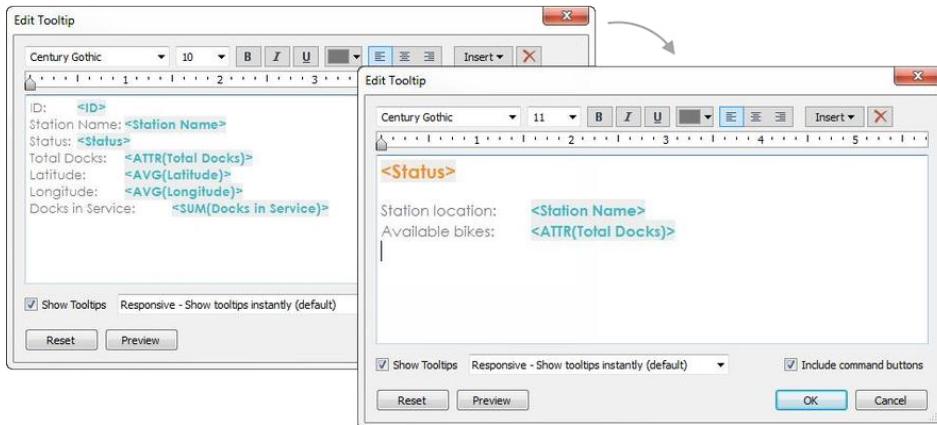
Tooltips appear when you hover the mouse pointer over a mark. Tooltips appear by default for most views. They're a great way to reinforce your brand and tell a better story.

Here's what the map tooltips looked like by default:

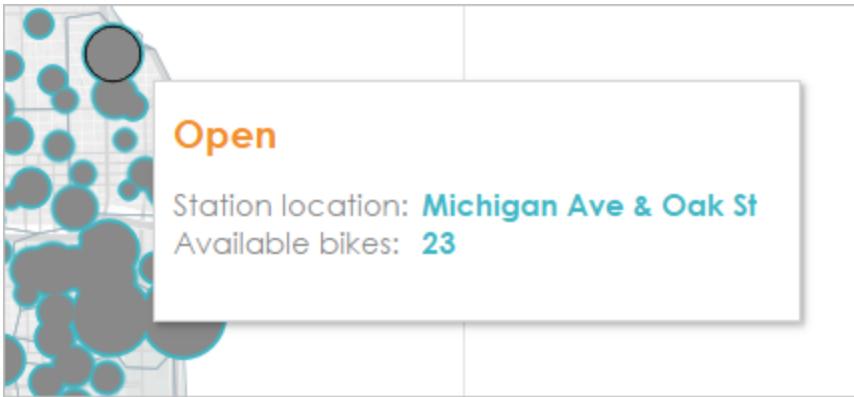


1. Go to the view's worksheet and click **Worksheet > Tooltip**.
2. Confirm that the tooltip has the details you want to display. You can change fonts, order, wording, alignment, and color.

For this rebranding scenario, we reduced the number of items displayed in the tooltip so that it's easier for users to quickly see what they're most interested in. We also reworded a few items and added the color orange as a contrasting, brand-appropriate color:



Here's what the map tooltips look like now:

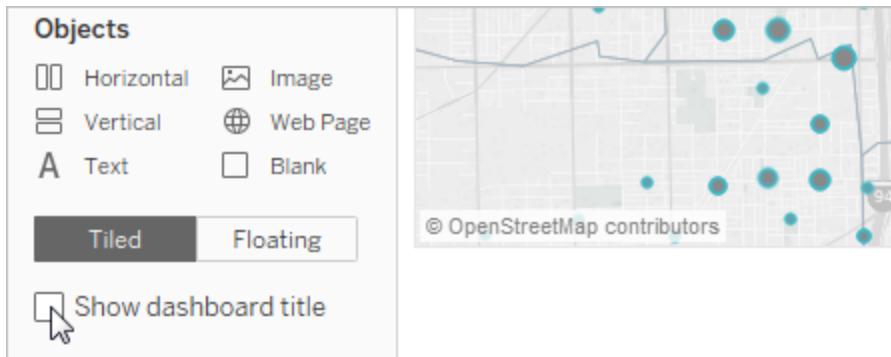


Add an image or logo

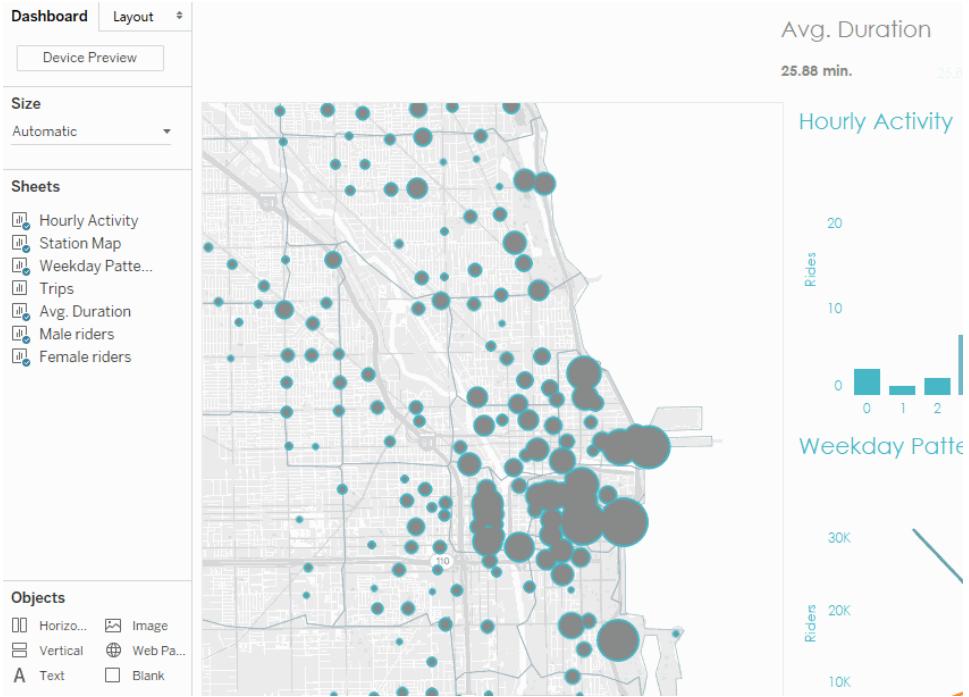
Adding your own images can also be a way to bring your organization's brand into a workbook. For example, you can import a logo and use it at the top of your dashboard instead of displaying the default dashboard title.

To remove the title and add an image:

1. In the Dashboard pane, stop displaying the dashboard title by clearing the **Show dashboard title** check box in the lower left corner.



2. Drag the **Image** object to the area where you want to display your logo. You may also need to adjust or remove other dashboard objects.



Click the image to replay it.

Here's our final, rebranded workbook:



Actions

Tableau allows you to add context and interactivity to your data using actions. Link to web pages, files, and other Tableau worksheets directly from your analytical results. Use the data in one view to filter data in another as you create guided analytical stories. Finally, call attention to specific results using highlighting.

For example, in a dashboard showing home sales by neighborhood you could use actions to help you quickly see relevant information for a selected neighborhood. Select a neighborhood in one view which then highlights the related houses in a map view, filters a list of the houses sold, and opens a web page showing census data for the neighborhood.

Note: Actions behave differently in workbooks that use cube data sources. Cube data sources do not accept actions from relational or other cube data sources. For example, suppose you have a workbook that contains a view that uses a MySQL data source, a second view that uses a cube data source A, and a third view that uses a cube data source B. Actions in a view that use the MySQL data source will not affect the views that use the cube data sources. However, actions in the view that use the cube A data source can affect the view that uses the MySQL data source. The view that uses the cube B data source will not be affected in this case.

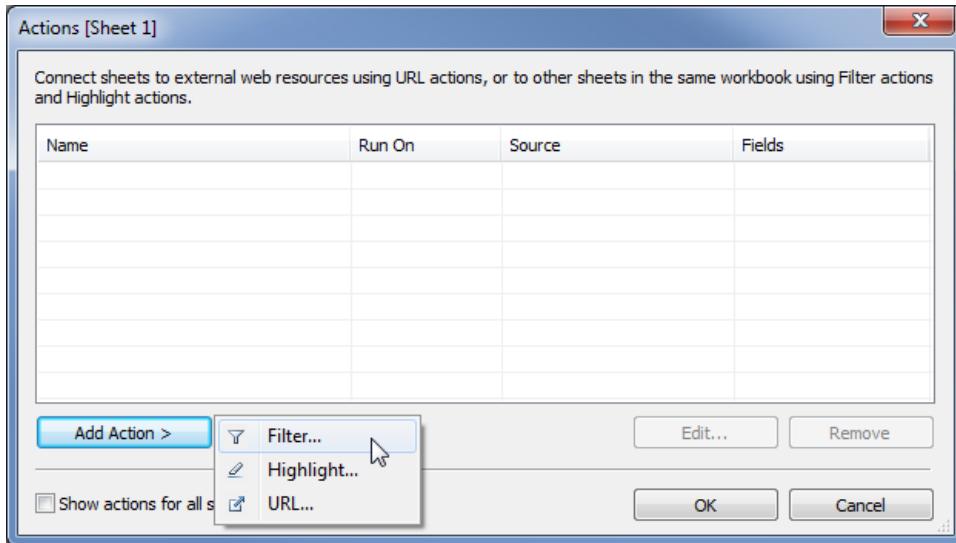
There are three kinds of actions in Tableau: Filter, Highlight, and URL actions.

Filter Actions

Filter actions are a way to send information between worksheets. Typically a filter action is used to send information from a selected mark to another sheet showing related information. For example, when looking at a view showing the sales price of houses, you may want to be able to select a particular house and show all comparable houses in a different view. You could define a filter action to accomplish this task. First you need to decide what comparable means. In this case, say that comparable houses are houses with a similar sale price and square footage. A filter action to show comparable houses can be defined by selecting a destination worksheet and defining filters on sales price and square footage.

Filter actions work by sending the data values of the relevant source fields as filters to the destination sheet. If you launch the filter action described in this example from a house that sold for \$450,000, the destination sheet will have a filter to only show houses that sold for the same amount.

1. On a worksheet, select **Worksheet > Actions**. From a dashboard, select **Dashboard > Actions**.
2. In the Actions dialog box, click **Add Action** and then select **Filter**.

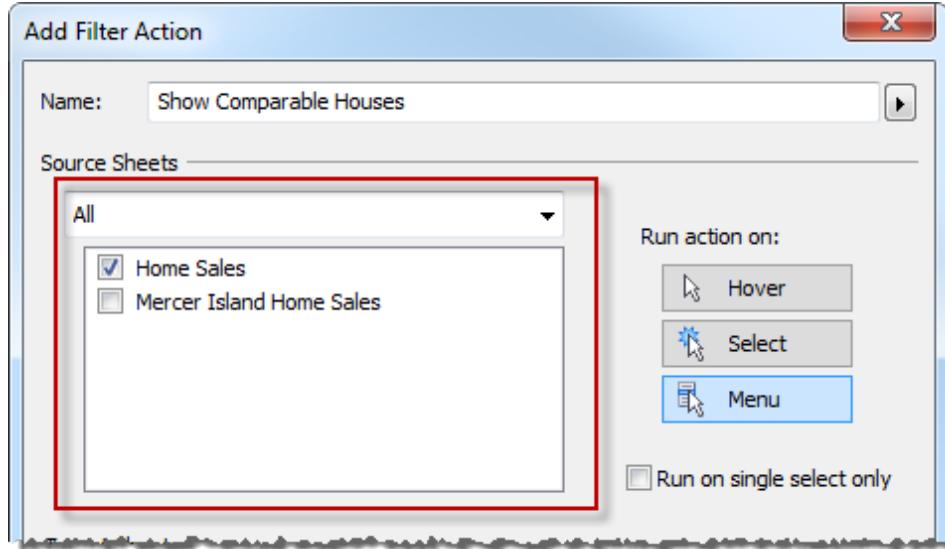


3. In the subsequent dialog box specify a name for the Action.

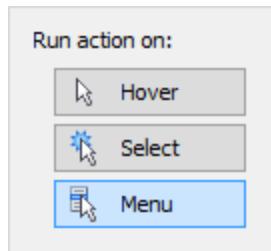
Use a name that defines the action. If you choose to run the action using the menu the name is the option that shows on the menu. For example, when sending housing information from one sheet to a map, the name could be "Map all comparable houses sold in February" You can use variables in the name that will be filled in based on the values of the selected field.



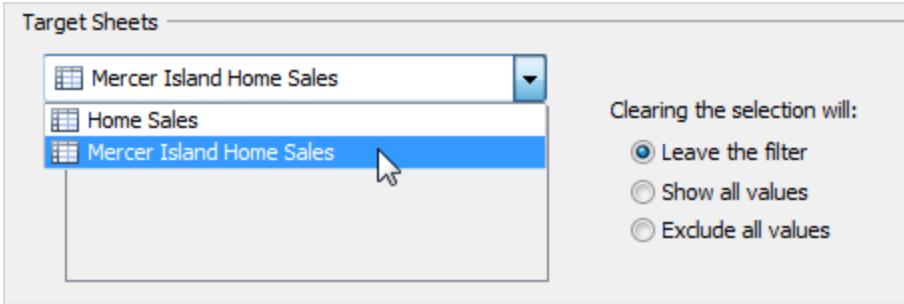
4. Use the drop-down list to select a source sheet or data source. When you select a data source or dashboard sheet you can further refine by selecting the individual sheets you want to launch the action from.



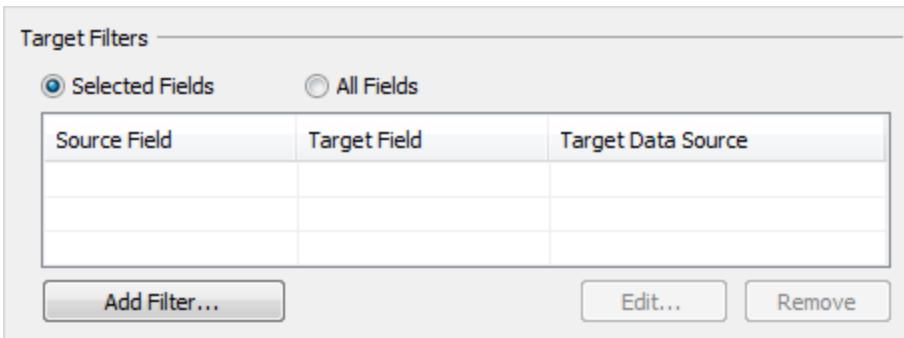
5. Then select how you want to launch the action. Select one of the following options:
 - **Hover** - rest the pointer over a mark in the view to run the action. This option works well for highlight and filter actions within a dashboard.
 - **Select** - click on a mark in the view to run the action. This option works well for all types of actions.
 - **Menu** - right-click (control-click on Mac) a selected mark in the view and then select an option on a the context menu. This option works well for filter and URL actions.



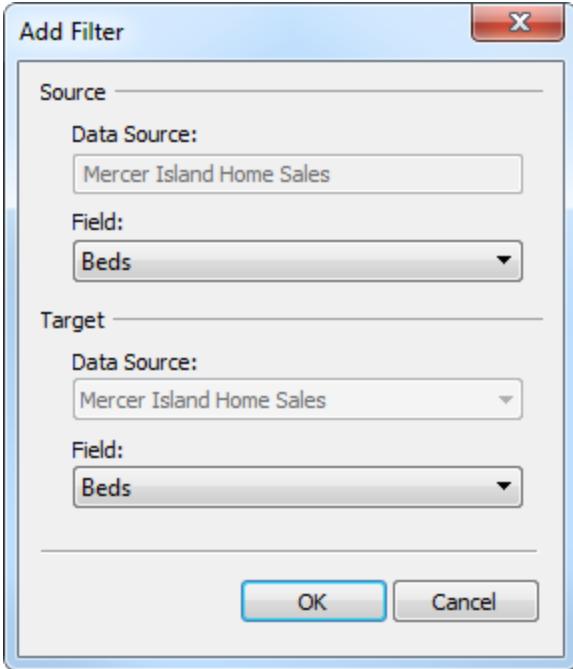
6. Use the second drop-down list to select a target sheet. When you select a dashboard sheet you can further refine the target by selecting one or more sheets within the dashboard.



7. Specify what to do when the select is cleared in the view. You can select from the following options:
 - **Leave the filter** - leaves the filter on the target sheets. The target views in the dashboard will show the filtered results.
 - **Show all values** - changes the filter to include all values.
 - **Exclude all values** - changes the filter to exclude all values. This option is useful when you are building dashboards that only show some sheets if a value in another sheet is selected.
8. Setup one or more filters to specify the data that you want to show on the target sheets. You can filter on **All Fields** or define filters on **Selected Fields**.
9. If you are defining filters for specific fields click **Add Filter**.



10. In the Add Filter dialog box, select a source and target data sources and fields. When you run the action from a specific mark on the source sheet, a filter is added to the target sheet that only includes values for the target field that match the values of the source field. In the comparable houses sheet link example, the Source Field is Beds and the Target Field is Beds. That means when you launch the sheet link for a house that has three bedrooms, the destination worksheet will only show houses that also have three bedrooms.



11. When finished, click **OK** three times to close all open dialog boxes and return to the view.

If you are connected to a relational data source, you can add sheet links across data sources even if the field names are not exactly the same. One data source may have a field titled Latitude while another has a Lat field. Using the drop down lists in this dialog box, you can associate the Latitude field to the Lat field. When using a multidimensional data source, the destination sheet must use the same data source as the source sheet. Moreover, the source field names must match the destination field names. In Tableau, multidimensional data sources are supported only in Windows.

The fields available in the Target Field drop-down list are dependent on what you selected as the Source Field. Only fields with the same data type as the source field can be selected as a destination field.

Highlight Actions

Highlight actions allow you to call attention to marks of interest by coloring specific marks and dimming all others. You can highlight marks in the view using a variety of tools. For example, you can manually select the marks you want to highlight, use the legend to select related marks, use the highlighter to search for marks in context or create an advanced highlight action.

The following table describes the different methods you can use to highlight marks in a view, dashboard, or story.

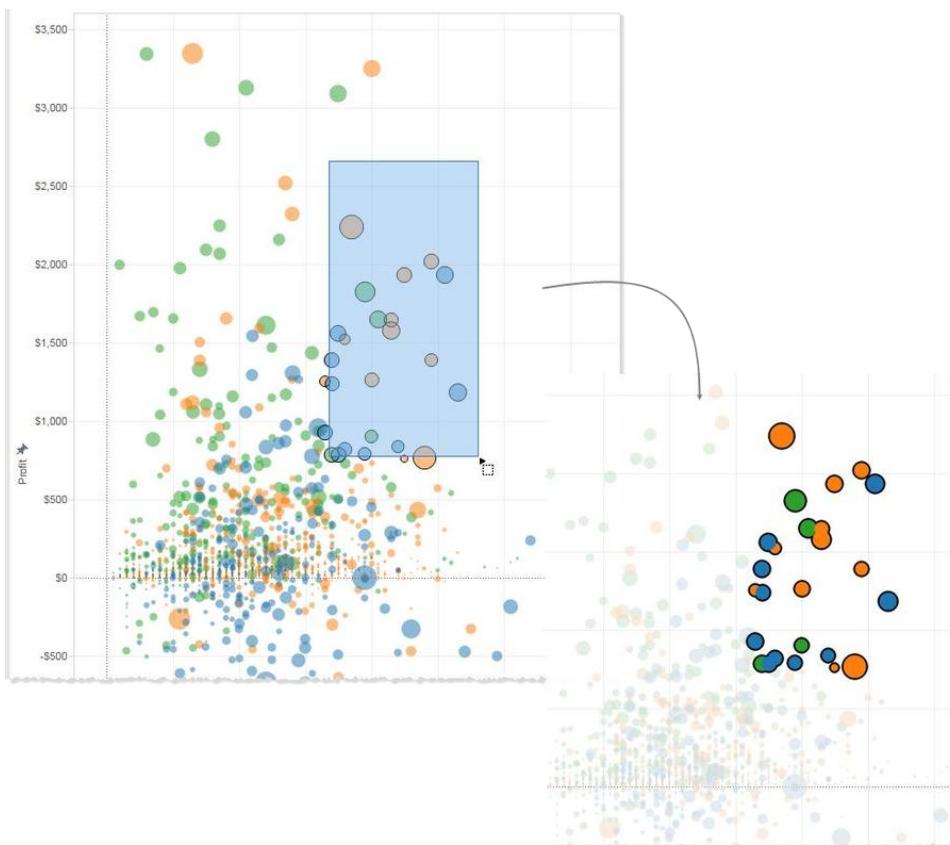
Highlight method	Benefits	When you might use this
Select marks in a view.	<ul style="list-style-type: none"> Manually select a group of marks to highlight in a view. Your selection is saved with the workbook and can be included in stories and when publishing. 	<ul style="list-style-type: none"> When you want to manually highlight a selection of marks and dim all others. Works well with small domains or views with a small amount of data.
Legends	<ul style="list-style-type: none"> Supports one-way and two-way highlighting. Highlight on color, size, or shape. You can disable or enable the highlighting action for the workbook or sheets from the toolbar. Your selection is saved with the workbook and can be included in dashboards and stories and when publishing. 	<ul style="list-style-type: none"> When you want to focus on select members in a view and dim all others. When you want to highlight using only the legend or the legend and the view. Works well with small domains or views with a small amount of data.
Highlighter	<ul style="list-style-type: none"> Search for data points in a view using keywords or select from a drop-down list. Highlight marks while maintaining the context of the other data points. Values automatically update when the underlying data is updated. Highlighters added to worksheets also appear on dashboards and stories. 	<ul style="list-style-type: none"> When you want to highlight a mark or group of marks for a discrete field that is included in the view. When you want to do ad hoc comparisons with instant highlighting. Works well with large domains and large amounts of data.
Actions	<ul style="list-style-type: none"> Highlight data based on criteria that you define. Specify the source and target sheets to apply the highlight action to. 	<ul style="list-style-type: none"> When you want to build interactive exploration into a dashboard. When you want to highlight data points in a dashboard using specific fields.

- | | | |
|--|---|--|
| | <ul style="list-style-type: none"> Specify the fields to use for highlighting. You can specify different types of actions to run on the same click (for example, filter and highlight). | |
|--|---|--|

Selecting Marks to Highlight

The simplest way to add highlighting to a view is to select the marks you want to highlight. When you manually select a mark in the view, all other marks are dimmed to draw attention to the selection. Your selection is saved with the workbook and can be included in stories or when publishing.

You can select multiple marks by holding down the Ctrl key on your keyboard (Command key on Mac) while you select each mark. You can also use one of the advanced selection tools to select marks within a specific shape. For more information on how to use advanced selection tools, see [Select Marks](#) on page 566.



Legend Highlighting

Legend highlighting is a powerful analytical mode that you can use to focus on specific members in a legend that is shown in the view. When you turn on legend highlighting you can use the members in the legend to highlight marks associated with the selected items in the legend and dim all other marks.

You can enable either one-way  or two-way  highlighting to highlight marks in the view. This icon shows at the top of the legend to indicate the mode that you are using.

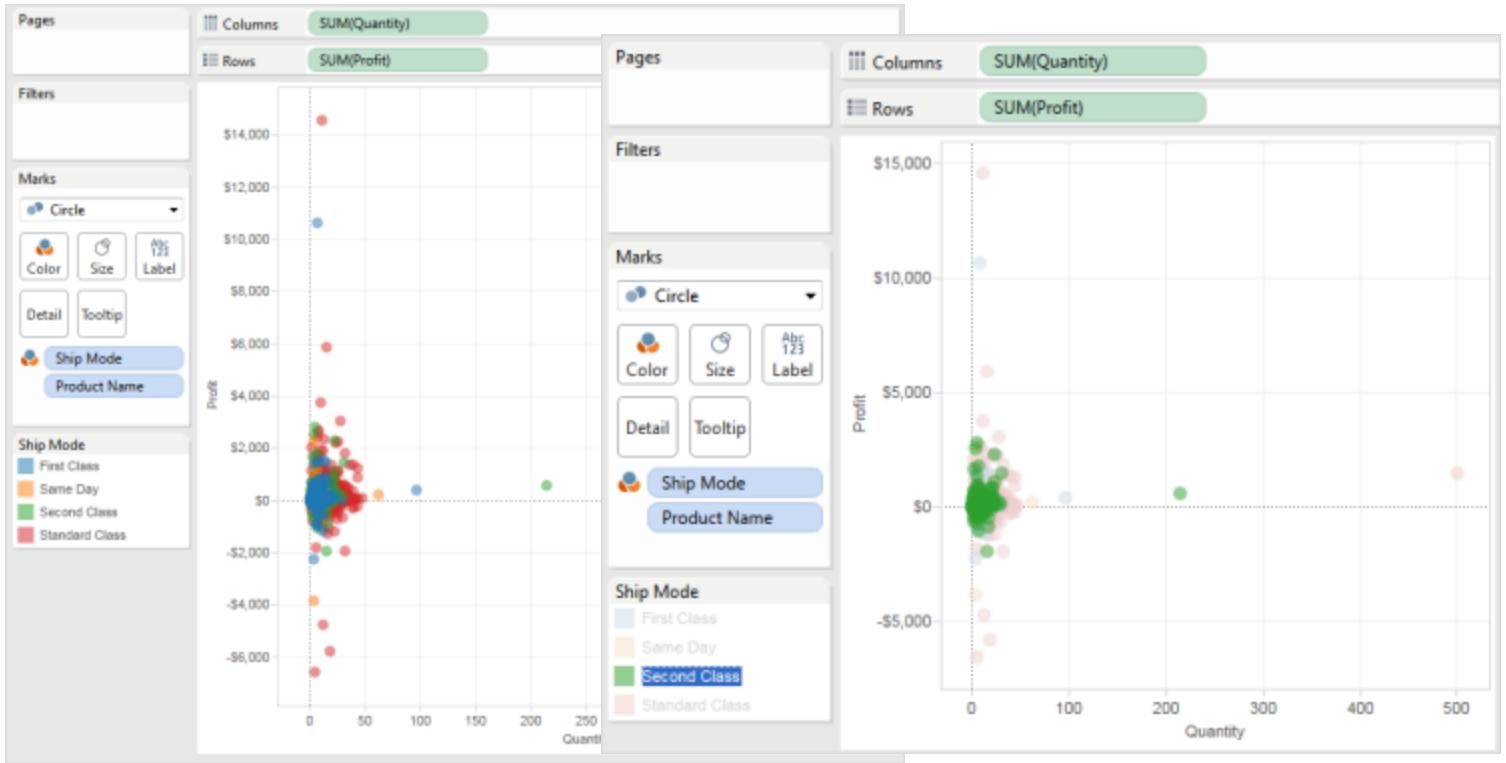
- One-way highlighting enables you to highlight marks using the values in the legend.
- Two-way highlighting enables you to highlight marks using either the legend or the view. This is the default. When you highlight marks in the view, the matching member in the legend is also highlighted.

You can also turn off highlighting for the workbook or the sheet from the toolbar. This action hides the highlight icon on the legend. For more information about using the toolbar highlighting option, see [Highlight Toolbar Button](#) on page 1128.

In this example, the views below show the relationship between order quantity and profit for several products. The view on the left uses the standard functionality for the color legend, where all marks are colored based on their shipping mode. The view on the right uses legend highlighting to call out the products that were delivered via Second Class shipping.

Normal Color Legend

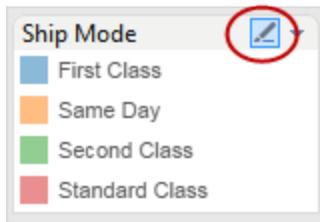
Color Legend Highlighting Enabled



You can easily switch between legend highlighting and normal modes using the legend card menu. For color legends, if you like how a view is highlighted, you can assign the highlight colors to the color palette. The old colors are replaced with the highlight colors.

To turn on legend highlighting

1. Click the **Highlight** button at the top of the legend or select **Highlight Selected Items** on the legend card menu. This example shows highlighting using the color legend. Color legend highlighting is turned on by default.



2. Select an item in the color legend.

After legend highlighting is turned on, you can quickly focus on specific data in the view by selecting different items in the color legend. When you turn on legend highlighting, a highlight action is created that you can modify in the Actions dialog box.

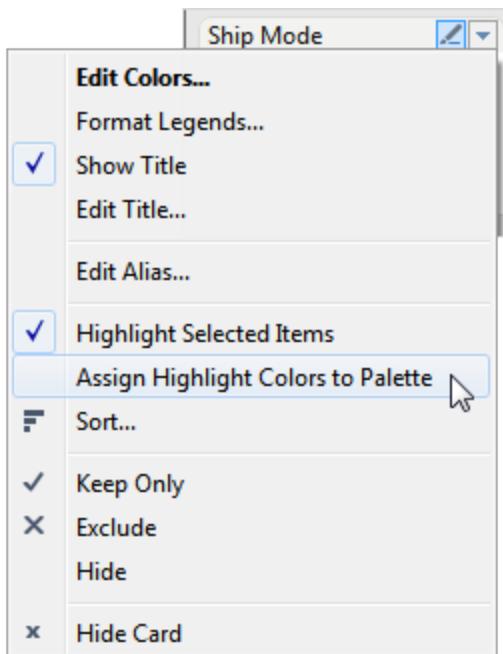
To turn off legend highlighting

- Click the **Highlight** button  at the top of the legend or select **Highlight Selected Items** on the legend card menu. This changes highlighting to one-way and you can use the legend to highlight matching marks in the view.

When you turn legend highlighting off the action is removed from the Actions dialog box.

If you like how the view is highlighted and want to keep a specific member highlighted even when you turn off legend highlight mode, you can assign the highlight colors to the existing color palette. The original color legend is discarded and the highlight colors become the new color palette for the legend.

To assign the highlight colors to the color palette, select **Assign Highlight Colors to Palette** on the color legend card menu.



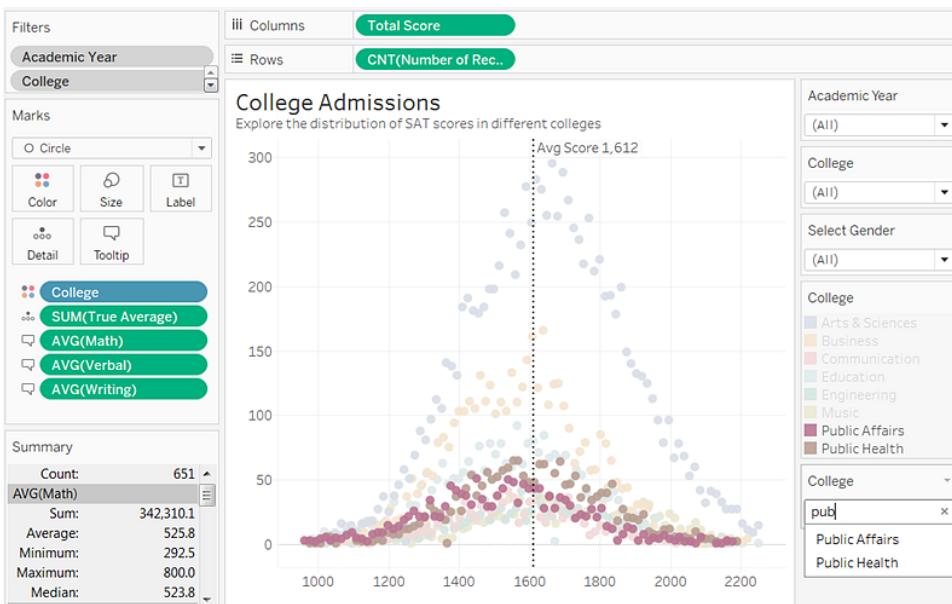
Highlight Data Points in Context

When you have a view with a large amount of data you might want to explore your data interactively and highlight a specific mark or group of marks while still maintaining the context of where those marks show in your view.

To do this you can turn on the Highlighter for one or more discrete fields that are included in your view and that affects the level of detail. For more information about setting the level of detail in a view, see **Dimensions and the Level of Detail** on page 277.

You can use keywords to search for matching data points. The highlighter immediately highlights the marks that match or partially match your keyword search. If you update the underlying data source for your view the data shown in the highlighter is automatically updated too.

In the example below, the Highlighter is turned on for the **College** field. Entering a partial keyword search for **Public** returns two possible matches. In the view, Tableau highlights the group of marks that math the partial search; **Public Affairs** and **Public Health**.



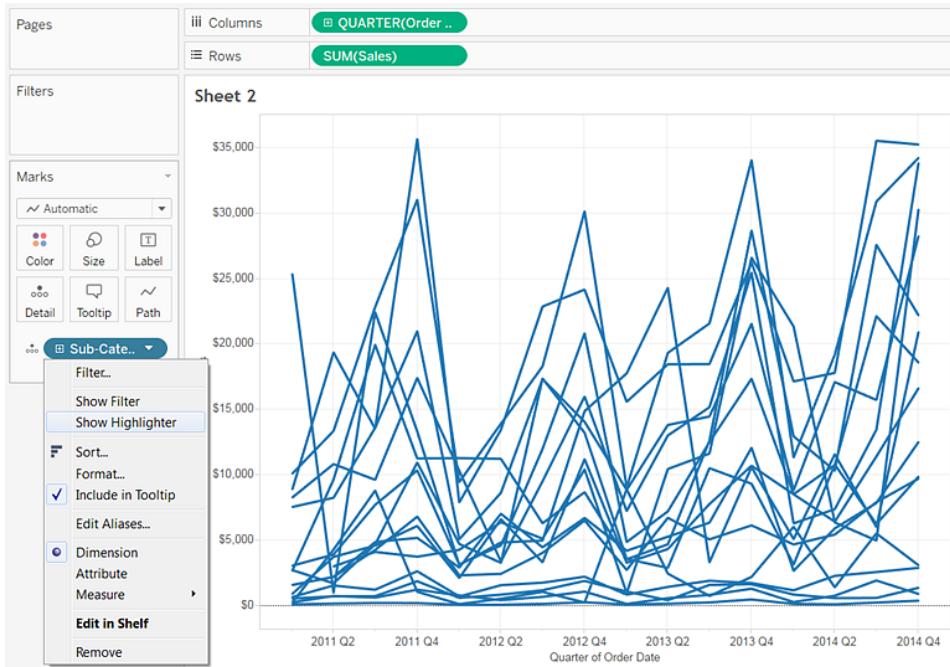
You can turn on highlighters for as many discrete fields in our view as you need. However, you can only highlight using one value set at a time. If you want to include the highlight option on dashboards and stories, turn them on in the worksheet before you add them to the dashboard, or select the worksheet on the dashboard, and then select **Analysis > Highlighters** from the top menu and select from the fields in the list.

If you want to change the format for the highlighter card after you turn it on, select **Format > Highlighter**.

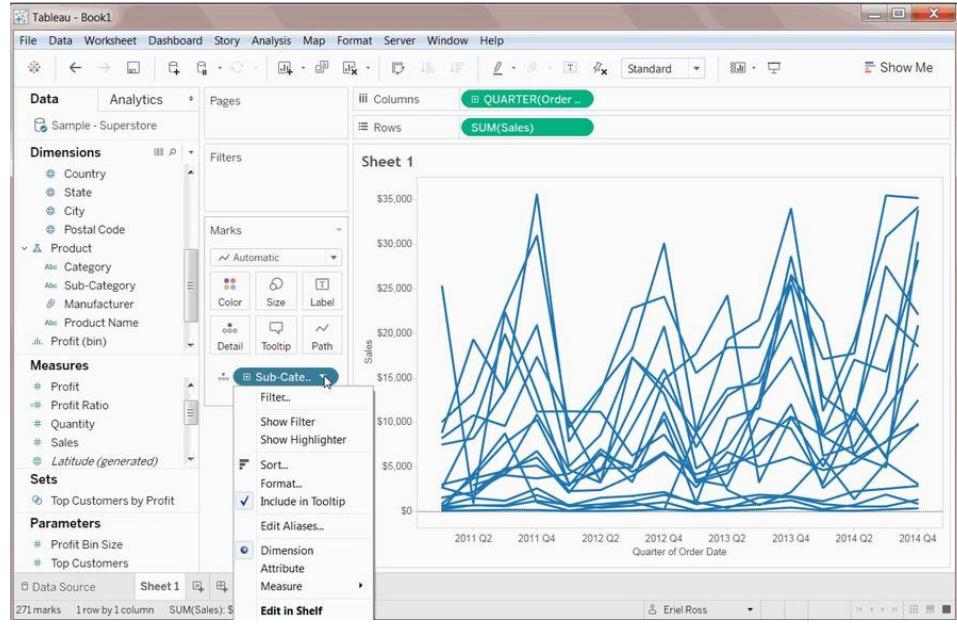
Note: Highlighters work across worksheets on a dashboard if the same field is included across those worksheets. If the same field is not found, then no match is found for the highlighted value, and the values in those worksheets are dimmed. If you have multiple highlighters showing on a dashboard, only one highlighter can be active at a time. In this situation, the highlight shows for the highlighter that was used last.

To turn on highlighting

1. Right-click on a discrete field (dimension) that is included in the view and impacts the level of detail in the view. Then select **Show Highlighter** from the context menu.



2. Click in the highlighter dialog box and do one of the following:
 - Enter a keyword to search for matching marks to highlight.
 - Enter a partial key word to search for all relevant matches that contain your search text.
 - Select an item in the drop-down list. You can select one item at a time.
 - Hover over items in the drop-down list to highlight marks in the view for an ad hoc analysis experience.



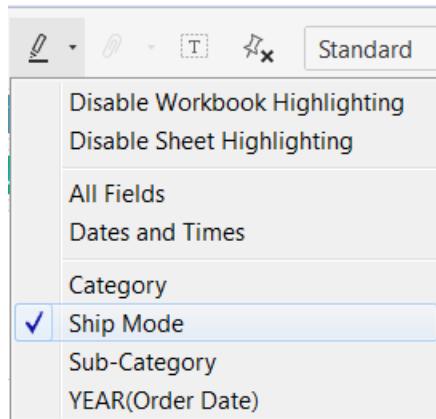
Click the image to replay it

3. Repeat steps 1 and 2 to add additional highlighters.

You can also show mark labels on highlighted marks. To turn on mark labels, on the Marks card, click **Label**, select the **Show mark labels** check box and select **Highlighted** in the Marks to Label section.

Highlight Toolbar Button

Another way to add a highlight action is to use the highlight button in the toolbar. Similar to legend highlighting, the toolbar button lets you highlight a collection of related marks in the view and functions the same way that two-way highlighting does. To turn on highlighting, select the fields you want to use for highlighting on the toolbar menu. The drop-down list shows the active fields in the view. Then select a mark in the view to see the related data.



For example, the view below shows **Sales** vs. **Profit** by **Region**. If you turn on highlighting for **Ship Mode**, when a mark is selected, all other marks that were shipped via that mark's ship mode are highlighted. In this example you can quickly see all products from across the United States that were shipped via Second Class.

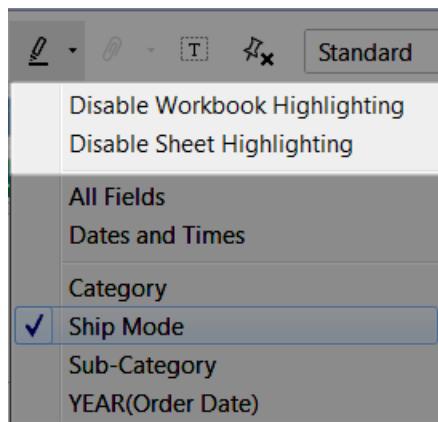


The toolbar menu also lets you highlight on **All Fields** or **Dates and Times**. **All Fields** considers all fields in the view when identifying matching records; **Dates and Times** considers all date and time fields in the view.

When you use the Highlight toolbar button an action is created in the Actions dialog box. You can modify the action to create more advanced highlighting behavior. For more information about editing actions, see [Creating Advanced Highlight Actions](#) below.

Finally, you can use the toolbar button to disable highlighting across the entire workbook or for just the active sheet. These options disable highlighting for legends only. They do not turn off the ability to manually highlight marks or use the Highlighter control.

When you turn off highlighting from the toolbar button, the highlight icon is hidden on the legend, and the **Highlight Selected items** menu option is grayed out on the context menu for the legend.

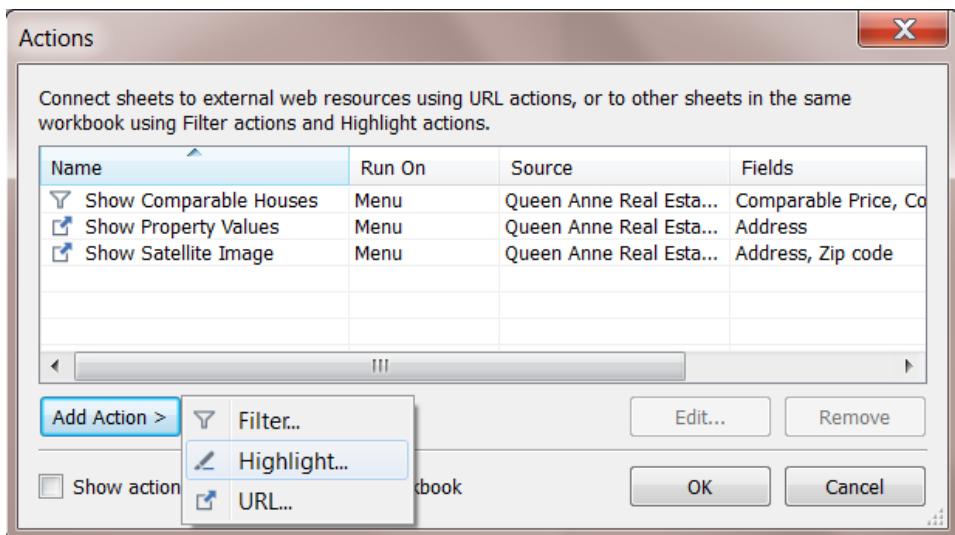


[Creating Advanced Highlight Actions](#)

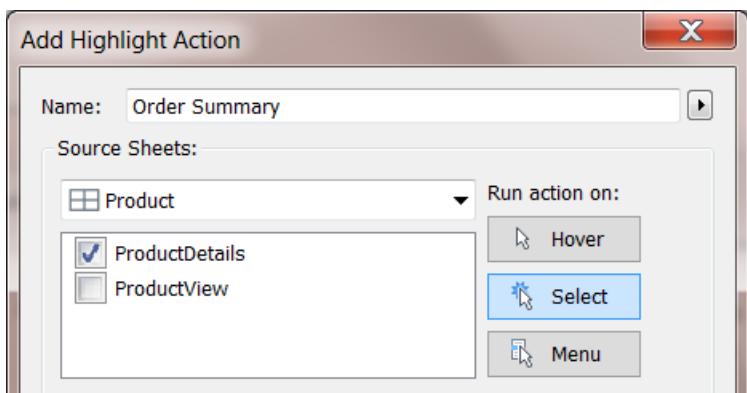
You can define more advanced highlight actions using the Actions dialog box. There you can specify source and target sheets and the fields you want to use for highlighting. Follow the steps below to create a highlight action.

To create a highlight action:

1. On a worksheet select **Worksheet > Actions**. From a dashboard, select **Dashboard > Actions**.
2. In the Actions dialog box click the **Add Action** button and then select **Highlight**.

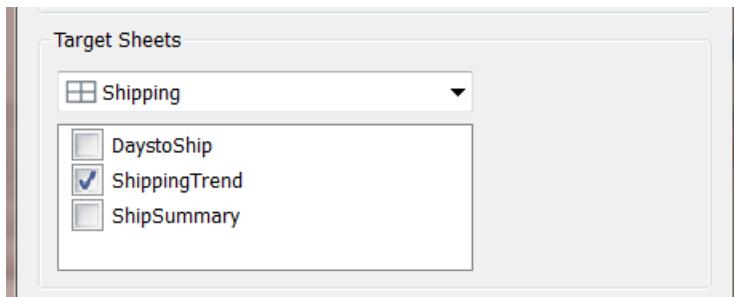


3. Name the action to identify it in the Actions dialog box. Try to make the name descriptive, such as, *Highlight Products Shipped by Delivery Truck*. You can select variables from a drop-down list and use them in the name. Then they are filled in based on the values of the selected field.
4. Use the drop-down list to select the source sheet or data source. If you select a data source or a dashboard sheet, you can further select individual sheets within them.

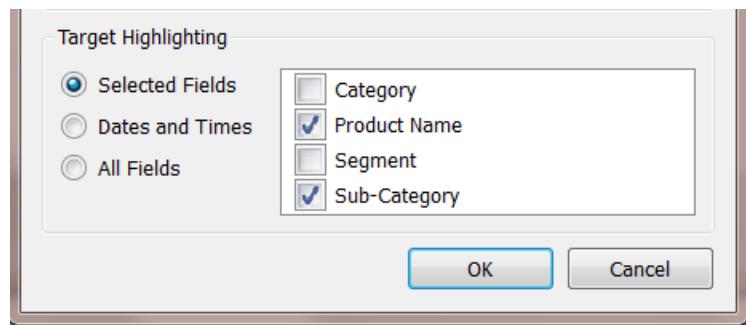


5. Select how you want to trigger the action. You can select from the following options:
 - **Hover** - Rest the pointer over a mark in the view to run the action. This option works well for highlight and filter actions within a dashboard.
 - **Select** - Click a mark in the view to run the action. This option works well for all types of actions.
 - **Menu** - Right-click (control-click on Mac) a selected mark in the view and then

- select an option on the context menu. This option works well for filter and URL actions.
- Select a target sheet. If you select a dashboard, you can further select individual sheets within the dashboard.



- Select the fields you want to use for highlighting. Select from the following options:
 - Selected Fields** - Marks in the target sheet are highlighted based on select fields. For example, highlighting using the Ship Mode field will result in an action that highlights all marks in the target sheet that have the same ship mode as the selected mark in the source sheet.
 - Dates and Times** - Marks in the target sheet are highlighted when their date and time match those of the marks selected in the source sheet. All dates and time fields are considered when determining a match.
 - All Fields** - Marks in the target sheet are highlighted when they match the marks selected in the source sheet. All fields are considered when determining a match.



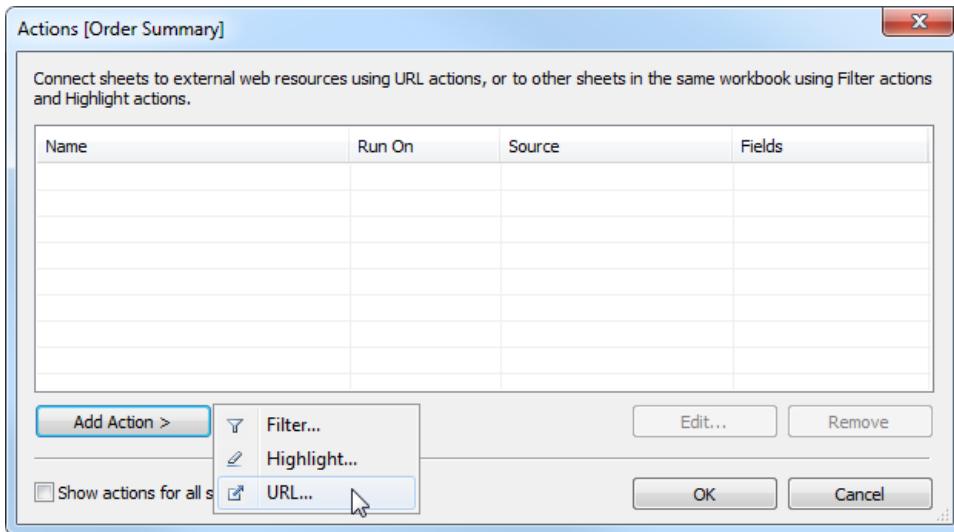
- When finished, click **OK** twice to close the dialog boxes and return to the view.

URL Actions

A URL action is a hyperlink that points to a web page, file, or other web-based resource outside of Tableau. You can use URL actions to create an email or to link to more information about your data that may be hosted outside of your data source. To make the link relevant to your data, you can substitute field values of a selection into the URL as parameters.

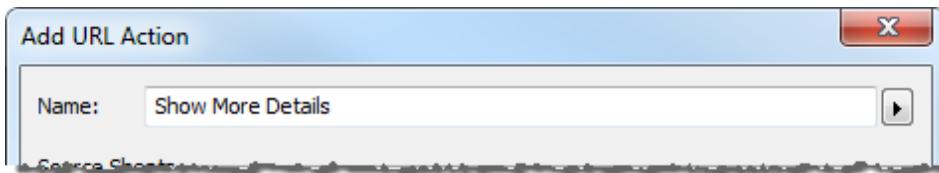
Make a URL action that opens a web page

1. On a worksheet, select **Worksheet > Actions**. From a dashboard, select **Dashboard > Actions**.
2. In the Actions dialog box, click **Add Action** and then select **URL**.

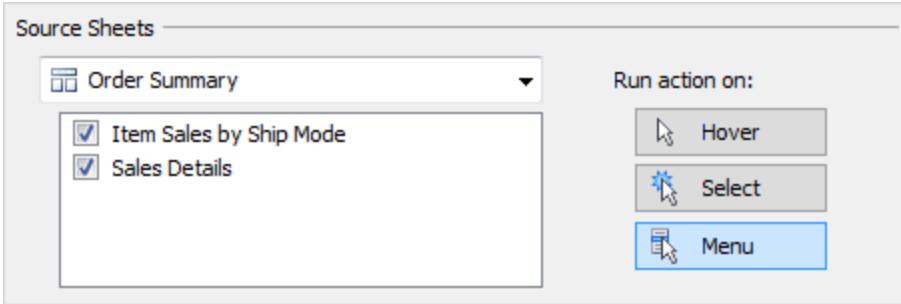


3. In the next dialog box, specify a name for the link.

Make the name descriptive of the action. If you choose to run the action using the menu, the name you specify here becomes the option that shows on the menu. For example, when linking to more product details, the name could be "Show More Details." You can use variables in the name, which will be filled in based on the values of the selected field.



4. Use the drop-down list to select a source sheet or data source. If you select a data source or dashboard you can select individual sheets within it (see [Filtering Across Multiple Data Sources Using a Parameter](#) in the Tableau Knowledge Base for more information about filtering data sources).



5. Select how users will run the action.

If you choose this The action is run when the user... option...

Hover	Mouses over a mark in the view. This option works well for highlight and filter actions within a dashboard.
Select	Clicks a mark in the view. This option works well for all types of actions.
Menu	Right-clicks (control-clicks on Mac) a selected mark in the view, then selects an option on the context menu. This option works well for filter and URL actions.

6. Specify the URL. You can use any URL that your browser can recognize including web pages, FTP resources, and files.

Just as you can use variables in the name of the URL, you can also use field values and filter values as parameters in the URL. That means that you can send information about each selected mark or filter setting to a given website.

In the URL you specify, include the appropriate prefix to ensure that the resulting hyperlink directs to the correct location. For example, if your URL links to a web page, include the `http://` prefix. If you publish workbooks to a Tableau Server, and your URL includes a protocol other than `http`, `https`, `gopher`, `news`, `ftp`, or `mailto`, you will need a server administrator to add the protocol to the server's whitelist. For more information, see [tabadmin set options](#) in the Tableau Server help.

When using a URL action to point to an external file (rather than a web page), use the universal naming convention (UNC) path for the URL action. A UNC path is a full path of a resource or file that is stored on the computer. A UNC path includes the computer name, drive letter, path to the file, and file name. For example, for a file `D:\myfile.txt`, specify the

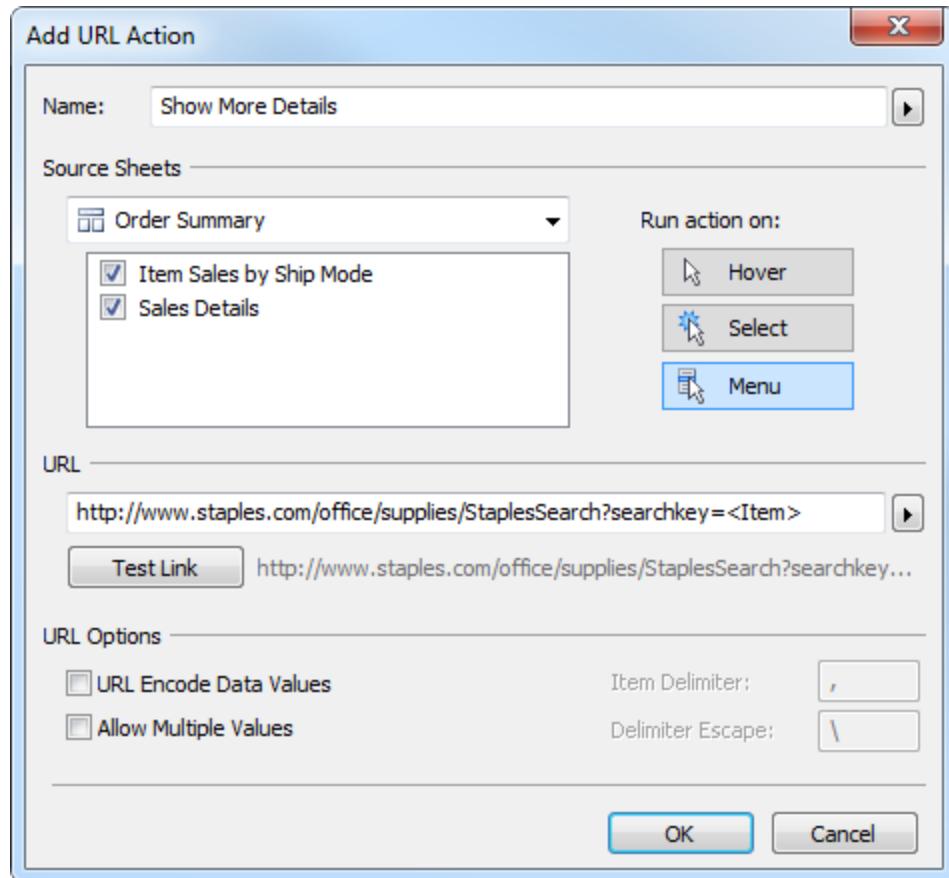
following UNC path: \\workstation1\d\$\myfile.txt, where "workstation1" is the computer name in your company domain.

7. (Optional) Select any of the following options:

- **URL Encode Data Values** - select this option if your data contains values that use characters that are not allowable in a URL. For example if one of your data values contains an ampersand, such as "Sales & Finance," the ampersand must be translated into characters that your browser understands (URL encoded) if you want to include that value in the URL.
- **Allow Multiple Values** - select this option if you are linking to a webpage that can take lists of values as parameters in the link. For example, say you select several products in a view and you want to see each product's details hosted on a webpage. If the server can load multiple product details based on a list of identifiers (product ID or product name), you could use multi-select to send the list of identifiers as parameters.

When you allow multiple values, you must also define the item delimiter, which is the character that separates each item in the list (for example, a comma). You must also define the Delimiter Escape, which is used if the delimiter character is

used in a data value.

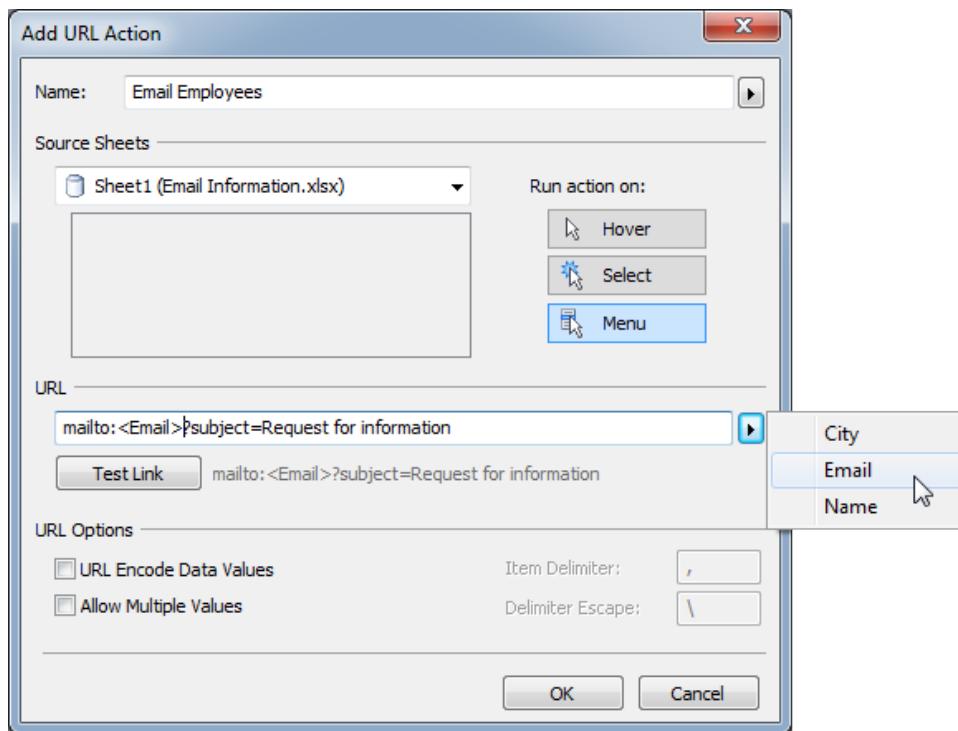


8. Click **OK** twice to close the dialog boxes and return to the view.

URL actions can also point to a web page object in a dashboard. Refer to [Actions and Dashboards on page 1139](#) to learn more about how actions work with dashboards.

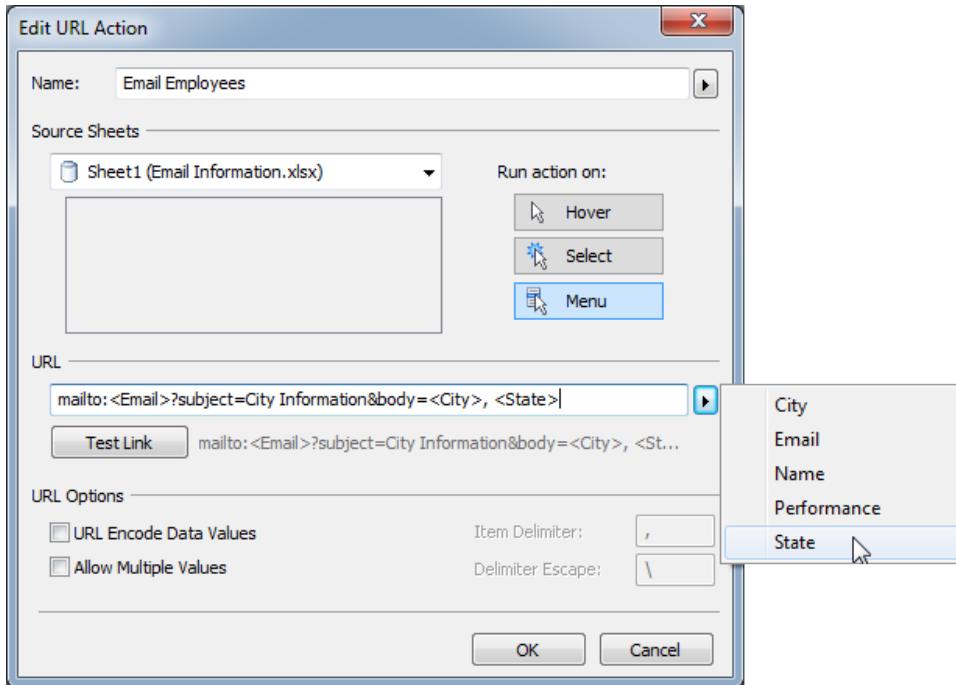
Make a URL action that creates an email

1. Select **Worksheet > Actions**.
2. In the Actions dialog box, click the **Add Action** button and select **URL**.
3. In the Source Sheets drop-down list, select the sheet that contains the field with the email addresses you want to send to.
4. In the URL textbox, type **mailto:**, click the arrow button to select the field in your data that contains the email addresses. Type **?subject=** and enter text for the Subject line if you want to. For example, in dialog below, the subject is Request for information.



5. Type **&body=**, and click the arrow button to select the fields of information that you want to include in the body of the email.

In the example below, the “Email” field contains the email addresses, the subject is “City Information”, and the body text of the email consists of the city and state information that is associated with the email address.



6. (Optional) Display data from your workbook in the body of your email as a vertical list instead of the default horizontal list. For example, suppose you have a horizontal list of cities, such as Chicago, Paris, Barcelona, which you would rather display vertically, like this:

Chicago
Paris
Barcelona

To make the list vertical, in the Edit URL Action dialog box, do the following:

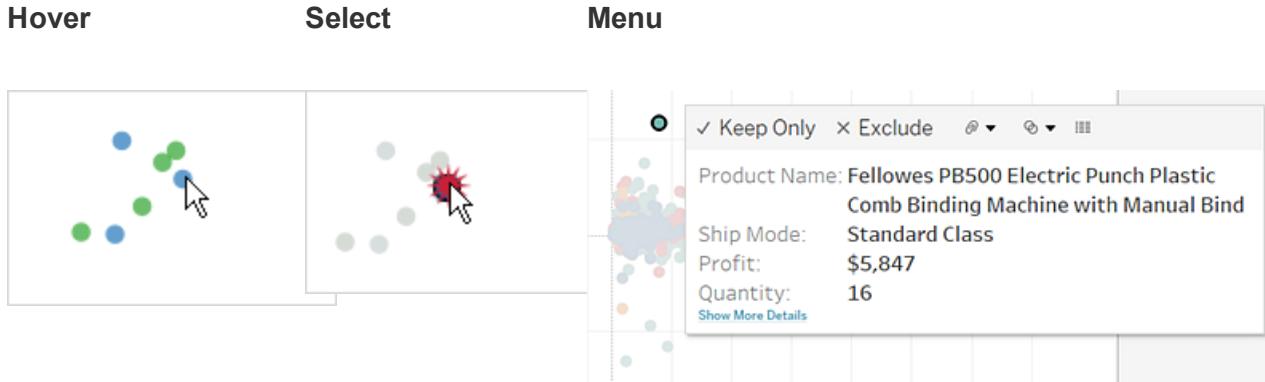
- Under URL Options, verify that the **URL Encode Data Values** check box is clear.
- Under URL Options, select the **Allow multiple values** check box.
- Type **%0a** in the **Item Delimiter** text box to add a line break. This set of characters is the URL-encoded character for a line break.

Running Actions

Depending on how the action is created, you can run an action using one of the following three methods:

- Hover** - rest the pointer over a mark in the view to run the action. This option works well for highlight and filter actions within a dashboard.
- Select** - click on a mark in the view to run the action. This option works well for all types of actions.

- **Menu** - right-click (control-click on Mac) a selected mark in the view and then select an option on the context menu. This option works well for filter and URL actions.



Links are not always visible for every worksheet and mark. Because links are mapped to specific fields in the data source, links will only be available for the worksheets that use the mapped fields. For example, if you add a hyperlink that uses both Latitude and Longitude as parameters in the link, the link will only be available to worksheets that use Latitude and Longitude in the view. Additionally, the link is only available on marks and headers that contain relevant values.

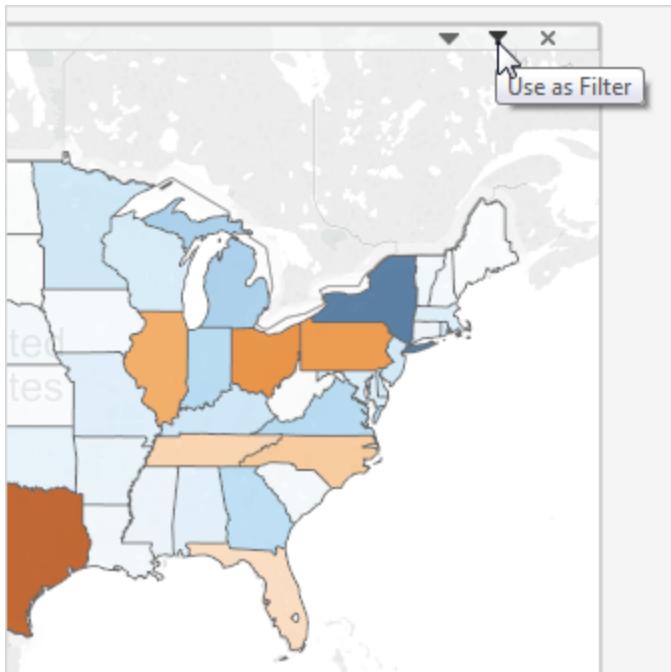
Actions and Dashboards

Actions often have unique behavior when the source or destination is a dashboard. This is because a dashboard can contain multiple views—so a single filter or highlight action can have broad impact. A dashboard is also unique in that it can contain web page objects. Web page objects are associated with URL actions, which you can use to embed a web page in a dashboard.

Use a Single View to Filter Other Views in a Dashboard

Imagine you have a dashboard that contains three views about profitability: a map, a bar chart, and a table of customer names. You can use a filter action to make one of the views in your dashboard, such as the map, the "master." When your users select a region in the map, the data in the other views is filtered so that it relates to just that region.

1. On the dashboard, select the view you want to use as a filter.
2. On the view's shortcut menu, choose **Use as Filter**. You can perform the same action by clicking the Use as Filter icon .



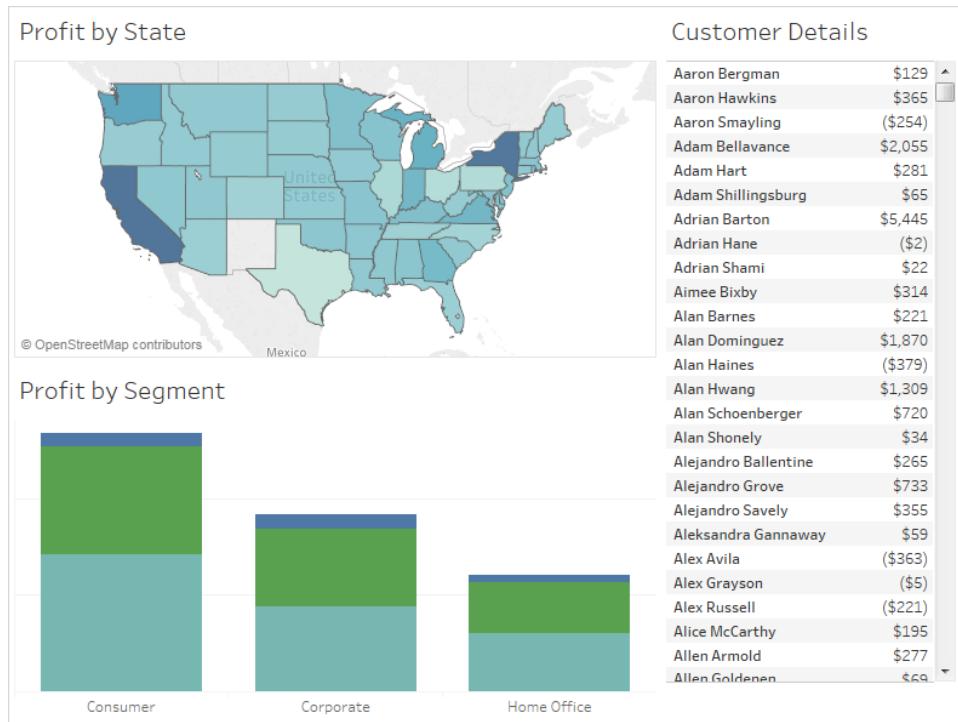
You can also use filter actions to filter the data on a dashboard when the data comes from multiple data sources. For more information, see [Filtering Across Multiple Data Sources Using a Parameter](#) in the Tableau Knowledge Base.

For more information about filter actions, see [Filter Actions](#) on page 1116.

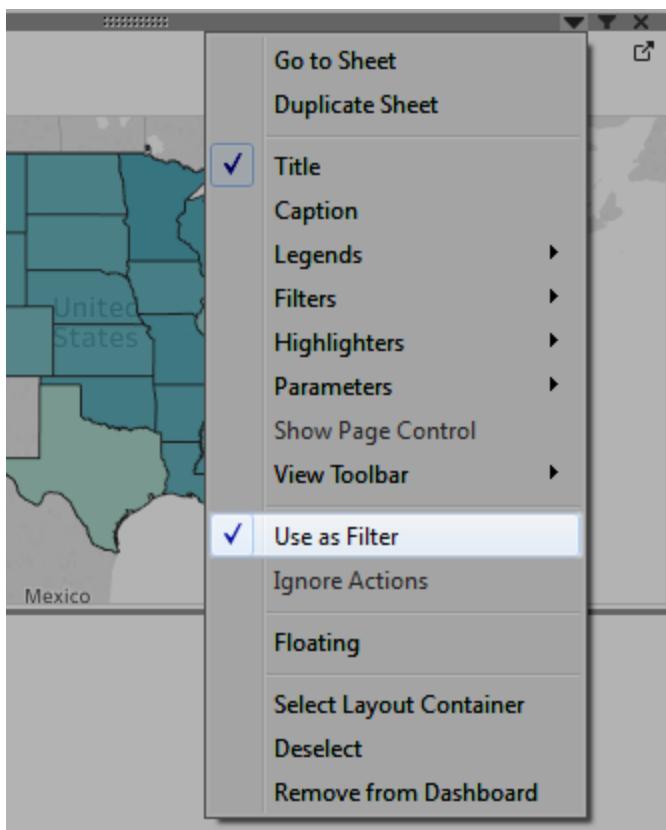
Use Multiple Views to Filter Other Views in a Dashboard

Similar to how you can use a single view to filter other views in a dashboard, you can also use multiple views as a filter. This is sometimes called having a "multi-master" dashboard. The trick is to not only use the master views as filters, but to also disable their ability to be filtered themselves.

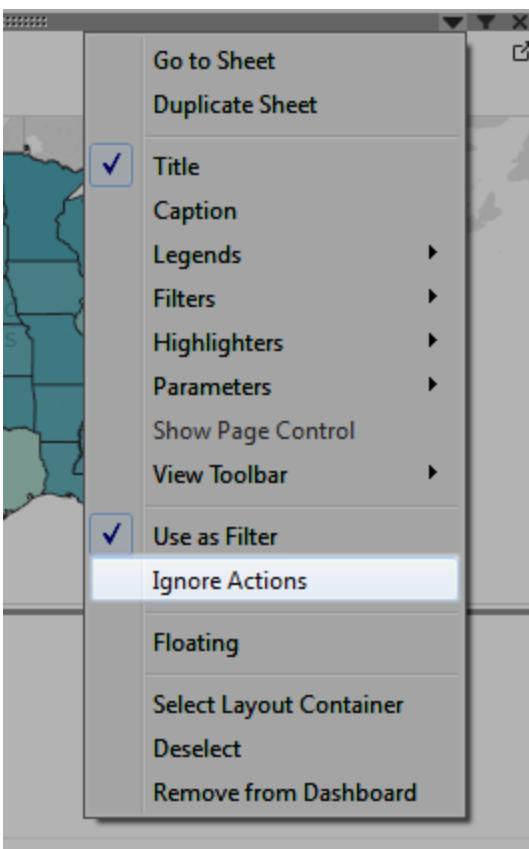
1. Create or open a dashboard that has at least three views.



2. Select the first view that you want to use as a filter (such as a map), and from its shortcut menu, select **Use as Filter**.

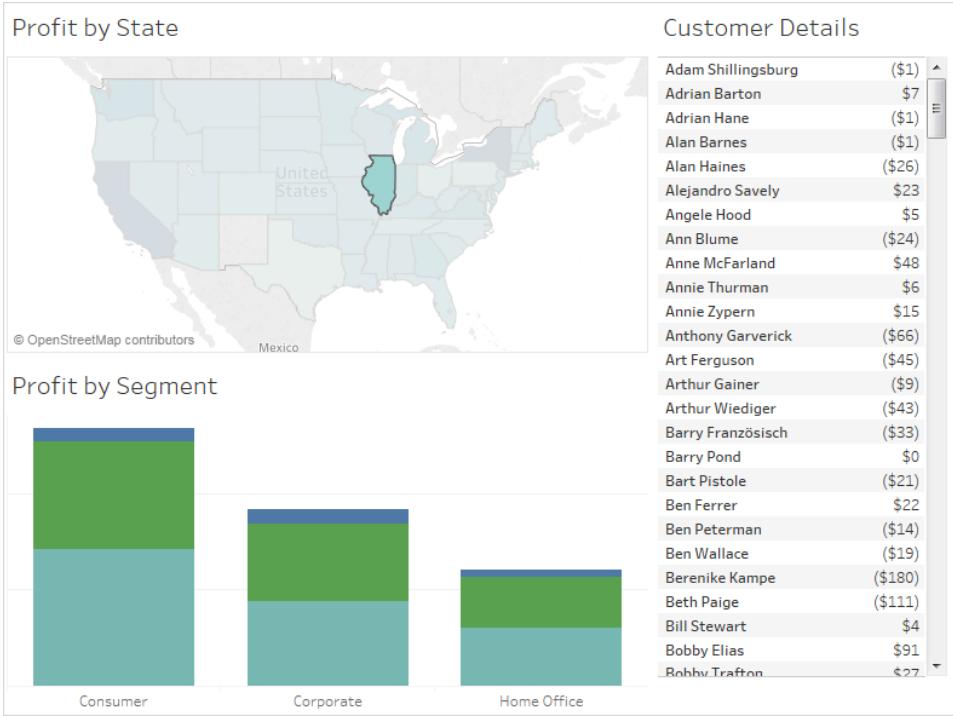


3. Open the same view's shortcut menu again and select **Ignore Actions**. This ensures that other filter actions, including the one you'll create next, will not affect this view.



4. Repeat the steps 2 and 3 for any other views you want to use as a filter.

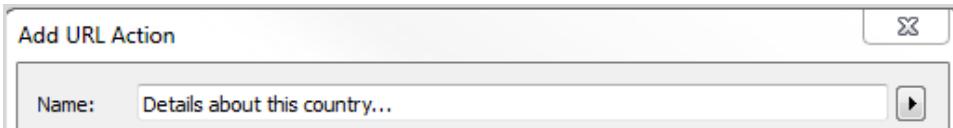
Now, selecting marks in a master view filters data in one or more detail views—all without affecting any other masters.



Use a URL Action to Display a Web Page Inside a Dashboard

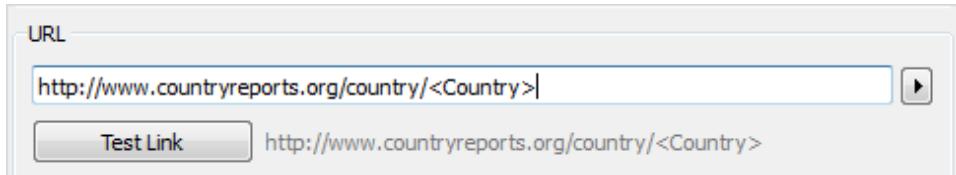
You can use a URL action with a web page object to display information from the web inside your dashboard. For example, you might have a dashboard that shows profits by country. In addition to showing the profit data in your dashboard, you also want to display supplemental information about the countries, and that information is on a web site. Here's how to do it.

1. From your dashboard, select **Dashboard > Actions**.
2. In the Actions dialog box, click **Add Action** and then select **URL**.
3. In the next dialog box, specify a name for the link. If you choose to run the action using a menu, such as a menu option on a tooltip, the name you specify here is what's displayed.



4. Under Source Sheets, select the view or data source that will initiate the action. For example, if you want the action to be initiated when a user clicks a link on a map's tooltip, select the map view.
5. Select how the people viewing your dashboard will run the action, such as **Menu**.
6. Enter the URL, starting with the `http://` or `https://` prefix, such as

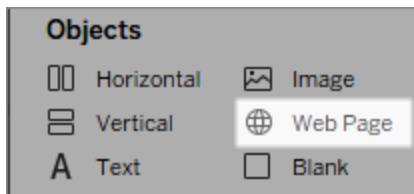
`http://www.example.com`. You can use field values as parameters in your URL. For example, if Country is a field used by a view in your dashboard, you can use `<Country>` as a parameter in your URL (see [URL Actions](#) on page 1132 for more details on protocols and parameters you can use).



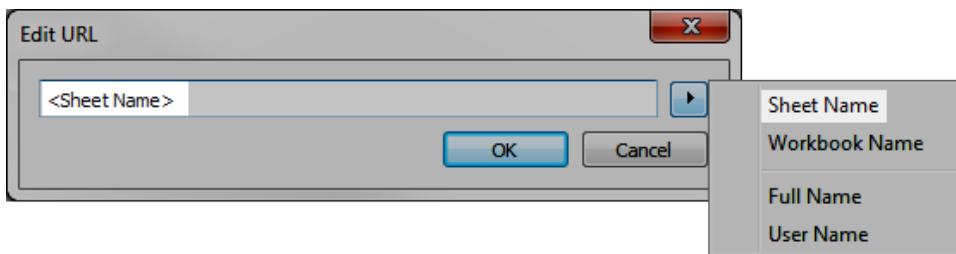
7. Click OK.

At this point, if you use this action (for example, by clicking a link on a tooltip), a web browser window will open and load the page you specified in step 6.

8. To make the web page load within your dashboard, drag a **Web Page** object onto your dashboard.



9. In the Edit URL dialog, select **Sheet Name**. This tells the web page object to receive its URL from the view that you specified in step 4.



10. Click **OK**.

Now when you launch the action, a web page automatically loads within the dashboard rather than opening a separate browser window.



Using Field and Filter Values in Actions

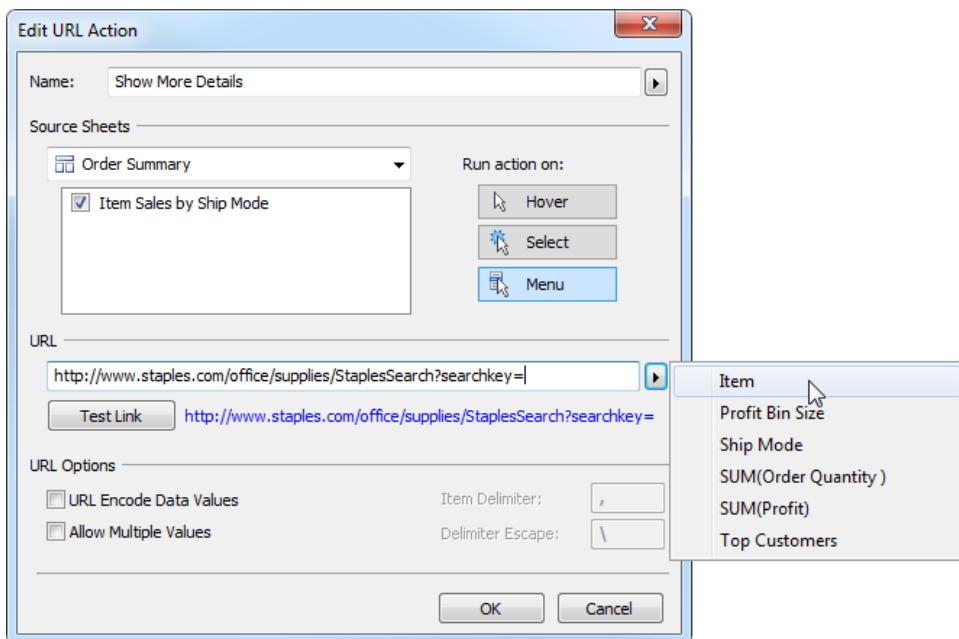
When you add an action in Tableau you often want to use values from your data as parameters in the name of the action as well as the action itself. Using fields as variables in the action name makes the menu item that launches the action specific to the selected mark. More commonly, using field and filter values as parameters in the URL of a URL action allows you to send information about a specific data point or filter setting to the destination webpage.

Using Field and Filter Values in URLs

Tableau lets you add field, filter information, and parameter values as variables into URL actions so when you follow the link the values of those fields and filters are included. For example, when linking to an online mapping service, you can insert the address field into the

URL so launching the link from a specific data point shows the address associated with that record on a map.

1. In the Add URL Action dialog box, begin typing the URL for the link.
2. Place the cursor where you want to insert a field, parameter, or filter value.
3. Click the arrow to the right of the text box and select the field, parameter, or filter you want to add to the URL. The field, parameter, or filter name is added to the URL between angle brackets. You can continue adding field and filter values parameters as many times as you need to create the URL.



Note: The list of available fields only includes non-aggregated fields. To use aggregated field values as a parameter in a link, you must first create a calculated field and then use the name of that field in the link. The calculated field must also be used in the view in order for the link to be available. A good way to use these fields is by placing them on the Level of Detail shelf.

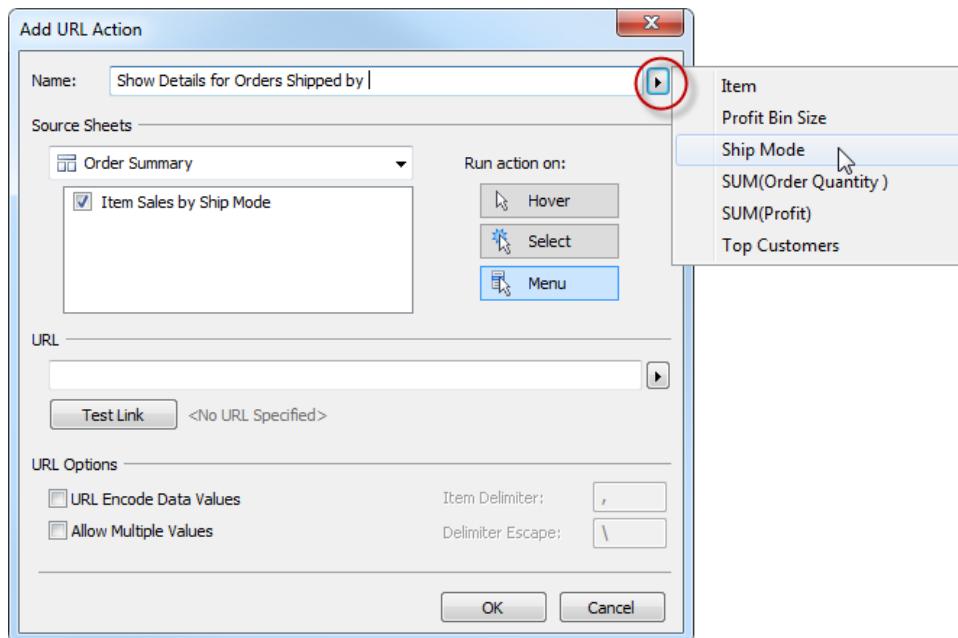
When inserting parameter values, the Display As value is what is sent with the URL action. To send the actual value instead, you should add the character ~na after the parameter name. For example, you may have a parameter that includes IP addresses. The Actual Value strings for the parameter include values such as 10.1.1.195 while the Display As strings include more friendly values such as Computer A (10.1.1.195). Inserting the parameter into the URL like this:
http://<IPAddress>/page.htm will send the display values so the final

destination URL would be `http://Computer A (10.1.1.195)/page.htm`. Obviously, that is not likely to create a valid URL. To send the actual value, the parameter in the URL should look like this: `http://<IPAddress~na>/page.htm`.

Using Field and Filter Values in Action Names

In addition to using field, parameter, and filter values in URLs, you can use this information as variables in the action names. The name of the action displays on the context menu when an action is launched using the menu. Using field and filter variables in the name is useful in making the action specific to the selected mark. In a view showing real estate information, you could name a URL action that points at satellite images from an online mapping service, "Show satellite image of <Address>." When you right-click (control-click on Mac) a specific mark, the <Address> tag is replaced with the location value associated with that mark.

1. In the Add Action dialog box, begin typing the name for the action.
2. Place the cursor where you want to insert the field or filter value.
3. Click the arrow to the right of the text box and select the field or filter you want to add as a variable. The field or filter name is added between angle brackets.



Dashboards

A dashboard is a collection of several worksheets and supporting information shown in a single place so you can compare and monitor a variety of data simultaneously. For example, you may have a set of views that you review every day. Rather than flipping through each worksheet, you can create a dashboard that displays all the views at once.

Similar to worksheets, dashboards are shown as tabs at the bottom of the workbook and update with the most recent data from the data source. When you create a dashboard, you can add views from any worksheet in the workbook. Each view you add to the dashboard is connected to its corresponding worksheet. That means when you modify the worksheet, the dashboard is updated and when you modify the view in the dashboard, the worksheet is updated.

Best Practices for Effective Dashboards

A well-designed dashboard can align your organization's efforts, help uncover key insights, and speed up decision-making. Use this topic for tips on best practices for creating effective dashboards in Tableau.

What's your goal?

- [Know your purpose and audience](#) below
- [Leverage the most-viewed spot](#) on the next page

Design for the real world

- [Author at your final display size](#) on the next page
- [Limit the number of views](#) on page 1151
- [Be security-savvy](#) on page 1152

Add interactivity to encourage exploration

- [Show filters](#) on page 1153
- [Enable highlighting](#) on page 1153

Save time

- [Use the Go to Sheet command](#) on page 1155
- [Hide sheets as you go](#) on page 1156

What's your goal?

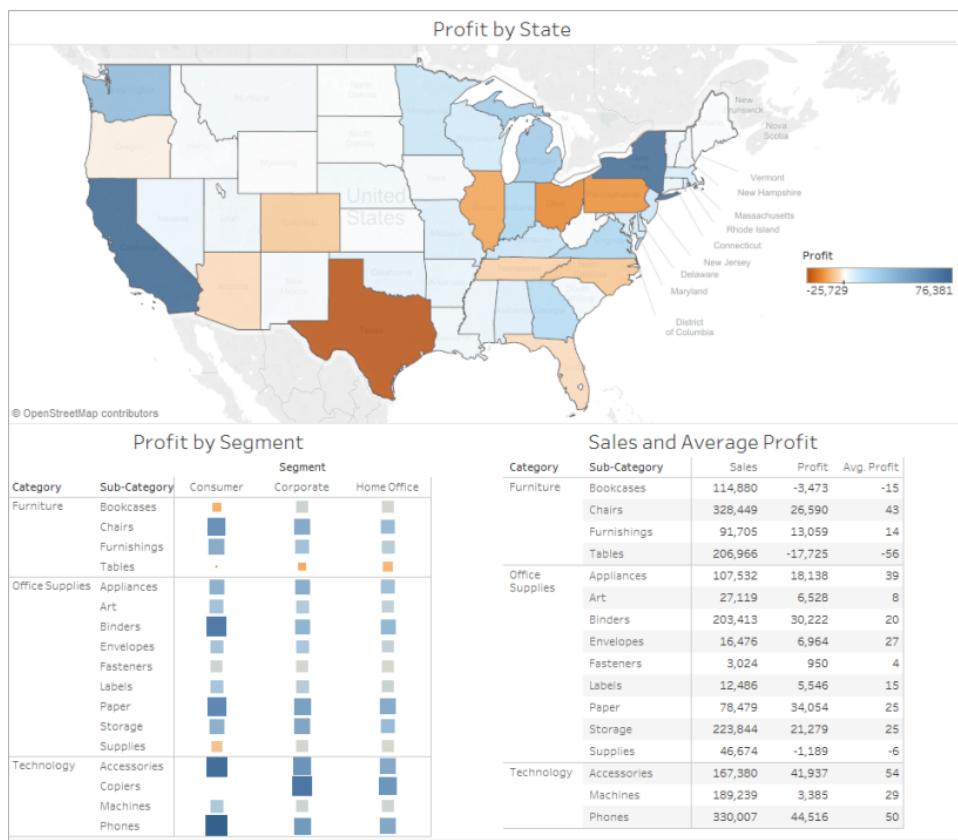
Know your purpose and audience

The best visualizations have a clear purpose and work for their intended audience. What will you be trying to say with this dashboard? Are you presenting a conclusion or a key question?

In addition to knowing what you're trying to say, it's important to know who you're saying it to. Does your audience know this subject matter extremely well or will it be new to them? What kind of cues will they need? Thinking about these questions before you head into the design phase can help you create a successful dashboard.

Leverage the most-viewed spot

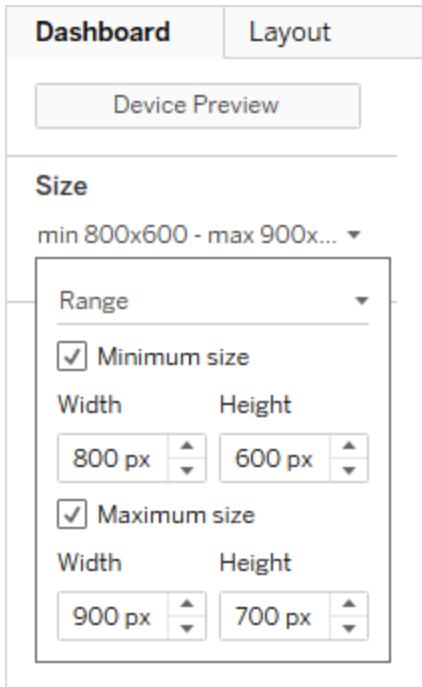
Most viewers scan web content starting at the top left of a web page. Once you know your dashboard's main purpose, be sure to place your most important view so that it occupies or spans the upper-left corner of your dashboard. In the dashboard below, the map view holds the key message.



Design for the real world

Author at your final display size

By default, Tableau dashboards are set to use a fixed size and if you keep this setting, be sure to construct your visualization at the size it will be viewed at. You can also set **Size** to **Automatic**, which makes Tableau automatically adapt the overall dimensions of a visualization based on screen size. This means that if you design a dashboard at 1300 x 700 pixels, Tableau will resize it for smaller displays—and sometimes this results in scrunched views or scrollbars. The **Range** sizing feature is helpful for avoiding this.

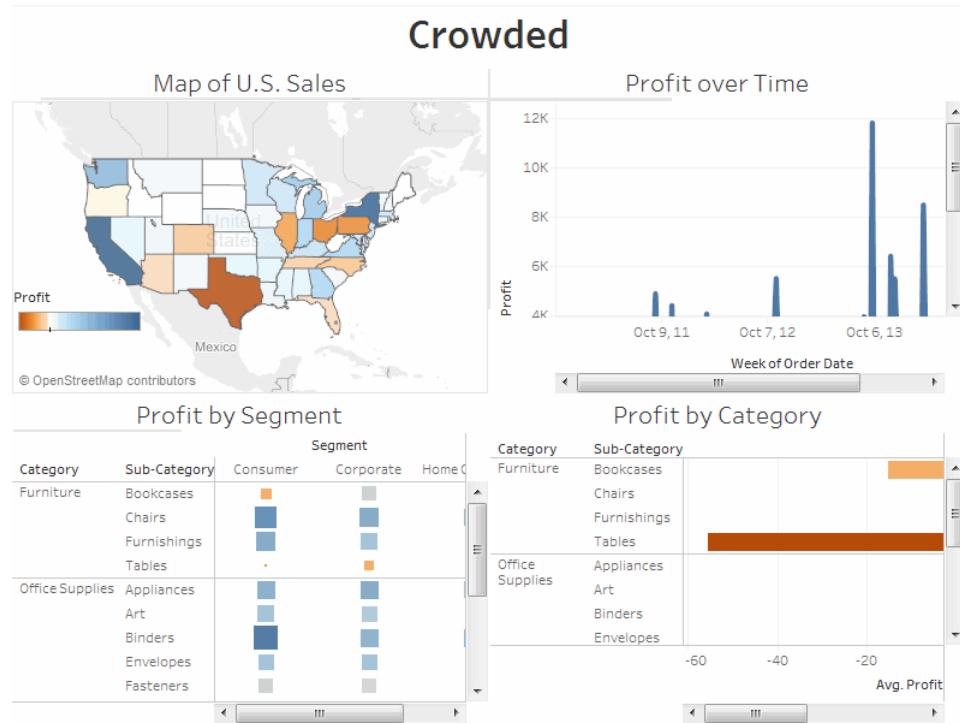


You can also design for specific device layouts, so that on tablets, for example, your dashboard contains one set of views and objects, and on phones it displays another. See [Create Dashboard Device Layouts](#) on page 1160 for steps.

Limit the number of views

In general, it's a good idea to limit the number of views you include in your dashboard to two or three. If you add too many views, visual clarity and the big picture can get lost in the details. If you find that the scope of your story needs to grow beyond two or three views, you can always create more dashboards.

Too many views can also interfere with the performance of your dashboard after it's published. See [Make Visualizations Faster](#) on page 1827 for more details on performance.



Click the image to replay it.

Be security-savvy

A dashboard can include more than just views, it can include objects—including embedded web pages. If you include web page objects in your dashboard, here are some tips to keep in mind:

- **HTTPS:** When you add a web page object to your dashboard, you're prompted to specify a URL. It's a best practice to use HTTPS (<https://>) in your URL. This ensures that the connection from your dashboard to the web page is encrypted. Also, if Tableau Server is running HTTPS and you use HTTP in the URL, your users' browsers won't be able to display the web page that the URL points to.
- **Plug-ins:** If the web page object requires a plug-in, such as Adobe Flash, the plug-in must be installed on the computer that's running Tableau. The Tableau component that handles plug-ins requires that Safari be installed as well.
- **Web view security options:** You can control whether the web page object has JavaScript and plug-ins enabled, among other things. To adjust them, select **Help > Settings and Performance > Dashboard Web View Security**, and then clear one or more of the options listed below.:
 - **Enable JavaScript.** Allows JavaScript support in the web view. Clearing this option may cause some web pages that require JavaScript to function improperly in the dashboard.
 - **Enable Plug-ins.** Enables any plug-ins the web page uses, such as an Adobe Flash or QuickTime player.
 - **Block Pop-ups.** When selected, blocks pop-ups.

- **Enable URL Hover Actions.** Allows URL hover actions. For more information, see [URL Actions](#) on page 1132.

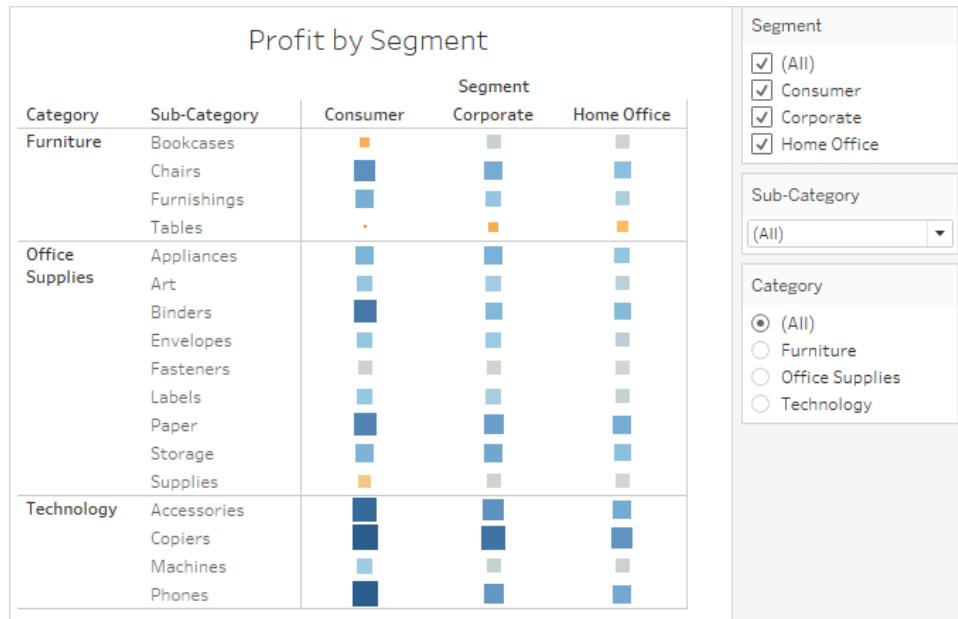
Note: Any changes you make to the security options apply to all web page objects in your workbook, including new web page objects you create, and all subsequent workbooks you open in Tableau Desktop. To see your changes, you may need to save and reopen the workbook.

Add interactivity to encourage exploration

Show filters

Filters help users specify which data is shown in the view. To turn on filters for a field, right-click the field in the Data window and select **Show Filter**.

You can customize each filter for different types of data. For example, you can show filters as multi-select check boxes, single select radio buttons, or drop-down lists, etc. You can include a search button, the option to show all fields, null controls, and more. You can also edit the title of a filter to give your viewers clear instructions for interacting with the data.



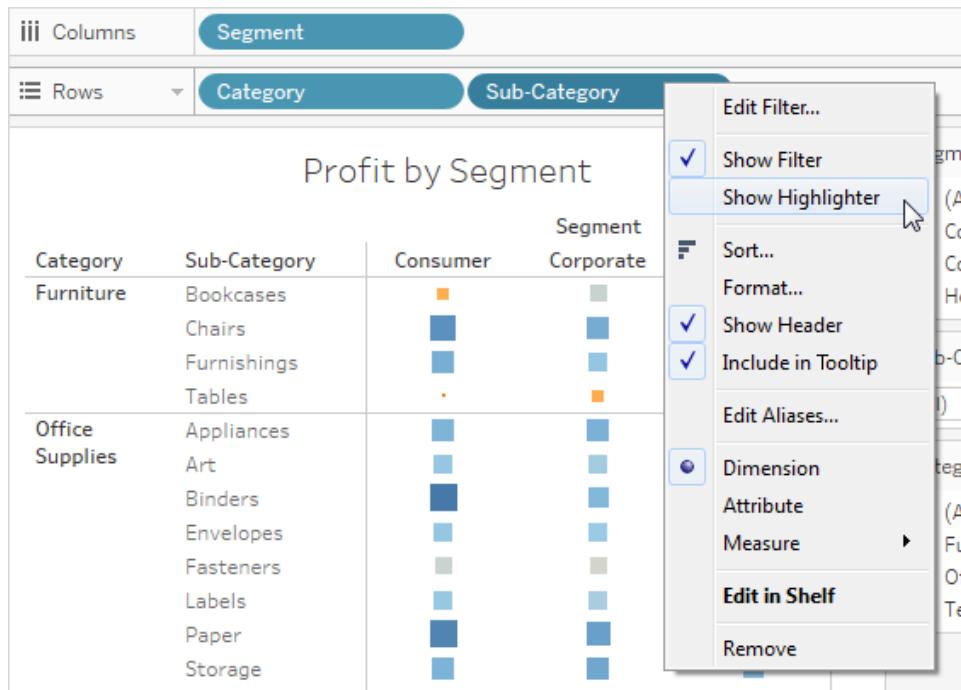
Enable highlighting

You can use the **Highlight** button on the toolbar to set up highlighting between views. When highlighting is turned on, a selection in one view will highlight related data in the other views. You can turn on highlighting for all fields or select specific fields. For more information about the different methods you can use to highlight data, see [Highlight Actions](#).

You can also display a highlighter that allows your customers to highlight parts of a view based on what they enter or select.

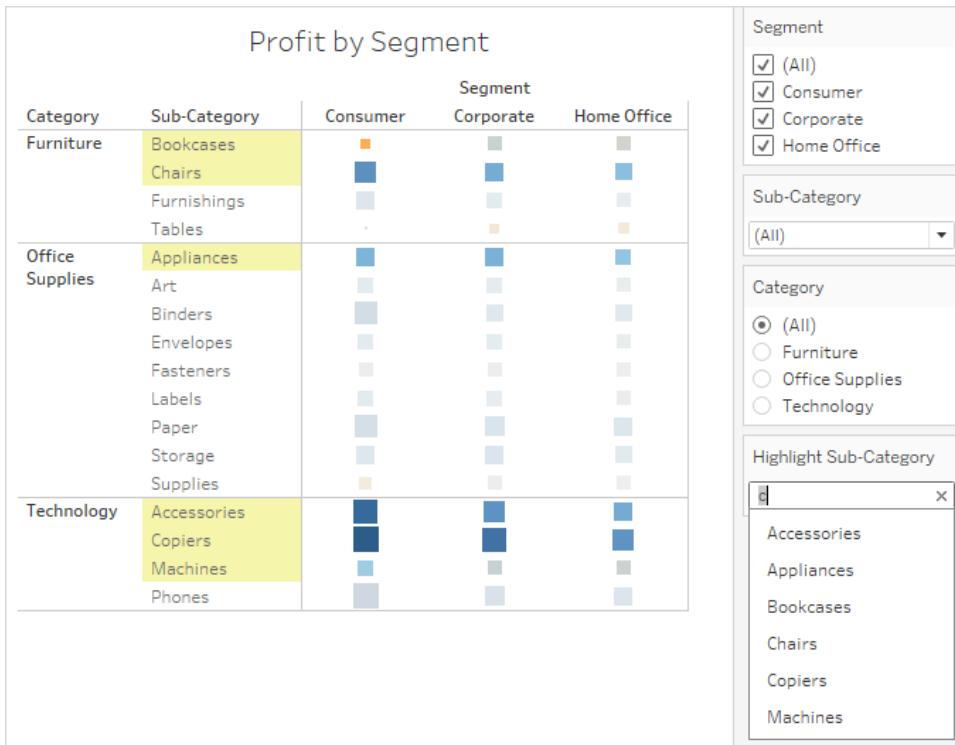
To display a highlighter:

1. Go to the worksheet where the view is (or select **Go to Sheet** from the dashboard).
2. Right-click the field you want to highlight and choose **Show Highlighter**:



In the highlighter, your users will be able to select or enter terms to highlight data in the

view:



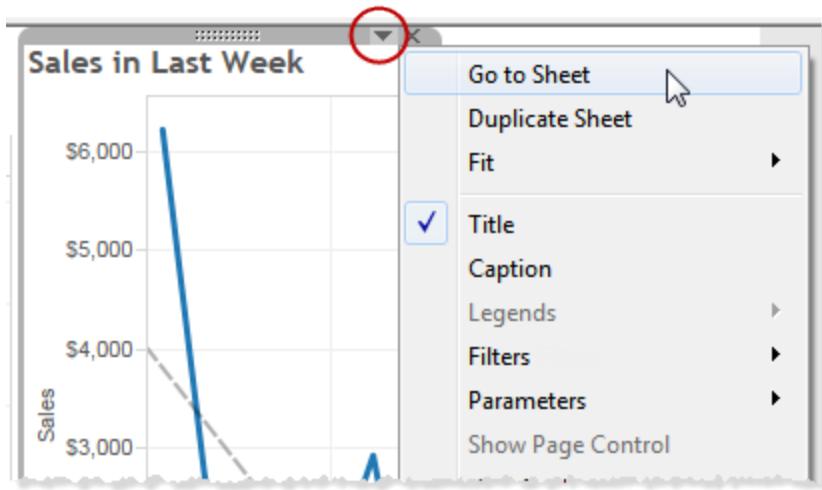
Save time

Use the Go to Sheet command

The views in a dashboard are connected to the worksheets they represent. Tableau provides short-cut menus to help you quickly accomplish basic tasks, like jumping from the dashboard you're working on to the original view, where you can perform other actions.

To jump to a sheet:

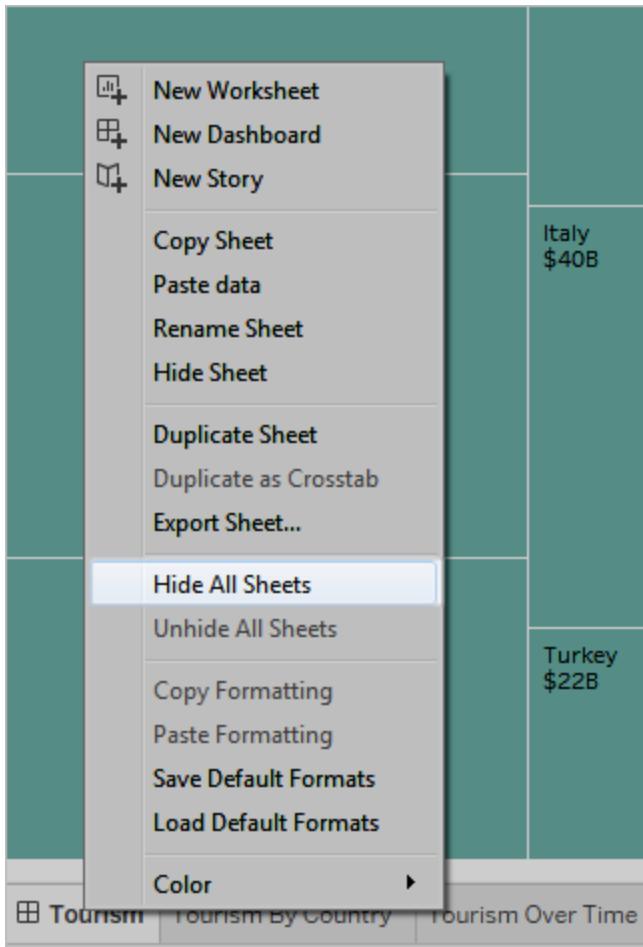
1. In your dashboard, select the view you want to go to.
2. Select **Go to Sheet**.



Hide sheets as you go

When you're working with a large number of worksheets to build dashboards, as you finish off a dashboard, you can hide the sheets that went into the dashboard. This can make your workbook easier to navigate.

To hide a dashboard's sheets, right-click (control-click on Mac) the dashboard's tab at the bottom of the workbook and select **Hide All Sheets**.



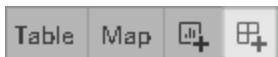
Create a Dashboard

After you've created one or more views, you can pull them into a dashboard, add interactivity, and much more. This topic walks you through the steps.

Open a dashboard sheet

You create a dashboard in much the same way you create a new worksheet. After you create a dashboard you can add views and objects.

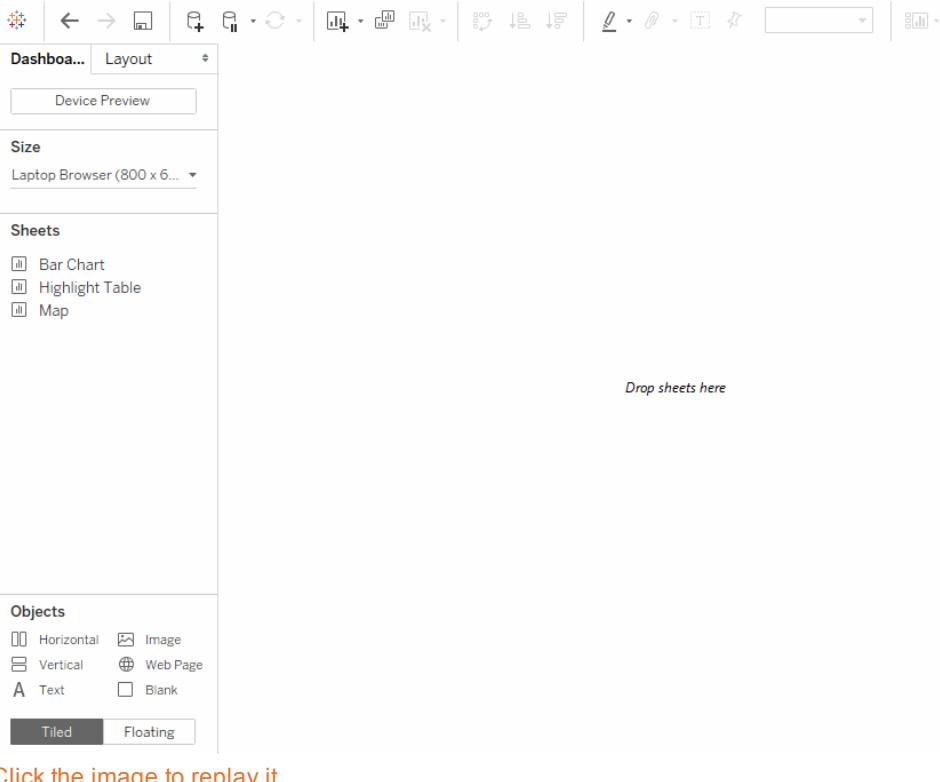
To open a new dashboard sheet and start creating a dashboard, click the **New Dashboard** icon at the bottom of the workbook:



The **Dashboard** tab appears on the left and lists the sheets in your workbook.

Add views

After you have a dashboard sheet, click the views you built (listed under **Sheets** on the left) and drag them to your dashboard sheet on the right. A gray, shaded area indicates where you can drop views:



Click the image to replay it.

Add objects

In addition to adding views to your dashboard, you can add objects, including web pages, images, text, blank space, and layout containers.

Layout containers are helpful for fine-tuning how your dashboard resizes itself when users interact with it. See [Refine Your Dashboard on page 1172](#) for tips on when and how to use layout containers.

To add an object:

Select an item under **Objects** on the left and drag it to the dashboard sheet on the right:

The screenshot shows a dashboard with a sidebar on the left containing object selection tools. The sidebar includes options for Horizontal, Vertical, Text, and Blank objects, with 'Tiled' selected. A 'Show dashboard title' checkbox is also present. To the right is a table with three rows:

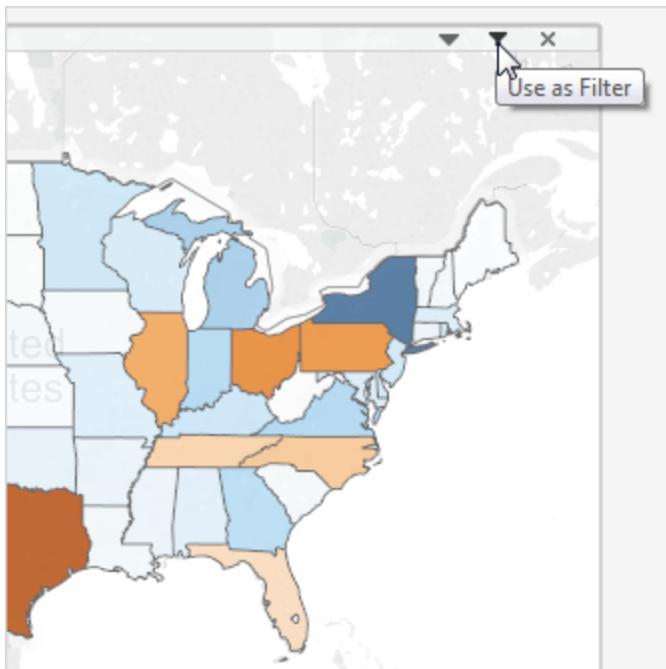
Copiers	\$24,083.71	\$18,990.28
Machines	\$2,141.06	\$703.02
Phones	\$23,837.11	\$11,766.22

Depending on the object you're adding, you may be prompted for additional information. If you are adding a Web Page object, see [Best Practices for Effective Dashboards](#) on page 1149 for tips on web view security options.

Add interactivity

Part of the power of dashboards is that you can set up filters and interactivity to associate the different views on your sheet and enhance your users' analysis. The people looking at your dashboard can stay in a single spot—they don't need to click outside the world you've created for them.

In a dashboard you can use the **Use as Filter** option to make one of the views act as a filter on all the other views in the dashboard. To do this, select the filter button.



You can also select the drop-down arrow and choose **Use as Filter**.

There are many other actions you can add to your dashboard as well. For details, see [Actions and Dashboards](#) on page 1139 and [Using Field and Filter Values in Actions](#) on page 1146.

Create Dashboard Device Layouts

After you've built a dashboard you can create layouts for it that are specific to particular devices.

You can tailor a dashboard's composition and contents according to different browser window sizes. When you publish the dashboard to Tableau Server or Tableau Online, people who interact with it experience a dashboard expressly designed for their screen display, whether it's a phone, tablet, or desktop. As the author, you only have to create and maintain a single dashboard, and deliver a single URL.

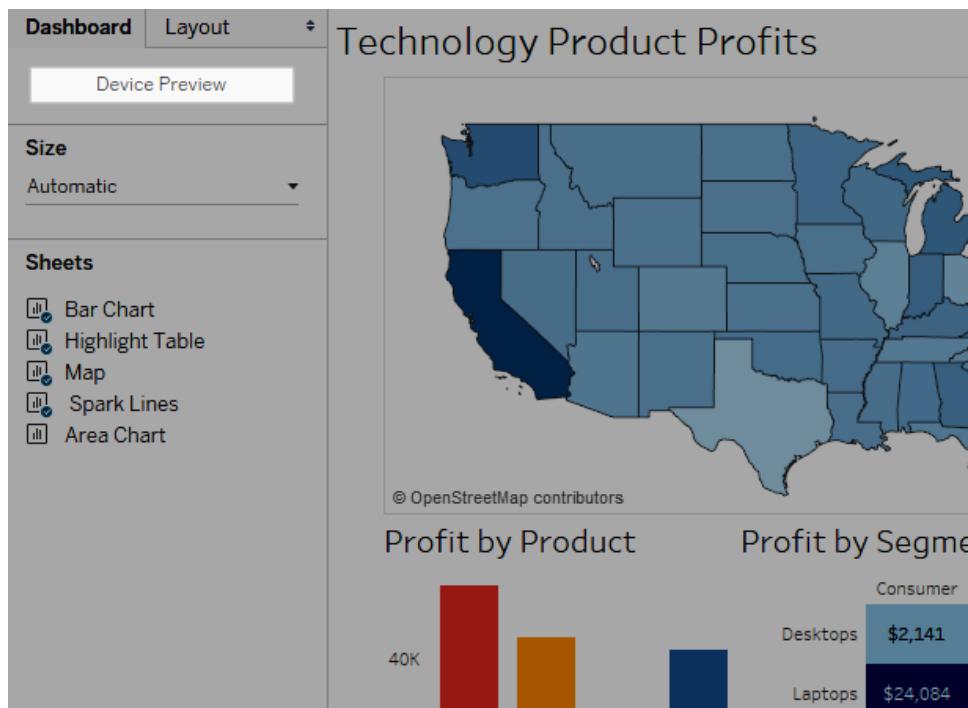
Preview and add a device layout

Creating a device layout starts with a preview phase.

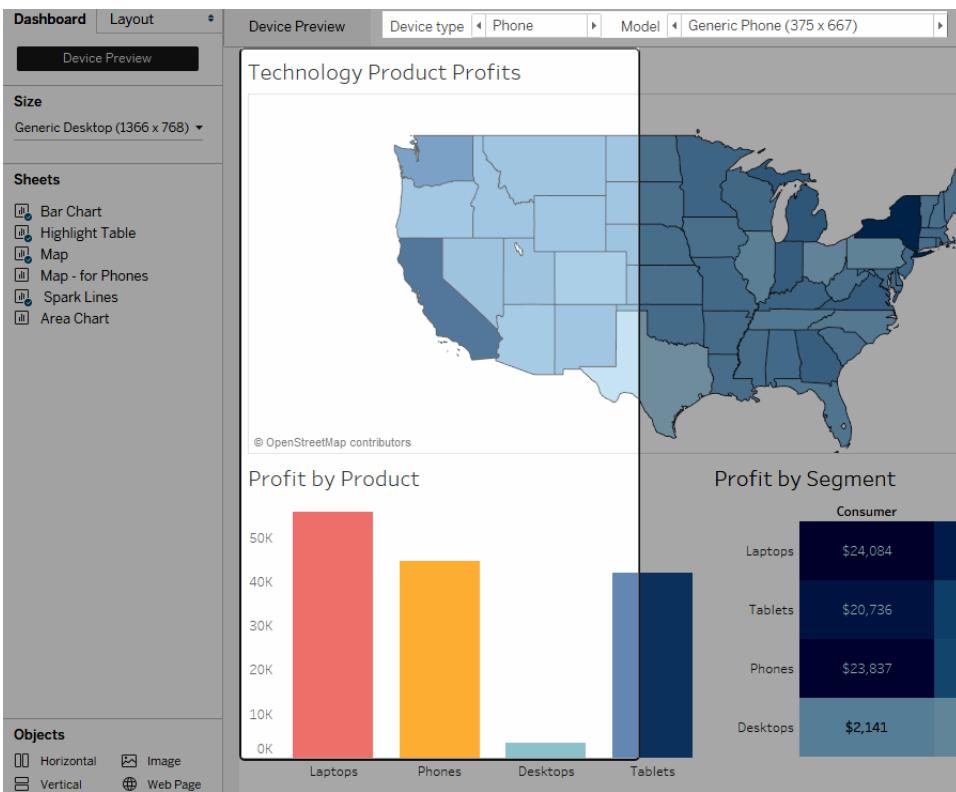
1. Open a dashboard.



2. On the **Dashboard** tab on the left, click **Device Preview**.



When a dashboard is in device preview mode, a preview bar appears above the dashboard and a black frame shows you the screen size options. The preview bar displays the **Device type** and **Model**.

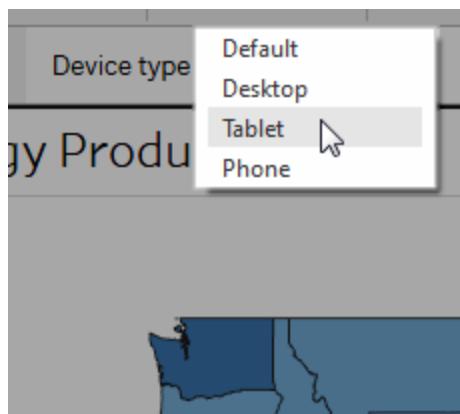


- Take a moment to click through the **Device types** and **Models** and explore the different screen sizes. Setting the **Size** option to **Fit all** is the best way to see the resize behavior. **Default** shows you the dimensions and layout of the dashboard you started with.

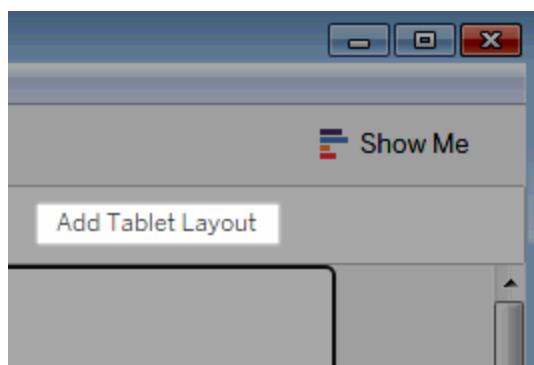
To see how the dashboard will look in landscape vs. portrait mode, click . Usually, landscape is optimal for tablets and portrait is best for phones.

Select the **Tableau Mobile app** check box to see the screen size you have to work with if the dashboard is viewed on a device where Tableau Mobile is installed. This option is available for some, but not all device models.

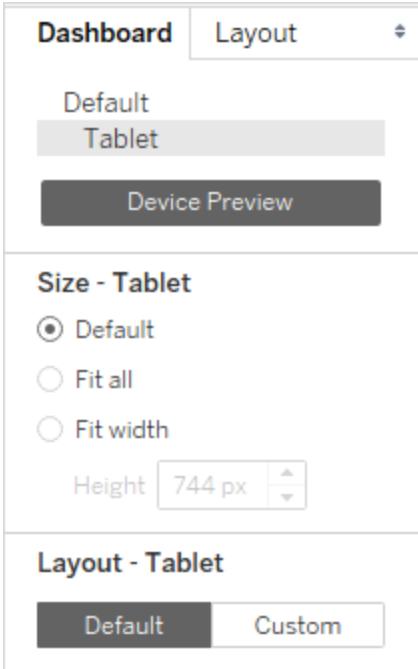
- Choose a **Device type**, such as **Tablet**.



5. After you select the **Device type**, click the **Add Layout** button in the upper-right corner (for example, **Add Tablet Layout**).



After you add a device layout, you'll see it appear on the **Dashboard** tab, under Default. The new device layout contains every item that the Default dashboard does and its size and layout are derived from Default as well.



Another change that takes place after you add a device layout is the default dashboard appears.

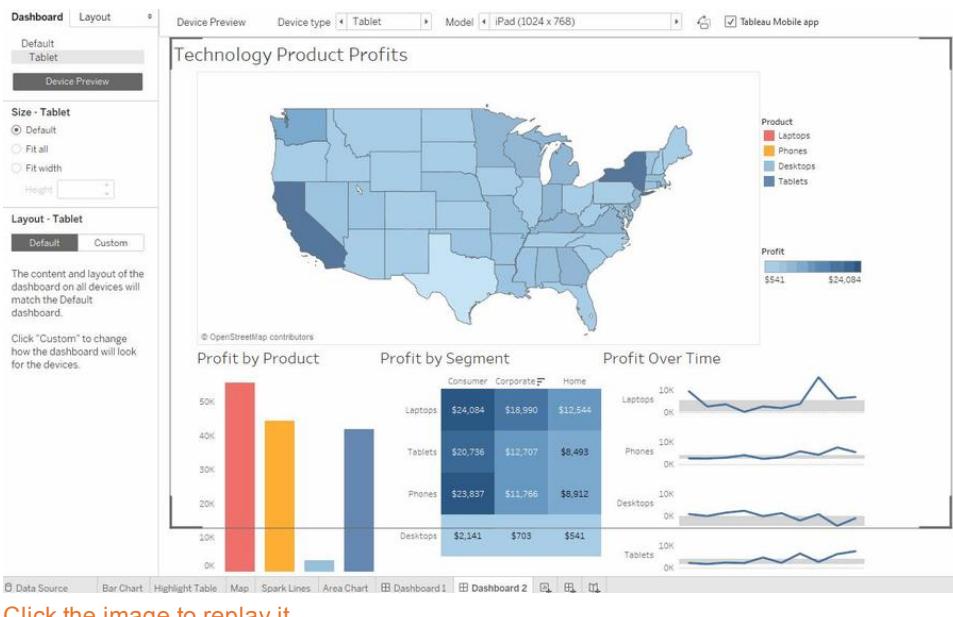
Think of the default dashboard as the parent dashboard, and the device layouts (desktop, tablet, and phone) as its children.

Any view, filter, action, legend or parameter that you want to add to a device layout must first exist in the default dashboard. Web, image, text, and layout objects can be added independently.

Customize a device layout

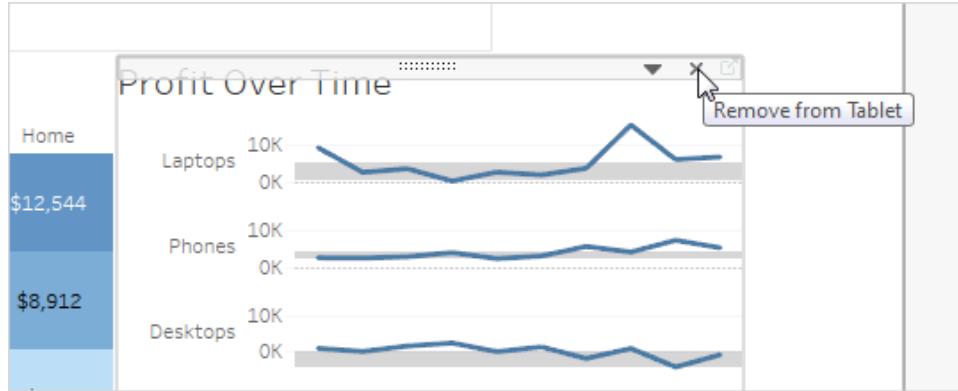
After you've added a device layout to your dashboard, you can customize it.

1. Click **Custom** and start removing and rearranging views and objects to create the look you want.

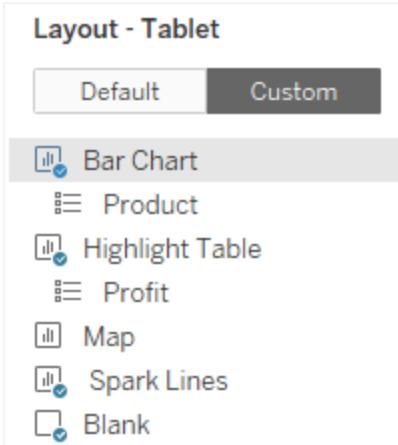


Click the image to replay it.

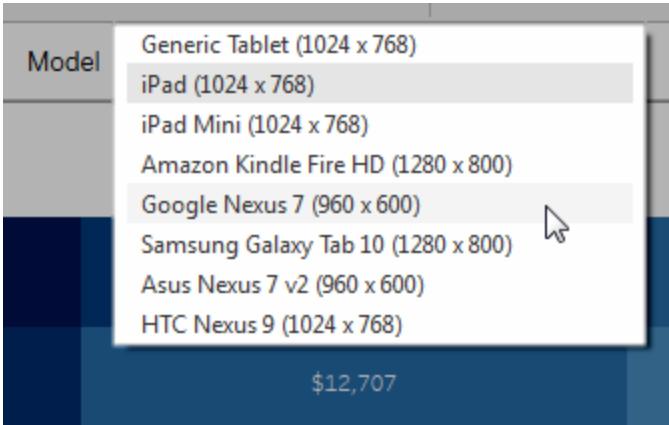
If you remove an item from the device layout, it's only removed from that layout. It still exists on the default dashboard and can be added back to the layout again, or to other device layouts.



Conversely, if a view, legend, filter, or object doesn't exist in the default dashboard, it can't be added to a device layout. Anything you can add to your layout is listed on the left, under **Layout**. If an item has a blue check mark, it means that it's part of the device layout you're currently working on.



2. Click through the **Device model** options to see how the layout will appear on different models. The Generic models (for example, Generic Tablet (1024 x 768)) are designed to fit most models.

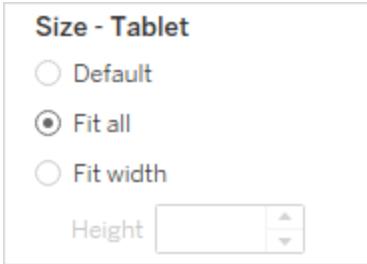


Ultimately, it's the size of the web browser that loads the dashboard that determines which layout appears on the device. See [Test the dashboard](#) on page 1171 for details.

3. Click to see how the dashboard will look in landscape vs. portrait mode. Usually, landscape is optimal for tablets and portrait is best for phones.

Selecting the **Tableau Mobile app** check box shows you the screen size you have to work with if the dashboard is viewed on Tableau Mobile. This option is available for iOS and Android phones and tablets that support Tableau Mobile.

4. Confirm your sizing choice by exploring the options under **Size**.



Default: The height and width of the device layout will mimic whatever the default dashboard is using. For example, if you're creating a tablet layout and the default dashboard is set to a fixed size of **Desktop Browser (1000 x 800)**, setting **Size** to **Default** for the tablet layout will make it use 1000 x 800 as its size.

Fit all: All items are automatically resized to fit the device frame size. The device frame size is determined by **Device type**, **Model**, orientation (portrait or landscape), and whether **Tableau Mobile app** is selected.



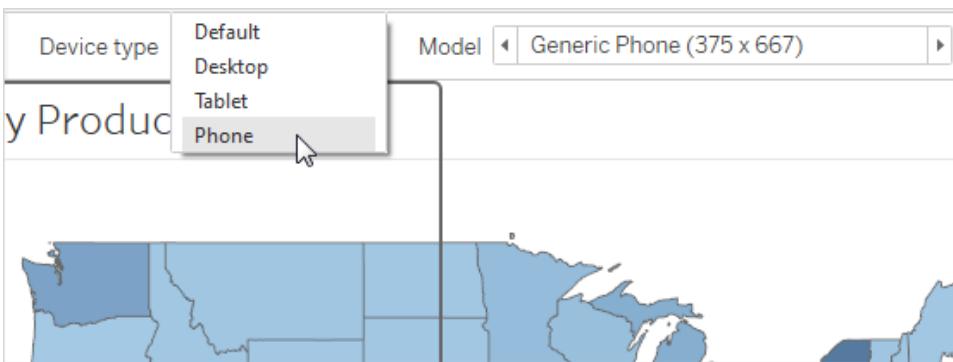
Fit width (recommended for phones): Items are automatically resized to fit the width of the device frame, but the height is fixed. This is a great option for phone layouts and vertical scrolling.

The next procedure describes how to create and optimize a layout for phone.

Add more content and optimize for phone

The following procedure steps you through adding a phone layout, including some tips for optimizing it.

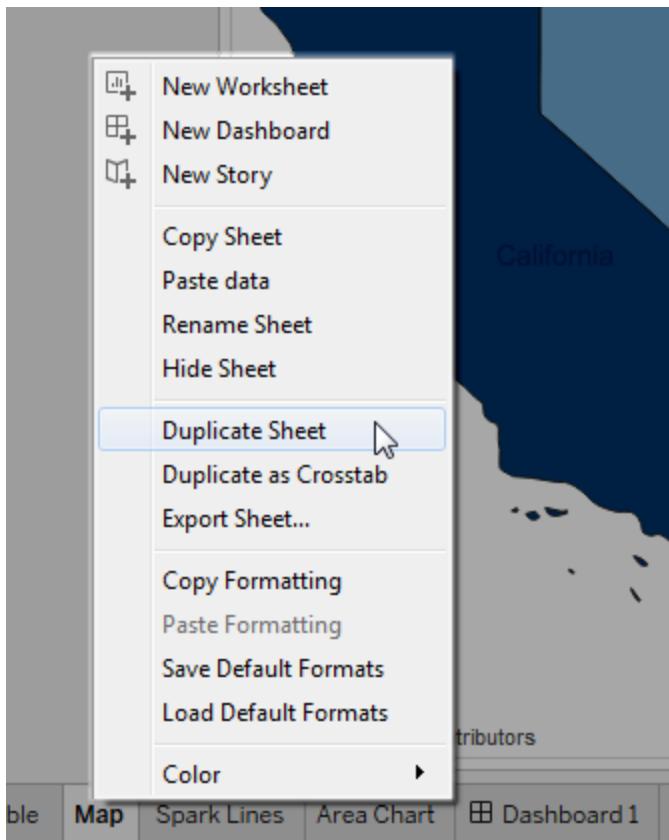
1. Add an additional layout by selecting a new **Device type**:



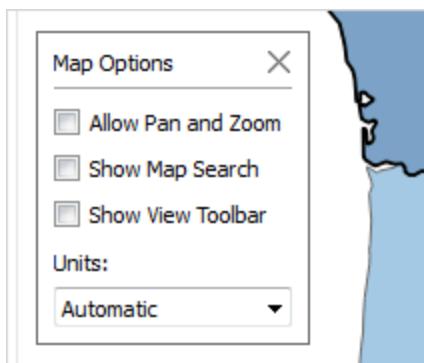
2. Click **Add Layout**.
3. (Optional) Because not every view or object is optimized for mobile viewing, one technique for creating a dashboard layout for phones is to create duplicates of certain

views in the default dashboard—as in, one for viewing on a desktop and laptop and a second for viewing on phones.

For example, to create a duplicate of a map view that has different functionality, go to the worksheet for the view, click its tab, and select **Duplicate Sheet**.

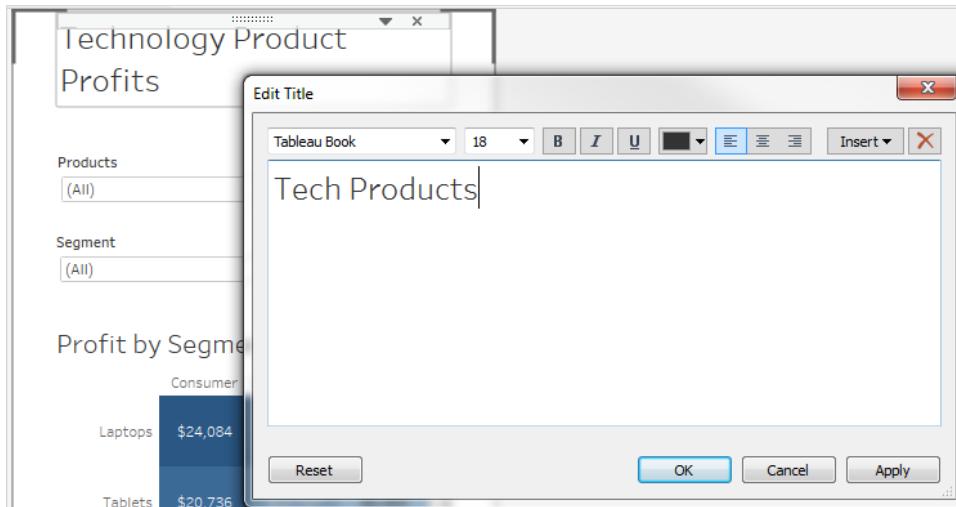


Next, customize the view for mobile viewing. For example, you may want to the map to be zoomed in to a specific region by default, or you may want to disable panning, zooming, and other maps functionality (**Maps > Map Options**).



Finally, add the new view to the default dashboard so that it can be available to the device layouts you're creating.

- As you optimize for different layouts, dashboard titles are another area to look at. Short titles work best for mobile viewing. To edit the title, double-click it:



Titles on a phone layout have the added benefit of providing a safe spot for users to initiate scrolling or swiping.

- White space is another visual element to consider if you're designing for phones. While screen real estate on a phone is scarce and you want to make the most of it, you may also want to provide additional safe places for your users to tap or initiate scrolling, so they don't select filters and other items unintentionally. To add white space, drag the **Blank** object onto the layout:

Dashboard | Layout

Device Preview Device type ▶ Phone ▶ More

Default

Tablet

Phone

Device Preview

Size - Phone

- Default
- Fit all
- Fit width

Height 1100 px

Layout - Phone

Default **Custom**

- Blank
- Blank
- Blank
- Blank
- Tech Profits

Clear all

Objects

- Horizontal Image
- Vertical Web Page
- Text Blank

Tiled **Floating**

Show dashboard title

Tech Profits

Products (All)

Segment (All)

Profit by Segment

	Consumer	Corporate	Home
Laptops	\$24,084	\$18,990	\$12,544
Tablets	\$20,736	\$12,707	\$8,493
Phones	\$23,837	\$11,766	\$8,912
Desktops	\$2,141	\$703	\$541

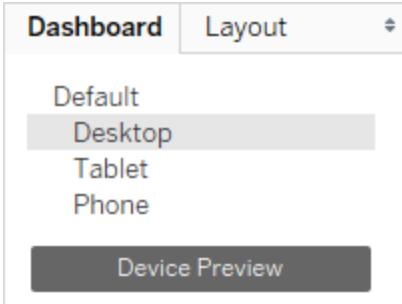
Profit Over Time

The chart displays two line graphs side-by-side. The top graph is for 'Laptops' and the bottom graph is for 'Phones'. Both graphs show profit levels from 0K to 10K. The 'Laptops' graph shows a general upward trend with some fluctuations. The 'Phones' graph shows a more stable, lower-level profit range.

Click the image to replay it.

In most cases, you should leave **Tiled** selected (the default). While floating objects can be used on device layouts, it's difficult to predict and control their exact position across multiple layouts.

6. Add device layouts until you have all three possible layouts under Default: Desktop, Tablet, and Phone.

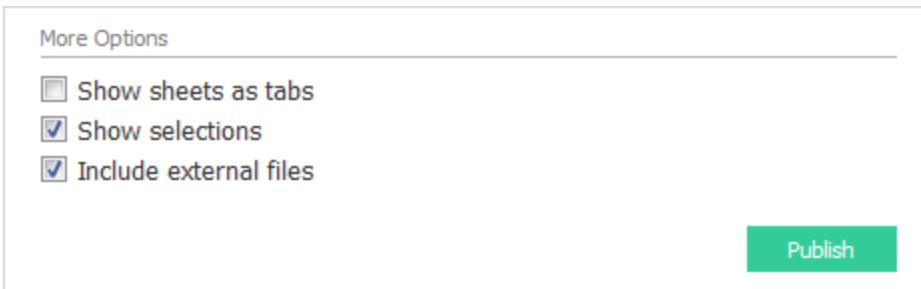


Creating a layout for each device type will give you the most control over your users' experience as they view your dashboard from different devices. This is especially true if you've created and added multiple versions of the same view to the default dashboard. After you have all three layouts, the default dashboard isn't viewable when the dashboard is published. Only the device-specific layouts for your dashboard are viewable.

Publish the dashboard

To publish the dashboard:

1. Click **Server > Publish Workbook**. If you're not already signed in, you're prompted for your credentials.
2. In the Publish Workbook to Tableau Server dialog box, make sure the **Show sheets as tabs** check box is cleared.



When this check box is selected for device-specific dashboards, the tabs' sizing requirements interfere with the server's ability to correctly detect the size of the web browser and load the correct layout.

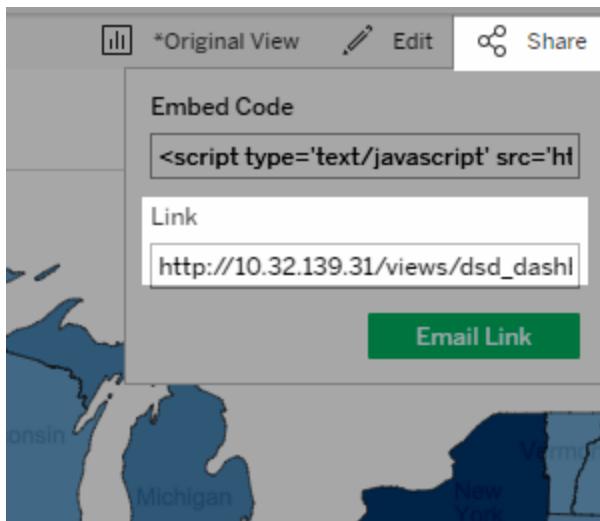
3. Click **Publish**.

Test the dashboard

After you publish the dashboard to Tableau Server or Tableau Online, test the dashboard by viewing it from different browser sizes. The best way to test is by using the embed code under **Share**.

To test the dashboard:

1. Open the dashboard on Tableau Server or Tableau Online.
2. In the upper-right corner of the page, click **Share** and copy the contents of the **Link** text box.



3. Paste the string into a web browser URL. The string should include the following:
embed=y
4. With the embed code string as your browser URL, test the different layouts by changing the size of your web browser window and refreshing it.

Tableau uses the size of the web browser that loads the dashboard to determine which dashboard layout to display on a device. Specifically, it uses the minimum dimension of the iFrame, whether it's the height or the width. In a few cases, a device type's label (Desktop, Tablet, etc.) may not map perfectly to the type of device it will actually display on. For example, a Tablet layout, not a Desktop layout, may display on a desktop device if the web browser is minimized, or if the desktop is simply very small.

If the smallest web browser dimension is...	This device layout is loaded...
Less than or equal to 500 pixels	Phone
Between 501 and 800 pixels	Tablet
Greater than 800 pixels	Desktop

Refine Your Dashboard

After you've created your dashboard, take a moment to step back and evaluate it. This topic lists some areas to check and refine.

Is everything in the right spot?

- **Confirm placement below**
- **Check label orientation and readability** on the next page

Review sizing and layout

- **Test your work and adjust sizing** on the next page
- **Customize image display** on page 1175
- **Use a layout container** on page 1175
- **Buy space with a floating layout** on page 1178

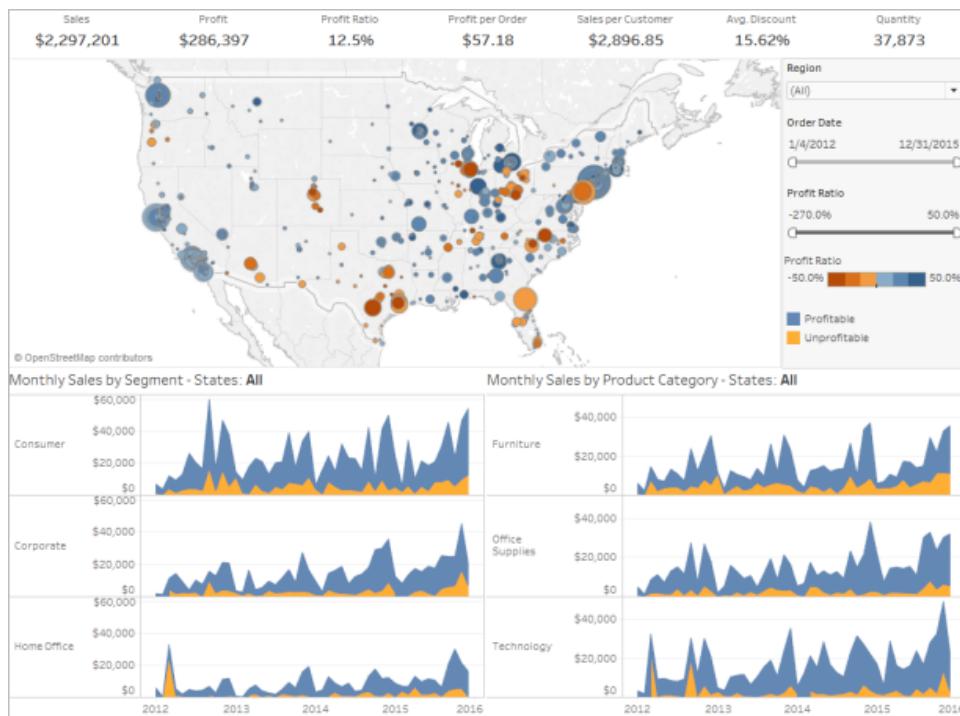
Highlight the essentials

- **Show just what your users need** on page 1180
- **Eliminate clutter** on page 1182
- **Make a sheet sorter** on page 1183

Is everything in the right spot?

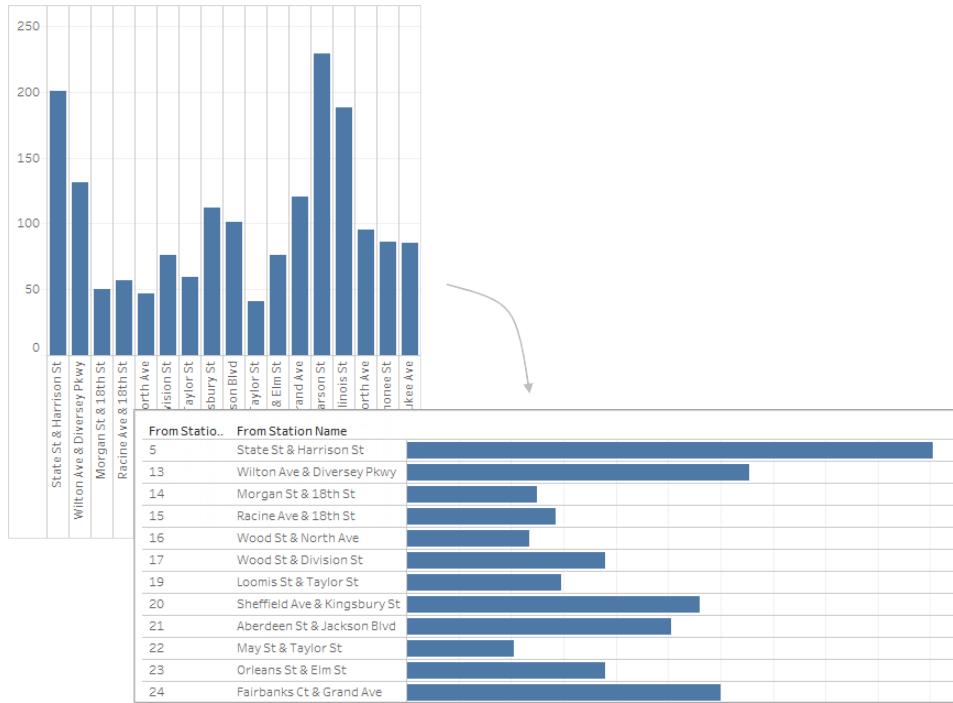
Confirm placement

Make sure that the view with your key finding or main question spans or occupies the upper-left corner of your dashboard. In your views, the most important data should be on the X or Y axis and your less important data should be on color, size, or shape.



Check label orientation and readability

Important labels should be horizontal so that your viewers can easily read them.



Review sizing and layout

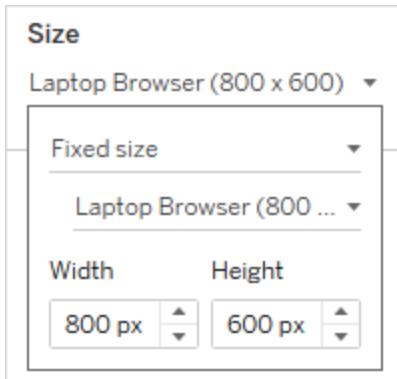
Test your work and adjust sizing

Once you finish your dashboard, it's a best practice to test it and experience it as your users will. For example, will they be using Tableau Reader on a laptop to view your dashboard, will the dashboard be published on Tableau Server, or will it be embedded in a web application and viewed from a tablet?

After you test your dashboard, you'll probably find some things to change—sizing and how and where items are positioned are usually among them.

Tableau dashboards are set to a fixed default size that's intended to work well on a typical desktop. However, when you publish (to the web, in a blog, for a presentation, etc.) you may find yourself more limited.

You can specify the overall size of the dashboard using the settings under **Size** in the **Dashboard** area on the left. Select a new size using the drop-down menu.



- **Automatic** - The dashboard automatically resizes to fill the window it's displayed in.
- **Fixed size** - The dashboard always remains a specific size. If the dashboard is larger than the window the dashboard becomes scrollable.
- **Range** - The dashboard scales between minimum and maximum sizes that you specify, after which scroll bars or white space display.

Customize image display

When you add an **Image** object to a dashboard, you can customize how the image displays by selecting an option on the image's menu. For example, you can select whether to **Fit Image**, which scales the image to the size of image object on the dashboard.

By default, any high-resolution images that you use in a dashboard are displayed in standard resolution and will maintain the same size whether they're viewed on standard or Retina displays.

To take advantage of high-resolution images in a dashboard that are intended to be viewed on a Retina display, you can use the @2x naming convention:

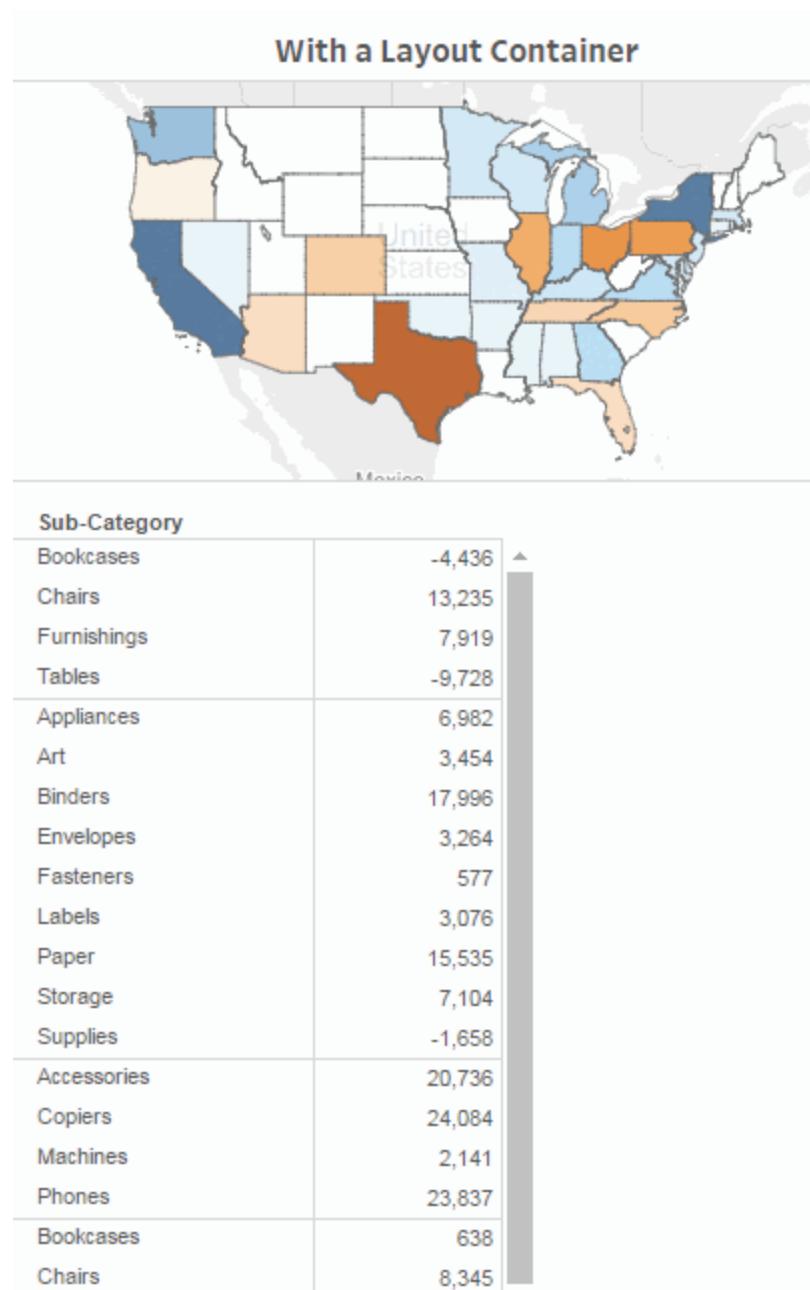
1. Navigate to the high-resolution image on your computer with Retina display.
2. Rename your high-resolution image to include the @2x modifier by using <image-name>@2x<file-extension> format. For example, logo@2x.png.
3. In your workbook, click and drag an **Image** object onto your dashboard.
4. When you're prompted, select the image you just renamed using the @2x naming convention and then click **Open**.

Note: EMF image file formats are not compatible with Tableau Desktop on the Mac.

Use a layout container

A layout container can create a smoother experience for your users by helping dashboard objects reposition and resize when one view is filtering another.

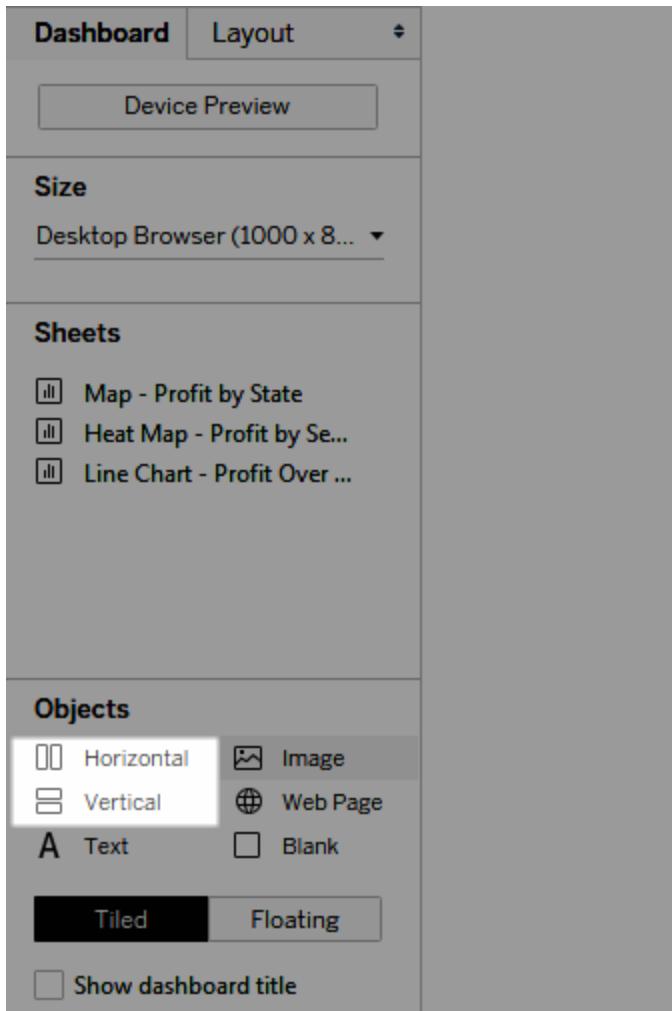
The image below shows how a dashboard behaves when two views are placed in a vertical layout container versus how they behave when they are not placed in a layout container. Notice how, in the dashboard that uses a layout container, the views adjust vertically as different filters are applied.



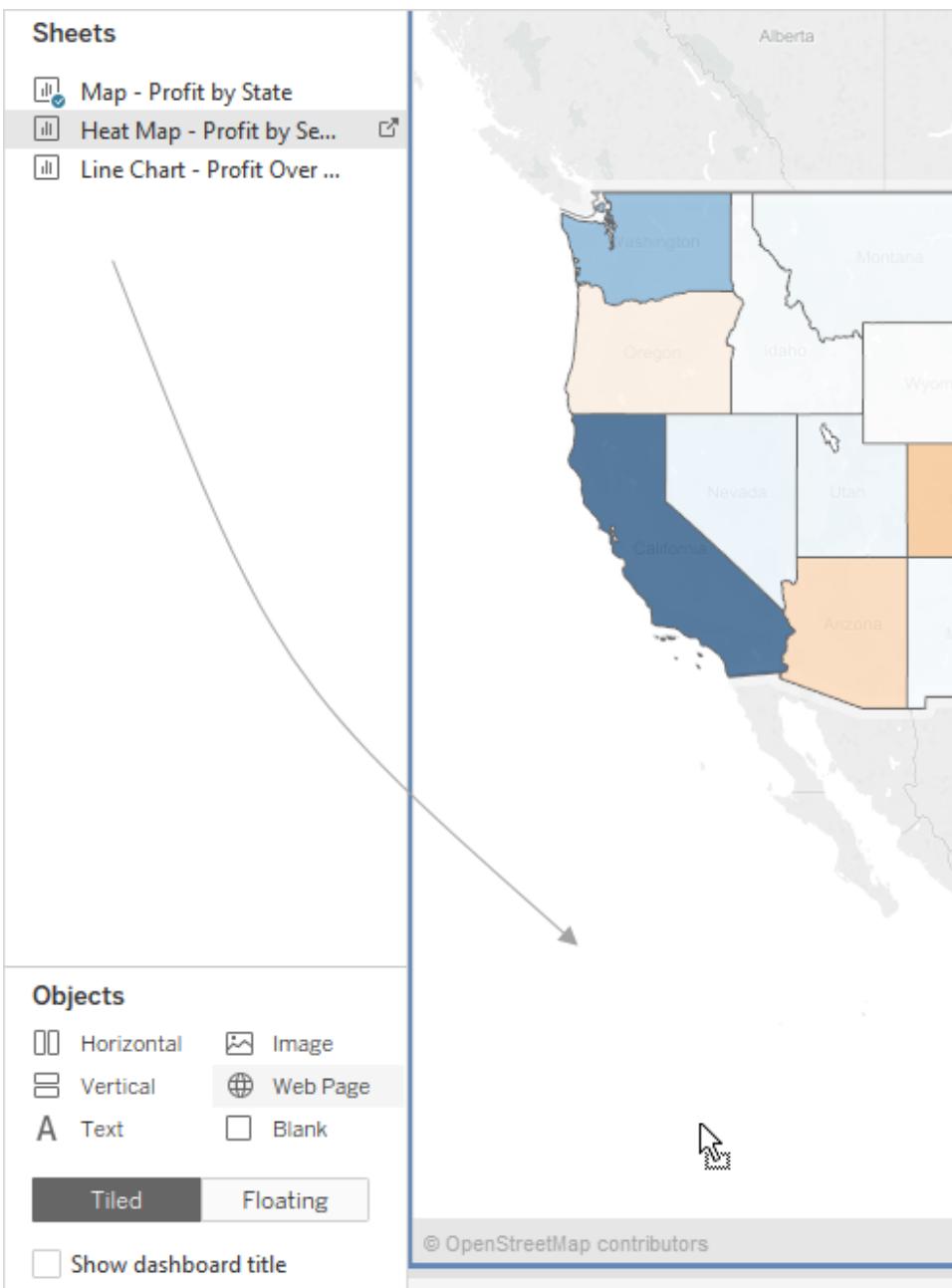
[Click the image to replay it.](#)

To add a layout container

1. Drag a horizontal or vertical layout container to the dashboard.

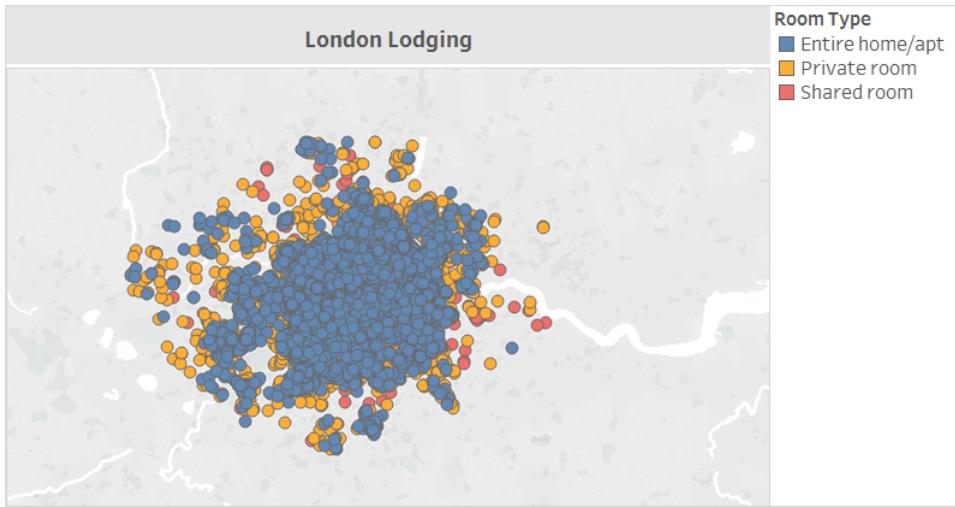


2. Add sheets and objects to the layout container. As you hover over the layout container, a blue box indicates that the object is being added to the flow of the layout container.

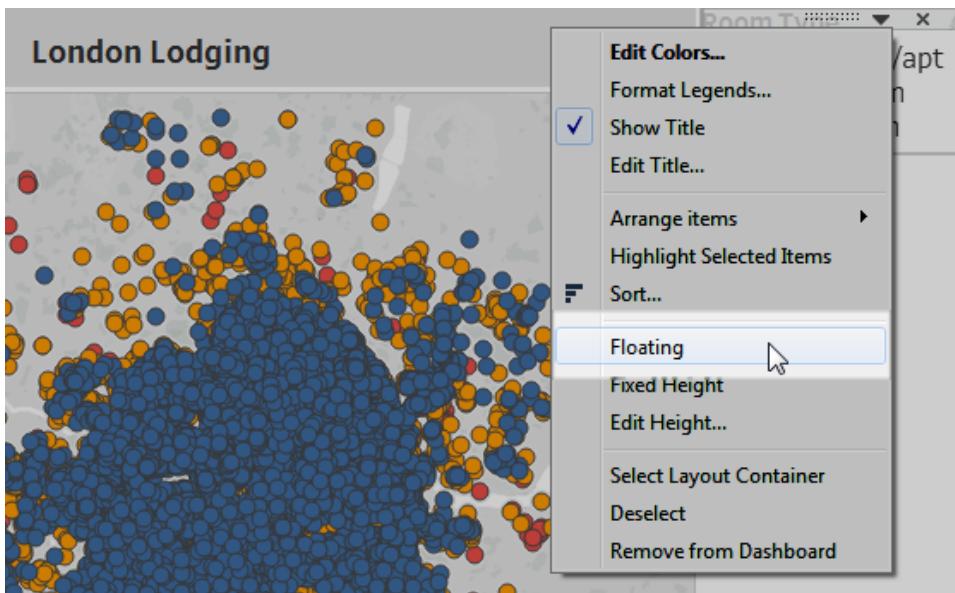


Buy space with a floating layout

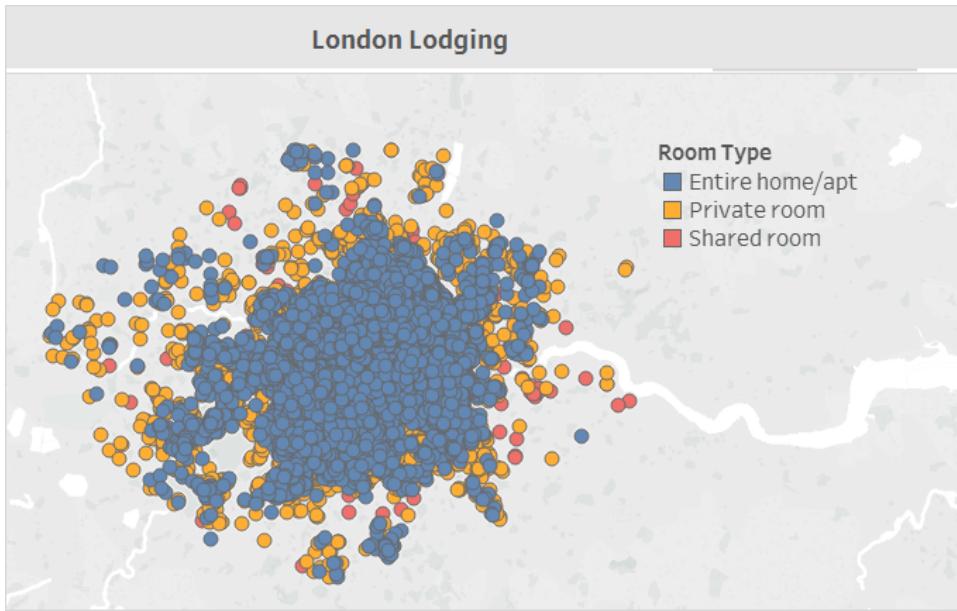
Tableau uses a tiled layout by default in dashboards, which means that each view, legend, and object is arranged in a single-layer grid, similar to a tiled floor.



If you want an item to overlap another, you can use a **Floating** layout for one or more items instead:



Legends can sometimes be floated over other items instead of remaining tiled.



See [Resize and Lay Out Your Dashboard](#) on page 1184 for more details.

If you create device layouts for a dashboard, use a **Tiled** layout. It will give you the most control over where objects appear. See [Create Dashboard Device Layouts](#) on page 1160 for steps on how to create device layouts.

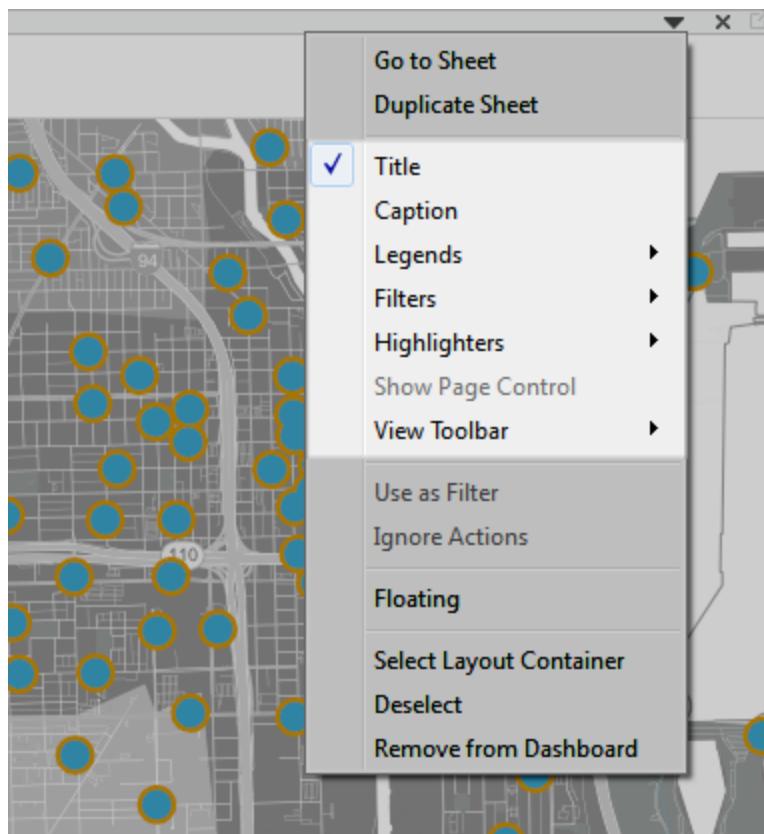
Highlight the essentials

Show just what your users need

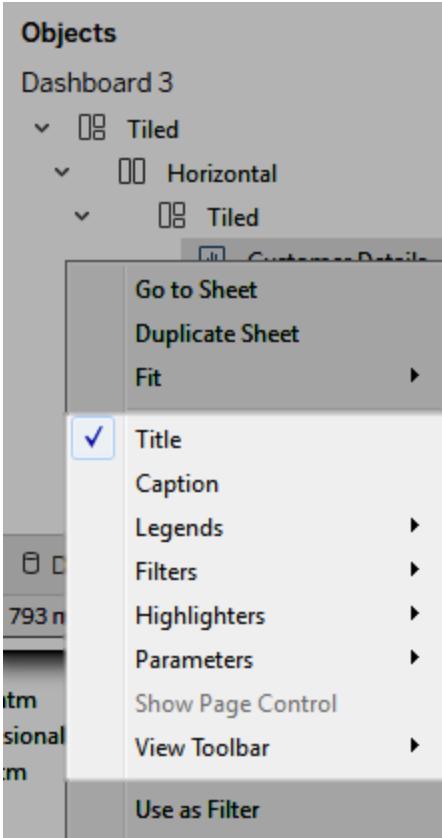
Legends, titles, captions, and filters can be helpful for your viewers. Make sure you give your users what they need and no more.

To change what's displayed for your users:

1. Select a view in the dashboard.
2. Click the drop-down menu in the upper-right corner of the selected view and select the items you want to show. For example, you can show the title, caption, legends, and a variety of filters.



3. Alternatively, you can right-click (control-click on Mac) an item in the **Layout** section of the **Dashboard** pane to access all of these same commands.

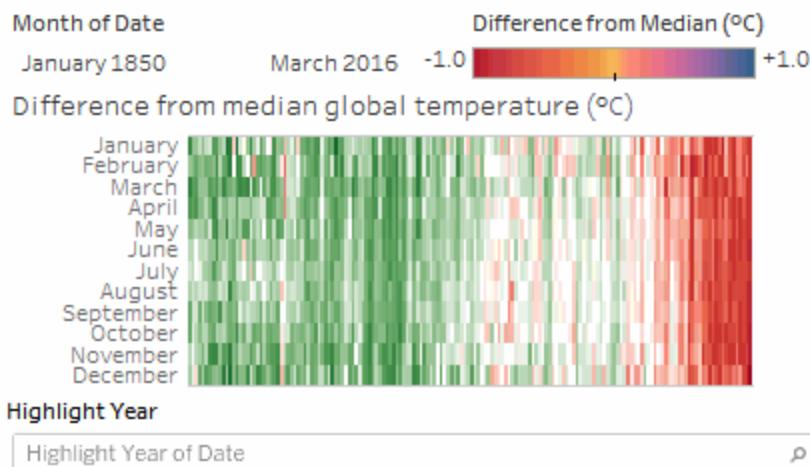


Note: Filters are only available for the fields used in the original view.

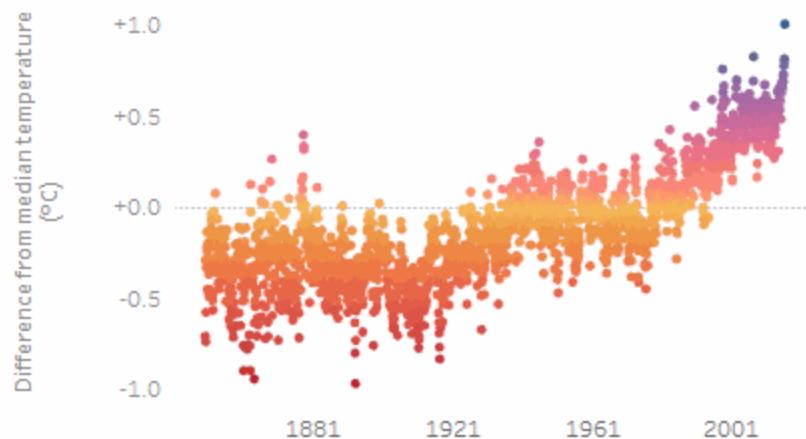
Eliminate clutter

In general, you should try to use no more than two color palettes in a single dashboard, and ideally, just one palette if your data is quantitative. Remove unnecessary text, lines, or shading that don't provide actionable information for your viewers. Make sure every legend you provide is really needed, remembering to take into account the final, published size of your dashboard.

Cluttered



Scatter Plot



The trend of Median for Date. Color shows Median. Details are shown for Year of Date and Month of Date. The data is filtered on Month of Date, which ranges from January 1850 to March 2016 and keeps Null values. The view is filtered on Year of Date, which excludes 2016. The trend of Median for Date.

Click the image to replay it.

Make a sheet sorter

A sheet sorter can be a time-saver for the people viewing your dashboard. It allows them to quickly display individual worksheets on the dashboard. For information, see [Creating a Sheet Selector for a Dashboard](#).

Resize and Lay Out Your Dashboard

After you create a dashboard, you might need to spend some time resizing it and reorganizing its contents so that it works for your users. This topic walks you through the different approaches you can take and gives you tips on when to use each.

Control overall dashboard size

- [Dashboard size options](#)
- [Set overall dashboard size](#)

Group items using layout containers

- [Layout container types](#)
- [Add a layout container](#)
- [Format a layout container](#)

Layer or tile individual items

- [Tiled versus floating layouts](#)
- [Assign a tiled or floating layout to a new item](#)
- [Assign a floating layout to an existing Item](#)

Size, position, and reorder individual items

- [Set an item's size and position](#)
- [Reorder floating objects](#)

Control overall dashboard size

Dashboard size options

Here are the settings you can use for controlling the overall size of a dashboard:

Fixed size (default): The dashboard remains the same size, regardless of the size of the window used to display it. If the dashboard is larger than the window, it becomes scrollable. You can pick from a preset size, such as Desktop Browser (the default), Small Blog, and iPad.



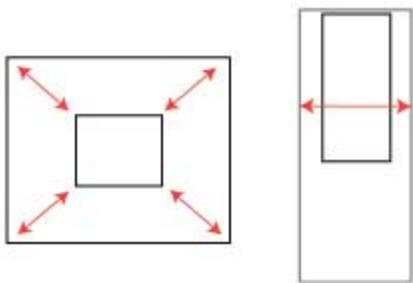
This setting allows you to specify the exact location and position of objects, which can be useful if there are floating objects in the dashboard. Select this setting if you know the size at which your dashboard will be displayed, and it's a single size.

Published dashboards that use a fixed size can load

faster than dashboards with variable sizes because they're more likely to be loaded directly from the server cache versus needing to be freshly rendered for every browser request.

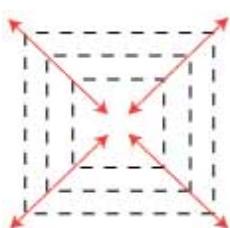
For other performance tips, see [Optimize Workbook Performance](#) on page 1817.

Range: The dashboard scales between minimum and maximum sizes that you specify. If the window used to display the dashboard is smaller than the minimum size, scroll bars are displayed. If it's larger than the maximum size, white space is displayed.



Use this setting when you're designing for two different display sizes that need the same content and have similar shapes—such as small browser windows and medium browser windows. It also works well for mobile dashboards with vertical layouts, where the width may change to account for different mobile device widths, but the height is fixed to allow for vertical scrolling.

Automatic: The dashboard automatically resizes to fill the window used to display it.



Use this setting if you want Tableau to take care of the resizing.

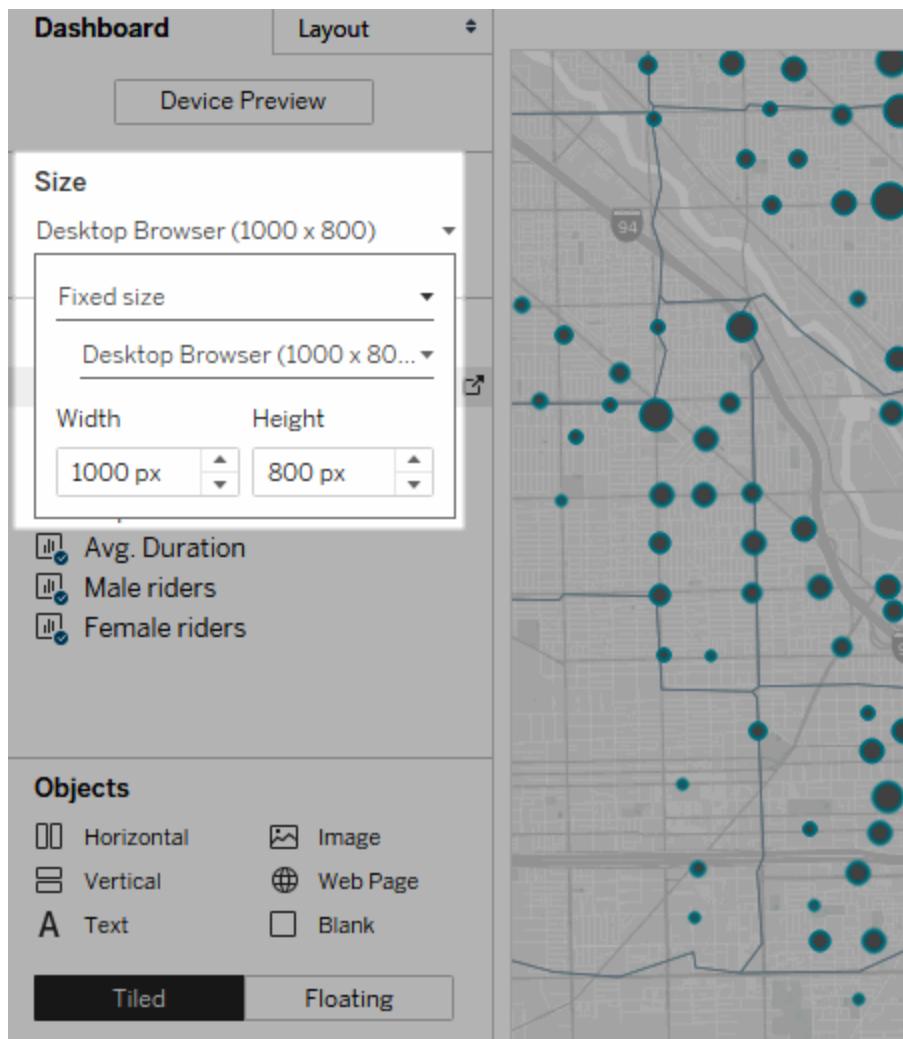
For best results with this setting, place views and objects in the dashboard using a **Tiled** layout.

Device layouts are another approach to creating a dashboard that will work for a variety of browser window sizes. In addition to being different sizes (desktop, iPad, mobile), each layout can contain different items. See [Create Dashboard Device Layouts](#) on page 1160 for details.

Set overall dashboard size

To set the overall size of a dashboard:

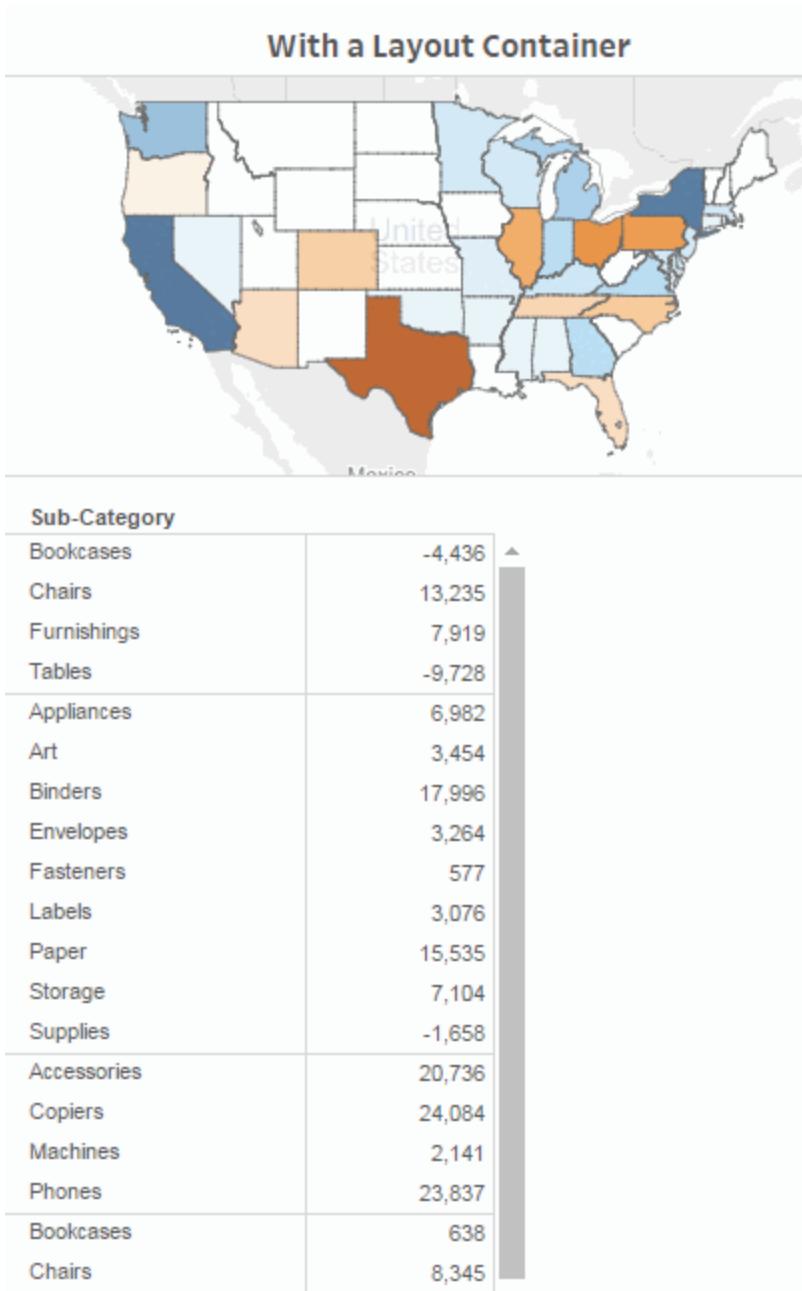
- Under **Size** on the Dashboard pane, select the dashboard's dimensions (such as **Desktop Browser**) or sizing behavior (for example, **Automatic**).



Group items using layout containers

A layout container is an optional way to keep views and objects together. It helps groups of items resize and change position when users interact with your dashboard so that the dashboard doesn't include white space that you don't want.

The image below shows how a dashboard behaves when two views are placed in a vertical layout container versus how they behave when they are not placed in a layout container. Notice how, in the dashboard that uses a layout container, the views adjust vertically as different filters are applied.



Layout container types

A horizontal layout container resizes the width of the views and objects it contains; a vertical layout container adjusts height.

Horizontal layout container

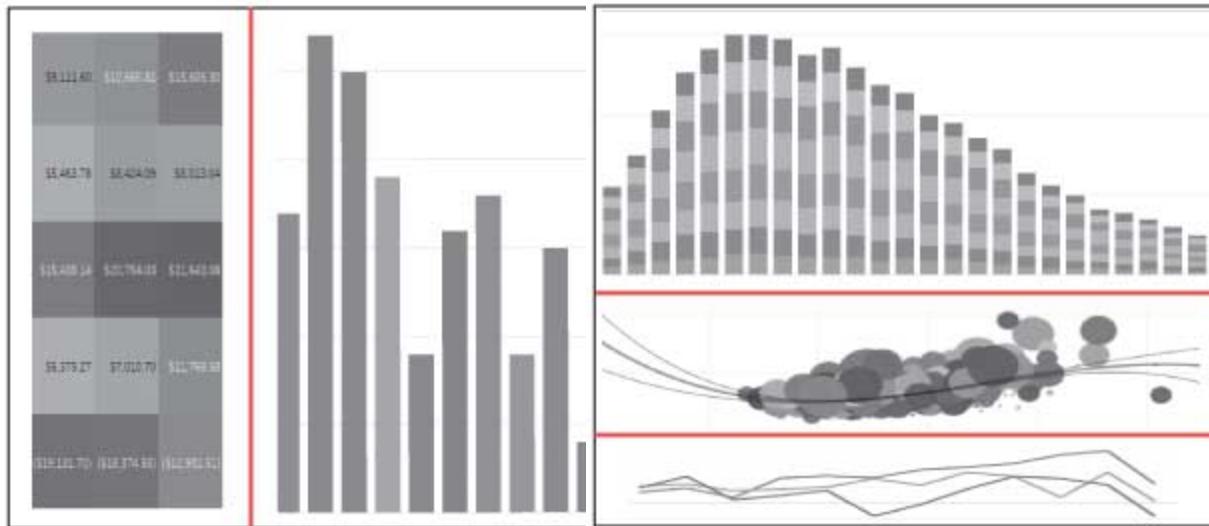
The two views below are in a horizontal

Vertical layout container

The three views below are stacked in a

layout container.

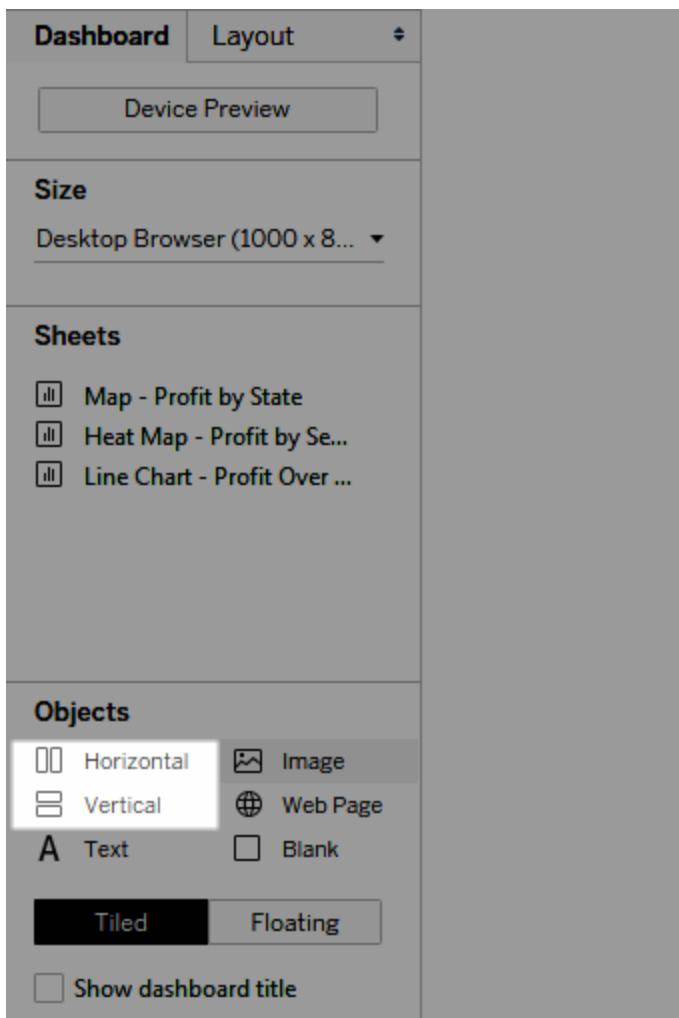
vertical layout container.



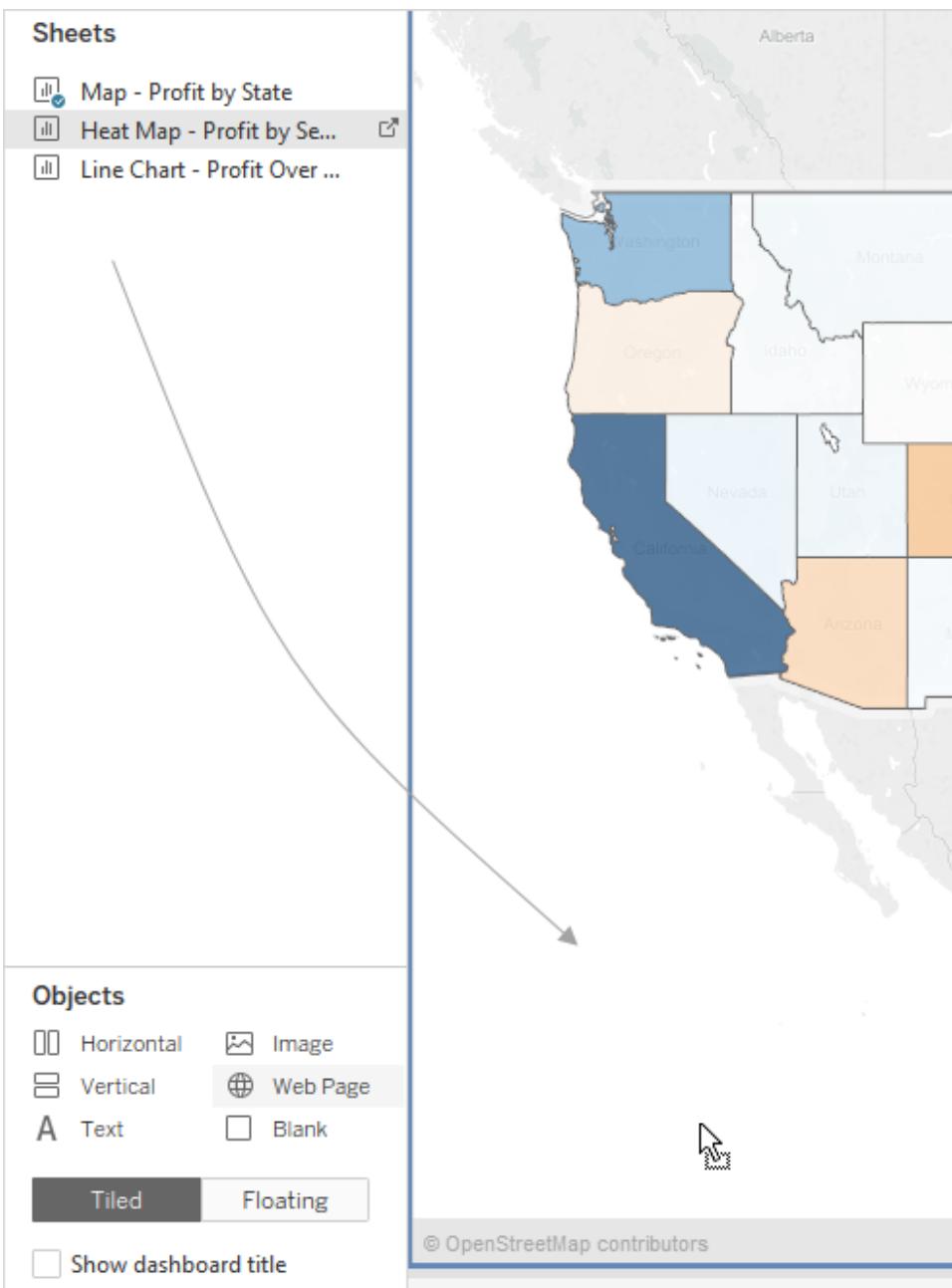
Add a layout container

To add a layout container to a dashboard:

1. Under **Objects** on the Dashboard pane, select **Horizontal** or **Vertical**.
2. Drag the container to the dashboard.



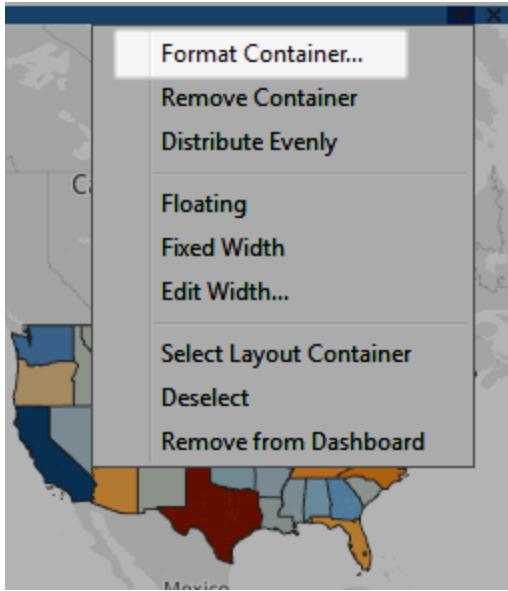
3. Add views and objects to the layout container.



Format a layout container

You can specify shading and border style for layout containers so that you can visually group objects in the dashboard. By default, layout containers are transparent and have no border style.

Open the shortcut menu for the layout container you want to format, and then select **Format Container**.



Layer or tile individual items

Tiled vs. floating layouts

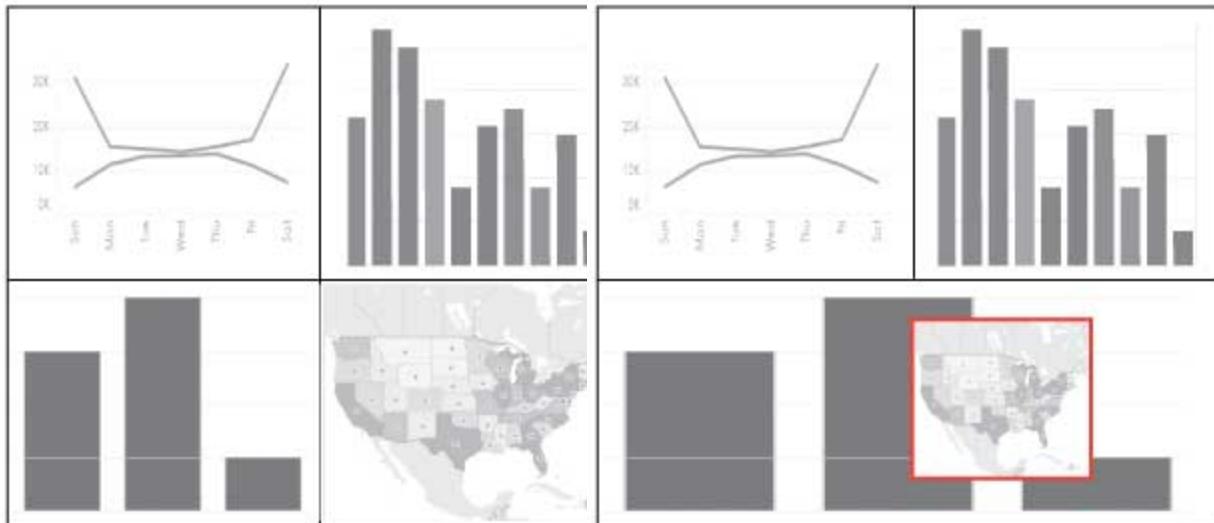
Each object and view that you place on a dashboard has a layout attribute, either tiled (the default) or floating.

Tiled layout

When a view or object is placed with this attribute, it becomes part of a single-layer grid that resizes based on the overall dashboard size. Tiled items don't overlap one another.

Floating layout

Items placed with this attribute can be layered on top of other objects. In the graphic below, every view is tiled except for the map view, which is floating.



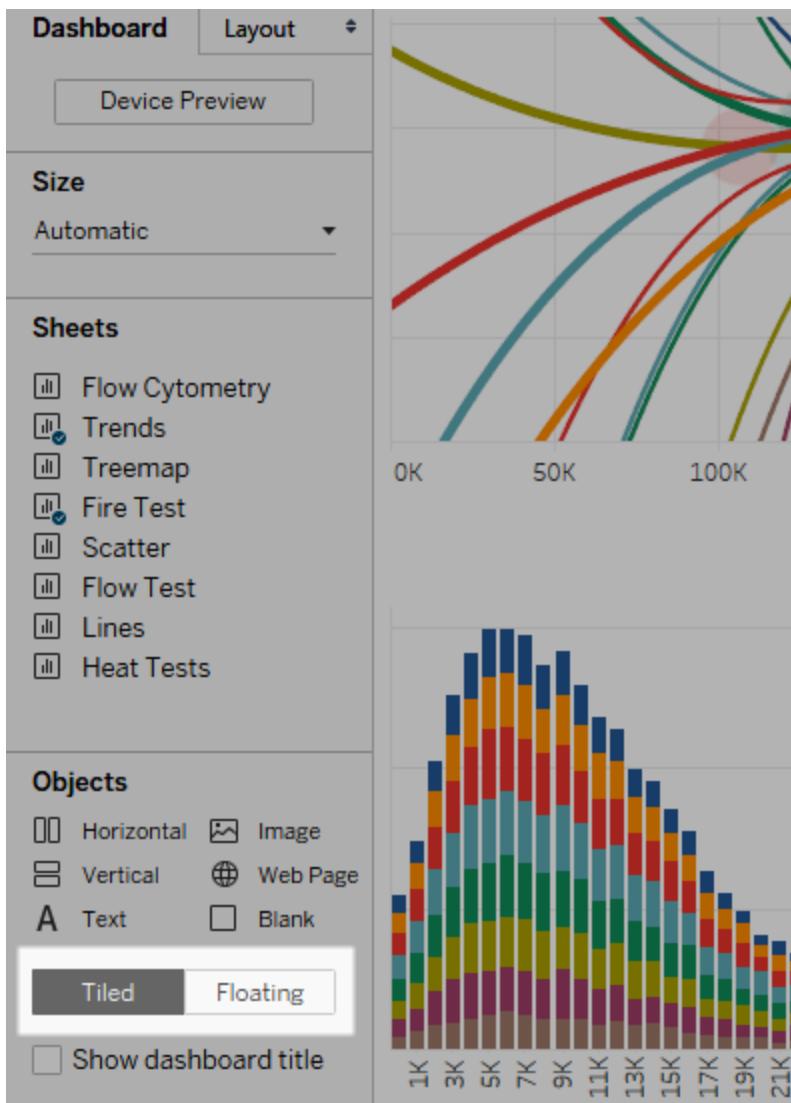
For best results, give floating objects and views a fixed size and position.

Layout containers can also be floating or tiled. For example, you may have a group of key performance indicators that you want to float over a view in your dashboard. Putting them in the same layout container keeps them together, and floating them as a group saves space. If you do make a floating layout container, the items within it must be tiled.

Assign a tiled or floating layout to a new item

To assign a tiled or floating layout to a view or object that isn't already in your dashboard:

1. Before you drag the view or object onto your dashboard, under **Objects** in the Dashboard pane, click the layout option you want to use: **Floating** or **Tiled**.



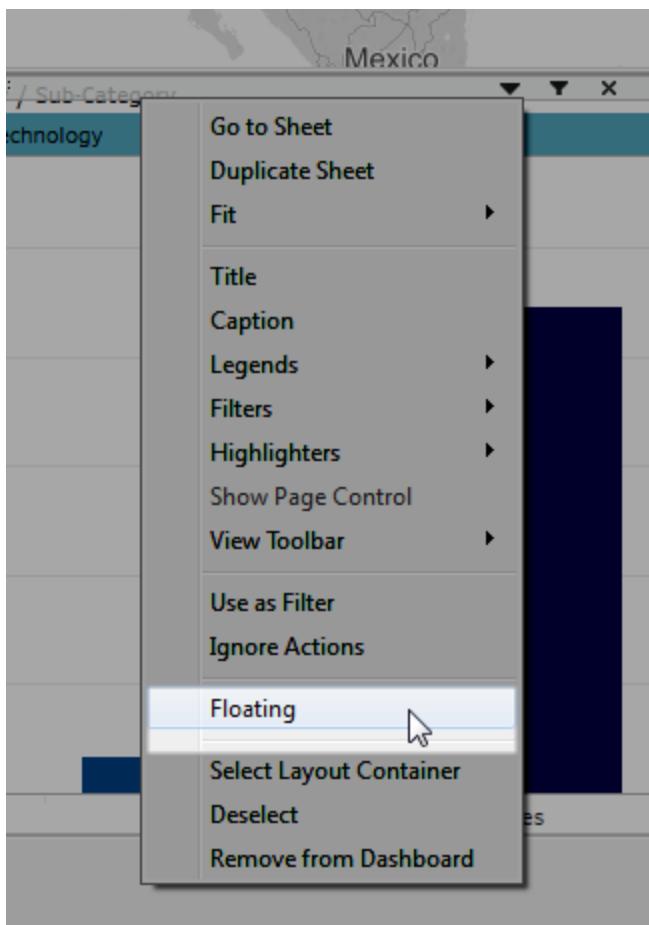
2. Drag the view or object onto the dashboard on the right.

Tip: You can also assign a floating layout to an item by holding down **Shift** on your keyboard as you drag it onto the dashboard.

Assign a tiled or floating layout to an existing item

To switch an item that's already in your dashboard from **Tiled** to **Floating**:

1. Select the item in the dashboard.
2. In the item's shortcut menu, select **Floating**:



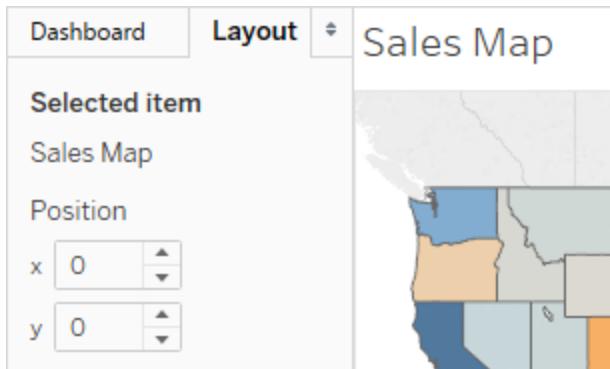
Size, position, and reorder individual items

If you want a high degree of control over the size and position of every item in a dashboard, and if you're also using **Fixed size** or **Range** for overall dashboard sizing, use the settings on the Layout pane. This is where you can control an object or view's exact position, size, and other characteristics.

Set an item's size and position

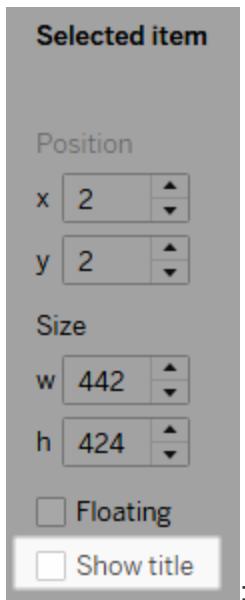
1. Select the dashboard item you want to position and size. The item's name appears under **Selected item**.
2. Define the item's position **x** and **y** position in pixels as an offset from the top left corner of the dashboard.

For example, to place an object in the top left corner of the dashboard, specify x = 0 and y = 0.



The values you enter can be positive or negative and must be whole numbers.

3. To resize an item, click and drag it in the dashboard. You can also use the settings under **Size**.
4. To hide the title of the item you have selected, clear the **Show title** check box. The **Show dashboard title** check box on the Dashboard pane controls just the dashboard title.

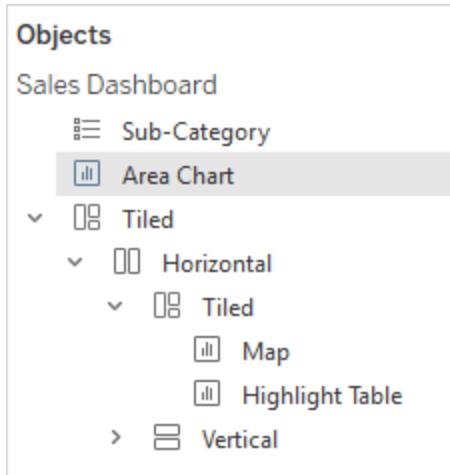


Reorder floating objects

The **Objects** area on the Layout tab displays tiled objects and any floating objects in a hierarchy.

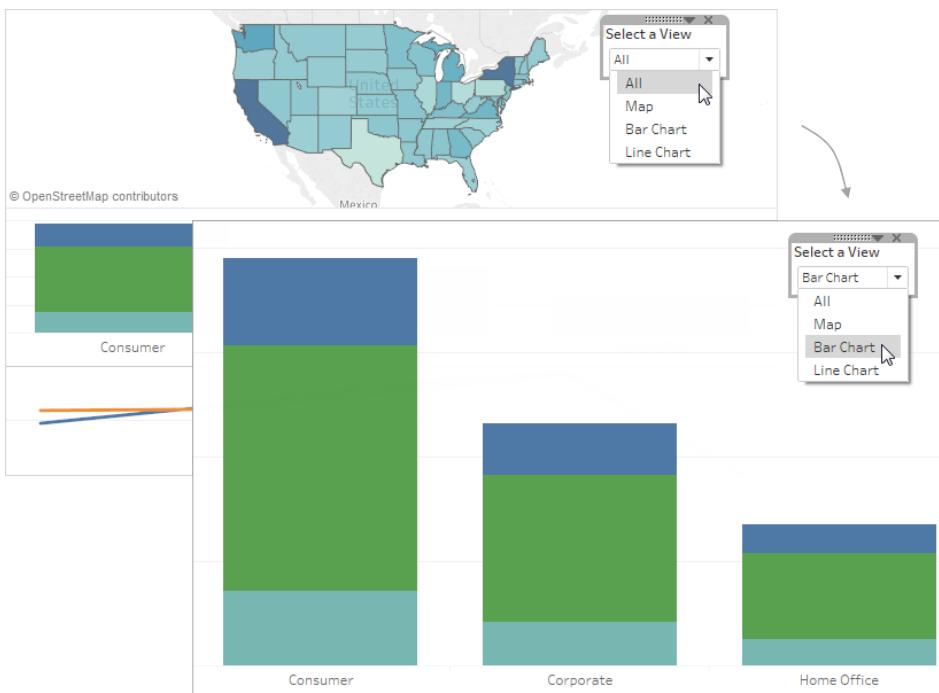
Drag floating items in the hierarchy to change the order in which they are layered in the dashboard. Items shown at the top of the list are in the front while items at the bottom of the list are in the back.

Note: Tiled layout items cannot be reordered.



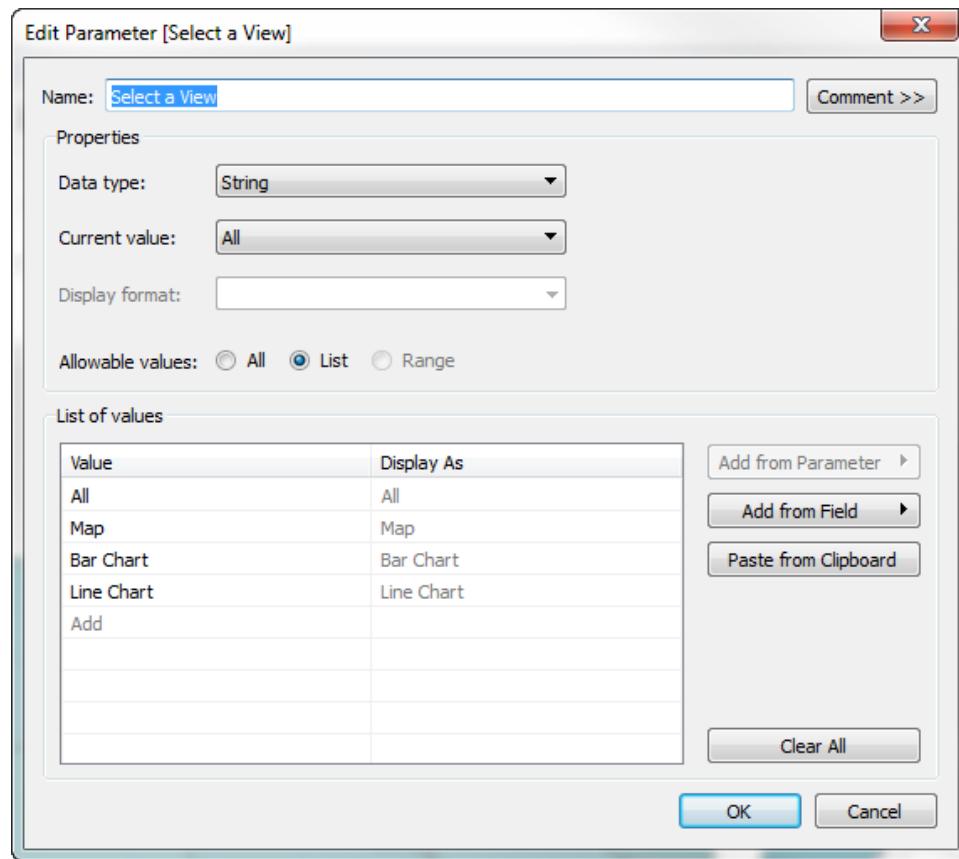
Create a View (Sheet) Selector for Your Dashboard

You can create a drop-down selector for your dashboard that lets your users select an individual view from the dashboard. When they select a view, all the other views on the dashboard are replaced with just the view they've selected, and the view fills the entire dashboard.



1. Right-click the Data window, and select **Create Parameter**.
2. Create a string parameter by doing the following tasks in the Create Parameter dialog box:
 - In the Name text box, type a name for the selector. For example, type **Select a View**.
 - In the Data type drop-down list, select **String**.
 - For **Allowable values** select **List**.
 - Under **List of values**, type **All** in the first text box, and then type the name of each view in the dashboard. In the example below, the view names are **Map**, **Bar Chart**,

and Line Chart.



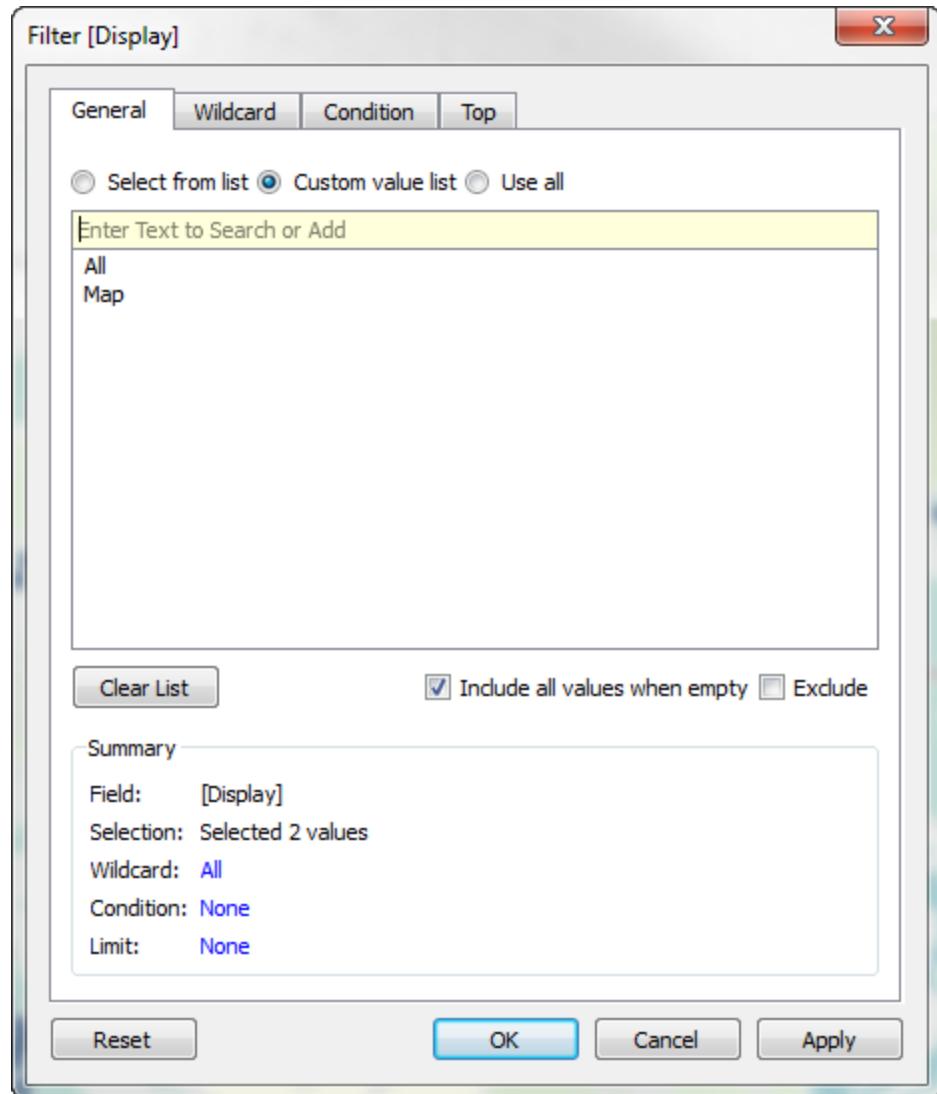
3. Click **OK**.
4. Right-click the Data window, and select **Create Calculated Field**.
5. Give the calculation a name, such as **Display**, then in the formula text box, type the name of the parameter you created in step 2.



6. For each view you plan to add to your dashboard, drag the calculation you created in step 4 to the Filters shelf and do the following tasks in the Filter dialog box:

- Select the **Custom Value List** option.
- Type **All** in the text box.

- Type the view name in the text box and click the **Add Item** button.



7. Click **OK**.
8. For each view you plan to include in your dashboard, right-click the upper-right corner of the worksheet, select **Parameters**, and then select the parameter you created in step 2 (for example, Select a View).
9. Select **Dashboard > New Dashboard**.
10. Under **Objects** on the Dashboard pane, drag a **Vertical** or **Horizontal** layout container to the dashboard. Then, drag each view's sheet to the dashboard.
11. On the dashboard sheet, right-click the title area of each view and select **Hide Title**.

Now you should have a dashboard with a drop-down selector that lets you select individual views. When a view is selected, it fills the entire dashboard, replacing the other views.

Stories

A **story** is a sheet that contains a sequence of worksheets or dashboards that work together to convey information. You can create stories to show how facts are connected, provide context, demonstrate how decisions relate to outcomes, or simply make a compelling case.

A story is a sheet, so the methods you use to create, name, and otherwise manage worksheets and dashboards apply to stories. For more information, see [Sheets on page 242](#). At the same time, a story is also a collection of sheets, arranged in a sequence. Each individual sheet in a story is called a **story point**.

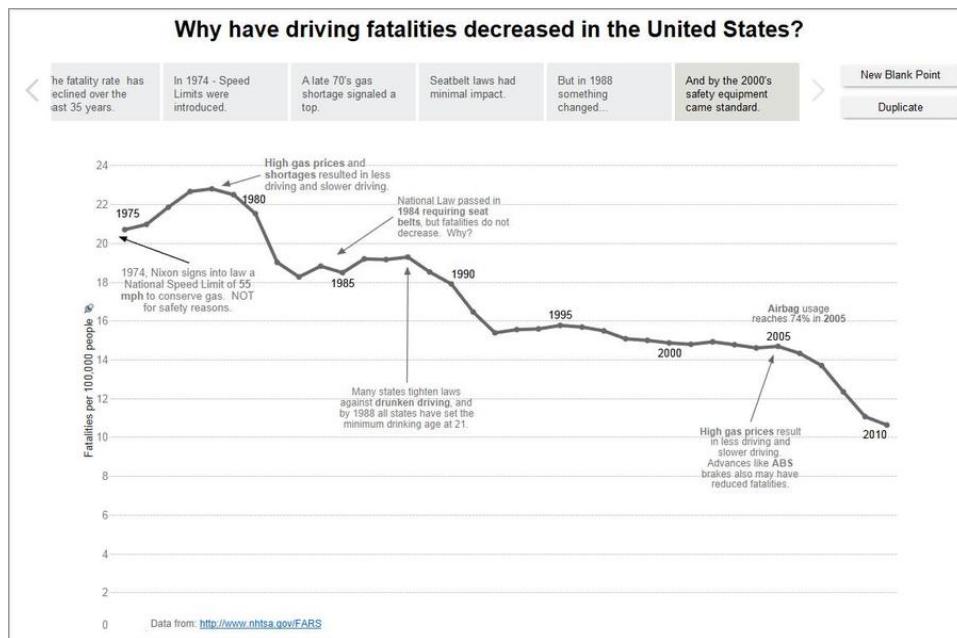


Tableau stories are not collections of static screen captures—your story points remain connected to the underlying data and change as the data changes—or as the views and dashboards that you use in your story change. When you share a story—for example, by publishing a workbook to Tableau Server or Tableau Online—users can interact with the story to reveal new findings or ask new questions of the data.

There are many different ways to use stories. For example:

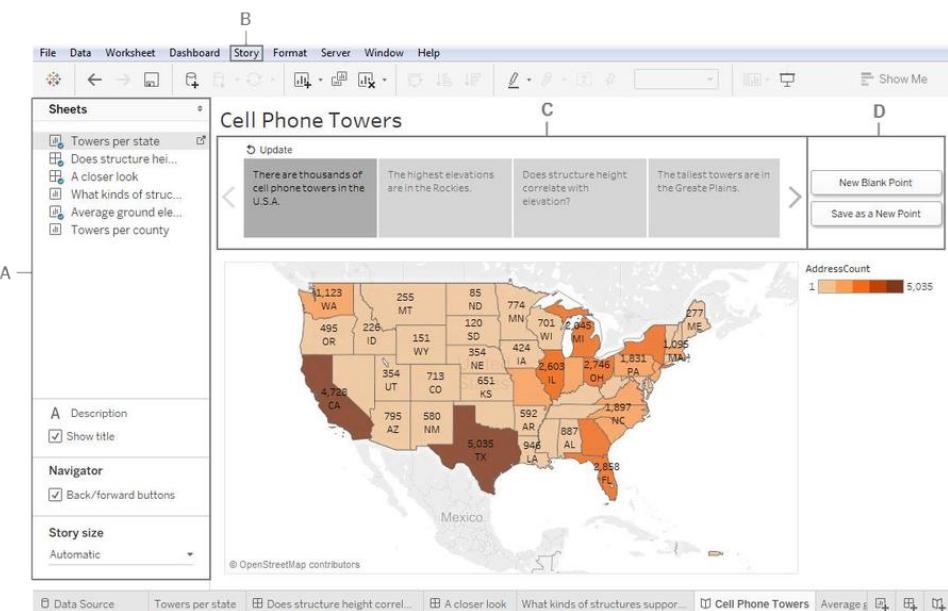
- **Use stories for collaborative analysis:** You can use stories to assemble a sequenced analysis, for yourself or for collaboration with your colleagues. Visualize the effects of data changes over time, or perform what-if analysis.
- **Use stories as presentation tools:** You can use stories to present a narrative to an audience. Just as dashboards provide spatial arrangements of views that work together,

stories present sequential arrangements of views or dashboards that create a narrative flow for your audience.

There are different ways to build a story. For example, each story point in a story can be based on a different worksheet or dashboard. Conversely, each story point can be based on a single worksheet or dashboard, which you customize for each story point, perhaps adding more information in each new story point. Often you will want to combine these approaches, using new sheets for some story points, and customizing the same sheet for other story points.

The Story Workspace

As you work on a story, you can use the following controls, elements, and features.



A. The Sheets pane

In the Sheets pane you can do the following.

- Drag dashboards and worksheets to your story.
- Add a description to a story point.
- Select to show or hide the navigator buttons.
- Configure the story size.
- Select to show the story title.

B. The story menu

When you select the Story menu, you can do the following.

- Open the Format Story pane.
- Copy the current story point as an image.
- Export the current story point as an image.
- Clear the entire story.
- Show or hide the navigator buttons and the story title.

C. The navigator

The navigator serves as a way to edit, organize, and annotate all the points in your story. You can also move through the story using the navigator buttons.

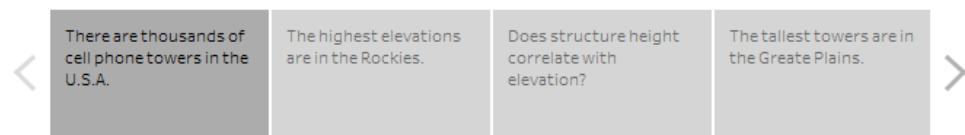
Navigator buttons

Click the forward arrow  on the right of the navigator to move forward one story point , and

the backward arrow  on the left of the navigator to move back one story point. You can also use the slider that appears when you hover over the navigator to quickly scroll through all the points in the story, and then select one to view or edit.

Story points

The current story point in the navigator highlights a different color to indicate it is selected.



When you add or make changes to a story point, you can choose to update the story point to save the changes, revert any changes, or delete the story point. For more information, see [Customize a story point on page 1208](#).



D. Options for adding a new story point

After you create a story point, you have several different options for adding another point. To add a new story point, you can do the following.

- Add a new blank point.
- Save the current story point as a new point.
- Duplicate the current story point.

For more information on these options, see [Create a Story](#) below.

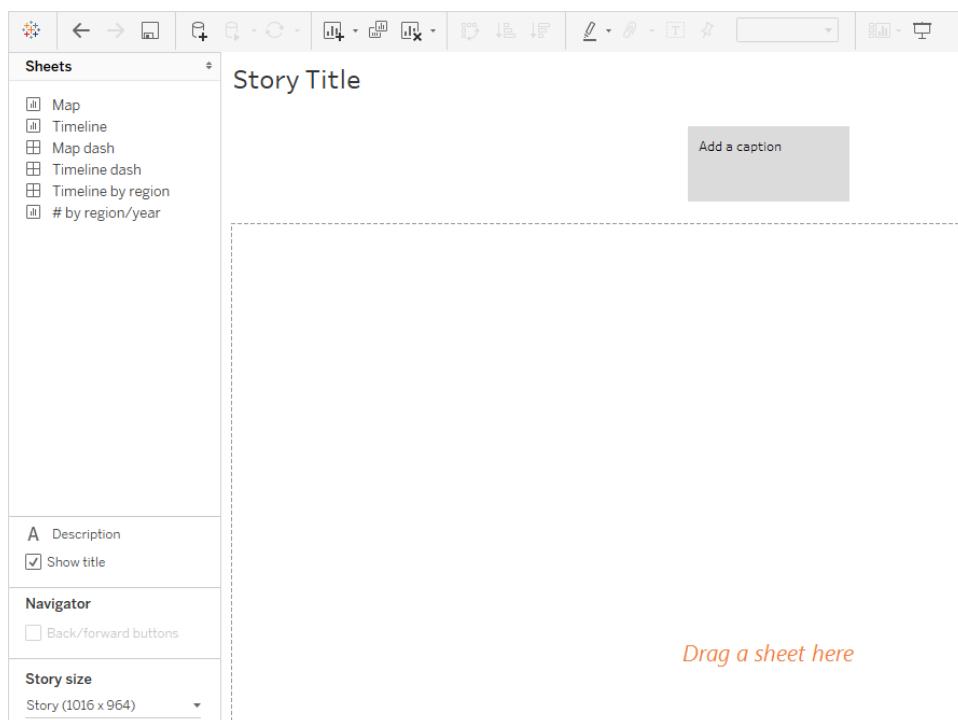
Create a Story

You create a story from existing worksheets and dashboards.

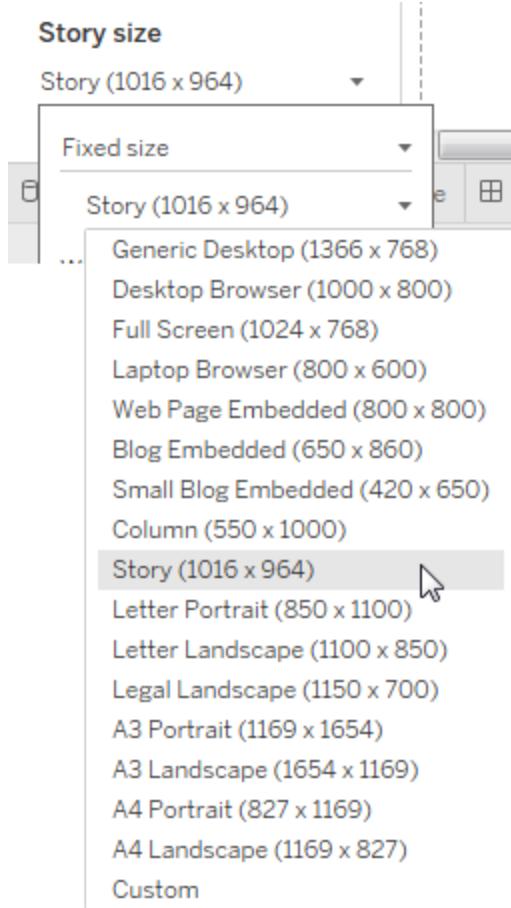
1. Click the **New Story** tab.



Tableau opens a new story as your starting point:

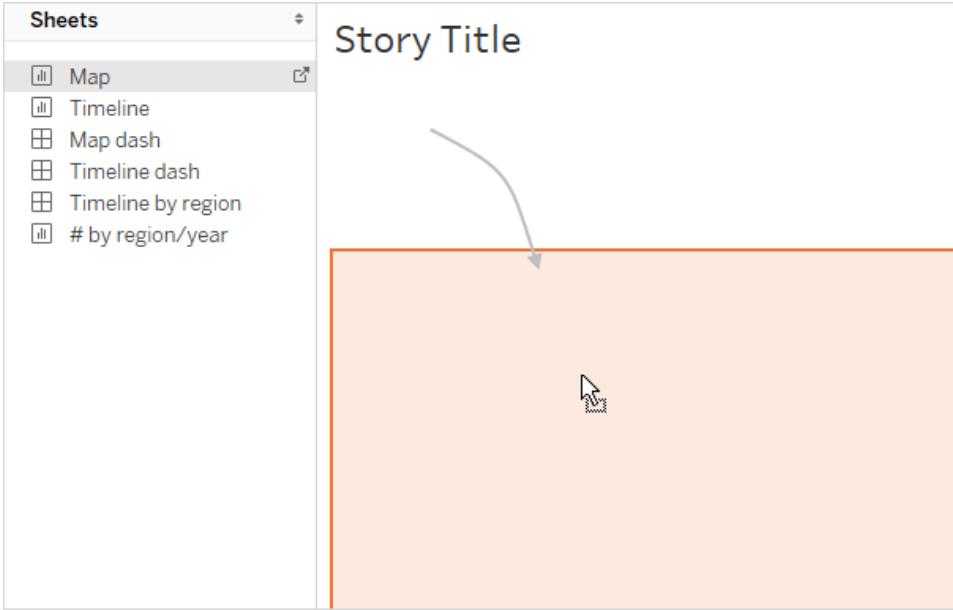


2. In the lower-left corner of the screen, choose a size for your story. Choose from one of the predefined sizes, or set a custom size, in pixels:



Note: Choose a size with your target platform in mind, rather than on the platform on which you are creating the story.

3. To add a title to your story, double-click **Story Title** to open the Edit Title dialog box. You can type your title in the dialog box, as well as choose a font, color, and alignment. Click **Apply** to view your changes.
4. Drag a sheet from the **Dashboards and Worksheets** area to the story, and drop it into the center of the view:



5. Click **Add a Caption** to summarize the story point.

Note: You can add descriptions and annotations within each story point if you want to provide more information.

6. Customize the story point. For more information, see [Customize a story point](#) on the next page.
7. Click **Update** above the navigator box to save your changes to the story point:



8. Add another story point. There are several way to add an additional story point:

- Click **New Blank Point** if you want to use a different sheet for the next story point.



- Click **Duplicate** if you want to use the current story point as the starting point for a new story point. Then customize the view or worksheet in the second story point to make it different in some way from the original.
- Click **Save as a New Point**. This option only appears when you begin customizing a story point. After you do so, the **Duplicate** button becomes the **Save as a New Point** button. Click **Save as a New Point** to save your customizations as a new story point. The original story point remains unchanged.

9. Continue adding story points until your story is complete.

Customize a story point

You can customize a story point in any of the following ways:

- By selecting a range of marks
- By filtering a field in the view
- By sorting on a field in the view
- By zooming in or panning on a map
- By adding a description box
- By adding annotations
- By changing the value of a parameter in the view
- By editing a dashboard text object
- By drilling down or up in a hierarchy in the view

After you modify a story point, do one of the following:

- Click **Update** to save your changes.
- Click the ⌂ (circular arrow) to revert the story point to its previous state.



Note: When you drag a sheet from the **Dashboards and Worksheets** pane to a story point, that sheet remains connected to the original sheet. If you modify the original sheet, your changes will automatically be reflected on the story points that use it. Changes you make in a story point do not automatically update the original sheet, however.

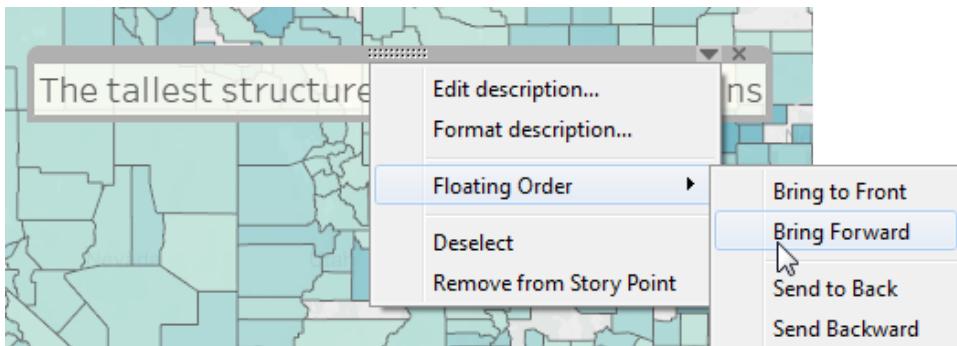
Add a description

You can add descriptions to a story point. To add a description, double-click **Description** in the **Sheets** pane. You can add as many descriptions to a story point as you want.

Note that descriptions:

- Do not attach to marks, points, or areas in the story point. You can position them wherever you like.
- Exist only on the story point where you add them. They do not affect the underlying sheet nor any other story point in the story.

After you add a description box, click it to select and position it. When you select a description box, you can open the menu by clicking the drop-down arrow:



Use the commands on this drop-down menu to edit the description, format the description text, set its floating order relative to any other description boxes it may overlap, deselect it, or remove it from the story point.

Format a Story

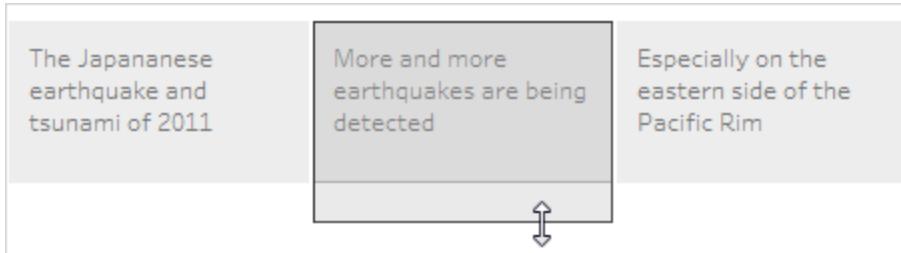
You can format a story in any of the following ways.

Resize captions

Sometimes the text in one or more of your captions is too long to fit inside the height of the navigator. In this case, you can re-size the captions vertically and horizontally.

1. In the navigator, select a caption.
2. Drag the border left or right to resize the caption horizontally, down to resize vertically, or select a corner and drag diagonally to resize the caption both horizontally and vertically.

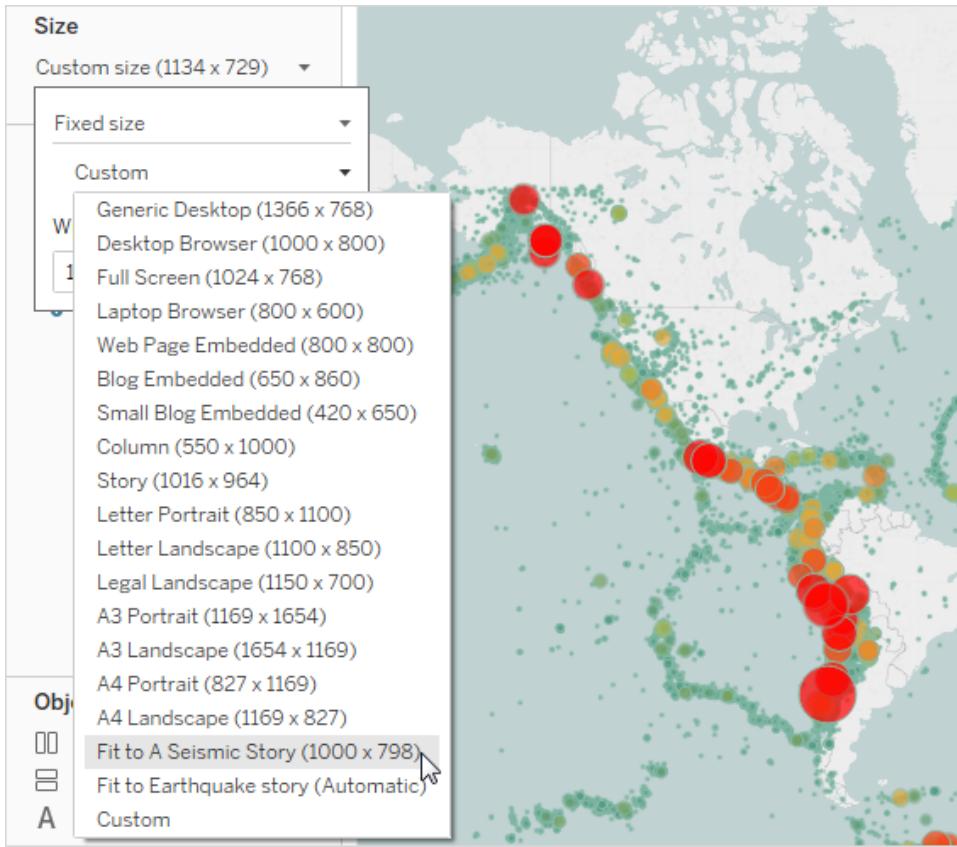
All captions in the navigator update to the new size.



Note: When resizing a caption, you can only select the left, right, or bottom border of the caption.

Fit a dashboard to a story

You can fit a dashboard to the exact size of a story. For example, if your story is exactly 800 by 600 pixels, you can shrink or expand a dashboard to fit inside that space. To fit a dashboard to a story, in the dashboard, click the **Size** drop-down menu and select the story you want the dashboard to fit inside.

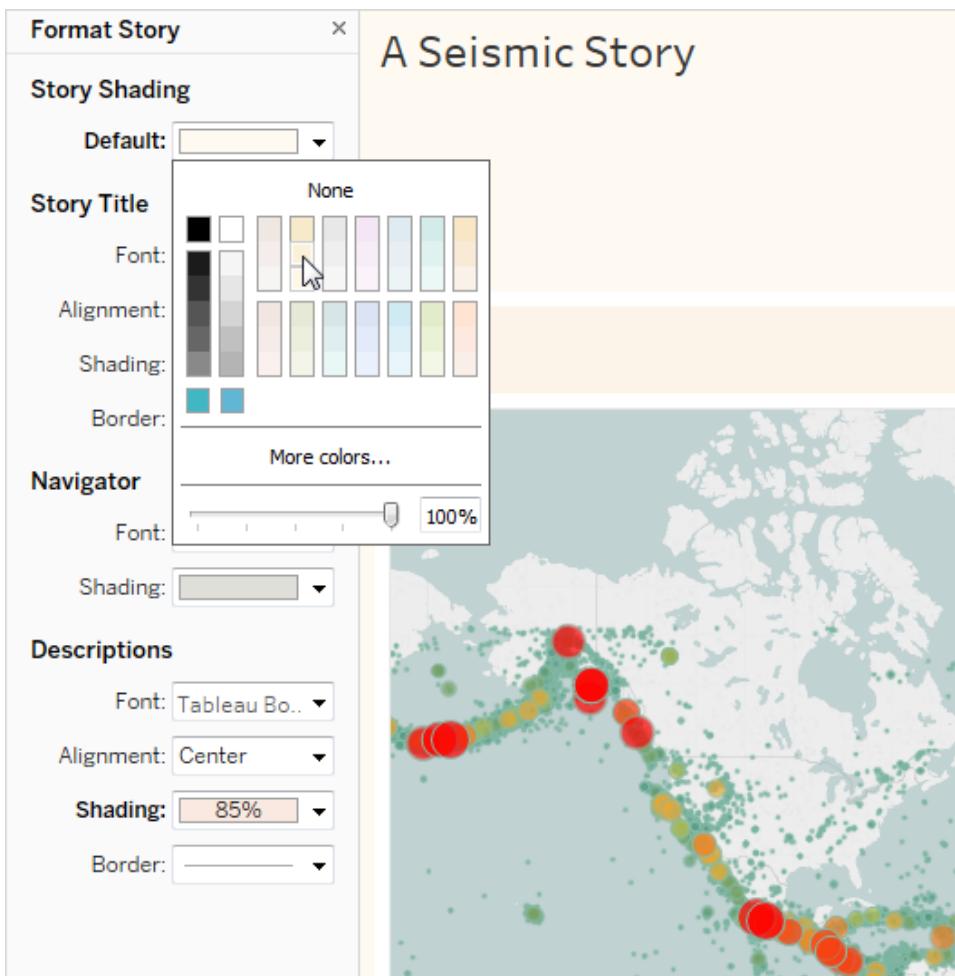


Format story pane

To open the **Format Story** pane, select **Format > Story**. In the **Format Story** pane, you can format any of the following parts of the story.

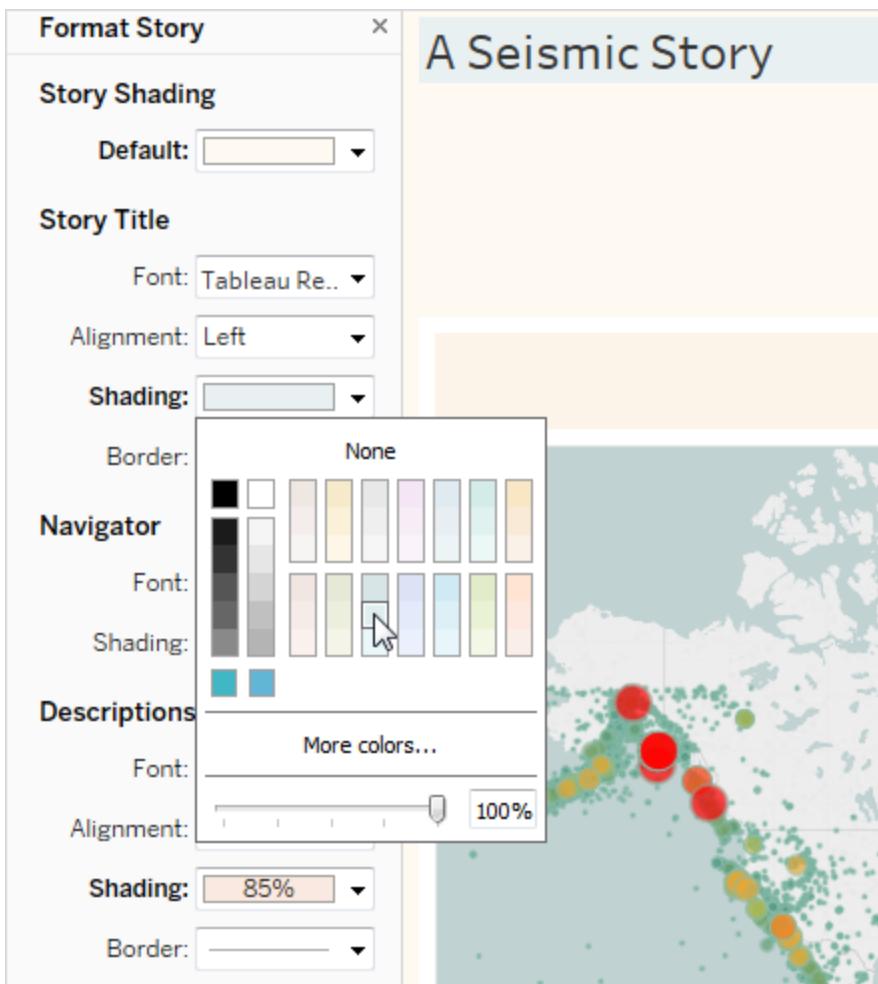
Story shading

To select a shading for your story, in the **Format Story** pane, click the **Story Shading** drop-down control. You can select a color and transparency for the story.



Story title

You can adjust the font, alignment, shading, and border of the story title. To format the title, click one of the drop-down controls in the **Story Title** section of the **Format Story** pane.

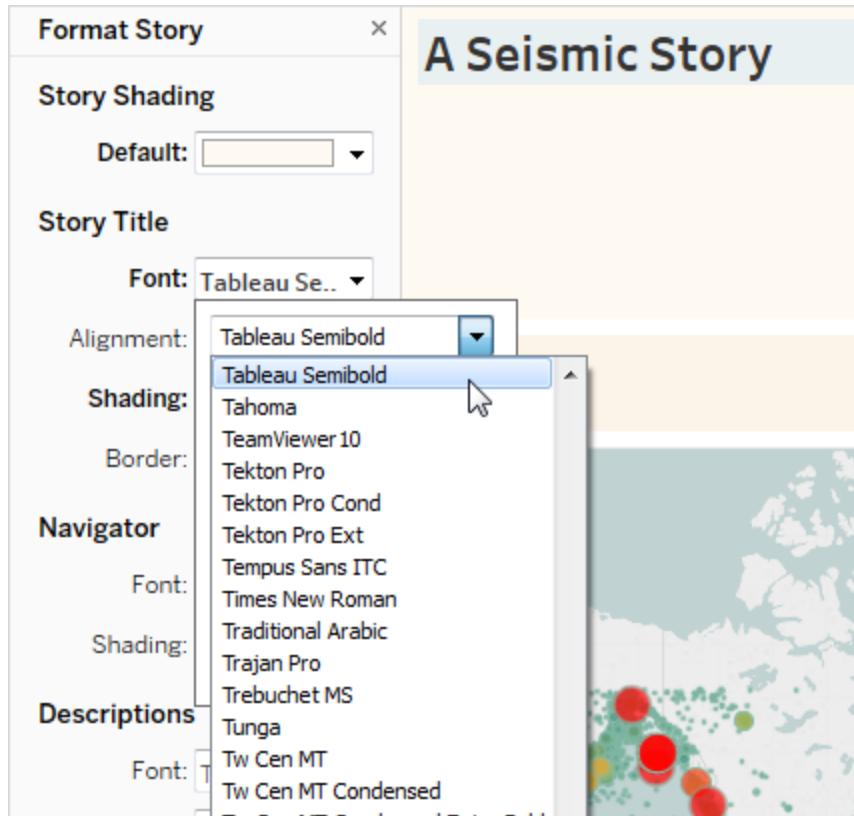


The navigator

You can adjust the font and shading of the navigator in the **Navigator** section of the **Format Story** pane.

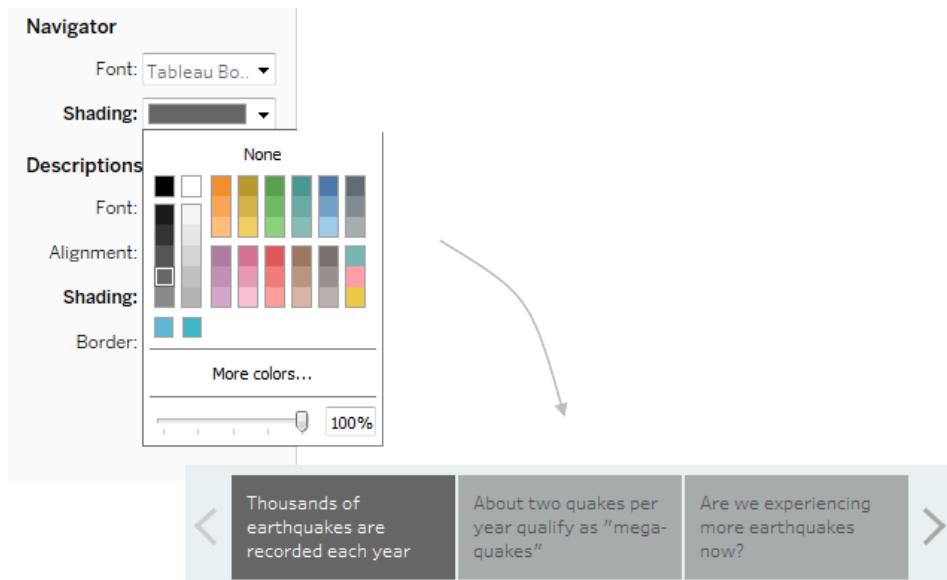
Font

To adjust the navigator font, click the **Font** drop-down control. You can adjust the style, size and color of the font.

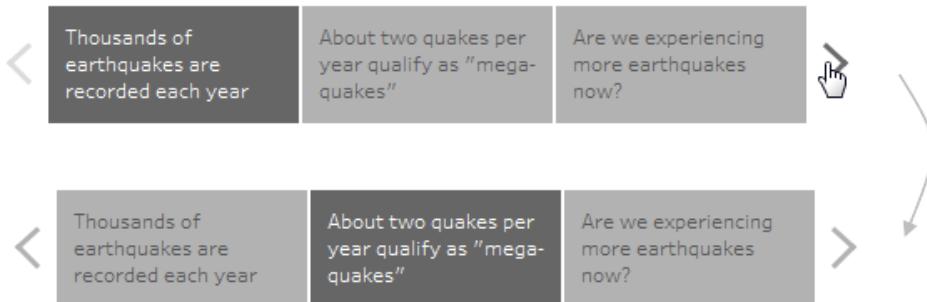


Shading

To select a shading for the navigator, click the **Shading** drop-down control. You can select a color and transparency for the navigator.



As you move through the navigator, the caption color and font updates to indicate the currently selected story point.



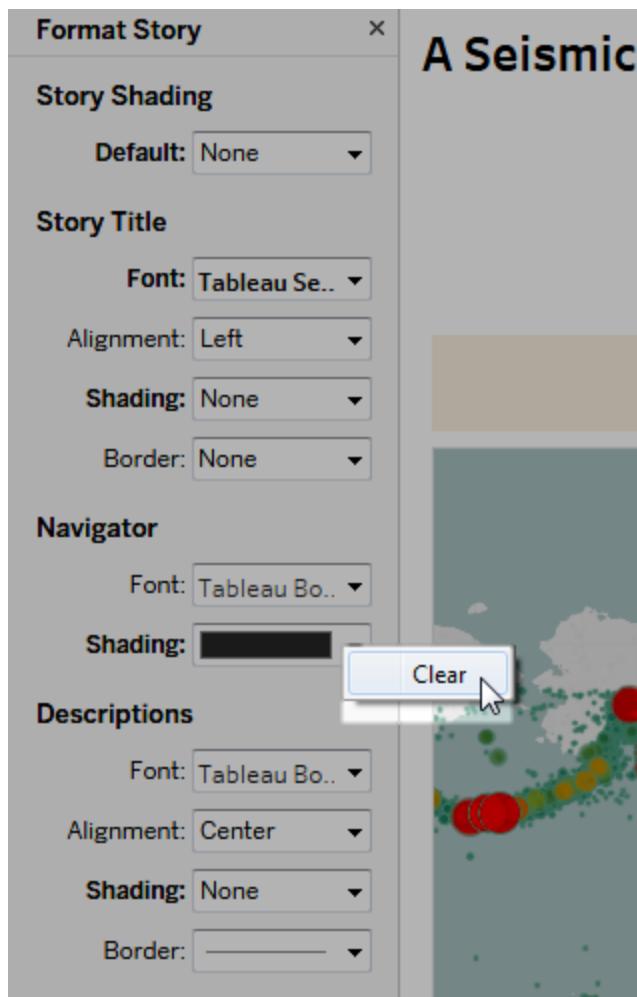
Descriptions

If your story contains any descriptions, you can format all the descriptions in the **Format Story** pane. You can adjust the font and add shading and border to the description.

Clear all formatting

To reset the story to the default format settings, click the **Clear** button at the bottom of the **Format Story** pane.

To clear a single format setting, right-click (control-click on Mac) the format setting you want to undo in the **Format Story** pane, and then select **Clear**. For example, if you want to clear the alignment of the story title, right-click (control-click on Mac) **Alignment** in the **Story Title** section, and then select **Clear**.



Update a Story

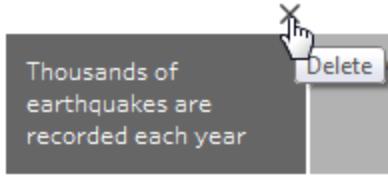
You can update a story in any of the following ways.

Revise an existing story point

To revise an existing story point, click it in the navigator, and then make changes. For more information, see [Customize a story point on page 1208](#). You can even replace the underlying sheet by dragging a different sheet from the **Sheets** area onto the story pane.

Delete a story point

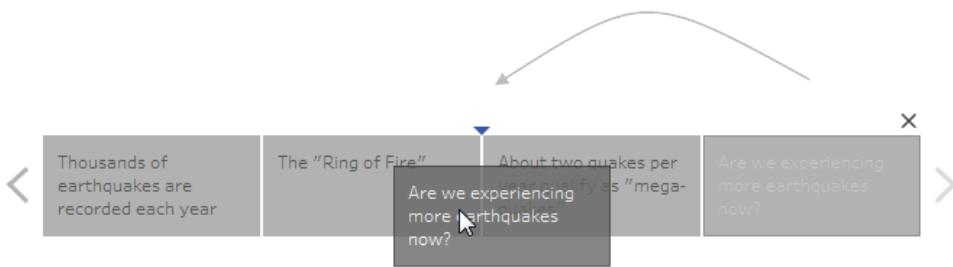
To delete a story point, click it in the navigator, and then click the (delete icon) just above the box.



If you delete a story point by accident, click the Undo button to restore it.

Insert a story point

To insert a new story point somewhere other than at the end of a story, add a story point, then drag it to the desired location in the navigator and drop it.



Alternatively, if you are dragging a sheet into the story, you can just drop it in the navigator between two existing story points.

Rearrange story points

You can use the navigator to drag and drop story points within a story as needed.

Present a Story

To present a story, use presentation mode. To switch in and out of presentation mode, click the

Presentation Mode button  on the toolbar. To exit Presentation Mode, press **Esc** or select the Presentation Mode button in the bottom right corner of the view. For more information on Presentation Mode, see [Reorganizing the Workspace on page 253](#).

You can also publish a workbook that contains a story to Tableau Server, Tableau Online, or save it to Tableau Public. For more information on publishing workbooks, see [Publish Data Sources and Workbooks on page 1225](#). Once you publish a story, the user can then open the story and navigate from story point to story point, or interact with stories just as they would interact with views and dashboards. Web users cannot, however, author stories, or permanently modify published stories.

Example – Earthquake Trend Story

This example shows how you can use Tableau to tell a story with data. The story described below considers the question: Are serious earthquakes becoming more common, or is it just that some strong earthquakes in recent years are creating that impression?

This story is available on Tableau Public:

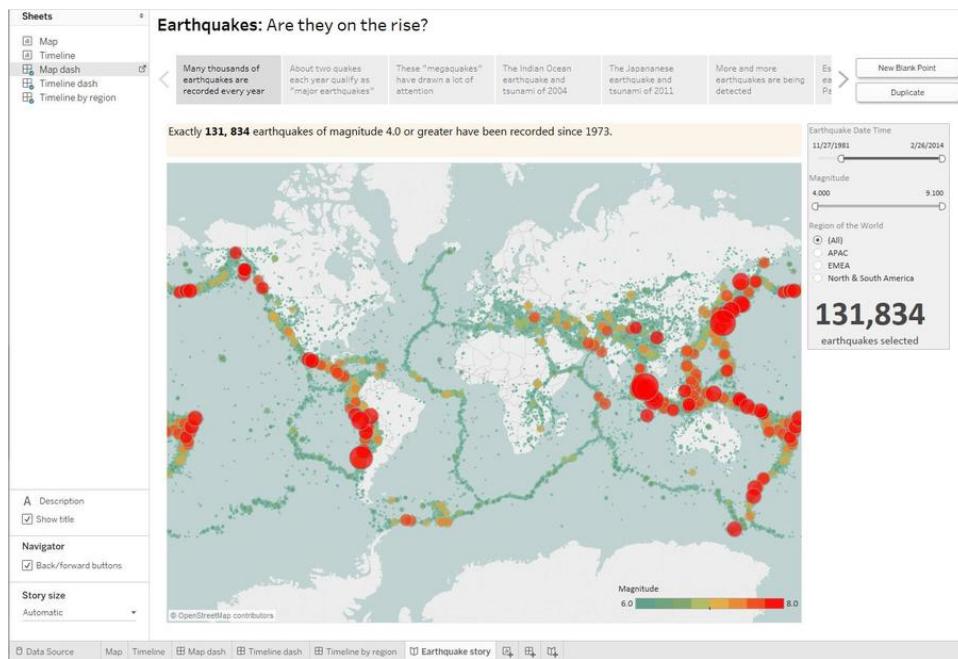
<https://public.tableau.com/profile/tableau.docs.team#!/vizhome/EarthquakeTrendStoryExample/Earthquakesstory>

The story was built from three simple dashboards:

- A map of earthquakes recorded around the world since 2004.
- A line chart showing the number of earthquakes recorded, by year, since 1973 (timeline).
- A different version of same timeline visualization, with earthquakes broken out by region, and with trend lines added.

Story point 1

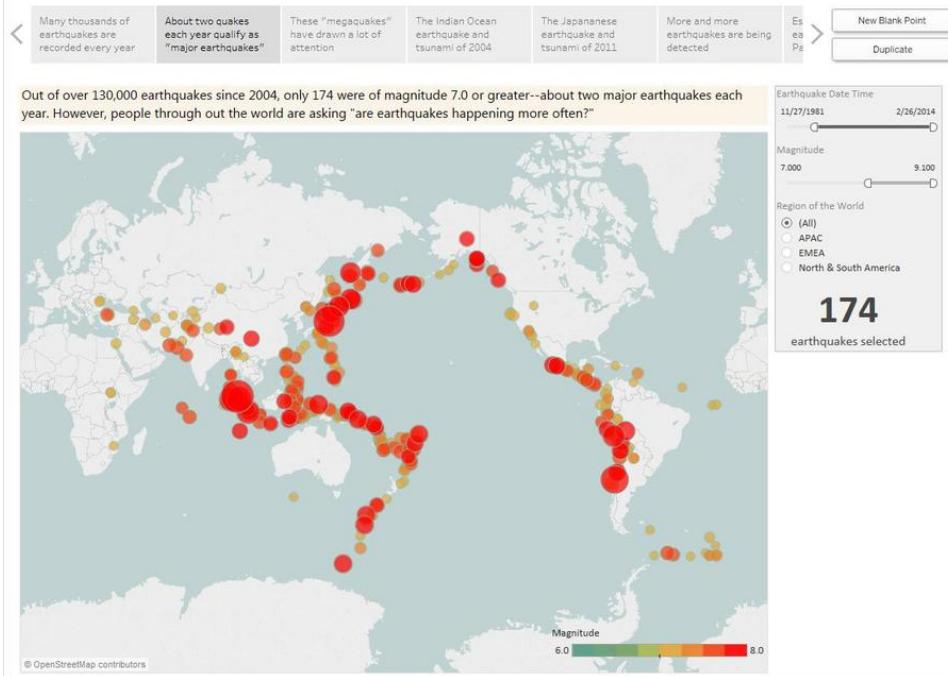
The first story point shows the map with all data included—all earthquakes, across the entire planet.



Story point 2

The second story point compresses the time frame and filters out smaller earthquakes—everything below magnitude 7. The map pans to show the Pacific "Ring of Fire," where the majority of the large earthquakes occurred.

Earthquakes: Are they on the rise?



Story point 3

In story point 3, the magnitude filter is adjusted to show only "megaquakes" (megathrust earthquakes of magnitude 8 or greater). The words work with the pictures to move the narrative forward.

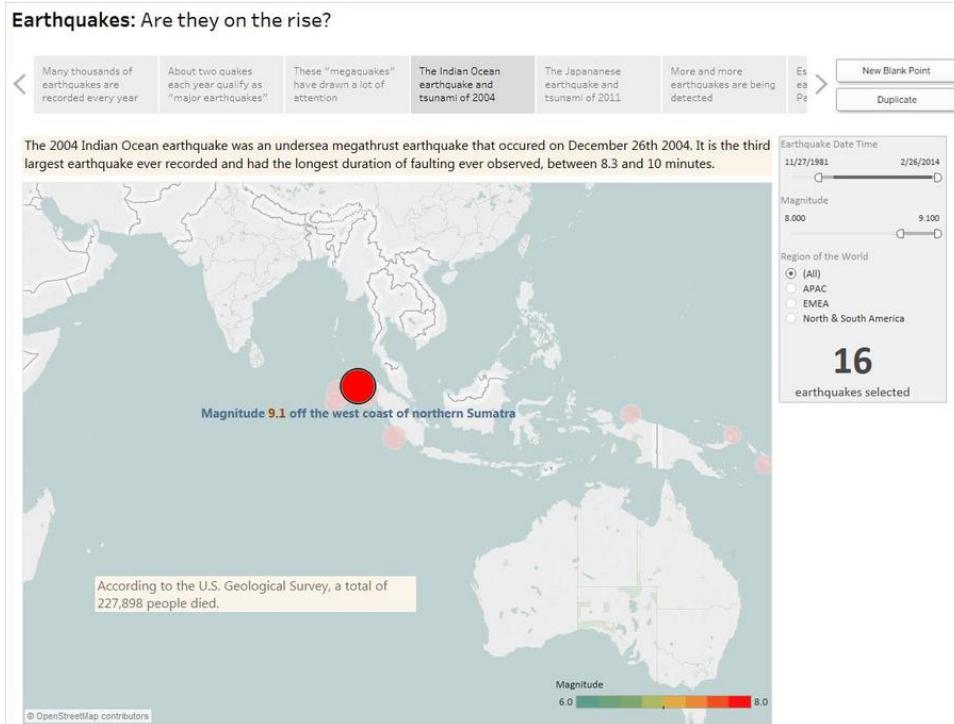


Story points 4 and 5

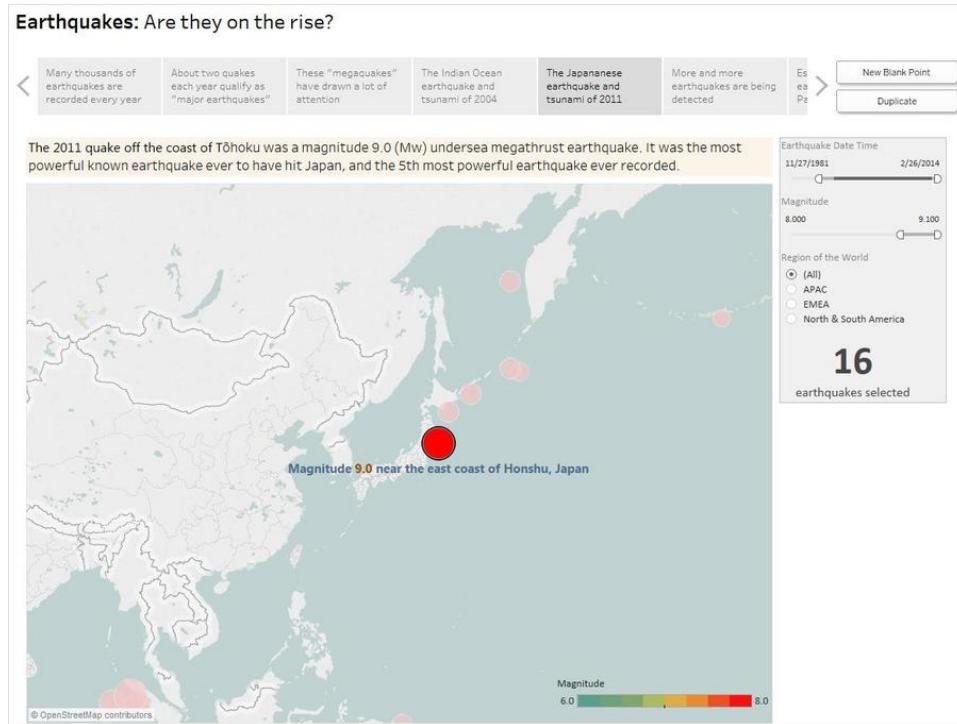
Story points 4 and 5 revisit the two most deadly earthquakes in recent history: the 2004 Indian Ocean earthquake and tsunami, and the 2001 Japanese earthquake and tsunami. You can view them below.

The first five story points have all used the same underlying dashboard—the author is creating a compelling visual story just by filtering the data and zooming and panning the map.

Story point 4

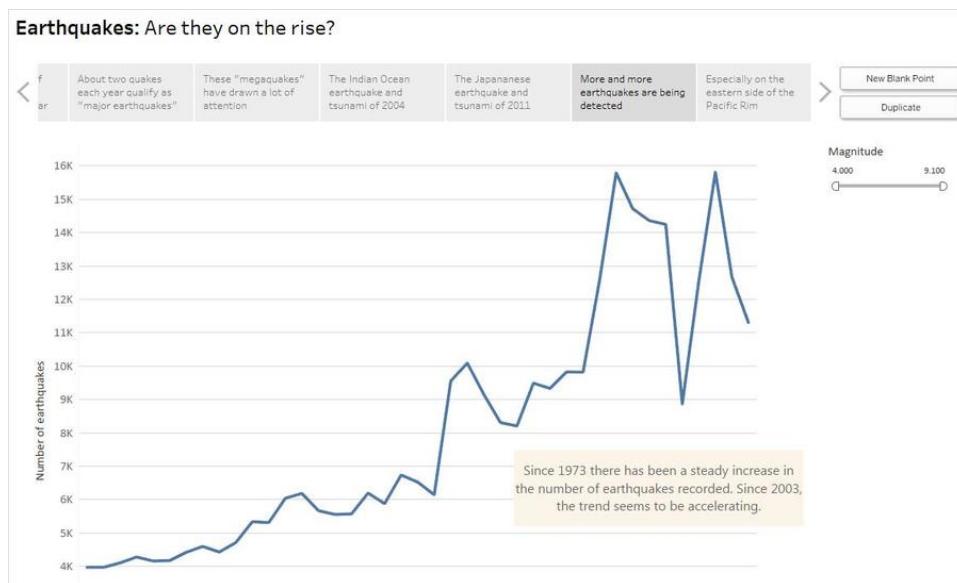


Story point 5



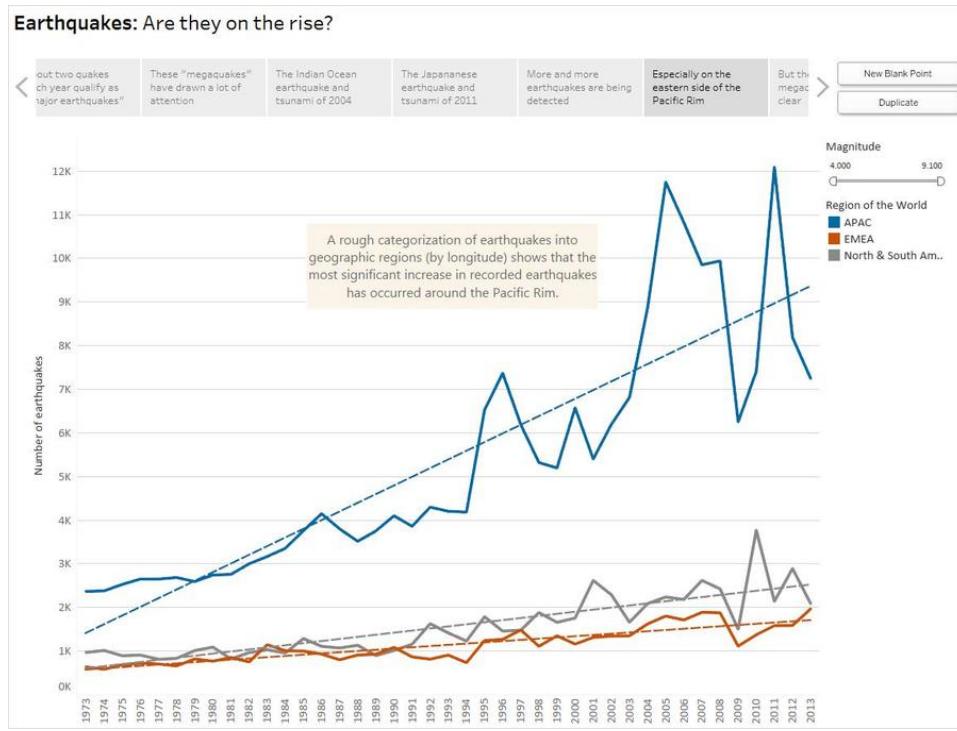
Story point 6

Story point 6 uses a line chart to show that more quakes are being reported over time since 1973—a three-fold increase.

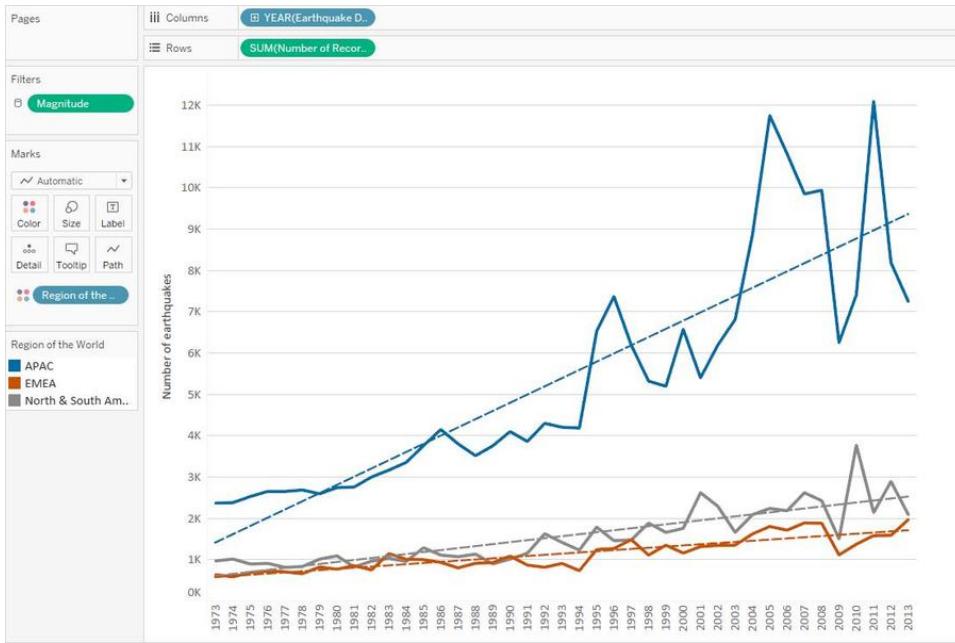


Story point 7

Story point 7 adds two kinds of detail to the simple line chart from story point 6: it breaks out earthquakes by region, and it adds trend lines, which reduce the variability in the data.

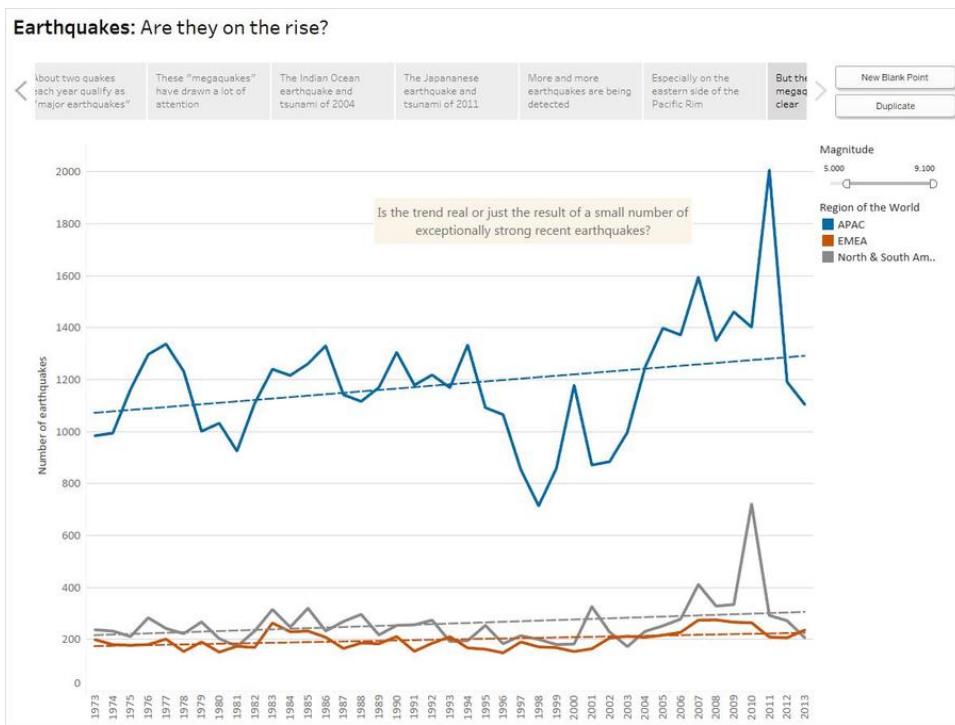


Specifically, the author has duplicated his original worksheet and then dragged region to color and added a trend line. Here is what the underlying worksheet looks like:



Story point 8

The final story point filters out weaker earthquakes and shows that what appeared to be a very clear trend is maybe not so clear. Yes, more earthquakes have been reported in recent years, especially in the Asia-Pacific region, but could that just be natural variation? Can we be confident that the trend will continue into the future?



Publish Data Sources and Workbooks

Suppose you create a view that exposes a new range of questions in the data you're using, and you want to share this analysis with other people using this data. Or maybe you are your team's Data Steward, in charge of building the data models approved for use by analysts, and meeting your organization's requirements for security, compliance, performance, and so on.

You can share your work with the rest of your team by publishing it to Tableau Server or Tableau Online. After it's published, you and your team can access it through your web browser or the Tableau mobile app. Publishing data sources can also help you to centralize data management.

In this topic

- [Why publish below](#)
- [What you can publish on the next page](#)
- [Who can publish on the next page](#)

In other resources

For the steps on how to publish, see either of the following topics:

- [Publish a Data Source on page 1234](#)
- [Publish a Workbook on page 1238](#)

Note: If you do not have Tableau Online or Server, you can share analysis on Tableau Public, a free cloud service. As the name suggests, views published to Tableau Public are publicly accessible. For more information, see public.tableau.com.

Why publish

You can publish data sources and workbooks when you want to widen the audience for your data analysis within your organization. By publishing you can begin to do the following:

- Collaborate and share with others
 - Allow people in your organization to view, interact with, download, subscribe to, share, edit, and save published views, even if they do not use Tableau Desktop. Incorporate views into blog posts or websites.
- Centralize data and database driver management
 - Create and publish data models that everyone can use. Centralized data management allows for sharing a single source for your Tableau data. All workbooks connected to the published data reflect updates to it.

In addition, when you publish and connect to data on the server, people connecting to the data from Tableau Desktop do not need to install and maintain database drivers on their own computers.

- Support mobility

Access your data from a different computer or location, through a web browser or the Tableau Mobile iOS app. Sign in to your organization's Tableau Server from a private network offsite.

What you can publish

Content types you can publish include:

- **Data sources:** You can publish data sources that others can use to build new workbooks. A data source can contain a direct (or live) connection to your database or an extract you can refresh on a schedule.
For information, see [Best Practices for Published Data Sources](#) below.
- **Workbooks:** Workbooks contain your views, dashboards, and stories, and data connection. You can include local resources, such as background images and custom geocoding, if they reside in a location that the server or other Tableau users cannot access.

Who can publish

To publish to Tableau Server or Tableau Online, your server or site administrator must grant you the following capabilities:

- A *site role* of **Publisher** on the site you're publishing to.
- **View** and **Save** capabilities set to **Allowed** on the project into which you publish.

If you use Tableau Desktop and are not sure whether you can publish to a server, or you are having trouble publishing, see your Tableau administrator. If you're an administrator, see [Content Access and Ownership](#) in the Tableau Server help (or the [Tableau Online version](#)) for more information about site roles and permissions.

Best Practices for Published Data Sources

Publishing data sources to Tableau Online or Tableau Server is integral to maintaining a single source for your data. Publishing also enables sharing data among colleagues; including those who don't use Tableau Desktop, but have permission to edit workbooks in the web editing environment.

Updates to a published data source flow to all connected workbooks, whether the workbooks themselves are published or not.

In this topic

- [What makes up a published data source](#) below
- [Preparing a data source for publishing](#) below
- [When to publish an extract](#) on the next page
- [Publishing data separately or embedded in workbooks](#) on page 1229
- [Keeping extracts up-to-date](#) on page 1230
- [Additional resources](#) on page 1230

What makes up a published data source

A Tableau data source consists of the following:

The data connection information that describes what data you want to bring in to Tableau for analysis. When you connect to the data in Tableau Desktop, you can create joins, including joins between tables from different data types. You can rename fields on the Data Source page to be more descriptive for the people who work with your published data source.

An extract, if you decide to create one. Guidelines for when to create an extract are included below, as well as in the additional resources.

Information about how to access or refresh the data. The connection also includes access information. Examples of this type of information include:

- The path to an original Excel file.
- Embedded credentials or OAuth access tokens for accessing the data directly.
- Alternatively, no credentials, so that users are prompted to enter them when they want to access the data (whether it's to view a workbook that connects to it, or to connect a new workbook to it).

For more information, see [Set Credentials for Accessing Your Published Data](#) on page 1248.

Customization and cleanup that helps you and others use the data source efficiently. When you're working with your view, you can add calculations, sets, groups, bins, and parameters; define any custom field formatting; hide unused fields; and so on.

All of these refinements become part of the metadata contained in the data source that you publish and maintain.

Preparing a data source for publishing

When you publish a data source, consider these best practices:

- Create the connection for the information you want to bring into Tableau and do any customization and cleanup that will help you and others use the data source efficiently.
- If appropriate, create an extract of the data you want to publish. For more information, see the following section, [When to publish an extract below](#).
- Develop a data source naming convention.

After publishing a data source you cannot rename it directly. Instead, you need to publish a new copy with the new name, and then update all workbook connections. A well-considered naming convention can also help other users of the data deduce which data source to connect to.

- Consider designating the following roles among your Tableau users:
 - A data steward (or team) who creates and publishes the data sources for the Tableau community, which meet your organization's data requirements.
 - A site administrator who manages published content, extract refreshes, and permissions on the server you publish to (Tableau Server or Tableau Online).

Central management helps to avoid data source proliferation. Authors who connect to managed data can be confident that the answers they find in it reflect the current state of the business.

When to publish an extract

Under the following conditions you might be required or choose to publish an extract instead of connecting live.

[Publishing data to Tableau Online that it cannot reach directly](#)

Tableau Online in the cloud cannot reach data sources that you maintain on your local network. For these data sources, you must publish an extract and set up a refresh schedule using the Tableau Online sync client.

Some cloud-hosted data sources always require extracts. These include Google Analytics, Salesforce.com, Oracle, OData, and some ODBC data sources. You can set up refresh schedules for some of these data sources directly on Tableau Online; for others you use the sync client.

Web data connector data sources always require extracts. If you connect to the data source using standard user name and password authentication, you can refresh it using the sync client. If you connect to the data source using OAuth authentication, you will need to republish the data source to refresh it.

For more information, see [Get your Data to Tableau Online](#) in the Tableau Online Help.

[Improving performance](#)

Even if the server supports live connections to your data, an extract might make more sense. For example, if the database is large or the connection slow, you can extract a subset that

includes only the pertinent information. The extract can be easier and faster to work with than connecting live.

In cases where you can use a live connection or an extract that you refresh on a schedule, you might want to experiment with both options to see which works best for you.

[Enabling functionality the data source does not inherently support](#)

For example, suppose you want to use the Median function with SQL Server data.

To learn more about creating data extracts, see [Extract Your Data](#) on page 440.

Publishing data separately or embedded in workbooks

You can publish data sources as standalone resources that workbooks connect to, or you can publish workbooks with the data sources included within them.

Each way of publishing has its advantages. The table below shows a few common points of comparison. It is not a comprehensive list, and these are generalizations. How these and other factors apply to you are specific to your environment.

Published separately	Embedded in workbook
Publishing data sources centralizes data management, enables policies around “certified” data and governance, and can help to minimize data source proliferation.	Each embedded data source has a disparate connection to the data. Each has the potential to show something different than the other at any given time (and data source proliferation is common).
Meant to be shared; becomes available for other Tableau users to connect to.	Data is available only inside the workbook; it is not available for other Tableau Desktop users to connect to.
Extracts can be refreshed on a schedule. You set up one refresh schedule for the extract, and all workbooks that connect to it always show the most current data.	Embedded extracts that aren’t refreshed can be useful for showing snapshots in time. If you want to keep the data fresh, each workbook must have its own refresh schedule.
Generally helps you to optimize performance on the server or site.	Performance might be affected when the server contains multiple workbooks that connect to the same original data, and each workbook has its own refresh schedule.

Keeping extracts up-to-date

When you publish a data source with an extract, you can refresh it on a schedule. The way you schedule refreshes depends on the data source type and whether you're publishing to Tableau Server or Tableau Online.

For more information, see the following topics:

- [Keep Data Fresh](#) in the Tableau Online Help
- [Scheduled Refresh Tasks and Subscriptions](#) in the Tableau Server Help

Additional resources

- [Data Server](#)—Training video by Tableau, with a helpful overview of data sources and publishing.
- [Understanding Tableau Data Extracts](#)

A version-agnostic, three-part series by Gordon Rose on the Tableau blog. It includes an in-depth look at the extract's file structure, guidelines for when to use extracts, and best practices.

- [O Extract, Where Art Thou?](#) and [TDE or Live? When to Use Tableau Data Extracts \(or not\)](#)

Posts by Tableau Zen Master Jonathan Drummond on his blog Drawing with Numbers. Includes tips on extracts, explains the different file types, describes different publishing scenarios. (Read the comments, too.)

- [Tableau Extracts—What / Why / How etc](#)

From the blog maintained by The Information Lab, a Tableau Gold Partner.

Disclaimer: Although we make every effort to ensure these links to external websites are accurate, up to date, and relevant, Tableau cannot take responsibility for the accuracy or freshness of pages maintained by external providers. Contact the external site for answers to questions regarding its content.

Prepare for Publishing a Workbook

When you publish a workbook, you need to make decisions that determine how others will access the views and the data behind them. You also need to take into account the experience you and others will have of interacting with the views on the server. Before you publish your workbook, use the information in this topic to help you make these decisions and improve the workbook's performance.

Note: If you have a Tableau site administrator, check with them on whether your organization already has publishing guidelines that have already answered these questions.

In this topic

- **What makes up a typical published workbook below**
- **Decide how to access the data and keep it up-to-date** on the next page
- **Assess the workbook performance and ease of use** on page 1233

What makes up a typical published workbook

During the publishing process, you select settings that determine how the pieces that make up the workbook interact with each other. The following image shows a simplified overview of a workbook's structure.



- A. The underlying (original) data that you connected to when you created the workbook. When you publish, you specify whether to include credentials for accessing this data, or whether to require users to enter credentials to access it.
- B. The Tableau data source. This is where the action is. It contains the XML metadata that describes how to access the underlying data (A), field customizations or calculations you made in Tableau, when to refresh the extract if there is one.
In this image, the data is embedded in the workbook, and it contains an extract connection and refresh schedule, some calculations, and so on.
- C. A view showing data from B, which you want to make available for your colleagues to edit

or interact with on the server. When you publish, you can select the sheets—which include views, dashboards, and stories—you want to share.

Decide how to access the data and keep it up-to-date

During the publishing steps, you need to answer the following questions about the data connection. For help with the answers, see **Best Practices for Published Data Sources** on page 1226.

- Will you publish with a live connection to the data, or will you create an extract?
- Do you want to publish the workbook's connections as separate, standalone Tableau data sources (and then connect the workbook to the published data); or embed the data into the workbook?

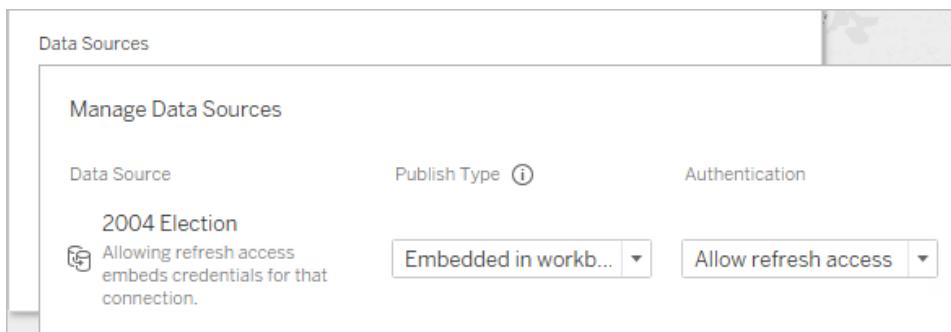
Tip: If the workbook already connects to a Tableau data source, you keep the existing connection. That means you've just answered this question.

- Do you want the workbook to show the data when other users open the workbook, or do you want to require them to provide database credentials? (Either way, they must also have access to the site and project you publish to.)

This can be complicated depending on the data the workbook connects to. In many cases, **the simple path to data** described below works great. If it doesn't work for you (or you're not sure), see **Set Credentials for Accessing Your Published Data** on page 1248.

The simple path to data authorization

When you go through the publishing steps, you specify how your workbook users will access the data the workbook connects to. You do this in the **Data Sources** section of the Publish Workbook dialog box.



Depending on the connection type, the simple path might work for your environment. You would do one of the following:

For a connection to a Tableau data source: Keep the existing connection and embed the password.

For other data connections:

1. Before you open the Publish Workbook dialog box, create an extract and include only the data you need for the workbook.
2. In the Publish Workbook dialog box, embed the extract in the workbook and select **Allow refresh access**. The latter embeds the credentials in the connection.
3. After you specify remaining settings and click Publish, set up a refresh schedule.

For the complete steps, see [Publish a Workbook](#) on page 1238.

Scenarios where data authorization requires some extra effort

The simple path doesn't work best for every situation. Some reasons you might need to divert from it include:

- Instead of embedding data into workbooks, your existing policies mandate publishing data sources separately and connecting workbooks to the Tableau published data sources.

This requires a few more steps than embedding the data; however, where there is a choice between the two, publishing data separately and managing it on the server is considered a better practice.

- You added a user filter to enforce row-level security. In this scenario, publishing live connections is more common, and other steps are required to secure the filter.
- You're publishing to Tableau Server, and you use SAP HANA or Impala single sign-on, or your workbook connects to data that can use impersonation-based authorization (SQL Server or Kerberos-enabled data).
- Other reasons that have to do with your organization's data security or authorization policies, which your IT staff can help you with.

Even if whatever you decide turns out to be not the best path, you can easily correct it by republishing. In some cases, your site administrator can change the settings on the server directly.

Assess the workbook performance and ease of use

Keep performance and interaction best practices in mind as you build workbooks you want to publish. If changes take a long time to display while you're working in Tableau Desktop, they will take as long or longer to display on the server. Simple steps you can take that can have a big impact include limiting the number of marks you add to a view, limiting the number of views you add to a dashboard, and removing unused fields from your data source.

For additional tips, see the following topics:

- [Make Visualizations Faster](#) on page 1827
- [Design for Performance While You Build a View](#) on page 1821

- [Designing Efficient Workbooks](#), starting with “Writing Efficient Workbooks,” on page 6.

This is a white paper by Tableau Sales Consultant Alan Eldridge. Viewing it requires signing in to the Tableau website using a free Tableau ID. This is the same ID you use for the Tableau community forums or watching our training videos.

Publish a Data Source

When you are ready to make a data source available to other Tableau Desktop users, you can publish it to Tableau Server or Tableau Online.

Note: If you haven't yet read about best practices for creating data sources and when to create an extract, see [Best Practices for Published Data Sources](#) on page 1226.

In this topic

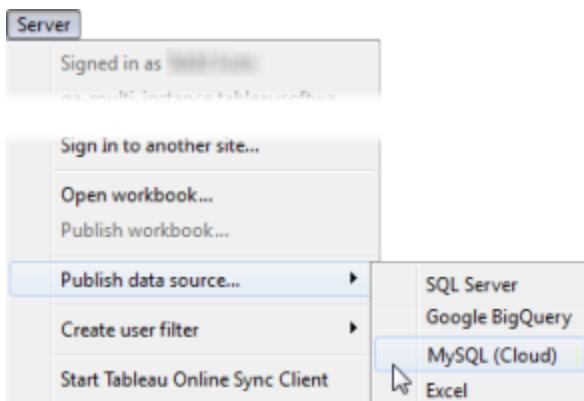
- [Publishing steps below](#)
- [Publishing with a Web Data Connector](#) on page 1237
- [Using hidden fields in workbooks](#) on page 1237

Publishing steps

The following steps give an overview of the publishing flow you will use regardless of the type of data or the server you publish to. After these steps you can find supplemental information for authentication types and using the Tableau Online sync client to refresh extracts of local data sources.

1. Select **Server > Publish Data Source**.

If your workbook is connected to multiple data sources, select the one you want from the submenu.



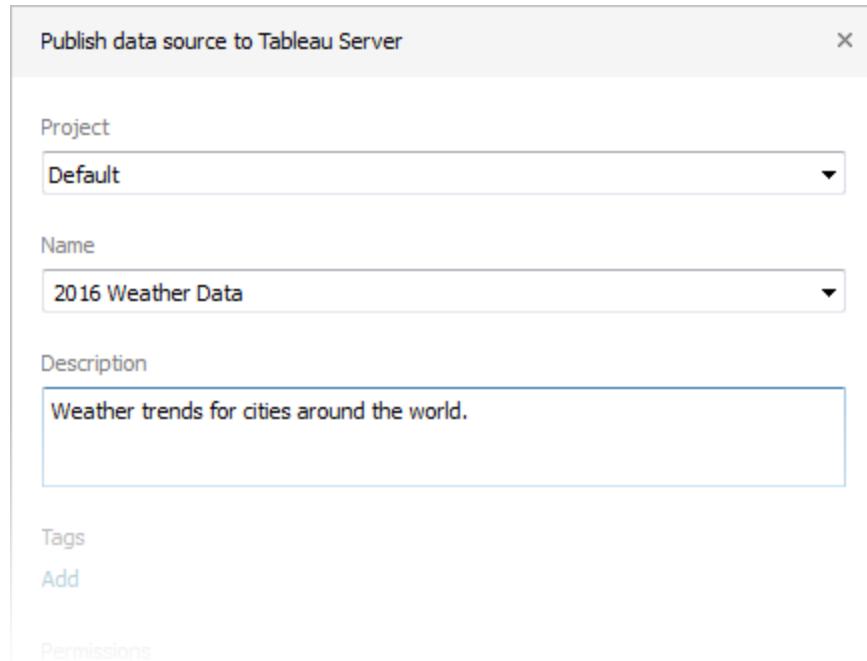
2. If you're not already signed in to Tableau Server or Tableau Online, sign in now.

How you sign in depends on how your administrator set up your environment. For information, see [Sign in to Tableau Server or Online](#) on page 1242.

3. In the **Publish data source to Tableau Server** dialog box, do the following:

- Select the project you want to publish to and enter the data source name.
- Add a description and tags that will help you and other users find it.

Separate tags using either a comma or a space. To add a tag that contains a space, put it in quotation marks (e.g., “Sales Quotes”).

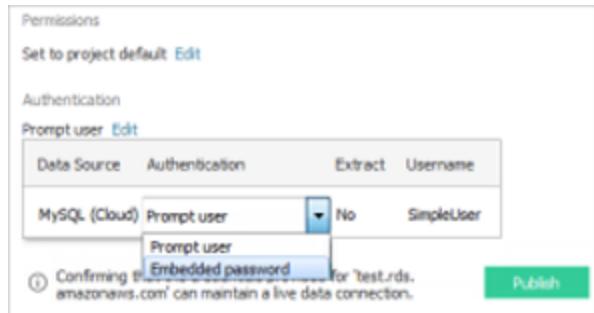


- Under **Permissions**, select **Edit** to change permissions indicated.

Note: Do this only if you are confident that they need to be something different than what is shown, which is the default setting on the server.

For more information, see [Set Permissions as You Publish a Data Source or Workbook](#) on page 1244.

- For **Authentication**, if you need to provide credentials to access your data, you can specify how authentication should be handled when the data is published to the server.



The options available for accessing the data source depend on the type of data you publish and whether you are publishing to Tableau Server or Tableau Online.

Information appears at the bottom of the dialog box to let you know whether you need to take further action, such as adding Tableau Online to your data provider's authorized list.

For information about the authentication types, see [Set Credentials for Accessing Your Published Data](#) on page 1248.

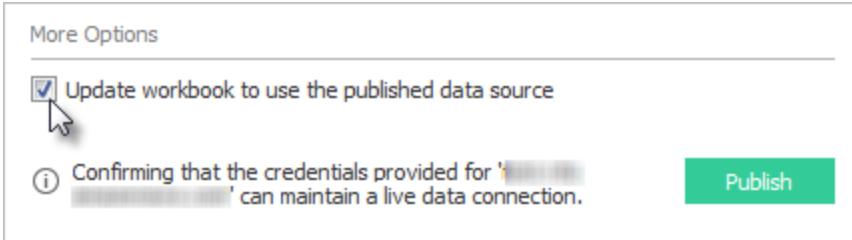
4. If you are publishing file-based data that is on a Windows mapped drive, or using images that will not be available from the server, select **Include External Files**.

When you include external files, copies of the files are put on the server as part of your data source. Copies of files are also put on the server and included as part of the data source when you publish extracts of multi-connection data sources that contain a connection to file-based data such as Excel. For more information about the implications of publishing extracts of multi-connection data sources, see [Extracts of multi-connection data sources that contain connections to file-based data](#) on page 362.

If you do not want to publish the external files to the server, change the connection information so that the data source references a full UNC path. For example, rather than connecting to D:\datasource.xls, you would connect to \\filesrv\datasource.xls.

5. By default, during the publishing process, Tableau updates the workbook connection to use the new published data source. It also closes the local data source.

To continue using the local data source instead, clear the **Update workbook to use the published data source** check box.



For more information, see [Quick Start: Automatically Update Workbook to Use Published Data Source](#) on page 1650

Note: If you select **Undo** after publishing the data source, Tableau will revert to using the local data source, but the data source will remain published. In addition, Tableau does not replace a local data source when you publish a cube (multidimensional) data source to an on-premises Tableau Server. (Tableau Online does not support publishing cube data sources.)

6. Click **Publish**.
7. (Optional) Set up a refresh schedule on the server.

After you click **Publish**, you are redirected to the site you published to, open in the web browser. There, you are prompted to set up the refresh schedule. If you published an extract of a data source to Tableau Online that Tableau Online cannot reach directly, such as a SQL Server data source you maintain on your local network, you schedule refreshes using the sync client.

For more information about refreshes and schedules, see any of the following topics:

- [Schedule Refreshes on Tableau Server](#)
- [Schedule Refreshes on Tableau Online](#)
- [Schedule Refreshes through the Sync Client](#)

Publishing with a Web Data Connector

To publish a web data connector data source, you need to *import* the web data connector to the server before you can set up a refresh schedule. You can do this only on Tableau Server.

To refresh web data connector data sources on Tableau Online, you need to use the sync client.

For information, see [Web Data Connectors in Tableau Server](#) in the Tableau Server Help or [Refreshing Data Using the Sync Client](#) in the Tableau Online Help.

Using hidden fields in workbooks

Workbooks connected to a published data source respect the state of hidden fields in the published the data source.

- If you create a new workbook that uses a published data source with hidden fields, those fields remain hidden in the workbook and cannot be used in calculations, sets, groups, and other object creation.
- If you work with an existing workbook that uses a published data source with hidden fields, those hidden fields are displayed in red in the workbook to indicate that the fields, and therefore the views and calculations that use those fields, are invalid.

You can address this issue in one of the following ways:

- Show (unhide) the relevant fields in the data source, and then republish the data source.
- Update the relevant workbooks to exclude the hidden fields.

For information, see [Hide or Unhide Fields on page 313](#).

See also

- [Allow Direct Connections to Data Hosted on a Cloud Platform](#) (Tableau Online)
- [Keep Data Fresh](#) (Tableau Online)
- [Data Sources](#) (Tableau Server)

Publish a Workbook

The steps below describe the general publishing process. Use the information you gathered while reviewing [Prepare for Publishing a Workbook on page 1230](#) to adapt these steps for your workbook and data.

If this is your first time publishing, you might publish as a test first, and work out any glitches before you let other users know the workbook is ready for them.

1. In Tableau Desktop, open the workbook you want to publish, or create a new one and add some data to the view.
2. Select **Server > Publish Workbook**.

If the **Publish Workbook** option does not appear on the **Server** menu, make sure a worksheet or dashboard tab is active (not the Data Source tab).

If necessary, complete the process to sign in to the server. For Tableau Online, enter <https://online.tableau.com>. For more information, see [Sign in to Tableau Server or Online on page 1242](#).

3. In the **Publish Workbook** dialog box, select the project, enter a name for the workbook, and add search tags.

[Learn more](#)

Projects are containers for workbooks and data sources. Permissions set on projects determine who can access your workbook, so knowing which project to publish to is

important.

For the workbook name, enter a unique name unless you want to create a new revision of an existing workbook you own.

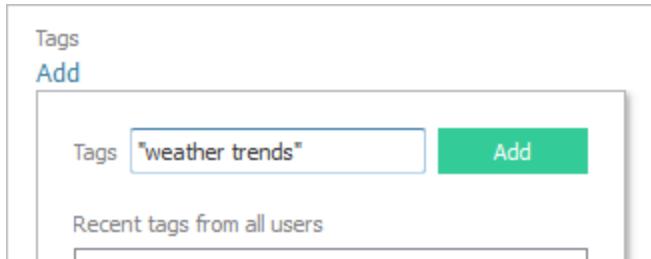
Note: If you're publishing to Tableau Server, and your administrator has turned off revision history, using an existing name overwrites the workbook. Revision history is enabled for all Tableau Online sites.

A good practice is to create a workbook naming convention that helps others find workbooks easily and know what each contains.

Add metadata keyword tags

Enter keywords that describe the workbook. Tags help users find related workbooks when they browse the server.

Separate tags using a comma or space. To add a tag that contains a space, put the tag in quotation marks.



4. For **Permissions**, accept the default project settings.

Generally a site administrator manages permissions on the server. If you think your workbook is an exception, work with your administrator to determine the best course of action, and see [Set Permissions as You Publish a Data Source or Workbook on page 1244](#).

5. For **Data Sources**, select **Edit** if you want to change the method shown for how people to access the underlying data your workbook connects to, or for how to publish the data (embedded in workbook or published separately).

- If you're publishing an extract, and you want to set up a refresh schedule, you must select **Embed password** or **Allow refresh access**, depending on whether your underlying data is on-premises or in the cloud.
- If your workbook connects to a Tableau data source, we recommend embedding the password. This is the default option if you are publishing one of the workbook's connections separately. Selecting Prompt users requires additional permissions on the Tableau data source.

6. Configure other settings that are available for this workbook.

For more information, see [Variable publishing options](#) below.

7. Click **Publish**.
8. (Optional) Set up a refresh schedule for each extract you published.

The publishing workflow guides you through these steps. For extracts of some data types you publish to Tableau Online, the publishing process starts the Tableau Online sync client on your computer.

For more information, see [Schedule Extract Refreshes as You Publish a Workbook](#) on page 1251.

Variable publishing options

The following sections describe publishing options that appear when they're appropriate for the workbook.

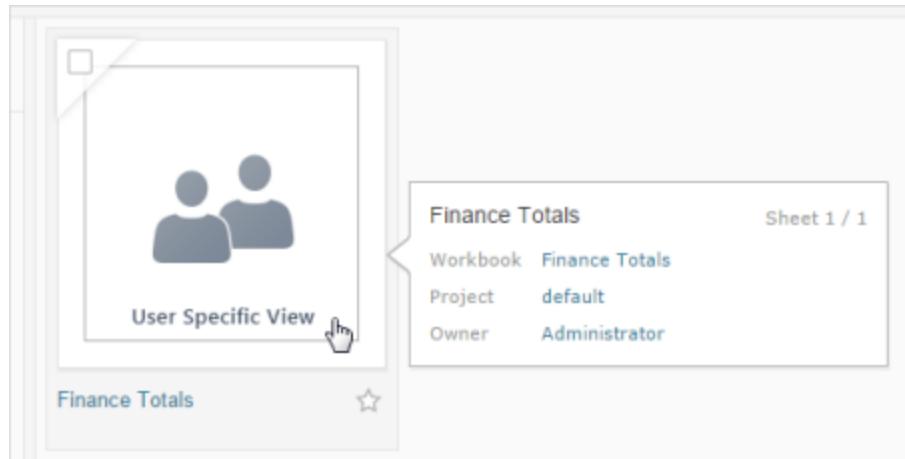
Specify how to generate thumbnails for workbooks with user filters

On the server where you publish, one of the options for browsing content is by thumbnail view. In this view, thumbnails are generated based on the workbook and its sheets. If your workbook contains user filter, you can specify which user's filter to use for creating the thumbnails.

For example, if you want the thumbnail image to show all regions of a sales forecast, you can generate thumbnails per the user who is allowed to see all regions.

In the following scenarios, a generic image appears in place of the view thumbnail.

- The user you select does not have permission to see the data.
- The data is from a Tableau Server data source that was published with data source filters, user calculations, impersonation, or other user reference.



To learn more about user filters, see [Restrict Access at the Data Row Level](#) on page 1252.

Show or hide sheets when you publish

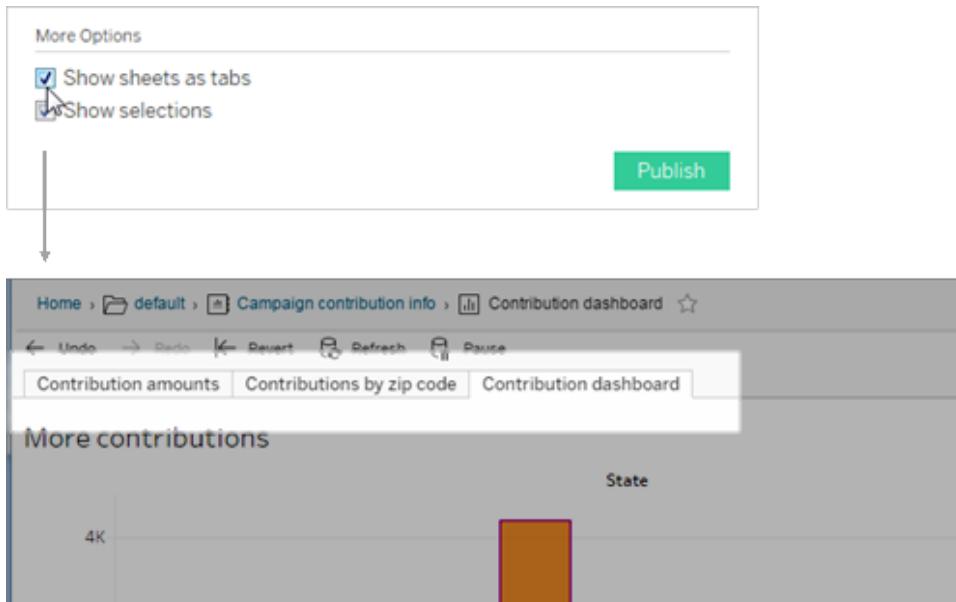
By default, Tableau Desktop publishes all sheets in a multiple-sheet workbook. In the **Sheets** section of the Publish Workbook dialog box, you can specify which sheets to include. Hiding sheets is useful when you want to publish a dashboard or story without showing the worksheets that were used to create it.

Important: Hiding sheets is not a security measure. Anyone who has the **Download/Web Save As** capability can access the hidden sheets. Other editing permissions can also allow access to hidden sheets. For more information, see [Content Access and Ownership](#) in the Tableau Server Help.

Show sheets as tabs

If you select multiple sheets to show, you can specify how users navigate them.

- Select the **Show Sheets as Tabs** check box to provide tab-based navigation.



- Clear the check box to allow people to open only one view at a time.

Show selections

You can select this if you want a particular portion of the view to be highlighted when others open the workbook. Make your selections in Tableau Desktop before you start the publishing process.

Include external files

If your views contain any information that isn't available to the server, or to other users who access the workbook on the server, the Publish Workbook dialog box will include a setting you

can select to include that information when you publish.

For example, you might use an Excel, CSV, or other data source local to your computer; image files; and so on.

If you are publishing to Tableau Server, and the workbook references data sources or images on a mapped drive, you can include external files when you publish, or you can change the connection information so that the workbook references the UNC path to the data source. For example, you could change **D:\datasource.xls** to **\filesvr\datasource.xls**.

If you are publishing to Tableau Online, and the workbook connects to a data source that Tableau Online cannot connect to directly, such as one that is stored on your local network, select the **Include External Files** check box.

Sign in to Tableau Server or Online

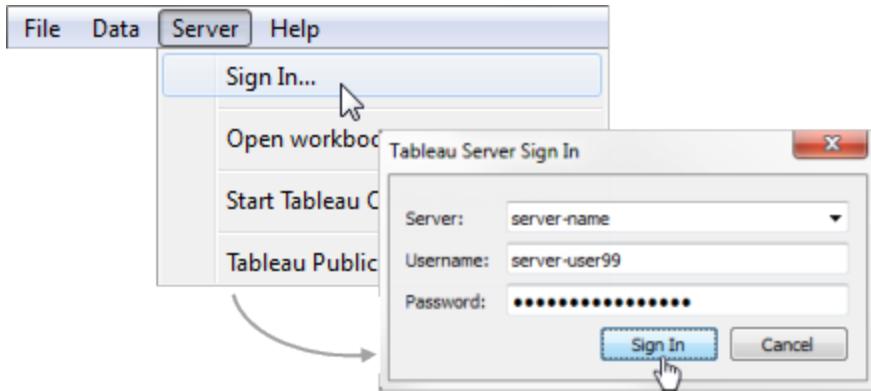
To publish data sources or workbooks, or to access data sources or workbooks that are already published, you need to sign in to your Tableau Server or Tableau Online site. The specific steps you take to sign in depend on the type of authentication your Tableau administrator has configured.

This topic describes the basic sign-in steps, includes variations for different authentication types, and links you to steps for clearing saved server sign-ins.

Basic sign-in steps

- Select **Server > Sign In**, and enter your server name, and user name and password.

For the Tableau Online server name, enter <https://online.tableau.com>.



Variations on signing in to Tableau Server

Note: For Tableau Online, see [Sign In to Tableau Online](#) in the Tableau Online Help.

With some Tableau Server configurations your sign-in experience varies from the basic steps. The sections here describe variations for single sign-on (SSO) and special issues related to working in a Kerberos environment.

Active Directory

If Tableau Server is configured to use Active Directory, enter your Windows user name. If you work in a multi-domain environment, and you are not currently on the default domain, enter your Tableau Server user name and password.

SAML or SAP HANA single sign-on

- If Tableau Server is configured to use SAML, your user name and password are managed by a third-party identity provider. When you enter your user name for Tableau Server, you are directed to the sign-in page for the identity provider.
- If Tableau Server is configured for SAP HANA SSO, Tableau connects to the server, and you do not need to provide credentials.

Mutual SSL

If Tableau Server is configured for mutual SSL and you have more than one client certificate, you are prompted to select which certificate to use the first time you connect.

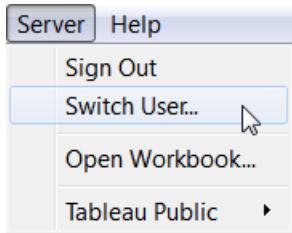
Kerberos environment

If Tableau Server is Kerberos-enabled and you are signed in to your computer using Active Directory credentials, Tableau Desktop connects to the server, and you do not need to provide credentials.

Switching user accounts (typically for testing purposes)

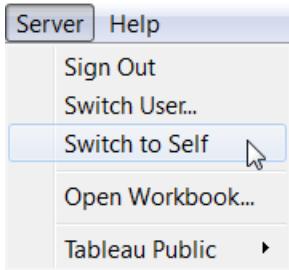
If Kerberos authentication does not succeed, you are prompted to provide a user name and password. If this occurs, and you need to change the user you're signed in as on Tableau Server, complete the following steps:

1. On the Server menu, select **Switch User**.



2. In the Tableau Server Sign In dialog box, provide the new user name and password.

If you sign in as a different user and then need to sign back in using your normal credentials, on select **Server > Switch to Self**.



Clearing saved server sign-ins

When you sign in to a server, Tableau Desktop stores your credentials in a secure token that remembers your Tableau Desktop connection with the server. When this token is in place, you can access the server directly, without having to sign in.

- If you want to forget a server connection, select **Server > Sign Out**.
- To remove all of your existing server connections, select **Help > Settings and performance > Clear saved server sign-ins**.

For more information, see [Quick Start: Stay Connected with Automatic Sign-In on page 1811](#).

If you never want server sign-ins to be saved, a Tableau Server administrator can change the server settings to disallow connected clients. This setting also affects Tableau Mobile and other Tableau clients. For information, see [Authentication for Connected Devices](#) in the Tableau Server Help.

Note: Automatic sign-in is available for servers configured for Windows Authentication only. Servers configured for Kerberos or SAML authentication cannot use automatic sign-in.

Set Permissions as You Publish a Data Source or Workbook

As the publisher of a workbook or data source, you can set permissions as part of the publishing process. Permissions allow or deny other users access to your published content on Tableau Server or Tableau Online. For example, who can interact with views in a workbook, edit them, or save changes to them, who can download a copy of a data source, and so on.

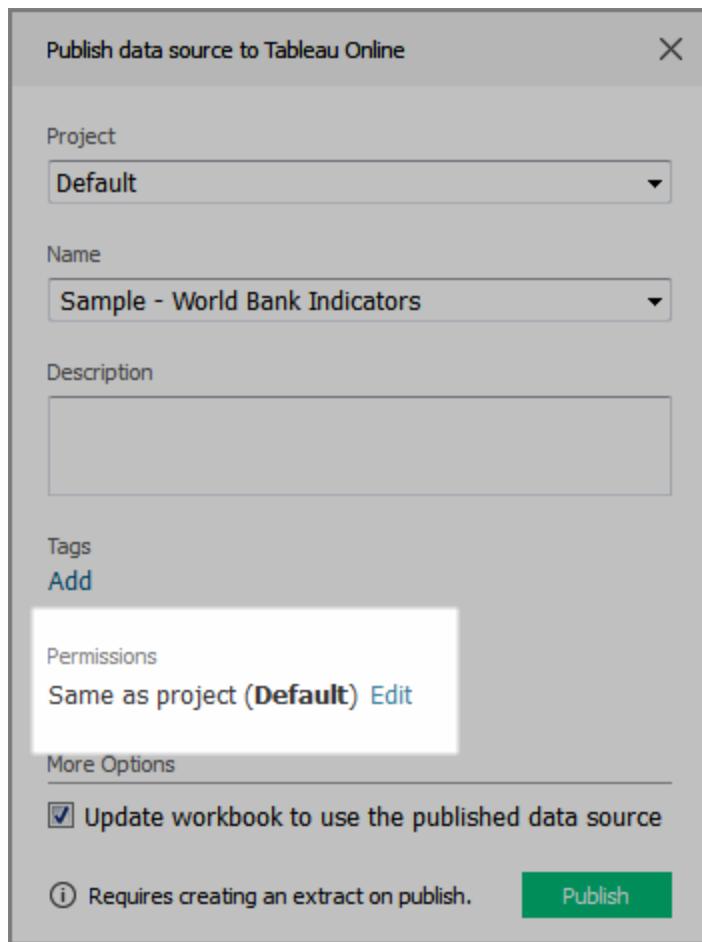
Permissions are separate from the type of access you set on the connection to the data. Accessing some data types requires signing in using a database name and password or embedding database credentials into the connection. For information about that, see [Set Credentials for Accessing Your Published Data on page 1248](#).

Back to permissions, with few exceptions, your best choice will be to accept the default settings in the publishing dialog box. If you agree, you can return to the [Publish a Workbook on page 1238](#), and save some time not reading the rest of this topic.

The remaining sections give context and tips for when you think your workbook or data source might be an exception to the default settings, and shows you how to change the permissions.

About setting permissions during publishing

When you start the publishing process, the dialog box shows the permissions that will be applied. By default, the content you publish takes the capabilities that are already set on the server, typically as they're set on the project you are publishing to.



When you change permissions in the publishing dialog box, you are setting capabilities *explicitly* for the content you're publishing. The effect is that the workbook or data source you publish no longer inherits changes made at its parent level, such as the project. Depending on your environment, this might be as you intend, or it might conflict with the guidelines your administrator has set and have unintended consequences.

Tips for deciding whether to set permissions explicitly

- **Learn your organization's practices**

Consult with your Tableau administrator to learn the guidelines for your organization. It's common (and recommended) practice for an administrator to manage permissions on your Tableau Server or Online site. If you work in such an environment, even if you set permissions during publishing, the person who manages permissions on the server might change these settings afterward.

- **Know the consequences of setting explicit permissions**

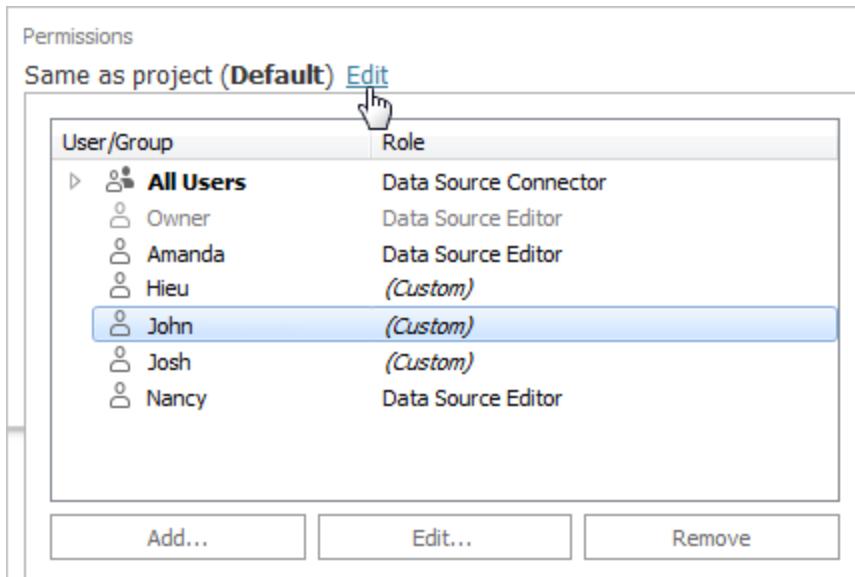
In addition to the potential conflicts described earlier, explicit permissions on some content requires extra maintenance to keep track of which content has exceptions, and which exceptions are applied.

- **Publish quickly by accepting the default permissions settings**

If necessary, you or your administrator can update permissions on the server afterward, where you have a more comprehensive view into the effects of your changes.

How to set explicit permissions during publishing

1. In the publishing dialog box, next to the summary that indicates the current settings, click **Edit**.

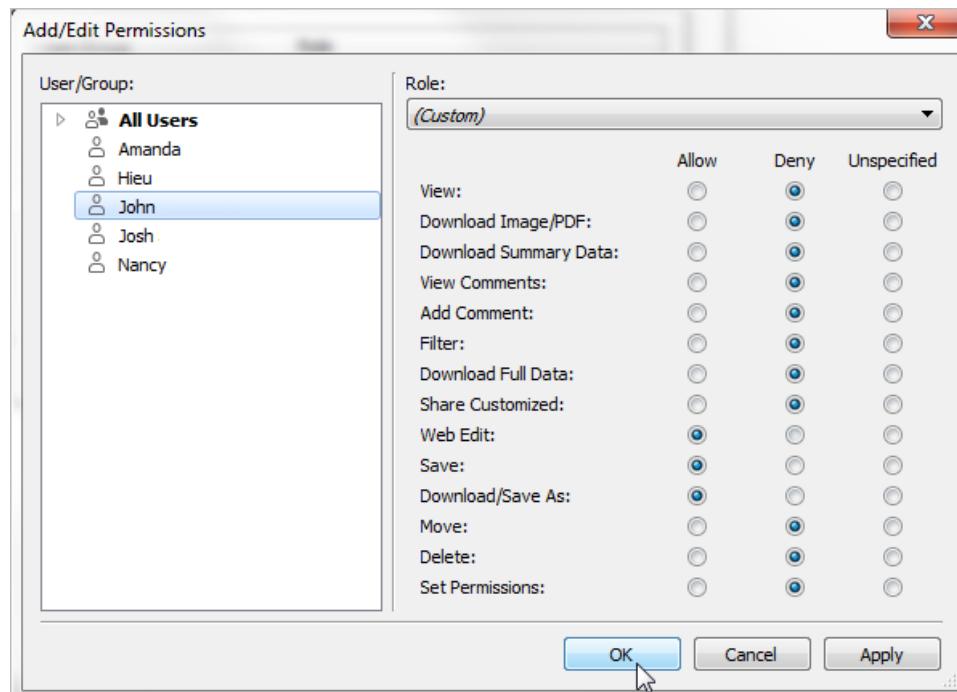


2. In the popup that appears, do one of the following:

- To set custom capabilities or assign a role explicitly, select an existing user or group and click **Edit**, or click **Add**.

In the **Add/Edit Permissions** dialog box, make your changes.

Click **Apply** to save changes and keep the dialog box open to configure another user or group. Click **OK** to close the dialog box.



- To remove roles or capabilities that are set explicitly, select the user or group, and then click **Remove**.

Predefined roles you can assign

When you publish a workbook, you can assign any of the following predefined roles to a selected user or group:

- Viewer:** Allows the user or group to view the workbook on the server, as well as add and view comments.
- Interactor:** Allows the user or group to view the workbook on the server, edit workbook views, apply filters, view underlying data, export images, and export data. All other capabilities are inherited from the user's group and project permissions.
- Editor:** Allows all capabilities to the user or group.

When you publish a data source, you can select from these roles:

- Data Source Connector:** Allows the user or group to connect to the data source on the server.
- Data Source Editor:** Allows the user or group to connect to the data source on the server and to publish, edit, download, delete, set permissions, and schedule refreshes

for the data source.

Note: If you are not the content owner or an administrator, you cannot schedule refreshes directly on the server.

Set Credentials for Accessing Your Published Data

When you publish a workbook to Tableau Online or Tableau Server, you can publish the data it connects to as an integrated part of the workbook (*embedded* into the workbook), or as separate, standalone data sources. In addition, if the data you're publishing requires anyone who wants to view it to enter a user name and password, you specify how others get access to the data when they open the workbook.

The latter is true also when you publish a data source on its own.

This type of authentication to your data is independent of how people sign in to your Tableau Online or Tableau Server site. For example, to give people direct access to the data in a workbook, you would embed a designated database user's credentials into the data source's connection. But anyone viewing the workbook would still need to be able to sign in to the site on Tableau Online or Tableau Server to open your workbook.

This topic describes how to set authentication on data connections as part of the publishing process.

In this topic

- [Set the authentication type below](#) (most connection types)
- [Kerberos environments](#) on page 1250
- [SAP HANA or Impala single sign-on](#) on page 1250
- [Salesforce, Google connections](#) on page 1250
- [Workbook connections to Tableau data sources](#) on page 1250

Note: This topic does not apply to connections to that do not require authentication, such as text files or Excel files. However, you can set authentication when you publish extracts of those data types.

Set the authentication type

For many types of connection you can embed a database user's name and password, or set a type of impersonation. Specific exceptions are described in the later sections in this topic.

The following steps describe how to set authentication as part of publishing a data source or workbook. You can do this for each connection in the data source.

1. In the Publish Workbook dialog box, go to the **Data Sources** area, which lists the workbook's connections, and click **Edit**.
2. In the **Manage Data Sources** popup, after you decide whether to publish the data source separately or as part of the workbook, select an authentication type for each connection in the data source.

The available authentication types depend on the connection type, and they can include one or more of the following:

- **Prompt user:** Users must enter their own database credentials to access the published data.
- **Embedded password:** The credentials you used to connect to the data will be saved with the connection and used by everyone who accesses the data source or workbook you publish.
- **Server run as account:** Tableau Server only. The server's Run As user account will authenticate all users.
- **Viewer credentials or Publisher credentials:** Tableau Server only. In a Kerberos, SAP HANA, or Impala environment, the user's domain user name and password are used. These are explained in more detail in the sections below this list.
- **Impersonate via embedded account or Impersonate via server Run As account:** (Available only when publishing SQL Server data to Tableau Server.) You can embed credentials through which SQL Server creates the connection and then impersonates the signed-in Tableau Server user. To create the connection you can specify the server's Run As account or a different database user that has the appropriate permissions for impersonation.

For more information, see the topics under [SQL Server Impersonation](#) in the Tableau Server Help.

- **Refresh not enabled or Allow refresh access:** These options appear when you publish an extract of cloud data such as from Salesforce.com, and database credentials are needed to access the underlying data. **Allow refresh access** embeds the credentials in the connection, so that you can set up refreshes of that extract on a regular schedule. Setting **Refresh not enabled** prompts users when they open the workbook.

Important: How you want to keep extracted data fresh is also a factor. If you want to set up an automatic refresh schedule, you must embed the password in the connection. In addition, if you're publishing a cloud data connection to Tableau

Online, the publishing steps will alert you if you need to add Tableau Online to the data provider's authorized list.

Kerberos environments

(Tableau Server on-premises only.) Select **Viewer credentials** to use the user's domain user name and password to access a Kerberos-enabled Teradata, PostgreSQL, Microsoft SQL Server, or Microsoft Analysis Services data source.

SAP HANA or Impala single sign-on

(Tableau Server only) Select **Viewer Credentials** to authenticate users who access the data source in these environments:

- You use single sign-on to SAP HANA and Tableau Server is configured for SAP HANA SSO.
- You use Impala with Cloudera Hadoop, and Tableau Server is configured for single sign-on.

Salesforce, Google connections

Salesforce and Google connections use a type of authentication in which the data provider sends a secure access token (using OAuth 2.0), so that the credentials themselves are not stored with Tableau. You need to create and embed these credentials after publishing, from the Data Connections page on the server.

- If you publish to Tableau Server, see [Edit Connections](#) in the Tableau Server Help.
- If you publish to Tableau Online and the workbook connects to Salesforce.com, see [Refresh Salesforce Extracts](#) in the Tableau Online Help.
- If you publish to Tableau Online and the workbook connects to Google Analytics, Google Sheets, or Google BigQuery data, see [Refresh Google Data](#) in the Tableau Online Help.

Workbook connections to Tableau data sources

When you publish a workbook that connects to a Tableau Online or Tableau Server data source, rather than setting the credentials to access the underlying data, you set whether the workbook can access the published data source it connects to. Regardless of the original data type, the choice for server data sources is always **Embedded password** or **Prompt users**.

If you select to prompt users, a user who opens the workbook must have **View** and **Connect** permissions on the data source to see the data. If you select embed password, users can see the information in the workbook even if they don't have View or Connect permissions.

See also

[Publish a Workbook](#) on page 1238

Schedule Extract Refreshes as You Publish a Workbook

When you publish workbooks that connect to extracts, you can set up a schedule for updates (refreshes) for those extracts, so the views in those workbooks stay current.

The ways you can set up and manage extract refresh schedules depends on where you publish and on the original data type.

- When you publish to **Tableau Server**, the schedule runs and is managed on the server.
- When you publish to **Tableau Online**:
 - If extracts are from cloud data (for example, Google Analytics or Salesforce), schedules run and are managed on Tableau Online.
 - If extracts are from *on-premises* data (data you maintain on your internal network) or web data connector data, you set up and manage refresh schedules using the Tableau Desktop sync client.

Note: In many organizations a server content manager or administrator manages all refresh schedules. We recommend that, before you publish, you find out from your administrator whether they have guidelines for these schedules.

Set up a schedule as you publish content

During the publishing process, after you click the **Publish** button, the scheduling options appropriate for your data types and publishing destination appear.

If you are publishing a multi-connection data source, you need to set a refresh schedule for each extract connection in the data source.

To use refresh schedules with connections to a database that requires you to sign in to it, you must store (embed) a database user's credentials with the connection. For more information, see [Set Credentials for Accessing Your Published Data](#) on page 1248.

Publish and refresh on Tableau Online

For the subtleties around refreshing each type of data you publish to Tableau Online, see [Keep Data Fresh](#). Here are examples of things you need to know:

- If your workbook connects to cloud data, add Tableau Online to your cloud data provider's authorized list (whitelist). The IP address range for your site location is listed in [Keep Data Fresh](#).
- The Tableau Online sync client starts during the publishing process if your data source or workbook connects to on-premises data.

The sync client supports standard database authorization and cannot refresh data you connect to through OAuth. To learn more, see [Refresh Data Using the Tableau Online Sync Client](#).

- When you publish a multi-connection data source to Tableau Online, if any one connection requires using the sync client, you must use the sync client to refresh *all* connections in the data source.

For example, say you publish a data source with connections to an extract of MySQL data hosted in the cloud, and one of a SQL Server database on your local network. Although Tableau Online supports refreshing hosted MySQL data, in this case, you would need to refresh both the SQL Server and MySQL connections through the sync client.

Refreshing web data connector extracts

When you publish a workbook with a web data connector (WDC) data source, you must *import* the web data connector to Tableau Server before you can set up a refresh schedule. You can do this only on Tableau Server. For information, see [Web Data Connectors in Tableau Server](#) in the Tableau Server Help.

For information about refreshing web data connector data sources published to Tableau Online, see [Schedule Refreshes Using the Sync Client](#) in the Tableau Online Help.

Restrict Access at the Data Row Level

When you share workbooks with others by publishing them to Tableau Server or Online, by default, all users who have access to the workbooks can see all of the data shown in the views. You can override this behavior by applying a type of filter that allows you to specify which data “rows” any given person signed in to the server can see in the view.

How user-based filtering works

Suppose you created a quarterly sales report for a set of products over several years, in different geographic regions.



When you publish this report, you want to allow each regional manager to see only the data relevant to his or her region. Rather than creating a separate view for each manager, you can apply a *user filter* that restricts access to the data based on users' characteristics, such as their role.

Restricting access to data in this way is referred to as *row-level security* (RLS). Tableau offers the following approaches to row-level security:

- [Create a user filter and map users to values manually](#) on page 1255.

This method is convenient but high maintenance, and security can be tentative. It must be done per-workbook, and you must update the filter and republish as your user base changes.

- [Create a dynamic filter using a security field in the data](#) on page 1257.

Using this method, you create a calculated field that automates the process of mapping users to data values. This method requires that the underlying data include the security information you want to use for filtering.

The most common way to do this is to use a reference (or “look-up”) table that contains this information. For example, if you want to filter a view so that only managers can see it, the underlying data must include user names and specify each user’s role.

Because filtering is defined at the data level and automated by the calculated field, this method is more secure than mapping users to data values manually.

[Adding user filters to data sources](#)

The two approaches in the previous section describe ways to add filters to data embedded in workbooks. If multiple workbooks connect to the same data, instead of wrangling filters on each

workbook, you can filter the data source, and then connect the workbooks to the data source after you publish it.

Workbooks that connect to your filtered data source expose only the data the user signed in to the server is allowed to see. In addition, all connected workbooks show data refreshes as they occur.

Extracts vs. live connections with user filters

The related topic describes how to publish a live connection with user filters. Publishing extracts with user filters has its own complications around row duplication and performance. These are not addressed here.

However, we can direct you a comprehensive discussion about row level security with extracts, on a blog maintained by a Tableau Sales Consultant who has extensive experience with this area. See the following two posts:

- [Defusing Row Level Security...Part 1](#)
- [Defusing Row Level Security...Part 2](#)

Part 2 contains what our Sales Consultant refers to as the “scalable” approach. This is described in the Advanced Solutions section of Part 2.

Disclaimer: Clicking these links will take you away from Tableau.com. Although we make every effort to ensure links to external websites are accurate and relevant, Tableau cannot take responsibility or provide support for the external content.

For further help with publishing extracts with user filters, we recommend that you work with the Tableau Professional Services team.

See also

[Data Security](#) in the Tableau Server Help

Create a User Filter and Secure it for Publishing

Using the example shown in the overview topic [Restrict Access at the Data Row Level on page 1252](#), this topic provides the steps for implementing each of the approaches Tableau takes toward row-level security. It includes additional steps for securing the filters for publishing with live connections, whether on data embedded in a published workbook, or in the a data source you want to publish separately.

If you haven't seen the example or read about the Tableau approach to row-level security, take a look at the overview topic before you complete the steps below.

In this topic

- [Create a user filter and map users to values manually on the next page](#)
- [Create a dynamic filter using a security field in the data on page 1257](#)

- [Filter a data source on page 1259](#)
- [Secure user filters on published content on page 1260](#)
- [Copy selected field values from one user to another on page 1261](#)

Create a user filter and map users to values manually

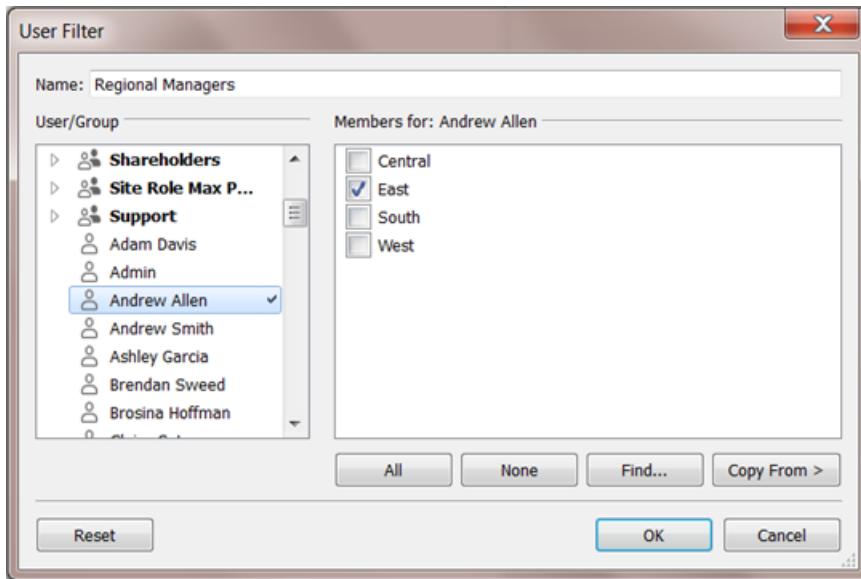
The steps in this section describe the simplest way to incorporate user-based filtering. This procedure can be sufficient if you have a small and fairly static set of users or groups to manage, and only a small number of workbooks that need user filters. Completing the steps is also a low-cost way to get more familiar with user filtering.

1. In Tableau Desktop, open the workbook, or create a new one, and set up the connection to the data you want to filter.
2. Select **Server > Create User Filter**. Then select the field you want to use for filtering the view. This example uses **Region**.
3. If prompted, sign in to your server or site. For information, see [Sign in to Tableau Server or Online on page 1242](#).
4. In the **User Filter** dialog box, type a name for the set of rules you are creating.

In this example, we'll use **Regional Managers**.

5. In the list on the left, select a user or group. On the right, select the individual members of the field you selected earlier, that you want the selected users to be able to see.

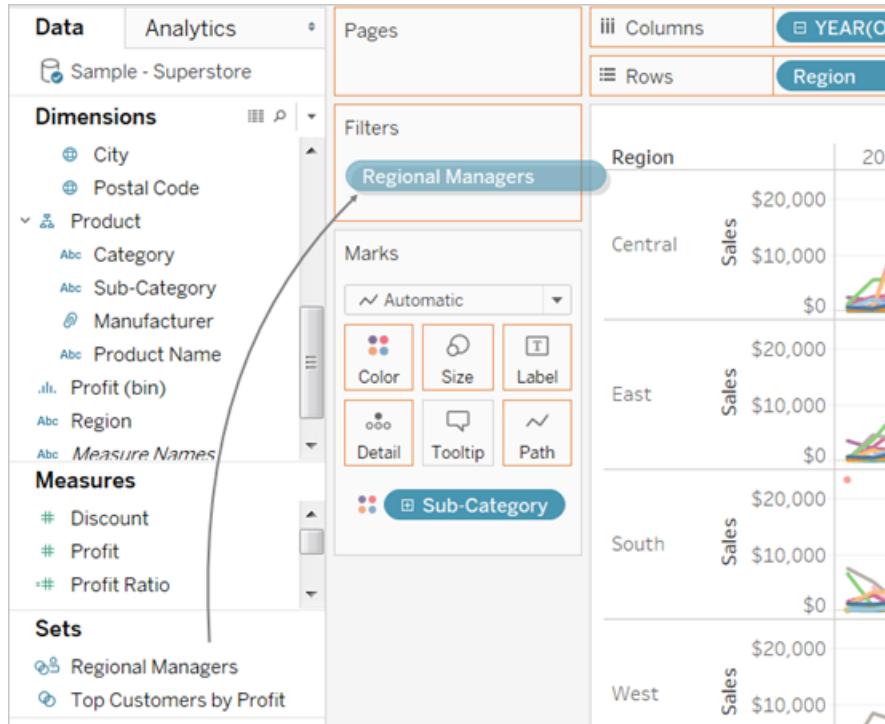
For this example, selected user Andrew Allen is the manager of the Eastern region, so in the field member list, you would select **East**.



Repeat this process for each user or group, and click **OK** when you're done mapping users to values.

After you create the user filter, it appears in the **Sets** area of the Data pane.

6. Drag the user filter to the **Filters** shelf.



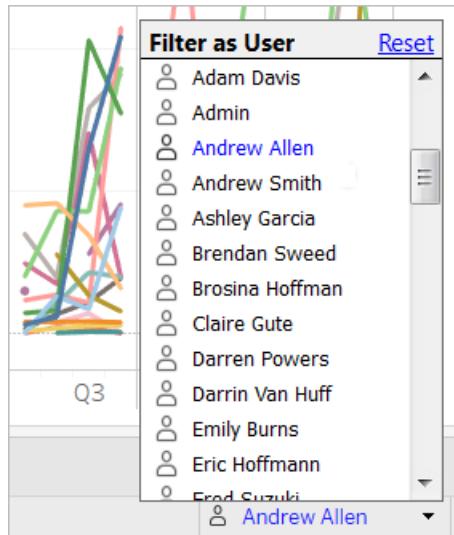
The filter becomes a context filter, and the view adjusts to show data that you are allowed to see.

7. Do any of the following to test or fine-tune the filter:

- If the view appears as a blank canvas, you need to allow yourself or a group you are a member of to see a region.

In the **Sets** area of the **Data** pane, open the drop-down menu on the user filter, and then select **Edit Set**.

- To preview how the filter works in the published view, in the lower-right corner of the workbook, open the **Filter as User** menu, and select the user or group from the list.



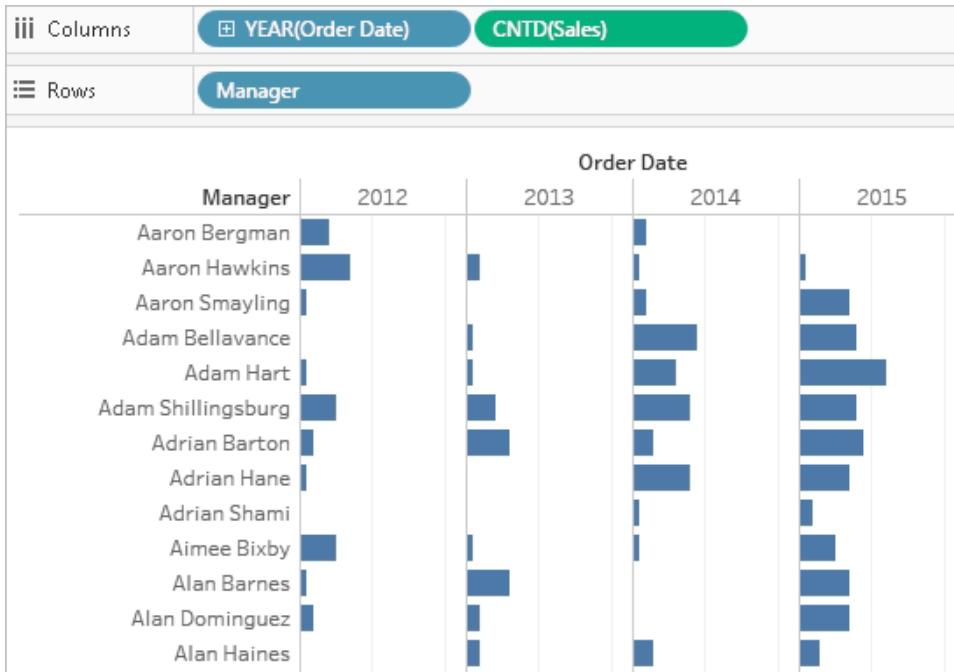
Note: Previewing is not available if the workbook connects to a Tableau Server data source.

- To return to viewing the workbook as yourself, in the top right corner of the **Filter as User** menu, select **Reset**.
- To copy the mapping selections you set on one user or group to another (rather than manually mapping the same settings), see [Copy selected field values from one user to another on page 1261](#).

When you publish the workbook, you need to take additional steps to make sure users cannot edit the workbook and remove the filter. For information, see [Secure user filters on published content on page 1260](#).

Create a dynamic filter using a security field in the data

The steps below are based on the following viz, which shows annual sales performance for a list of regional managers.



As described in [Create a User Filter and Secure it for Publishing on page 1254](#), if you want to take this approach, the database must include the field you want to use for filtering.

For this example, the data includes a reference table called **People**, which contains two columns: Region and Manager. Names in the Manager field match Tableau Server or Online user names, and we'll use this field for filtering.

You can follow along using the Superstore data that comes with Tableau Desktop, although the fields and values do not match exactly.

Connect to the data and set up the user filter

1. In Tableau Desktop, open the workbook you want to add user filtering to, or create a new one, connecting to the data you want to filter.

In this example, we use a table called **Orders**.

2. On the **Data Source** page, add the reference table, creating a left join. Here we add the **People** table and create a left join on the **Region** field.
3. Go to the worksheet, select **Analysis > Create Calculated Field**, and create the following field:
 - Name: User is a manager
 - Formula: USERNAME() = [Manager]



This new true/false field appears in the Dimensions pane. The formula returns TRUE if the user name of the person signed in to the server exists in the Manager column.

4. Add the **User is a manager** field to the **Filters** shelf.

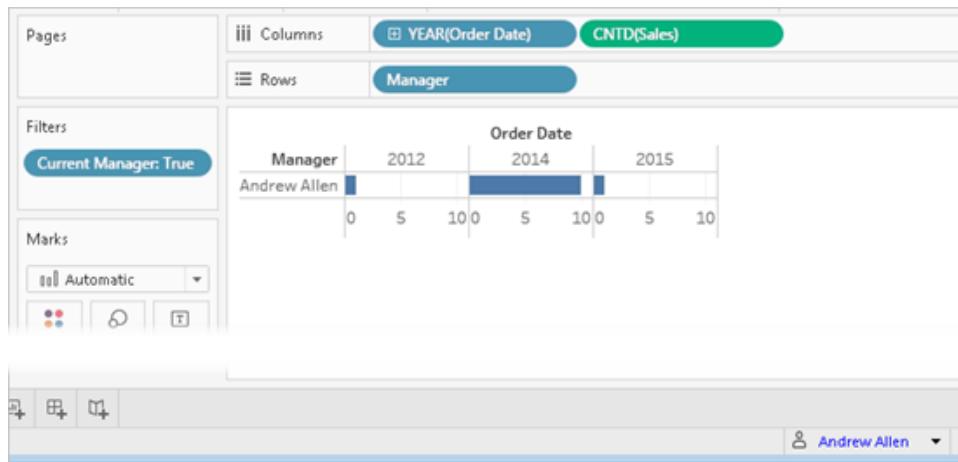
5. In the **Filter** dialog box, select **True**, and then click **OK**.

This sets the filter so that only people who are managers can see the data in the view.

If you are not listed in the Manager field, your view might appear as a blank canvas.

6. See how the view looks to a particular person: in the lower-right, open the **Filter as User** menu, and select someone you know is a manager.

The following image shows how the view shown earlier would look if Andrew Allen were signed in.



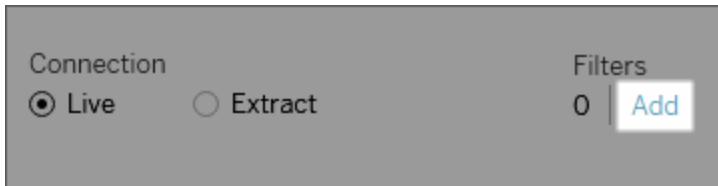
As with a manual user filter, you need to take steps to [Secure user filters on published content](#) on the next page.

Filter a data source

Rather than maintain user filters and special permissions on each published workbook, you can filter a data source, and then publish the data source to make it available as a shared, one-to-many resource for anyone who uses that data.

This procedure builds from the dynamic-filter approach described in this topic.

1. Complete the steps in [Create a dynamic filter using a security field in the data on page 1257](#).
2. In the lower-left area of Tableau Desktop, select the **Data Source** tab.
3. In the upper-right area of the **Data Source** page, under **Filter**, click **Add**.



4. In the **Edit Data Source Filters** dialog, click **Add**, add the calculated field you created for the dynamic filter, and set the filter to **True**.

In our examples, this is the **User is a manager** field.

Click **OK** until you get back to the **Data Source** page.

[Secure user filters on published content](#)

When you publish a workbook or data source with user filters, you need to set permissions to make sure that users opening your workbook or connecting to your Tableau Server data source cannot remove the filter, thereby gaining access to all of the data.

Notes

- Before publishing, we recommend that you consult with your Tableau administrator regarding any existing policies in your organization, particularly for setting permissions. See also [Set Permissions as You Publish a Data Source or Workbook on page 1244](#).
- This information applies publishing with **live connections**. Publishing extracts with user filters is not covered here. For a link to a comprehensive resource, see [Extracts vs. live connections with user filters on page 1254](#) in the overview.

Permissions settings

To secure user filters, the following capabilities must be set to **Deny**, either during publishing, or afterward on the server.

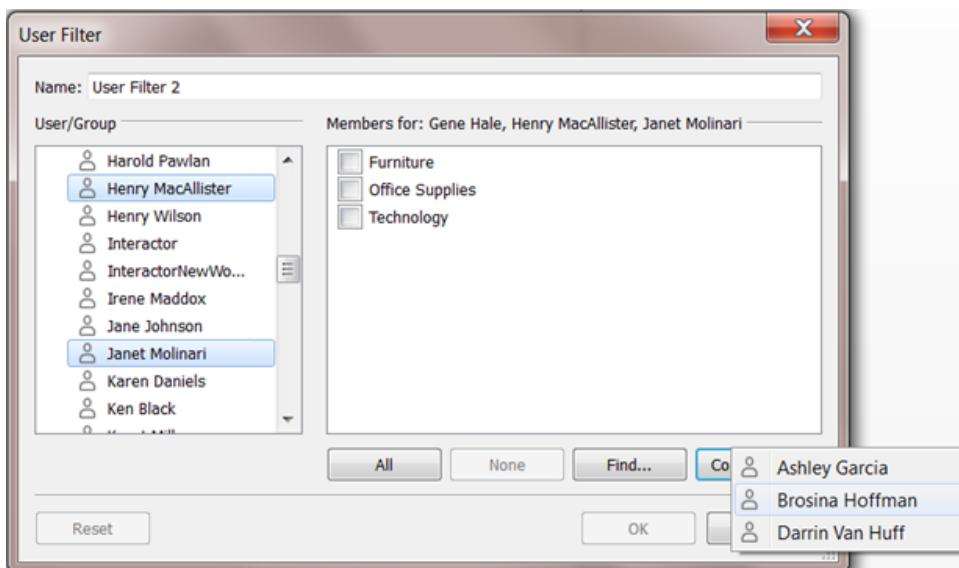
Workbooks	Data sources
<ul style="list-style-type: none">• Web Edit• Download Workbook/ Save As	<ul style="list-style-type: none">• Save• Download/ Save As

Workbooks	Data sources
• Set Permissions	• Set Permissions

Copy selected field values from one user to another

When you're creating a user filter manually, after you map a user or group to data values (members), you might want to map another user or group in the same way. You can do this by copying and pasting the settings.

1. In the **Data** pane, under **Sets**, select a user filter. Click the drop-down arrow and select **Edit Set**.
2. In the **User Filter** dialog box, select the user or group on which you want to paste settings from another user or group.
3. Click **Copy From**, and select the user or group whose settings you want to copy.



See also

[Data Security with User Filters](#), from the Tableau video learning library.

To watch the video, you need to sign in to Tableau.com (or create a free account).

Save Workbooks to Tableau Public

If you want to share your data discoveries with the world outside of your organization, you can save your workbook to Tableau Public, a free cloud service. On Tableau Public, anyone can interact with your views, or download your workbooks or data sources. For information, go to the [Tableau Public website](#).

When you save workbooks to Tableau Public, the publishing process creates an extract of your data source.

Save a workbook

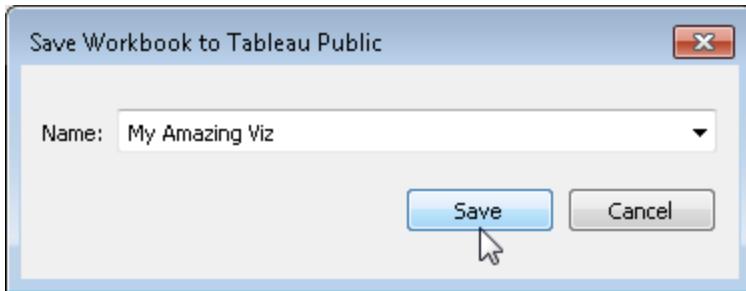
1. Select **Server > Tableau Public > Save to Tableau Public**.
2. Sign in using your Tableau Public account.



If you don't have one, select the link to create a new one.

3. Type a name for the workbook and click **Save**.

Tip: The title becomes part of your view's metadata. Use a unique title that will help others find it when they search. (The title shown in the image is a good example of how *not* to name your workbook.)



After the workbook is published, you are redirected to your account on the [Tableau Public website](#).

On the Tableau Public site, you can select a featured view, hide views, customize metadata and otherwise manage content you publish.

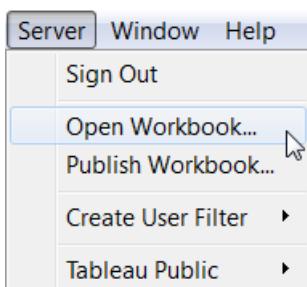
To get a link to share on social media or code to embed in a web page, display a view, and then click **Share** at the bottom of the view. (You can get links and embed code for other Tableau Public users' views this way, too.)

Opening Workbooks from the Server

If you have been granted the **Download/Web Save As** permission for a workbook, you can use Tableau Professional to open the workbook from the server. When you open a workbook from the server and make changes, you can either save it to your hard drive or, if you have been allowed the **Write/Web Save** permission, you can republish the workbook to the server.

To open a workbook from the server:

1. Select **Server > Open Workbook**.



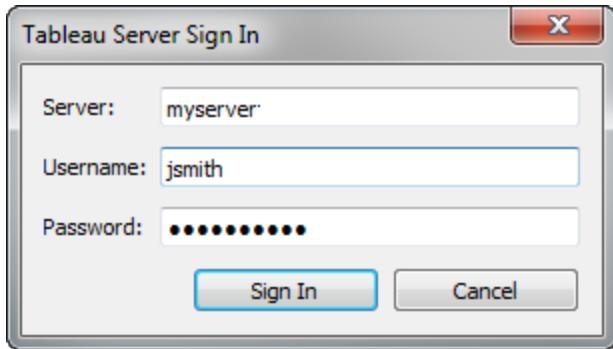
2. If you are not already signed in to Tableau Server, you see the Tableau Server Sign In dialog box.



Enter the server name or URL (for example, sales_server, or https://sales_server) and click **Connect**.

If Tableau Server is Kerberos-enabled and you are on a machine with valid Active Directory credentials, you will connect automatically to the server. In this case, you can skip step 3. If Kerberos authentication fails, you will be prompted to provide a user name and password for authentication.

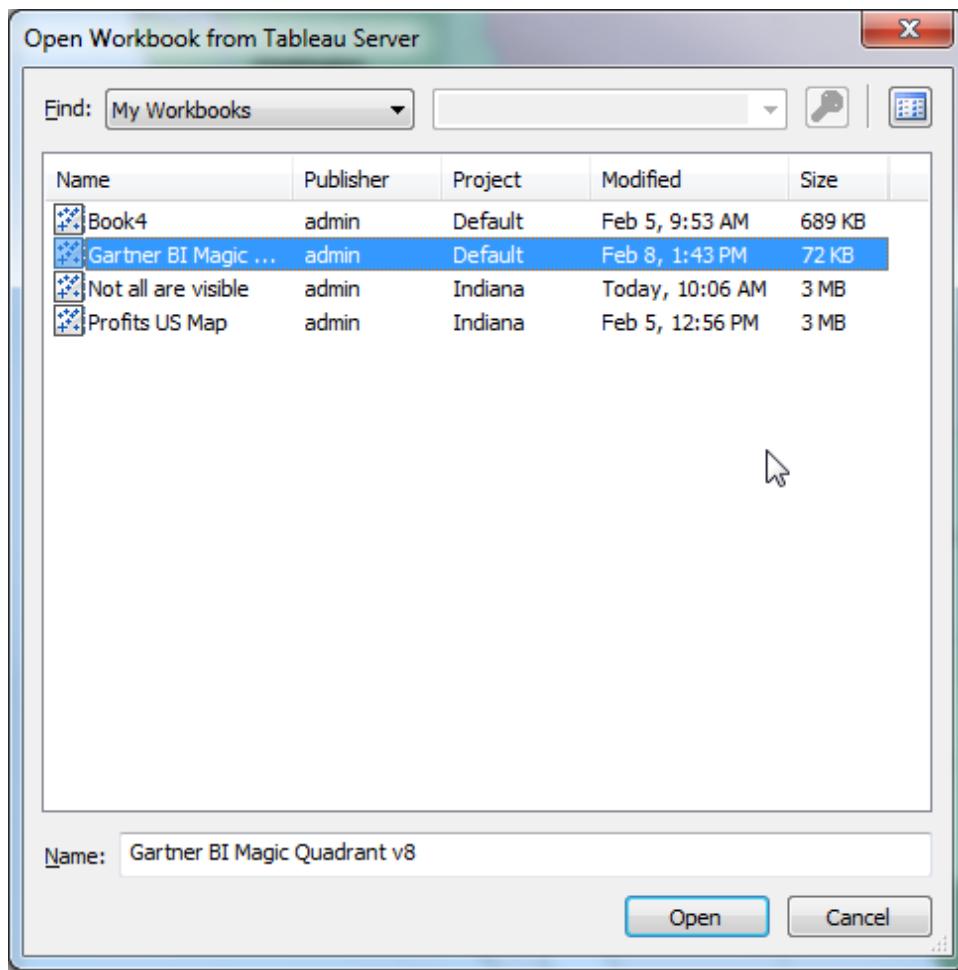
3. Next, enter your user name and password and click **Sign In**.



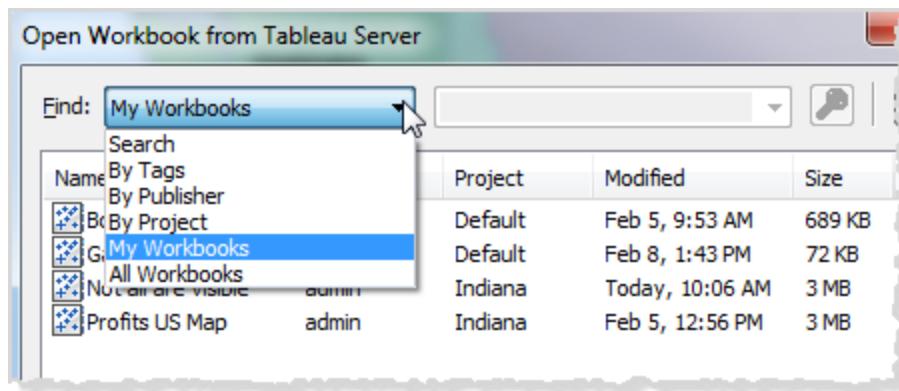
If Tableau Server is configured to use Active Directory, enter your Windows user name (the domain is not required—except in multi-domain environments where the user is not in the default domain); otherwise, enter your Tableau Server user name.

If Tableau Server is configured to use SAML for user authentication, you won't see the above dialog box. Instead, you'll see a sign in prompt from an external identity provider.

4. In the Open Workbook from Tableau Server dialog box, select the workbook you want to open.



You can find workbooks using the **Find** drop-down lists. You can search all workbooks on the server or find by tags, publisher, project, or workbooks that you published.



If you choose **Search** from the drop-down list, you can then enter a search string in the search box to the right. If you type an unquoted string in the search box, Tableau will search for workbook names that begin with the string that you type. If you type a quoted string, Tableau will search for workbook names that contain the string.

6. Click **Open**.

Save, Export, and Print

After you create views in Tableau, you may want to save or share your work. You can export your work in a number of different formats to be used in different applications such as Microsoft PowerPoint and Microsoft Excel.

Saving Your Work

After you create useful views of your data, you should save the results. Tableau provides three ways for you to save your work:

- Workbooks – Saves all open worksheets.
- Packaged Workbooks – Saves the workbook along with all referenced local file data sources and images into a single file.
- Bookmarks – Saves the current worksheet.

You can share workbooks and bookmarks with your co-workers, provided they can access the relevant data sources. If your co-workers do not have access to the data sources, you can save a packaged workbook.

Custom fields such as binned measures, calculated fields, groups, and sets are saved with workbooks and bookmarks.

Workbooks

When you open Tableau, it automatically creates a new workbook. Workbooks hold the work you create and consist of one or more worksheets. Each worksheet contains a particular view of your data.

To save a Tableau workbook:

1. Select **File > Save** or type **Ctrl + S** (**⌘S** on a Mac).
2. Specify the workbook file name in the **Save As** dialog box.

By default, Tableau saves the file with the **.twb** extension. The default location is the **Workbooks** folder in the Tableau repository. However, you can save Tableau workbooks to any directory you choose.

Tableau file names cannot include any of the following characters: forward slash (/), backslash (\), greater-than sign (>), less-than sign (<), asterisk (*), question mark (?), quotation mark ("), pipe symbol (|), colon (:), or semicolon (;).

To save an extra copy of a workbook that you already have open, select **File > Save As** and proceed by saving the file with a new name.

Packaged Workbooks

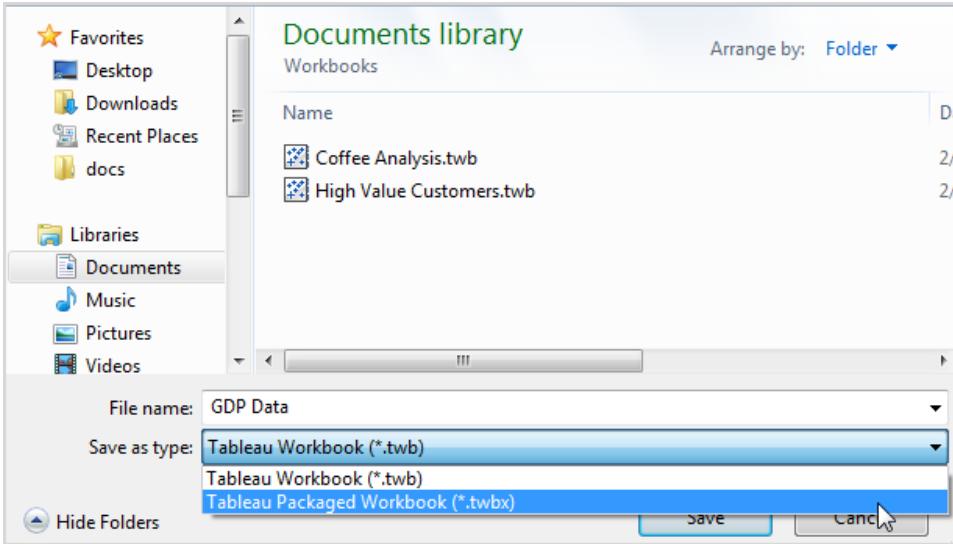
Workbooks often reference external resources. For example, workbooks might reference background images or local file data sources such as Excel, Access, and Tableau data extract files (.tde).

When you save a workbook, links to these resources are also saved. The next time you open the workbook, the views are automatically updated with any changes that may have occurred to the data and images. In most cases, you will want to save the workbook in this way. But if you plan to share the workbook with someone who does not have access to the referenced resources or to Tableau Server, you might want to save a packaged workbook instead.

Packaged workbooks contain the workbook along with a copy of any local file data sources and background images. The workbook is no longer linked to the original data sources and images. These workbooks are saved with a .twbx file extension. Other users can open the packaged workbook using Tableau Desktop or Tableau Reader.

[Create a .twbx with file-based data sources](#)

1. Select **File > Save As**.
2. Specify a file name for the packaged workbook in the Save As dialog box.
3. Select **Tableau Packaged Workbooks** on the **Save as type** drop-down list.



4. Click **Save**.

The default location is the **Workbooks** folder of the Tableau repository. However, you can save packaged workbooks to any directory you choose.

The following files are included in packaged workbooks:

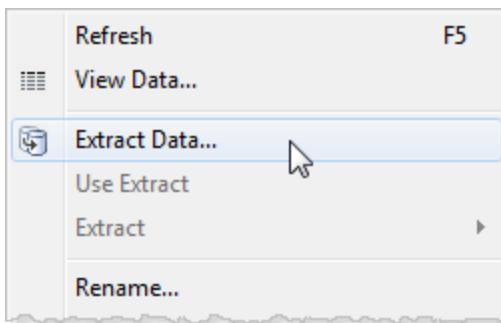
- Background images
- Custom geocoding
- Custom shapes
- Local cube files
- Microsoft Access files
- Microsoft Excel files
- Tableau data extract files (.tde)
- Text files (.csv, .txt, etc.)

If you are sharing packaged workbooks that contain Microsoft Excel or Access 2007 data sources, the people opening the workbook must either have Microsoft Excel and Access 2007 or the Office 2007 Data Connectivity Components installed on their computers. The data connectivity components are available on the Tableau [Tableau Drivers](#) page.

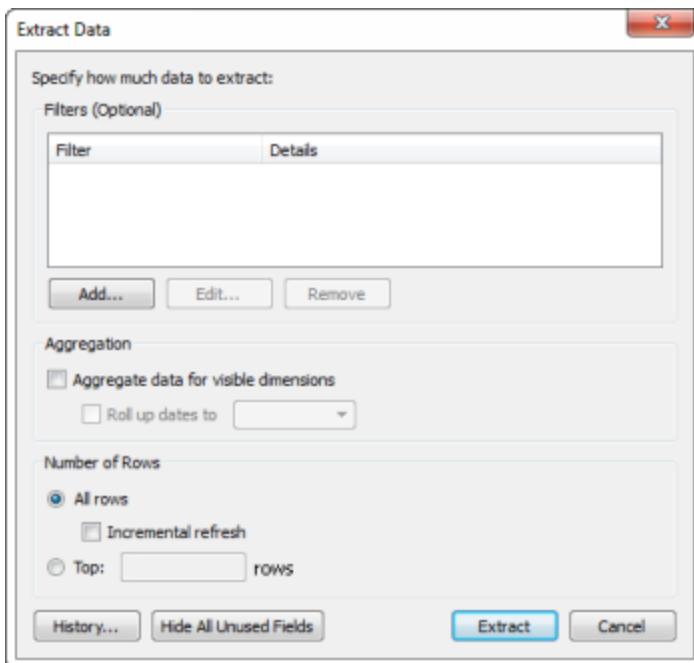
Create a .twbx with non-file-based data sources

If the workbook contains connections to enterprise data sources or other non-file-based data sources, such as Microsoft SQL, Oracle, or MySQL, the data must be extracted from the data sources for it to be included in a packaged workbook (.twbx).

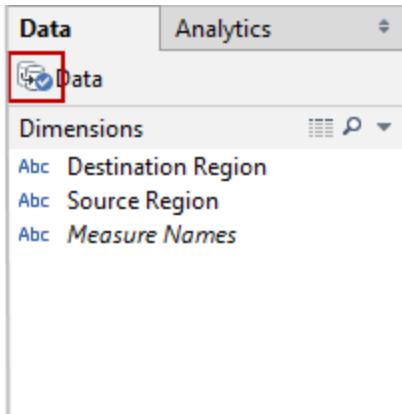
1. In the workbook, right-click the data source in the Data pane and choose **Extract Data**.



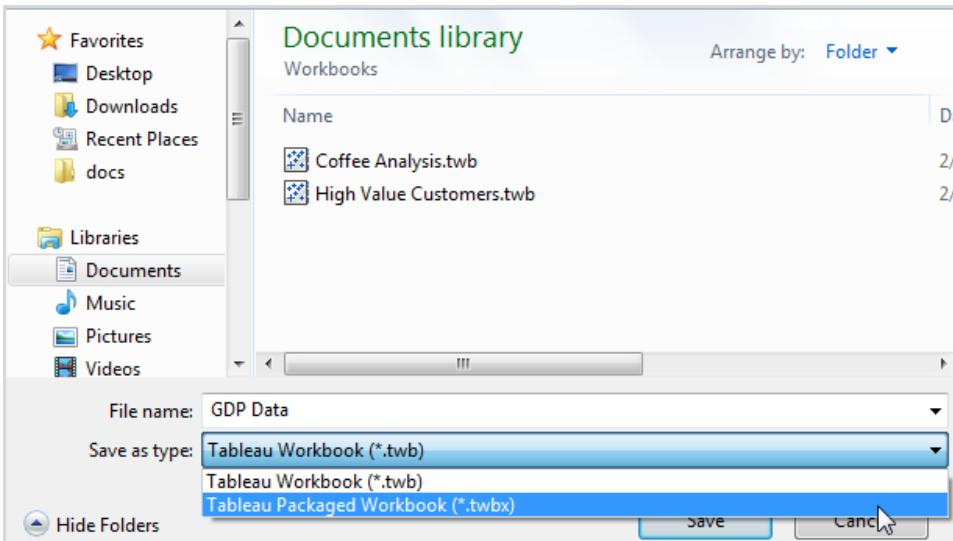
2. In the Extract Data dialog box, click the **Extract** button to extract all data from the data source.



After the extract completes, the data source icon changes to indicate that an extract is active for that data source. Instead of a single cylinder, there are two cylinders connected by an arrow.



3. **Optional:** Repeat the above steps for each data source in the workbook.
4. Select **File > Save As**.
5. From the **Save as type** drop-down menu, select Tableau Packaged Workbook (*.twbx).

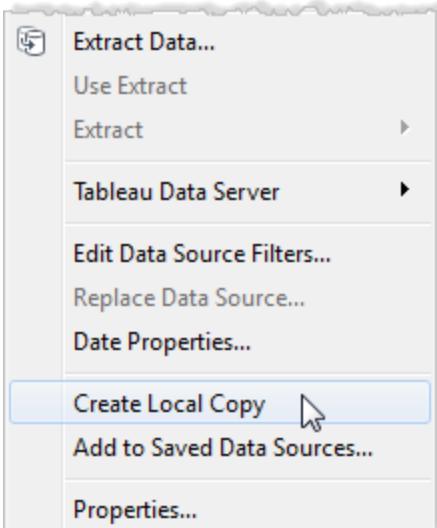


After the extracts have been created for all non-file-based data sources and the packaged workbook has been saved, you can send your workbook.

Create a .twbx with Tableau Server data sources

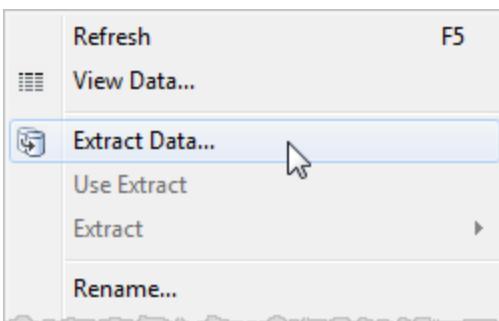
If the workbook contains connections to a published Tableau Server data source, you must download a local copy of the Tableau Server data source, take an extract of it, and then replace the connection to the local copy for it to be included in a packaged workbook (.twbx).

1. In the workbook, right-click the published data source in the Data pane, and then select **Create Local Copy**.

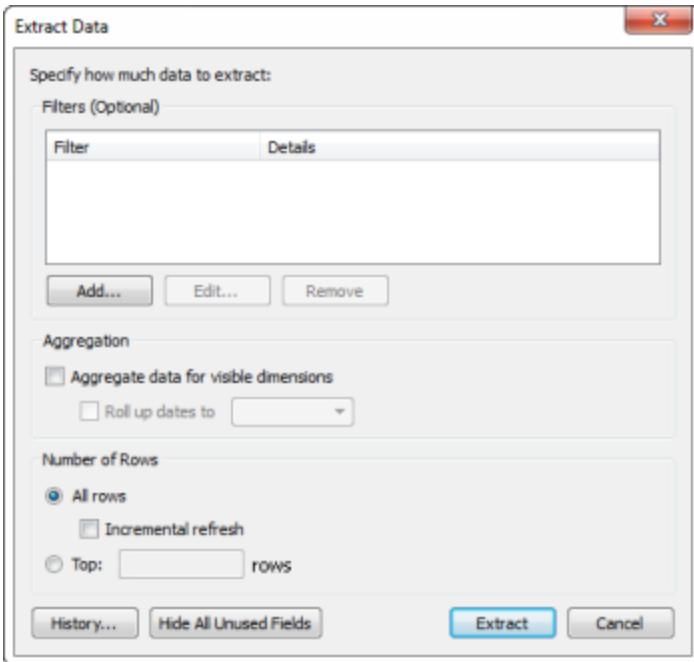


A copy of the published data source is added to the Data pane.

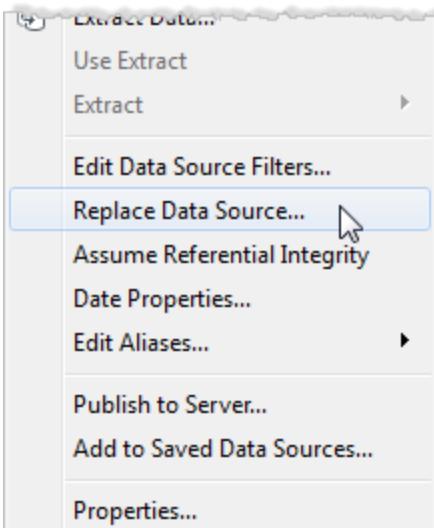
2. Right-click the local copy, and select **Extract Data**.



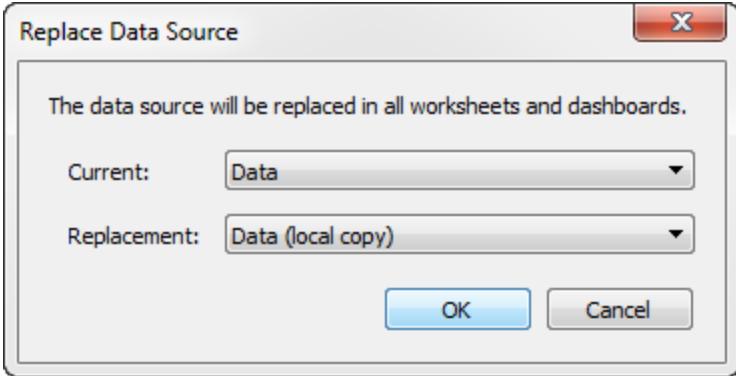
3. In the Extract Data dialog box, click the **Extract** button to extract all data from the data source. Creating an extract of the data source allows the person you are sharing the workbook with to have access to a copy of the data source.



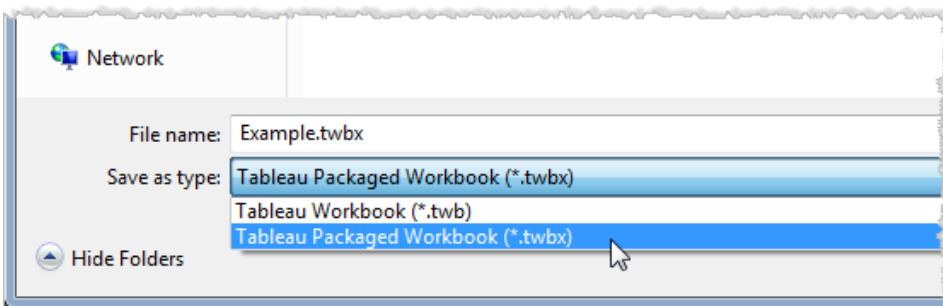
4. In the Data pane, right-click the published data source, and then select **Replace Data Source**.



5. Verify that the published data source will be replaced by the local data source, and then click **OK**.



6. Right-click the published data source, and then click **Close**.
7. Select **File > Save As**.
8. From the **Save as type** drop-down menu, select Tableau Packaged Workbook (*.twbx).



After the local copy and extract of the local copy is created and the packaged workbook saved, you can send your workbook.

Unpackage a .twbx

Packaged workbooks can be unpackaged.

- On a Windows computer, right-click the packaged workbook file (.twbx) in Windows Explorer and select **Unpackage**.
- On a Mac computer, rename the file with a .zip extension (for example, from myfile.twbx to myfile.zip) and then double-click it.

When you unpackage a workbook, you get a regular workbook file (.twb), along with a folder that contains the data sources and images that were packaged with the workbook.

Bookmarks

You can save a single worksheet as a Tableau bookmark. Bookmarks can be accessed from any workbook using the Bookmarks menu. Bookmarks are convenient when you have worksheets that you use frequently.

To save a Tableau bookmark:

1. Select **Window > Bookmark > Create Bookmark**.
2. Specify the bookmark file name and location in the Create Bookmark dialog box.

Tableau saves the file with a .tbo extension. The default location is the Bookmarks folder in the Tableau Repository. However, you can save bookmarks to any location you choose.

Bookmarks that are not stored in the Tableau repository do not appear on the **Bookmark** menu.

You can organize bookmarks into folders in the same way you organize files or documents. This can be useful when you have a large number of bookmarks to manage. For example, you might organize bookmarks based on employee name, product types, or sales results. You can organize bookmarks by creating a new folder, renaming an existing folder, renaming existing bookmark files, and so on.

Delete bookmarks the same way you would delete any other file on your computer. After you delete a bookmark from the Bookmarks folder in the Tableau Repository, it is removed from the **Bookmarks** menu the next time you start Tableau.

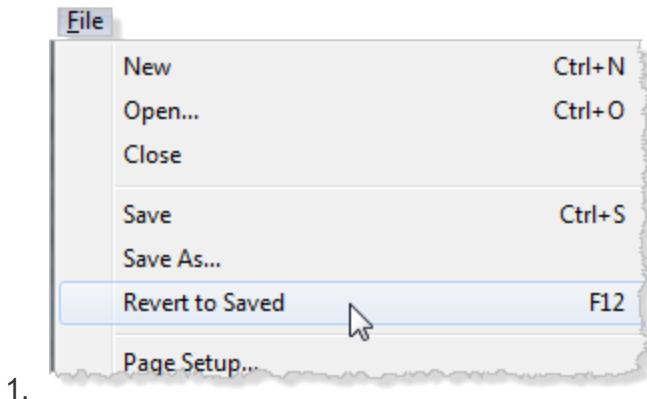
While bookmarks are generally a snapshot of the worksheet and include the data connection, formatting, etc., a bookmark does not include parameter values and the current page setting on the Pages shelf.

Reverting Workbooks

To revert a workbook is to undo all of the changes you've made to a workbook since you last saved it. You could click through your history using the **Undo** button, but there is an easier way.

To revert to the last saved state of a workbook:

Select **File > Revert to Saved**.



1.

2. Click **Revert** in the warning dialog box.

Or, just press F12.

The **Revert to Saved** command is only available for workbooks (.twb) that do not have connections to Extract data sources.

Exporting Your Work

After you have created some data views, you might want to export your results to other applications. Tableau provides several methods for you to export your work:

- Export Data – Copy the data from a view to an Excel worksheet or export as an Access database.
- Export as an Image – Copy images of your views into other applications such as Microsoft Office or PowerPoint. You can also include the images in web pages.

Exporting your results is a convenient way to share your work with coworkers who do not have access to Tableau, or to include your work as part of a presentation or document. You can also use Tableau to present your data.

Note: You can also export selected sheets to incorporate your results into other Tableau workbooks. For information, see **Export and Import Sheets** on page 1284.

Export Data

If you want to export data from Tableau to another application, or create a new data source that contains a portion of the records in your original data source, there are several options in Tableau.

When exporting data, you should keep these rules in mind:

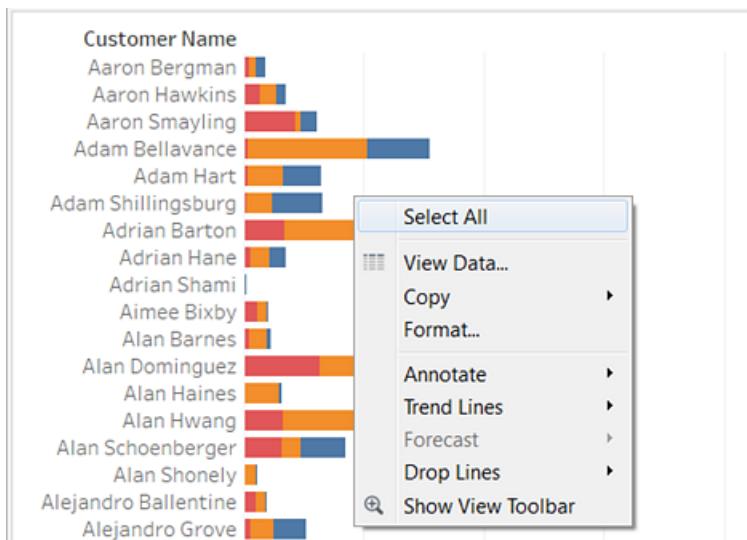
- You can select any portion of a data view to export. If you want to export all data in a view, right-click (control-click on Mac) in the view and choose **Select All**. Copying and exporting to a crosstab always exports all data in the view regardless of what you have selected.
- The fields that are exported to the new data source come from the fields on the worksheet shelves. The exception is fields that are external filters--that is, fields that appear only on the **Filters** shelf.
- If you want to include other fields (either dimensions or measures) with the exported data without changing the basic view, place those fields on the **Level of Detail** shelf.

Copy Records to Clipboard

Typically this function is used to copy records from Tableau into Microsoft Excel. To create an Excel spreadsheet from Tableau data, follow these three steps:

1. Select the data in Tableau that you want to copy.

Right-click (Control-click on Mac) the view and click **Select All**.



2. Select **Worksheet > Copy > Data**, or right-click (control-click on Mac) the view and select **Copy > Data** from the context menu.
3. Open an Excel worksheet and paste the data into a new sheet. Notice that the fields placed on the **Rows**, **Columns**, and **Color** shelves are copied into the sheet. However, the **Customer Segment** field is not copied because it is an external filter (it appears only

on the **Filters** shelf).

	A	B	C	D	E
1	Customer Name	Product Category	Region	SUM(Sales Total)	
2	Maurice Satty	TECHNOLOGY	CENTRAL	19,214.25	
3	Victor Price	TECHNOLOGY	CENTRAL	12959.78	
4	Deanra Eno	TECHNOLOGY	EAST	10455.7	
5	Speros Goranitis	TECHNOLOGY	EAST	14272.76	
6	Meg Tillman-Sact	FURNITURE	WEST	14584.2	
7	Skye Norling-Chri	TECHNOLOGY	WEST	11106.19	
8	Noel Staavos	TECHNOLOGY	WEST	29689.57	
9	Stephanie Ulprigh	TECHNOLOGY	EAST	51280.73	

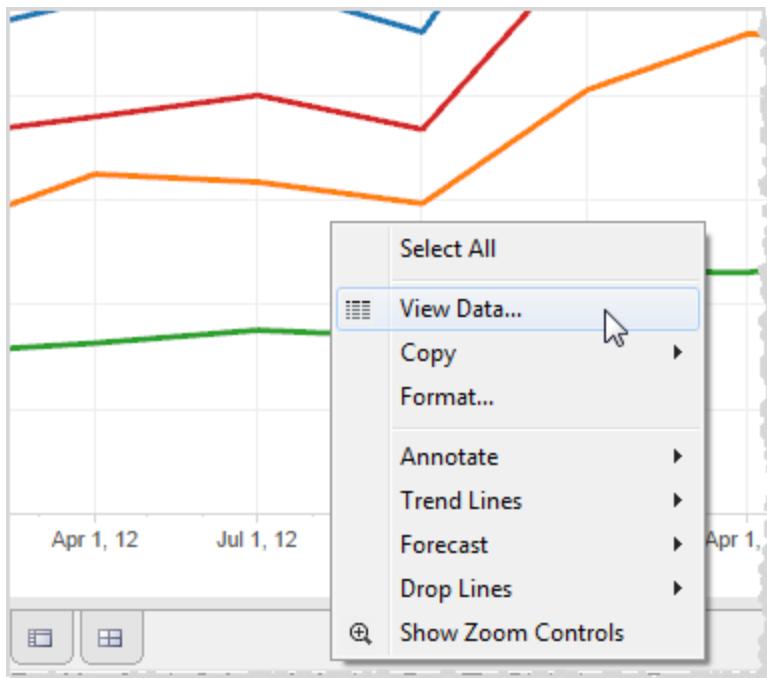
Copy Underlying Records to Clipboard

Copying underlying data can be used to copy the disaggregated data behind a view.

To copy underlying records:

1. Select the desired data in Tableau.

Right-click (Control-click on Mac) the selected records and select **View Data** on the context menu.



- 2.
3. The resulting dialog box shows the Summary data along with the Underlying Data. Select

the data you want to copy and then click **Copy** in the upper right corner of the dialog box.

Market	Quarter of Date	Profit
West	January 1, 2012	\$7,209
West	April 1, 2012	\$7,596
West	July 1, 2012	\$8,006
West	October 1, 2012	\$7,354
West	January 1, 2013	\$10,874
West	April 1, 2013	\$10,737
West	July 1, 2013	\$11,305
West	October 1, 2013	\$10,915
South	January 1, 2012	\$3,077
South	April 1, 2012	\$3,267
South	July 1, 2012	\$3,515
South	October 1, 2012	\$3,379
South	January 1, 2013	\$4,637
South	April 1, 2013	\$4,619
South	July 1, 2013	\$4,967
South	October 1, 2013	\$5,017
East	January 1, 2012	\$5,380

4. Open an Excel worksheet and paste the data into a sheet. Notice that the fields placed on the Rows, Columns and Color shelves are copied onto the worksheet.

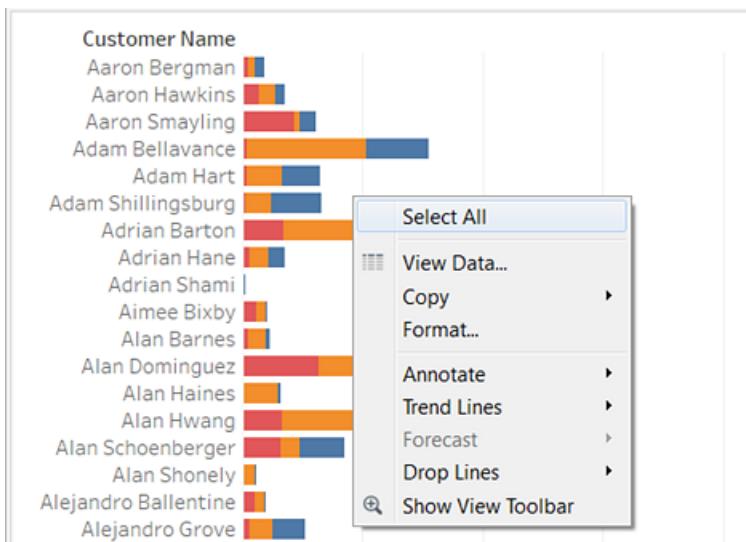
A	B	C	D
1 Market	Quarter	of Date	Profit
2 Central	January	1, 2012	\$9,109
3 Central	October	1, 2012	\$9,215
4 Central	April	1, 2012	\$9,826
5 Central	July	1, 2012	\$10,112
6 Central	October	1, 2013	\$13,686
7 Central	January	1, 2013	\$13,737
8 Central	April	1, 2013	\$13,888
9 Central	July	1, 2013	\$14,279
10 East	January	1, 2012	\$5,380
11 East	October	1, 2012	\$5,936
12 East	July	1, 2012	\$6,346
13 East	April	1, 2012	\$6,499
14 East	January	1, 2013	\$8,110
15 East	October	1, 2013	\$8,809
16 East	July	1, 2013	\$8,952
17 East	April	1, 2013	\$9,185
18 South	January	1, 2012	\$3,077

Export Records To Microsoft Access

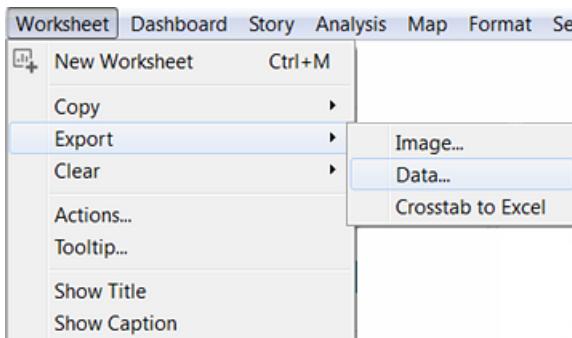
This topic describes how to create an Access database from Tableau data.

Note: Access is not available on the Mac.

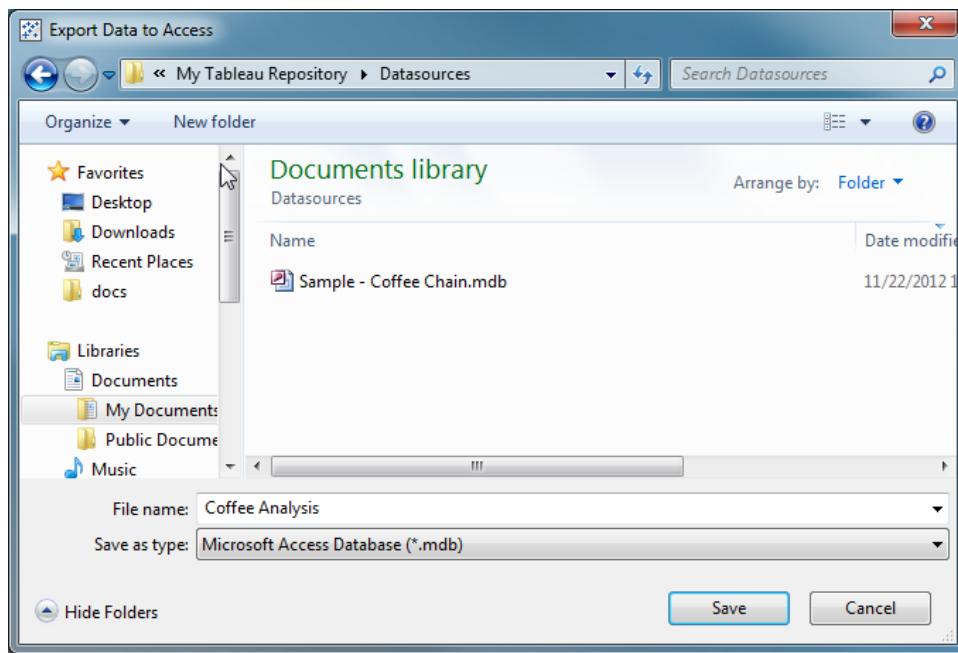
1. Select the desired data in Tableau. Right-click (Control-click on Mac) the view and then select **Select All**.



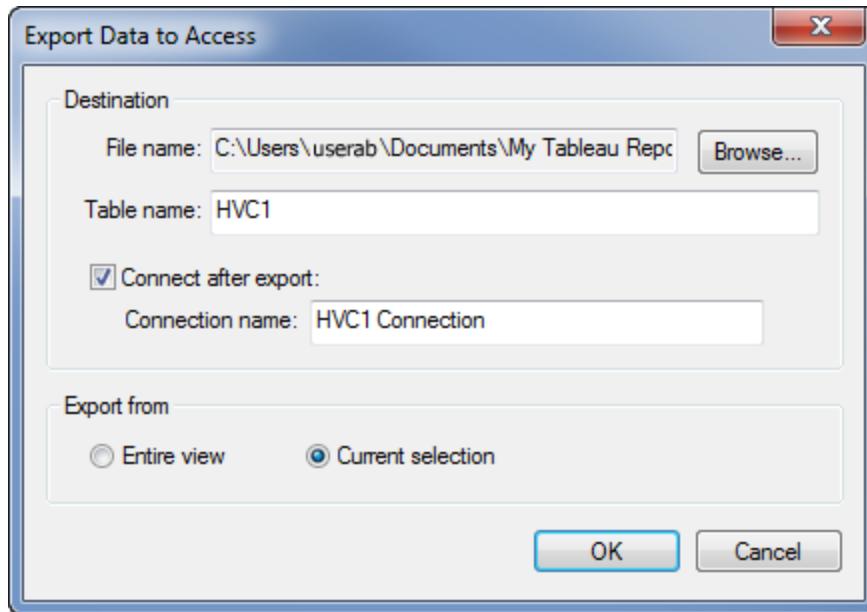
2. Select **Worksheet > Export > Data**.



3. Select a location and name for your Access database. Access databases end with a .mdb file extension.



4. Type a name for the database and click **Save**. The Export Data to Access dialog box is now displayed. The Connect after export option allows you to immediately connect to the new data source and continue working in Access without interrupting your work flow.



Copy Crosstab to Clipboard

You can copy a crosstab (text table) version of a view to your local Clipboard and transfer it from the Clipboard to another application. For instance, you might want to transfer a crosstab in Tableau to a crosstab in Microsoft Excel. Or you might want to transfer the data behind a

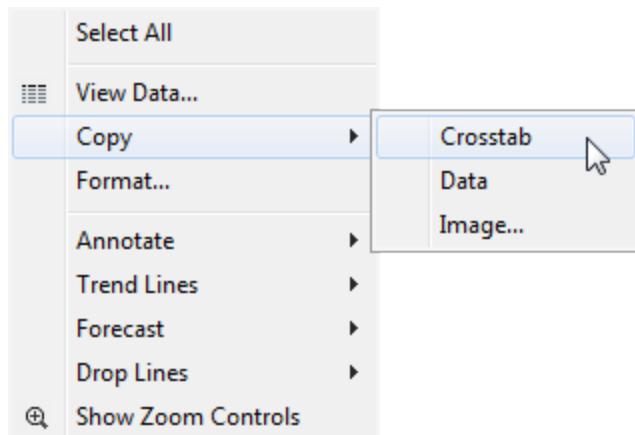
graphical view in Tableau to Excel in a crosstab format. Copying a crosstab to the Clipboard is restricted by the following general rules:

- You must copy all records in the view. You cannot copy a subset of records.
- This option is valid for aggregated views only. It cannot be used on disaggregated views of data, because a crosstab is by definition an aggregated view of data. In other words, the **Aggregate Measures** option on the Analysis menu must be selected in order for this option to work properly.

Other restrictions may apply depending on the data in your view. You cannot copy a crosstab if the view contains continuous dimensions such as continuous dates and times.

To copy a view as a crosstab to the clipboard:

1. Right-click (Control-click on Mac) any view in Tableau and select **Copy > Crosstab**.



2. Open an Excel worksheet and select Paste from Excel's Edit menu or press Ctrl-V (⌘V on a Mac).

	A	B	C	D	E	F	G	
1		Quarter of Date	Quar					
2	Market	1-Jan-12	1-Apr-12	1-Jul-12	1-Oct-12	1-Jan-13	1-Apr-13	
3	Central	\$9,109	\$9,826	\$10,112	\$9,215	\$13,737	\$13,888	
4	East	\$5,380	\$6,499	\$6,346	\$5,936	\$8,110	\$9,185	
5	South	\$3,077	\$3,267	\$3,515	\$3,379	\$4,637	\$4,619	
6	West	\$7,209	\$7,596	\$8,006	\$7,354	\$10,874	\$10,737	
7								

The pasted data always appears as a crosstab in Excel, even if the initial view of the data in Tableau was not in a crosstab format.

Export Crosstab to Excel

There is a more direct way to transfer a cross-tab view of data to Microsoft Excel on a Windows computer. Select **Worksheet > Export > Crosstab to Excel**.

Tableau automatically pastes a crosstab version of the current view into a new Excel workbook. This option automatically opens a new instance of the Excel application.

Although this is more direct, it can decrease performance because it is copying the formatting as well as the data. If the view you are exporting contains a lot of data, a dialog box opens asking whether you want to copy the formatting options. Disregarding the format can enhance performance.

On a Mac computer, **Export Crosstab to Excel** opens a dialog box where you can save the file. You must then manually open the file in Excel.

Extracting Data

Another way to export all data or a subset of your data to a new data source is to use Tableau Extracts. For details, see [Extract Your Data on page 440](#).

Export as an Image

If you want to transfer your Tableau results into a presentation, report or web page, Tableau provides the following several options.

[Copy your View to Another Application](#)

You can copy your view as an image and paste it into another application such as Microsoft PowerPoint, Word, or Excel. If you're using Tableau Desktop on a Mac, a TIFF (Tagged Image File Format) is created for pasting. On Windows, a BMP (Bitmap) is created. Either image format will include everything in your view, including Tableau fonts.

If you're running Windows, you can create an EMF (Enhanced Metafile) image instead of a BMP, but if your view includes a Tableau font (such as Tableau Regular, Tableau Semibold, etc.), a font that's similar to the Tableau font will be substituted. See [\(Windows only\): Create an EMF of your view below](#) for steps.

To create a vector-based file of your view that embeds the Tableau fonts, you can use **File > Print to PDF**. See [Print to PDF on page 1290](#) for details.

[Create an image of your view](#)

1. Select **Worksheet > Copy > Image**.
2. In the Copy Image dialog box, select what you want to include in the image, along with the legend layout (if the view contains a legend).
3. Click **Copy**. Tableau copies the current data view as a BMP (Windows) or a TIFF (Mac) and places it on the Clipboard.
4. Open the target application and paste the image from the Clipboard.

[\(Windows only\): Create an EMF of your view](#)

You can create an EMF of your view instead of a BMP by using the Export command. If your view uses a Tableau font, it will be substituted with a similar (non-Tableau) font.

1. Select **Worksheet > Export > Image**.
2. In the Export Image dialog box, select what you want to include in the image, along with the legend layout.
3. Click **Save** and in the Save Image dialog, save the image as Enhanced Metafile (*.emf).

Export to an Image File

The export image command saves the current view as an image file. You can export to an image file with the following three steps.

1. Select **Worksheet > Export > Image**.
2. In the Copy Image dialog box, select the contents you want to include in the image and the legend layout (if the view contains a legend).
3. Click **Save**.
4. In the Save Image dialog box, navigate to where you want to save the image file and type a file name into the text box. Select a file format from the Save as type drop-down menu.
5. Click **Save**.

Print to PDF

You can publish one or more views to PDF by selecting **File > Print to PDF**.

When printing a sheet, filters in the view are not included. To show filters, create a dashboard containing the sheet and print the dashboard to PDF. Note, when printing a dashboard to PDF, the contents of web page objects are not included.

Exporting the Data Source

After connecting to data, you can save the data source to your repository. Saving the data source creates a shortcut to the data and lets you avoid having to create a new connection every time you want to use that data source. Exporting the data source is useful if you have added custom fields such as groups or sets to the Data pane. See [Export Data Sources on page 461](#).

Copying Information Between Workbooks

You can combine resources from different workbooks into one, or pull information from a larger workbook into a standalone subset. To share or extract a subset of information, you can use any of the following actions on worksheets, dashboards, and stories:

- Copy and paste selected sheets to another workbook.
- Import a saved workbook into your current workbook.

- Export selected sheets to a new workbook.

What gets Copied or Saved with Selected Sheets

When you copy, save, or export selected sheets, other workbook items that the views in those sheets depend on are also included:

- Any associated visible or hidden sheets.
- Data sources for fields used in the selected sheets.
- Calculations, parameters, groups, sets, actions, custom shapes, and so on.

How Tableau Handles Duplicate Items

When you paste or import sheets from a different workbook, some items might already exist in the destination workbook, or some items might have the same name in both places. If Tableau encounters an exact duplicate item in the Data pane, such as a calculation, it does not paste or import that item into the destination workbook. However, if an item in the Data pane has the same name but is defined differently, Tableau imports and renames it.

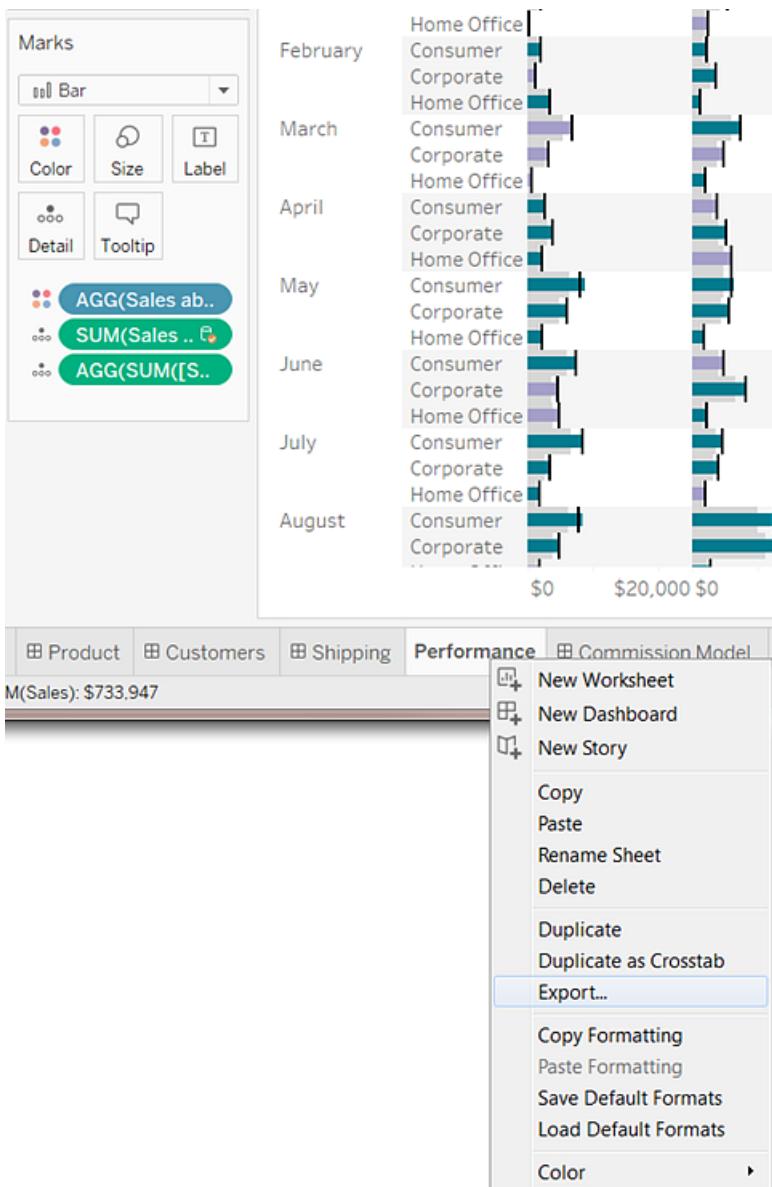
Tableau also pastes or imports sheets and data sources with duplicate names—whether only the name is identical, or their names and contents are identical—and it renames the newer copy.

Export and Import Sheets

If you want to extract a subset of information from a larger workbook to maintain as a standalone file, you can export or save selected sheets to a new workbook. You can then import that workbook into an existing one to incorporate its sheets and other objects into the existing workbook.

Note: These steps describe how to share your work among Tableau workbooks. You can also export views to use outside of Tableau. For information, see [Exporting Your Work on page 1275](#).

1. Open the workbook that contains the sheets you want to export to a new file.
2. Using the tabbed worksheet, filmstrip view, or sheet sorter view, right-click (Control-click on Mac) the sheet tab or thumbnail view, and then select **Export** to export a single sheet. Use Shift + click or Ctrl + click to select multiple sheets.
3. In the Save As dialog box, specify the file format you want to save (.twb or .twbx), select the location for the new workbook file, give it a name, and then click **Save**.

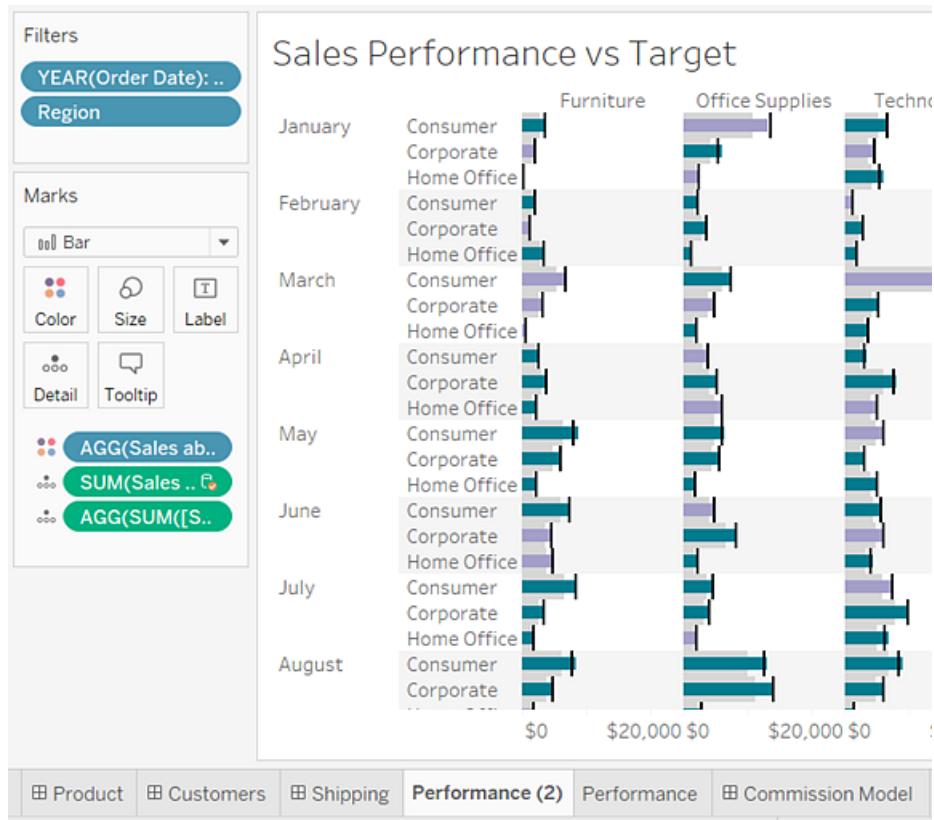


Import a Tableau Workbook

After you save or export selected sheets to a new workbook (.twb) file, you can import the information into another workbook.

1. With the existing workbook open, select **File > Import Workbook**.
2. Select the workbook that contains the sheets you saved from another workbook, and click **Open**.

The following image shows the result of importing a workbook that contains a sheet with the same name as a sheet in the existing workbook. Tableau adds a number after the name of the imported sheet.



Copy and Paste Sheets

Copying and pasting sheets is a quick way to combine information from different workbooks or create a new workbook. You can copy one or more sheets from the tabbed sheet view, filmstrip view or sheet sorter view. Use Shift + click or Ctrl + click to select multiple sheets.

To copy and paste a worksheet using the filmstrip view, do the following:

1. Open a workbook and click the **Filmstrip** button in the status bar.



2. Select the thumbnails of the sheets you want to copy, then right-click (Control-click on Mac) and select **Copy**.

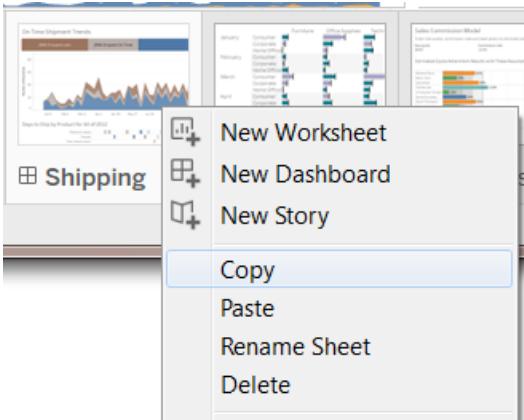


Tableau copies the information in the file format (.twb or .twbx) of the workbook.

3. Open the destination workbook, or create a new workbook. Right-click (Control-click on Mac) on the tab for any sheet, and select **Paste**.

Pasted sheets are placed after existing worksheets, dashboards, and stories.

Note: The **Paste** option is not available when the active sheet is a story.

4. Save the changes.

For information about what gets copied when you copy and paste sheets, see [Copying Information Between Workbooks on page 1283](#)

Print

After creating a view or several views in Tableau you can print them. The first thing you should do before printing is specify how you want the printed page to look using the Page Setup dialog box. Then you can print to a printer or publish to a PDF. You can also print the Tableau Help directly from your web browser or to a PDF.

Printing Results

Once you have a view or several views created in Tableau, you can print them or publish them as a PDF.

Page Setup

Before you print, there are several options you can set to specify how the worksheet will look when it is printed. For example, you can select which elements to include, set the printed page orientation, specify where you want to put the legend, margins, and more. You define these settings in the Page Setup dialog box. You can set different page setup options for each worksheet in the workbook. That way you can have different titles, captions, legend settings,

etc., for each worksheet you want to print. To open the Page Setup dialog box, select **File > Page Setup**.

The Page Setup dialog box has the following tabs:

General

Use the General tab to select the elements you want to show when you print. You can show or hide the title, view, caption, color legend, shape legend, size legend, and map legend.

Specify how to handle headers and breaks. The headers refer to the headers in each of your views. When you select **Repeat headers and legends on each page**, the row and column headers will appear at the top of each printed page when a view breaks across several pages.

Select **Break pages on pane boundaries** to prevent page breaks in the middle of a cell in a table.

If you have used the Pages Shelf to build your view, you can select whether to print the current page only or all pages.

Layout

Use the Layout tab in the Page Setup dialog box to specify the layout legend, page margins, and centering options for printing.

Legend Layout

If you include one or more legends, you can select one of the options for how you want the legends to appear on the printed page.

Margins

Specify top, bottom, left, and right margins by typing values into the text boxes.

Centering

Optionally, select whether to center the view horizontally or vertically--or both--on the page.

Print Scaling

Use the Print Scaling tab to fit the view to a certain size or to change the page orientation. These options only affect printed documents. The scaling options you specify here will not affect exported images or PDFs. However, the orientation settings will be used as the default when you publish the workbook to Tableau Server or Tableau Public.

Print Scaling

You can scale your view to fit within a single page or scale it across multiple pages. Select from the following options:

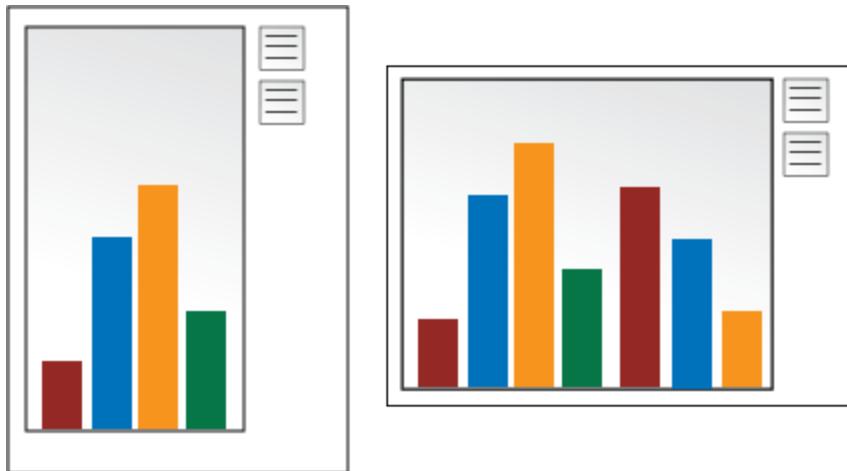
- Automatic – Scales the view automatically based on the paper size.
- Scale to – Scales the view to the specified percentage of its original size.
- Fit to – Scales the view to fit within the specified area. Select the number of printed pages across and down. For example, if you have a really wide view that is not very tall, you can specify three pages across by one page down.

Page Orientation

Use the page orientation settings to specify how you want the view oriented on the printed page. For example, if you have a view that is very wide but not very high, select **Landscape**. Select from the following page orientation options:

- Use Printer Setting – Use the page orientation that is already specified by the printer.
- Portrait – Presents the view so that it is oriented vertically on the printed page.
- Landscape – Presents the view so that it is oriented horizontally on the printed page.

The following diagram shows the difference between portrait and landscape page orientations.



These page orientation settings are used as the default settings when you publish the workbook to Tableau Server or Tableau Public.

Printing

After you have configured the [Page Setup](#) on page 1287 settings, you can print by selecting **File > Print**. In the Print dialog, select a printer, decide whether to show selections, specify a print range, and select the number of copies you want to print.

The following options in the Print dialog box are unique to Tableau.

Show Selections

When this option is selected any selections you've made in the views will be maintained while printing.

Changing the Print Range

When you print from a workbook with multiple worksheets, each worksheet represents one or more printed pages, depending on the page setup.

Select from the following print ranges:

- **Entire Workbook** - Prints all the worksheets in the workbook.
- **Active Sheet** - Prints only the sheet currently displayed in the workbook.
- **Selected Sheets** - Prints the selected sheets.

You can select multiple worksheets in a workbook by holding down the CTRL or Shift keys (or the ⌘ key on a Mac) while clicking the worksheet tabs that you want to select.

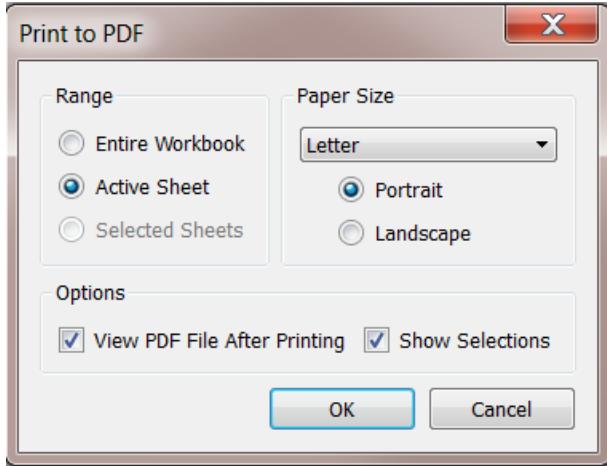
Print to PDF

In Tableau, you can publish views as PDF files rather than printing them as hard copies. You do not need to have Adobe Acrobat installed on your computer.

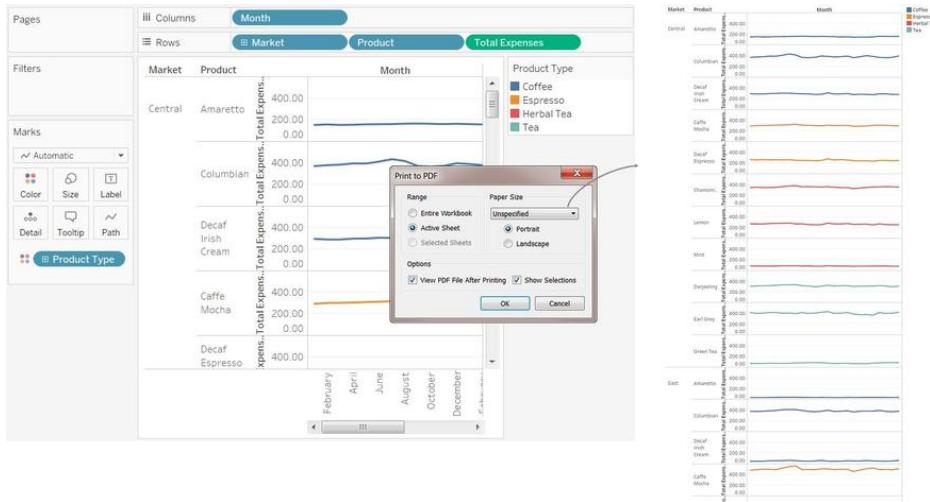
For information about how to use the **Pages** shelf to create a different view on each page, how to include page numbering and headers in the PDF, or how to troubleshoot print problems when exporting to a PDF, see [Tips for Printing and Print to PDF](#) in the Tableau Knowledge Base.

Publish as a PDF using a Windows computer

1. Specify [Page Setup](#) on page 1287 options for each sheet in your workbook.
2. Select **File > Print to PDF**.



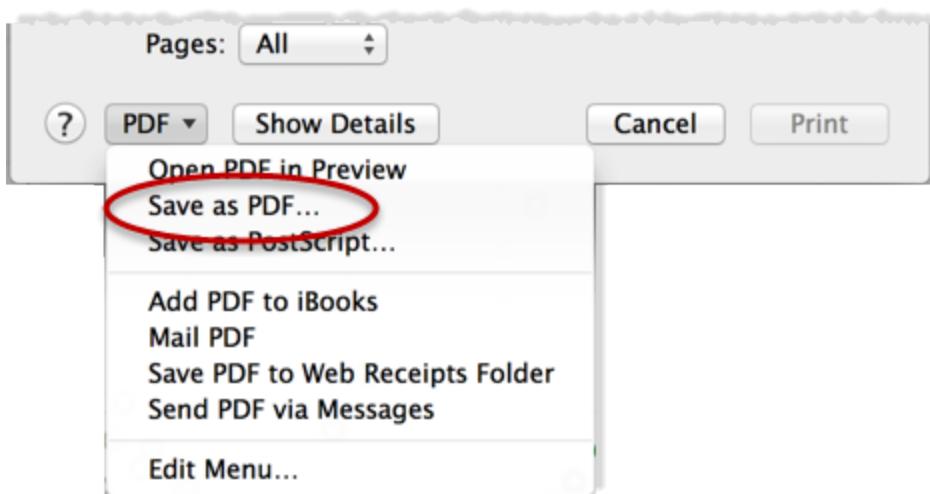
3. In the Print to PDF dialog box, select the print Range:
 - Entire Workbook - Publishes all the sheets in the workbook.
 - Active Sheet - Publishes only the sheet currently displayed in the workbook.
 - Selected Sheets - Publishes the selected sheets.
4. Select a **Paper Size**. If you select Unspecified, the paper size will expand to the necessary size to fit the entire view on a single page.



5. Select **View PDF File After Printing** if you want to automatically open the PDF after creating it. This option is only available if you have Adobe Acrobat Reader or Adobe Acrobat installed on your computer.
6. Select whether to **Show Selections**. When this option is selected the selections in the views are maintained in the PDF.
7. Click **OK** and specify where you want to save the PDF. Then click **Save**.

Publish as a PDF using a Mac computer

1. Specify **Page Setup** on page 1287 options for each sheet in your workbook.
2. Select **File > Print**.
3. In the Print dialog box, click **Show Details** to select a print range:
 - Entire Workbook - Publishes all the sheets in the workbook.
 - Active Sheet - Publishes only the sheet currently displayed in the workbook.
 - Selected Sheets - Publishes the selected sheets.
4. Click **PDF > Save as PDF**.



5. Specify where you want to save the PDF, then click **Save**.

Printing the Help

You can print an individual Help topic using the print options in your Web browser.

The screenshot shows the Tableau Desktop online help interface. At the top left is the Tableau logo followed by the word "help". A search bar is at the top right. On the left, a "Contents" tab is selected, showing a list of topics: Welcome to Tableau Desktop, Getting Started, Connect to Data, Building Data Views, Advanced Analysis, Publish and Share, Reference, Examples, Copyright, and Contact. The "Welcome to Tableau Desktop" topic is highlighted. The main content area on the right displays the "Welcome to Tableau Desktop" page with the following text:

Welcome to Tableau Desktop

This online Help system contains the complete product documentation for Tableau Desktop. Features such as full text search make it easy for you to find what you are looking for. The online Help is divided into sections, which are displayed as books in the Contents tab on the left.

[Getting Started](#) - learn the basic concepts in Tableau and complete a step-by-step tutorial.

[Connect to Data](#) - connect to data, join tables, extract data, and manage queries.

[Building Data Views](#) - drag and drop fields to create views of your data, then refine the view.

In addition to printing individual help topics, you can also download an offline help system and a printable PDF.

Reference

Upgrade Tableau Desktop, consult a Quick Start, find out how to connect to a particular data source, look up Tableau functions, keyboard shortcuts, connectors, glossary definitions, and more.

Connector Examples

Follow the link below for information on how to connect to your specific data source.

Connectors are listed in the order that they appear on the **Connect** pane.

Excel

This topic describes how to connect Tableau to Microsoft Excel file data and set up the data source. Tableau connects to .xls and .xlsx files.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Excel**.
2. Select the Excel workbook you want to connect to, and then click **Open**.

Alternatively, in Tableau Desktop on Windows, to connect to the Excel file using the Microsoft Jet-based connection, in the Open dialog box, click the **Open** drop-down menu, and then select **Open with Legacy Connection**. The Data Source page appears.

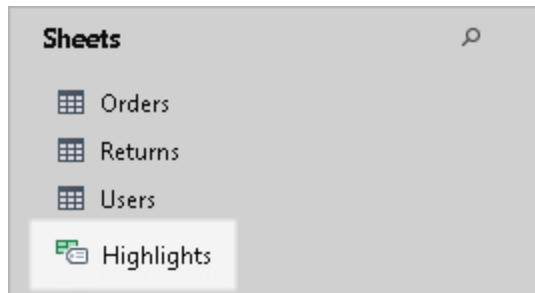
3. On the Data Source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. If your Excel file has one table, click the sheet to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

You can also connect to a named range or an Excel table (also known as an Excel list) in the same way you connect to a worksheet. Both the named range and Excel table function as a table in Tableau.

You create named ranges in Excel by selecting a range of cells and then selecting **Define Name** on the **Formulas** tab. Similar to named ranges, you can create an Excel table in Excel by selecting a range of cells, and then selecting **Insert >**

Table. When you connect to a named range or Excel table in Tableau, an icon appears next to the sheet in the Data Source page as shown below.



You can connect to multiple Excel workbooks at the same time as long as each connection in the data source has a unique name.

Note: Tableau does not support pivot tables in Excel.

Optional settings

You can set the following options before you build the view.

[Set Excel table options](#)

On the canvas, click the table drop-down arrow and then specify whether the data includes field names in the first row. If so, these names will become the field names in Tableau. If field names are not included, Tableau generates them automatically. You can rename the fields later.

[Clean your data with Data Interpreter](#)

If Tableau detects that it can help optimize your data source for analysis, it prompts you to use Data Interpreter. Data Interpreter can detect sub-tables that you can use and remove unique formatting that might cause problems later on in your analysis. For more information, see

[Clean Your Data with Data Interpreter on page 428](#)

[Review and manage the data source](#)

In addition to setting table options and cleaning your data, you can make the following configuration changes to the data source before you start your analysis:

- **View the data in the Tableau data source** – In the grid, click **Update Now** to preview the first 1000 rows of your live or extract data source. If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes. If you want changes to automatically appear in the grid, click **Automatically Update**.
- **Manage metadata** – Click the metadata grid button to quickly examine the general structure of the Tableau data source and its fields and to perform routine management tasks like sorting fields and rows, hiding multiple fields at once, or quickly renaming or

resetting fields.

- **Sort fields and rows** – From the **Sort fields** drop-down list, select how you want to sort the columns in the grid or metadata grid. You can sort the columns by data source or table order. Sort the row values by clicking the sort button next to the column name.
- **Hide fields** – Hide a field by clicking the column drop-down arrow and selecting **Hide**.
- **Rename fields and reset field names** – To rename the field, in the data preview grid, double-click the field name, or in the metadata grid, click the field name. You can also select a column or multiple columns, click the column drop-down, and then select **Reset Name** to revert back to the original name of the field
- **Pivot columns** – Pivot fields to transform data in a crosstab format into a columnar format. This is only available for non-legacy connection types. For more information, see [Pivot Data from Columns to Rows on page 423](#).
- **Split columns** – Split a string field into multiple fields. This is only available for non-legacy connection types. For more information, see [Split a Field into Multiple Fields on page 429](#).
- **Create calculations** – Create a new calculation based on existing fields in the Tableau data source. Click the column drop-down arrow and select **Create Calculated Field**.
- **Copy values** – Copy values in the grid by selecting the values and then pressing Ctrl+C on Windows or Command-C on a Mac. Alternatively, to copy values in the metadata grid, select the values, right-click on Windows or Control-click on a Mac, and then select **Copy**.
- **Connect live or use an extract** – At the top of the data source page, select a live or extract connection to the data source. If you choose to take an extract, an **Edit** link displays that you can use to set up filters that define a subset of the data to include in the extract.
- **Add data source filters** – At the top of the data source page, add data source filters to restrict the visibility and use of the fields in the data source.

Microsoft Excel data source example

An example of a Microsoft Excel data source is shown below.

Category	Q1	Q2	Q3	Q4
Appliances	13,130	38,266	15,543	64,655
Binders and Binder A...	85,668	47,569	51,891	78,370
Bookcases	17,061	22,741	14,490	36,947
Chairs & Chairmats	66,384	48,000	89,280	164,776
Computer Peripherals	17,979	36,158	31,308	69,716
Copiers and Fax	40,452	30,203	19,398	88,565
Envelopes	4,298	17,730	9,621	25,429
	742	857	2,660	2,210

Note: If the Excel file contains columns that are more than 254 characters wide, Tableau Desktop can't use these fields for workbooks that were created before Tableau Desktop 8.2. Also, you cannot use the legacy connection to connect to this data. Either remove the columns, modify them to fit within 254 characters prior to connecting in Tableau Desktop, or upgrade the Excel data source. For more information about upgrading data sources, see [Upgrade Data Sources](#) on page 464.

Tips for Working with Your Data

Data can be organized in various ways. To take advantage of Tableau Desktop, Tableau recommends that you connect to data that has not been formatted. That is, data that has not already been organized in a crosstab, aggregated, or contains other extraneous information. If your data is like this, options for viewing, aggregating, and grouping the data can become limited because Tableau doesn't have the information it needs to see or distinguish individual data points from the other information that might be associated with the data.

In most cases, if you have data like this, you can connect directly to it from Desktop and use Data Interpreter or other cleaning options to work around the problematic structure of the data. In other cases, you might need to edit the Excel data directly before connecting to it from Desktop.

The following sections highlight and provide suggestions for resolving some common formatting or issues that can make analyzing your data difficult.

In this topic:

- [Pivot data in crosstab format](#) below
 - [Remove pre-aggregated data](#) below
 - [Remove or exclude introductory text](#) on the next page
 - [Flatten hierarchical headers to a single row](#) on page 1300
 - [Remove blank rows](#) on page 1302
 - [Add missing headers](#) on page 1302
-

Pivot data in crosstab format

When data is formatted as in crosstab format, the table is column oriented. In a column oriented table, the variables are stored as column headers. However, Tableau Desktop is optimized for row oriented data. In a row-oriented table, the variables are stored in the row values.

For example, suppose you have column-oriented table, which shows math, science, and history scores for grade school students.

ID	Gender	School	Math	Science	History
1	M	West	90	80	80
2	F	South	50	50	50
3	M	Central	50	80	80

Tableau Desktop is optimized to connect to row-oriented tables, where math, science, and history values are organized under a column called "Subject" and the scores for each student are organized under a column called "Score." You can pivot the columns in the rows, by manually editing your Excel data. Alternatively, connect to your Excel data from Desktop and then use the pivot option. For more information about the pivot option, see [Pivot Data from Columns to Rows](#) on page 423.

ID	Gender	School	Subject	Score
1	M	West	Math	90
1	M	West	Science	80
1	M	West	History	80
2	F	South	Math	50
2	F	South	Science	50
2	F	South	History	50
3	M	Central	Math	50
3	M	Central	Science	80
3	M	Central	History	80

Remove pre-aggregated data

Data can often come pre-aggregated. That is, data can contain sums, averages, medians, etc. A common example of pre-aggregated data comes in the form of subtotals and grand totals.

Subtotals and grand totals data are computed from the raw data, but is not part of raw data itself.

For example, suppose you have a table that contains a row of subtotals information.

ID	Gender	School	Math	Science	History
1	M	West	90	80	80
2	F	South	50	50	50
3	M	Central	50	80	80
4	M	Central	100	90	80
5	F	West	90	100	80
Subtotals			380	400	370

In this case, pre-aggregated data needs to be removed. To use subtotals and grand totals in your analysis, manually remove this type of information from your table. Then, connect to your Excel data from Desktop and calculate subtotals and totals using the totals option. For more information, see [Totals on page 976](#). Alternatively, connect to your Excel data from Desktop, turn on Data Interpreter, and then use the totals option. For more information, see [Clean Your Data with Data Interpreter on page 428](#).

ID	Gender	School	Math	Science	History
1	M	West	90	80	80
2	F	South	50	50	50
3	M	Central	50	80	80
4	M	Central	100	90	80
5	F	West	90	100	80
Subtotals			380	400	370

[Remove or exclude introductory text](#)

Excel data that is delivered as a report can contain titles or blocks of introductory text. Because Desktop expects either column headers or row values in the first row of a table, this information can cause problems during your analysis.

For example, suppose you have a table that contains a report title and date.

Excel Report						
January, 2017						
	ID	Gender	School	Math	Science	History
	1	M	West	90	80	80
	2	F	South	50	50	50
	3	M	Central	50	80	80
	4	M	Central	100	90	80
	5	F	West	90	100	80

In this case, the title and date information needs to be removed. To use a title and date for a report, do one of the following:

- Manually remove this information from your Excel data. Then connect to your Excel data from Desktop and add a report title using the title option. For more information, see [Titles on page 234](#).
- Connect to your Excel data from Desktop, turn on Data Interpreter, and then use the title option. For more information, see [Clean Your Data with Data Interpreter on page 428](#).
- If you cannot remove this information from your Excel data, create a named range and connect to the named range from Desktop. For more information, see [Excel on page 1294](#).

Excel Report						
January, 2017						
	ID	Gender	School	Math	Science	History
	1	M	West	90	80	80
	2	F	South	50	50	50
	3	M	Central	50	80	80
	4	M	Central	100	90	80
	5	F	West	90	100	80

[Flatten hierarchical headers to a single row](#)

In general, Tableau Desktop expects only the first row in your Excel data to contain column headers. Data that contains multiple layers of column headers can cause problems during your analysis.

For example, suppose you have a table that contains one major header and multiple sub-headers.

California					
ID	Gender	School	Math	Science	History
1	M	West	90	80	80
2	F	South	50	50	50
3	M	Central	50	80	80
4	M	Central	100	90	80
5	F	West	90	100	80

In this case, the hierarchy of headers must be flattened or removed. To do this, you can manually create a new column for each header in the hierarchy directly in your Excel data. Alternatively, connect to your Excel data from Tableau Desktop and then turn on Data Interpreter. Verify that your headers are flattened correctly. For more information about Data Interpreter, see [Clean Your Data with Data Interpreter](#) on page 428.

ID	State	Gender	School	Math	Science	History
1	California	M	West	90	80	80
2	California	F	South	50	50	50
3	California	M	Central	50	80	80
4	California	M	Central	100	90	80
5	California	F	West	90	100	80

Make sure there are no blank cells

If you create new columns for your hierarchical headers, make sure that each cell in the new columns contains values.

ID	State	Gender	School	Math	Science	History
1	California	M	West	90	80	80
2		F	South	50	50	50
3		M	Central	50	80	80
4		M	Central	100	90	80
5		F	West	90	100	80
6	Oregon	M	North	70	80	100
7		F	East	80	80	100
8		F	West	50	80	80
9		F	West	100	80	100
10		M	West	80	80	90

While you might repeat the same value for each row, it's important that each row contains the data that associates it with the data that was stored in the hierarchical header. You must manually remove blank cells from your Excel data.

ID	State	Gender	School	Math	Science	History
1	California	M	West	90	80	80
2	California	F	South	50	50	50
3	California	M	Central	50	80	80
4	California	M	Central	100	90	80
5	California	F	West	90	100	80
6	Oregon	M	North	70	80	100
7	Oregon	F	East	80	80	100
8	Oregon	F	West	50	80	80
9	Oregon	F	West	100	80	100
10	Oregon	M	West	80	80	90

Remove blank rows

Make sure that there are no blank rows in your data. To fix blank rows, you must remove the blank rows from your Excel data.

ID	Gender	School	Math	Science	History
1	M	West	90	80	80
2	F	South	50	50	50
3	M	Central	50	80	80
4	M	Central	100	90	80
5	F	West	90	100	80
22	M	North	70	80	100
23	F	East	80	80	100
24	F	West	50	80	80
25	F	West	100	80	100
26	M	West	80	80	90

Add missing headers

Make sure that there are no missing column headers. To fix missing headers, you must manually add the missing headers directly to your Excel data.

	Gender	School	Math	Science	History
6	Oregon	M	North	70	80
7	Oregon	F	East	80	80
8	Oregon	F	West	50	80
9	Oregon	F	West	100	80
10	Oregon	M	West	80	80

Text File

This topic describes how to connect Tableau to a text file data and set up the data source. Tableau connects to delimited text files (*.txt, *.csv, *.tab, *.tsv).

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Text File**.
2. Select the file you want to connect to, and then click **Open**.

Alternatively, when using Tableau Desktop on Windows, to connect to the text file using the Microsoft Jet-based connection, in the Open dialog box, click the **Open** drop-down menu, and then select **Open with Legacy Connection**. The data source page appears.

3. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to. The default name is automatically generated based on the file name.
 2. Click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Note: For text files, custom SQL is available only when using the legacy connection or in workbooks that were created before Tableau Desktop 8.2.

Optional settings

You can set the following options before building the view.

Set text file options

On the canvas, click the table drop-down arrow and then select whether the first row contains column names. This option is selected by default. Alternatively, you can have Tableau generate names when you connect. These names can be changed later. You can also select **Text file properties** to specify the following:

- Select the character that is used to separate the columns. Select from the list of characters or select **Other** to type in a custom character.
- Select the text qualifier that encloses values in the text file.
- Select a character set that describes the text file encoding. The available encodings are based on the operating system you are using. For example, on Windows, ANSI is listed as windows-1252 and OEM is listed as 437.

Note: In workbooks created prior to Tableau Desktop 8.2 or that use the legacy connection, you can select ANSI, OEM, UTF-8, UTF-16, or Other. When you select Other, you must specify the character set in the provided text field. This value is verified when the connection is attempted.

- Select the locale by which the file should be parsed. This option tells Tableau which decimal and thousands separator to use.

Review the data, pivot, split, and create calculations

The first 1,000 rows of the data in the data source are automatically displayed below the grid. If you add tables, remove tables, or make changes to the join conditions, the grid updates. You can also do the following in the grid:

- Change the data type or geographical role of a column by clicking the data type icon.
- Hide a field by clicking the column drop-down arrow and selecting **Hide**.
- Rename a field by double-clicking the column name to rename it.
- Reset a field name by clicking the column drop-down arrow and selecting **Reset Name**.
- Sort fields in both the grid and metadata grid by selecting a sort option from the **Sort fields** drop-down list.
- Sort rows in the grid by clicking the sort button next to the column name.
- Pivot fields to transform data in a crosstab format into a columnar format. This is only available for non-legacy connection types. For more information, see [Pivot Data from Columns to Rows on page 423](#).
- Split a string field into multiple fields. This is only available for non-legacy connection types. For more information, see [Split a Field into Multiple Fields on page 429](#).
- Create new calculations based on existing field in the Tableau data source. Click the column drop-down arrow and select **Create Calculated Field**.
- Copy values in the grid by selecting the values and then pressing Ctrl+C (Command-C on a Mac). Alternatively, to copy values in the metadata grid, select the values, right-click (Control-click on a Mac), and then select **Copy**.

Examine the data source structure and perform management tasks

Use the metadata area to quickly examine the general structure of the Tableau data source and its fields. If you are working with a particularly large data source, use the metadata area to perform routine data management tasks such as hide multiple fields at once or quickly rename or reset fields.

Connect live or use an extract

At the top of the Data Source page, select **Live** or **Extract** mode. If you choose to create an extract, the **Edit** link displays. Click **Edit** to set up filters that define a subset of the data that you want to include in the extract.

Add data source filters

Add data source filters to restrict the visibility and use of the fields in the Tableau data source.

Best practices

Collect files in a single directory

Collect all relevant text files for a multi-table connection in a single directory, with nothing else in that directory, so that users cannot inadvertently select a file that is not appropriate for the connection.

Text file column width and size considerations

If the text file contains columns that are more than 254 characters wide, Tableau cannot use these fields for workbooks that were created before Tableau Desktop 8.2 or that use the legacy connection. Either remove the columns, modify them to fit within 254 characters prior to connecting in Tableau, or upgrade the text file data source.

Large text files often perform poorly as a data source, because the queries can take a long time.

Text file data source example

Here is an example of a text file data source:

Access

This topic describes how to connect Tableau to a Microsoft Access file (*.mdb, *.accdb) and set up the data source. Tableau supports all Access data types except OLE Object and Hyperlink.

Before you begin, gather this connection information:

- Access file name.
- If the file is password protected, you need the Database password.
- If the file has workgroup security, you need Workgroup security credentials:
 - Workgroup Information File name
 - User
 - Password

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Access**, select the Access file that you want to connect to, and then click **Sign In**.

Password protected

If the Access file is password protected, select **Database Password**, and then enter the password.

Workgroup security

If the Access file is protected by workgroup security, select **Workgroup Security**, and then enter the Workgroup Information File name, User, and Password into the corresponding text fields.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to. The default name is automatically generated based on the file name.
2. Drag a table to the canvas. You can drag a table or query.
3. Click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query](#) on page 404.

Optional Settings

You can set the following options before building your view.

Review the data and change data types

The first 1,000 rows of the data in the data source is automatically displayed below the canvas. If you add tables, remove tables, or make changes to the join conditions, the grid updates. You can also do the following in the grid:

- Change the data type of a column by clicking the data type icon.
- Rename or hide the column by clicking the column header drop-down menu and selecting the respective option.

Manage metadata

Click the metadata area button to perform routine management tasks.

Rename or hide columns

Rename or hide a column by clicking the column header drop-down menu and selecting the respective option.

Connect live or use an extract

At the top of the data source page, select a live or extract connection to the data source. If you choose to take an extract, an Edit link will display allowing you to set up filters that define the subset of the data to include in the extract.

Add data source filters

Add data source filters to restrict the visibility and use of the fields in the data source.

Access data source example

Here is an example of an Access data source:

The screenshot shows the Tableau Data Source setup process. In the top window, titled 'Sample - Coffee Chain', the 'Filename' field is set to 'H:\10.0\Sample - Coffee Chain.mdb'. A red arrow points from the 'Sign In' button to the 'Extract' radio button in the 'Connection' section of the main Tableau interface below. The main interface displays a 'Coffee Product' connection with the 'Extract' option selected. The preview pane shows a table with columns: Product Line, Product Type, Product, #, and Product Type. The data includes various coffee types like Beans, Espresso, and Decaf across different product lines and types.

Product Line	Product Type	Product	#	Product Type
Beans	Coffee	Decaf Irish Cream	3	Decaf
Beans	Espresso	Decaf Espresso	6	Decaf
Beans	Coffee	Amaretto	1	Regular
Beans	Coffee	Colombian	2	Regular
Beans	Espresso	Caffe Latte	4	Regular
Beans	Espresso	Caffe Mocha	5	Regular
Beans	Espresso	Regular Espresso	7	Regular

Note: If the Access file contains columns that are more than 254 characters wide, Tableau will not be able to use these fields. Either remove the columns from the table or modify them to fit within 254 characters prior to connecting with Tableau.

JSON File

This topic describes how to connect Tableau to a local JSON file and set up the data source. It also includes information about how folders are organized in the **Data** pane, why measures are

calculated, and tips for working with JSON data in Tableau.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **JSON File**, and then do the following:
 1. Select the file you want to connect to, and then click **Open**.
 2. In the Select Schema Level dialog box, select the schema levels you want to view and analyze in Tableau, and then click **OK**. For more information, see [Select schema levels on page 1311](#).
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. Click the sheet tab to start your analysis.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information see, [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add or remove tables, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button  to perform routine management tasks.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source, alphabetically, or alphabetically per table.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page, to add data source filters.

JSON file data source example

Here is an example of a JSON file as a data source using Tableau Desktop on a Windows computer:

The screenshot shows the Tableau Data Source Editor interface. On the left, there's a sidebar with 'Connections' and 'Files'. Under 'Connections', 'example_business' is selected. Under 'Files', 'example_business.json' is listed. The main area is titled 'example_business' and shows a preview of the data. At the top right, there are 'Connection' (radio buttons for 'Live' and 'Extract') and 'Filters' (0 | Add). Below the preview, there are filters for 'Sort fields', 'Data source order', and checkboxes for 'Show aliases' and 'Show hidden fields'. A row limit of '74' is set. The preview table has columns: '#', 'example_business.json Document Index (...', 'Category Index (g...', 'Category', 'example_business.json City', 'example_business.json Cost', 'example_business.json Id', 'example_business.json Name'. The data shows 74 rows of dessert items in Tucson.

Select schema levels

When you connect Tableau to a JSON file, Tableau parses the data in the JSON file and infers the schema from that process. Tableau flattens the data using this inferred schema. The JSON file schema levels are listed in the Select Schema Level dialog box.

The schema levels that you select in the dialog box determine which dimensions and measures are available for you to view and analyze in Tableau. If you select a child schema level, the parent level is also selected.

For example, here's a snippet of a JSON file:

The JSON file generates these schema levels:

The screenshot shows a JSON editor interface with a modal dialog titled "Select Schema Level". The dialog contains a table with columns "Schema" and "Example Value". The table lists schema levels and their example values, with some levels expanded to show child attributes. A message at the top of the dialog states: "The schema levels you select determine which dimensions and measures are available for analysis in the worksheet." Below the table, it says "4 schema levels selected". At the bottom right are "OK" and "Cancel" buttons.

	Schema	Example Value
<input checked="" type="checkbox"/>	example_business.json	
<input checked="" type="checkbox"/>	City	Tucson
	Cost	100
	Id	1
	Name	Yogurt Depot
	Revenue	2000
<input checked="" type="checkbox"/>	Category	["dessert", ...]
<input checked="" type="checkbox"/>	Stars_rated	
	customers_rated	10
	stars	5
<input checked="" type="checkbox"/>	Visits	
	day	Mon
	visit_count	300

Change schema levels

You can change the schema levels you selected by going to the data source page and selecting **Data > [JSON file name] > Select Schema Level**.

How dimension folders are organized for hierarchical JSON files

After you click the sheet tab, the selected schema levels of your JSON file show under **Dimensions** on the **Data** pane. Each folder corresponds with the schema level you selected, and the attributes associated with that schema level are listed as children of the folder.

For example, in the following screenshot, **Stars** is a dimension under the schema level **Stars rated** folder, and **Day** is a dimension under the schema level **Visits** folder. **Category** is also a schema level, but because it is a list of values and not a hierarchy of data, it doesn't require a folder.

The screenshot shows the Tableau Data pane with the 'example_business' dataset selected. The 'Dimensions' section is expanded, displaying various schema levels and their generated indexes:

- Example Business
 - Category
 - Category Index (generated)
 - City
 - Document Index (generated)
 - Id
 - Name
- Stars rated
 - Stars
 - Stars rated Index (generated)
- Visits
 - Day
 - Visits Index (generated)

At the bottom, there is a section labeled 'Measure Names'.

For each document, a unique index is generated and stored in the flattened representation of data. An index is also generated for each level in the schema.

For example, in the screenshot above, in addition to the **Document Index (generated)** entry, the **Category**, **Stars rated**, and **Visits** schema levels all have generated indexes.

Why measures are calculated in hierarchical JSON files

When a hierarchical JSON file is flattened, data might be duplicated. To keep measures consistent with their schema levels, Tableau creates level of detail (LOD) calculations to accurately represent the data at the schema level. The original measures are located in the **Source Measures** folder and you can use them, but we recommend that you use the calculated measures.

On the **Data** pane, calculated measures are labeled as `=#<measure name> per <parent name>`.

The screenshot shows the Tableau Data pane with the 'Measures' section expanded. It lists various calculated measures:

- Source Measures
 - Cost per Document
 - Customers Rated per Stars rated
 - Number of Records per Category
 - Number of Records per Stars rated
 - Number of Records per Visits
 - Revenue per Document
 - Visit Count per Visits
 - Latitude (generated)*
 - Longitude (generated)*
 - Number of Records
 - Measure Values

To view the LOD calculation for a measure, follow these steps:

1. Select the measure.
2. Click the drop-down arrow, and then select **Edit**.

The following example shows the LOD calculation for **Revenue per Document**. The formula selects the maximum revenue for each document index value.



The use of LOD calculations means that you can select multiple schema levels and be confident that the measures are not over-counted.

Tips for working with JSON data

These tips can help you work with your JSON data in Tableau.

- Do not exceed the 10x10 limit for nested arrays.

A high number of nested arrays creates a lot of rows. For example, 10x10 nested arrays result in 10 billion rows. When the number of rows Tableau can load into memory is exceeded, an error displays. In this case, use the Select Schema Level dialog box to reduce the number of selected schema levels.

- A data source that contains more than 100 levels of JSON objects can take a long time to load.

A high number of levels creates a lot of columns, which can take a long time to process. As an example, 100 levels can take more than two minutes to load the data. As a best practice, reduce the number of schema levels to just the levels that you need for your analysis.

- A single JSON object cannot exceed 128 MB.

When a single object top-level array exceeds 128 MB, you must convert it to a file where the JSON objects are defined one per line.

- The pivot option is not supported.

Statistical File

This topic describes how to connect Tableau to a statistical file and set up the data source. Tableau connects to SAS (*.sas7bdat), SPSS (*.sav), and R (*.rdata, *.rda) data files.

In this topic:

- **Make the connection and set up the data source** below
 - **(Optional) Review and manage the data source** on the next page
 - **R data file object and format support** on page 1317
 - **Change the character encoding for a statistical file** on page 1318
-

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Statistical File**, select the file that you want to connect to, and then click **Open**.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. Click the sheet tab to start your analysis.

For information about connecting to more than one table, see **Join Your Data** on page 354.

Statistical file data source example

Here is an example of a statistical file data source using Tableau Desktop on a Windows computer:

The screenshot shows the Tableau Desktop interface with a 'Loan' data source selected. The left sidebar has 'Connections' and 'Files' sections. The main pane shows a preview of the 'Loan.sas7bdat' file with the following data:

Loan Decision	Loan Status	Loan Amt	App Prior Customer	App Race	App Gen
REJECT	FINISHED	130.000	NO	CAUCASIAN	MALE
REJECT	FINISHED	141.000	NO	CAUCASIAN	MALE
REJECT	FINISHED	133.000	YES	CAUCASIAN	MALE
REJECT	FINISHED	160.000	NO	CAUCASIAN	MALE
REJECT	FINISHED	118.000	NO	CAUCASIAN	MALE
REJECT	FINISHED	94.000	NO	CAUCASIAN	MALE
REJECT	FINISHED	150.000	NO	CAUCASIAN	MALE

Note the following:

- The Statistical File connector does not support value labels.
- The Statistical File connector supports only one table per statistical file.

If an error message appears, "An error occurred while communicating with the data source," make sure that your statistical file contains only one object, and that the object is a data frame or a matrix. R files may contain hidden objects in what appears to be a clean workspace. To check for hidden objects, run `ls(environment(), all.names=TRUE)` from RStudio.
- You can change the character encoding for a statistical file. For more information, see [Change the character encoding for statistical files](#).

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

R data file object and format support

R data files should contain only one of the following types of objects:

- Two-dimensional matrices
- Vectors
- Factors
- Data frames

Note that if the R data file contains more than one object, Tableau will connect to the first one.

Supported R file formats are binary or ASCII. Compressed files are not supported in Tableau Desktop. You must decompress files first with another program, such as Gzip or WinZip, before you connect with Tableau.

Change the character encoding for a statistical file

Tableau reads the data in a statistical file (for example, an SAS or R file) based on the file's character encoding or on information in an input file. R files typically use the character encoding of the operating system, compared to SAS and SPSS files, which include the character encoding information in the file. Sometimes, you might need to specify a different encoding. For example, if a colleague sends you a statistical file with Greek character encoding, then you must specify a Greek character set to use the file with Tableau. If you need to use a different character set when reading from a statistical file, you can create a Tableau data source customization (TDC) file and specify the encoding to use.

Create the TDC file

A .tdc file is an XML file that applies to a single data source and contains vendor and driver name information of the data source provider. For the statistical file connector, the vendor and driver name is `stat-direct`.

To create a TDC file:

1. Open a plain text editor, such as Notepad.
2. Copy the information from the sample provided below, paste it into your text file, and then specify the `source-charset` value. (For a list of encodings, see [User-defined Encodings Supported by the Statistical File Connector](#) on page 1625.)
3. Save the file with a .tdc extension—for example, `r-statsfile.tdc`—to the My Tableau Repository\Datasources folder.

Sample TDC file sets the source-charset value

The following TDC file example sets the `source-charset` value to `shift-jis` for a statistical file data source.

```
<connection-customization class='stat-direct' enabled='true' version='10.0'>

<vendor name='stat-direct' />

<driver name='stat-direct' />

<customizations>

  <customization name='source-charset' value='shift-jis' />

</customizations>

</connection-customization>
```

Important: Tableau does not test or support TDC files. These files should be used as a tool to explore or occasionally address issues with your data connection. Creating and maintaining TDC files requires careful manual editing, and there is no support for sharing these files.

Other Files

This topic describes how to connect Tableau to supported file types, including Tableau Data Extract files and Tableau Workbooks.

1. On the start page, under **Connect**, click **Other Files**.
2. In the Open dialog box, navigate to and select a file.
3. Click **Open**.
4. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
5. Click the sheet tab to start your analysis.

Tableau Data Extract data source example

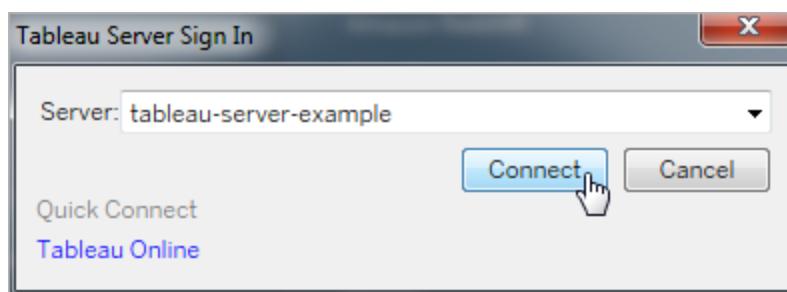
Here is an example of a Tableau Data Extract file data source using Tableau Desktop on a Windows computer:

The screenshot shows the Tableau Desktop application window. In the top left, there's a toolbar with icons for File, Data, Server, Window, and Help. Below the toolbar is a navigation bar with back, forward, and search buttons. On the left side, there's a sidebar titled 'Connections' with a 'Sample - Superstore orders (Extract)' entry selected. The main area is titled 'Superstore orders' and has tabs for 'Live' and 'Extract'. The 'Extract' tab is selected, showing a preview of the data. At the top of the preview pane, there are filters for 'Sort fields', 'Data source order', and checkboxes for 'Show aliases', 'Show hidden fields', and a row limit of '1,000 rows'. Below these filters is a table with columns: Row ID, Order ID, Order Date, Ship Date, Ship Mode, and Customer ID. The table contains 7 rows of data. At the bottom of the preview pane, there are buttons for 'Data Source', 'Sheet 1', and other sheet navigation.

Tableau Server or Tableau Online

This topic describes how to connect to a data source published to Tableau Server or Tableau Online.

1. On the start page, under **Connect**, click **Tableau Server**.
2. You can connect to Tableau Server or Tableau Online.
 - To connect to Tableau Server, enter the name of the server and then click **Connect**.
 - To connect to Tableau Online, click **Tableau Online** under **Quick Connect**.



Connected Desktop

You can stay connected to your Tableau Server and Tableau Online servers. Tableau Desktop saves each connection from session to session if you don't sign out. Next time

you launch Tableau Desktop, you are signed in to your most recent server connection. In this case, you can skip step 3.

You can also sign in to another Tableau Server and quickly switch between servers and sites from the Tableau Desktop **Server** menu. For more information, see [Quick Start: Stay Connected with Automatic Sign-In on page 1811](#).

Kerberos environment

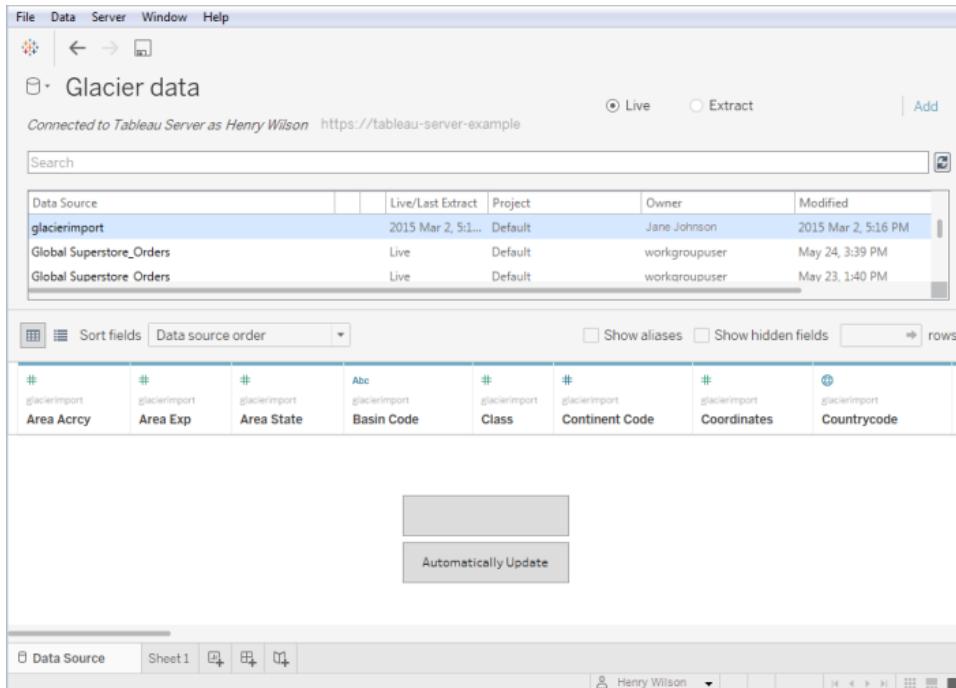
If Kerberos is enabled on Tableau Server and you are on a computer with valid Active Directory credentials, you will connect automatically to the server. In this case, you can skip step 3. If Kerberos authentication fails, you will be prompted to provide a user name and password for authentication. If you need to change the user you're signed in as, see [Switch Kerberos User](#).

3. To sign in:

- For Tableau Server, enter your user name and password.
- For Tableau Online, enter your email address and password.

4. Select a data source from the list of published data sources.

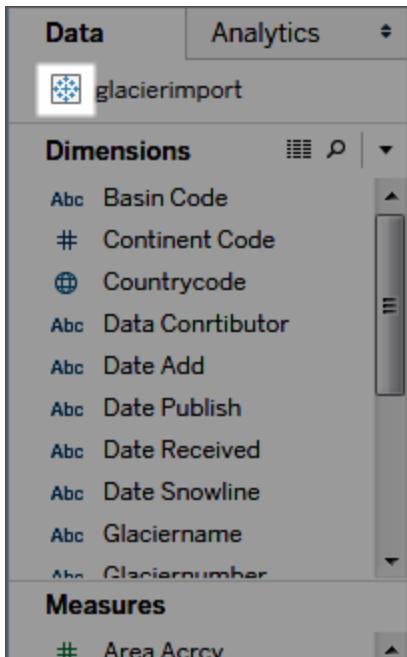
You can sort the list of data sources by clicking the column headers. Alternatively, search for a data source by using the search box. Click the refresh button  to refresh the list and show any new data sources.



Data Source	Live/Last Extract	Project	Owner	Modified
glacierimport	2015 Mar 2, 5:1...	Default	Jane Johnson	2015 Mar 2, 5:16 PM
Global Superstore_Orders	Live	Default	workgroupuser	May 24, 3:39 PM
Global Superstore Orders	Live	Default	workaroupuuser	May 23, 1:40 PM

Note: If you select a cube (multidimensional) data source, the Create Local Copy dialog box appears, and you must create a local copy of the data before you can start your analysis.

5. Click the sheet tab to start your analysis. Tableau Server or Tableau Online data sources are shown in the **Data** pane with a Tableau icon.



You may want to download a local copy of the data source so that you can, for example, work offline or make changes to a data source without modifying the original. To download a local copy, on the **Data** menu, select the data source, and then select **Create Local Copy**. A duplicate of the data source is added to the **Data** pane.

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

Actian Matrix

This topic describes how to connect Tableau to an Actian Matrix (formerly ParAccel) database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Database name
- User name and password

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Action Matrix**, and then do the following:
 1. Enter the name of the server that hosts the database and the name of the database that you want to connect to.
 2. Enter the user name and password, and then click **Sign In**.
If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, select a schema.
 3. Under **Table**, select a table or use the text box to search for a table by name.
 4. Drag a table to the canvas, and then click the sheet tab to start your analysis.
Use custom SQL to connect to a specific query rather than the entire data source.
For more information, see [Connect to a Custom SQL Query on page 404](#).

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

[View the data in the Tableau data source](#)

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

[Manage metadata](#)

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

[Sort fields and rows](#)

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

[Hide a field](#)

Click the column drop-down arrow and select **Hide**.

[Rename a field](#)

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

[Split columns](#)

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

[Create a new calculation based on an existing field in the data source](#)

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

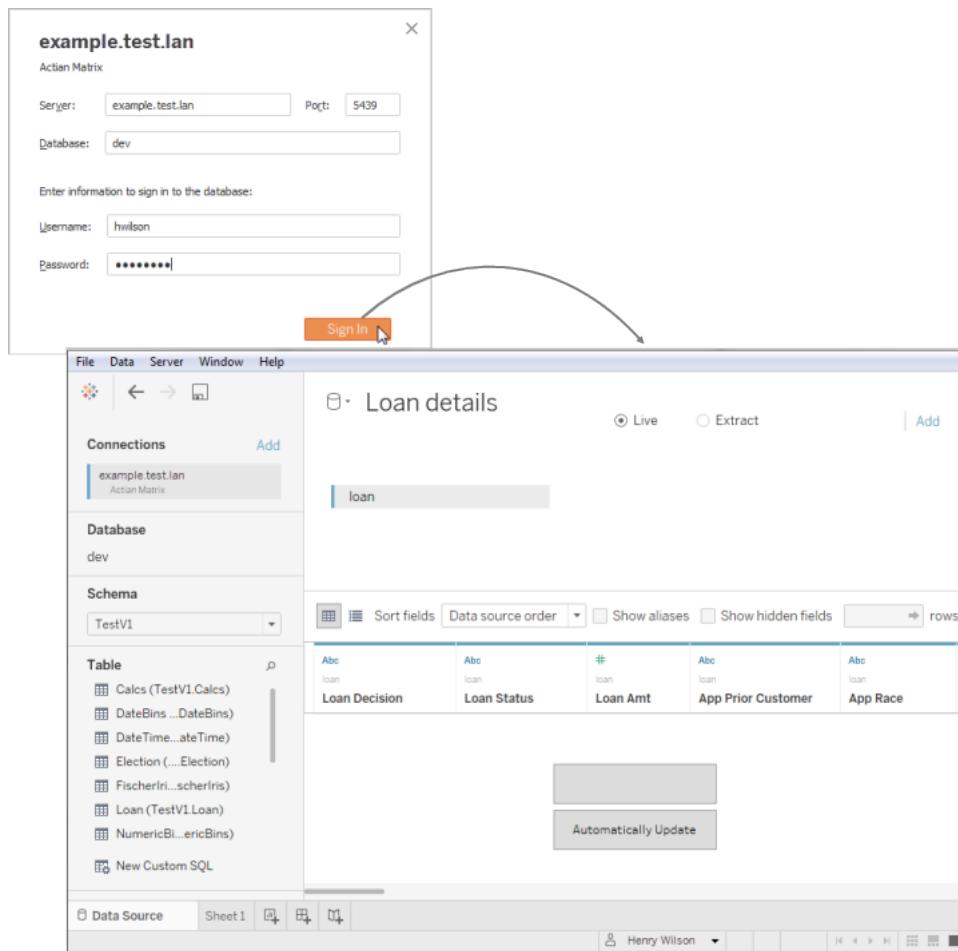
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Actian Matrix data source example

Here is an example of an Actian Matrix data source:



Actian Vector

This topic describes how to connect Tableau to an Actian Vector database and set up the data source.

Before you begin, gather this connection information:

- Virtual node name for the database you want to connect to
- Database name
- Authentication method: Authentication defined in the virtual node, or user name and password

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Action Vector**, and then do the following:
 1. Enter the name of the virtual node for the database and the name of the database you want to connect to.
 2. Specify whether to use authentication defined in the virtual node, or a specific user name and password.
 3. Click **Sign In**.If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, select a schema or use the text box to search for a schema by name.
 3. Under **Table**, select a table or use the text box to search for a table by name.
 4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query](#) on page 404.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

[Get more data from different databases](#)

- Click **Add** next to **Connections**. For more information, see [Join Your Data](#) on page 354.

- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data](#) on page 369.

[View the data in the Tableau data source](#)

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

[Manage metadata](#)

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

[Sort fields and rows](#)

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

[Hide a field](#)

Click the column drop-down arrow and select **Hide**.

[Rename a field](#)

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

[Split columns](#)

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields](#) on page 429.

[Create a new calculation based on an existing field in the data source](#)

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Actian Vector data source example

Here is an example of an Actian Vector data source:

The screenshot shows the Actian Vector interface. At the top, a connection dialog box is open for a connection named "vectorwise". It contains fields for "Virtual node" (set to "vectorwise") and "Database" (set to "TestV"). Below these, there's a section for entering login information: "Use authentication defined in the virtual node" (radio button) and "Use a specific username and password" (radio button, selected). The "Username" field is set to "test" and the "Password" field contains masked text. A "Sign In" button is at the bottom of this dialog. A curved arrow points from the "Sign In" button to the "Live" radio button in the main interface below. The main interface shows a "Loan report" section with a table titled "loan". The table has columns: "Loan Decision", "Loan Status", "#", "Loan Amt", "App Prior Customer", and "App Race". The "Loan Decision" column contains values like "Abc", "loan", and "Abc". The "Loan Status" column contains values like "Abc", "loan", and "Abc". The "#", "Loan Amt", "App Prior Customer", and "App Race" columns all contain "Abc". Below the table is a button labeled "Automatically Update". On the left side of the main interface, there's a sidebar with sections for "Connections" (showing "vectorwise" as selected), "Database" (set to "TestV"), and "Schema" (set to "dbadmin"). The "Table" section lists various tables such as "batters", "calcs", "datebins", "datetime", "election", "fischeris", "loan", and "New Custom SQL". At the bottom of the interface, there are tabs for "Data Source" and "Sheet1", along with other standard window controls.

Amazon Aurora

This topic describes how to connect Tableau to an Amazon Aurora database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- User name and password
- Are you connecting to an SSL server?

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Amazon Aurora**, and then do the following:

1. Enter the name of the server that hosts the database.
2. Enter the user name and password, and then click **Sign In**.

Select the **Require SSL** check box when connecting to an SSL server.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. From the **Database** drop-down list, select a database or use the text box to search for a database by name.
3. Under **Table**, select a table or use the text box to search for a table by name.
4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source.

For more information, see [Connect to a Custom SQL Query on page 404](#).

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

[Connect to live data or use an extract](#)

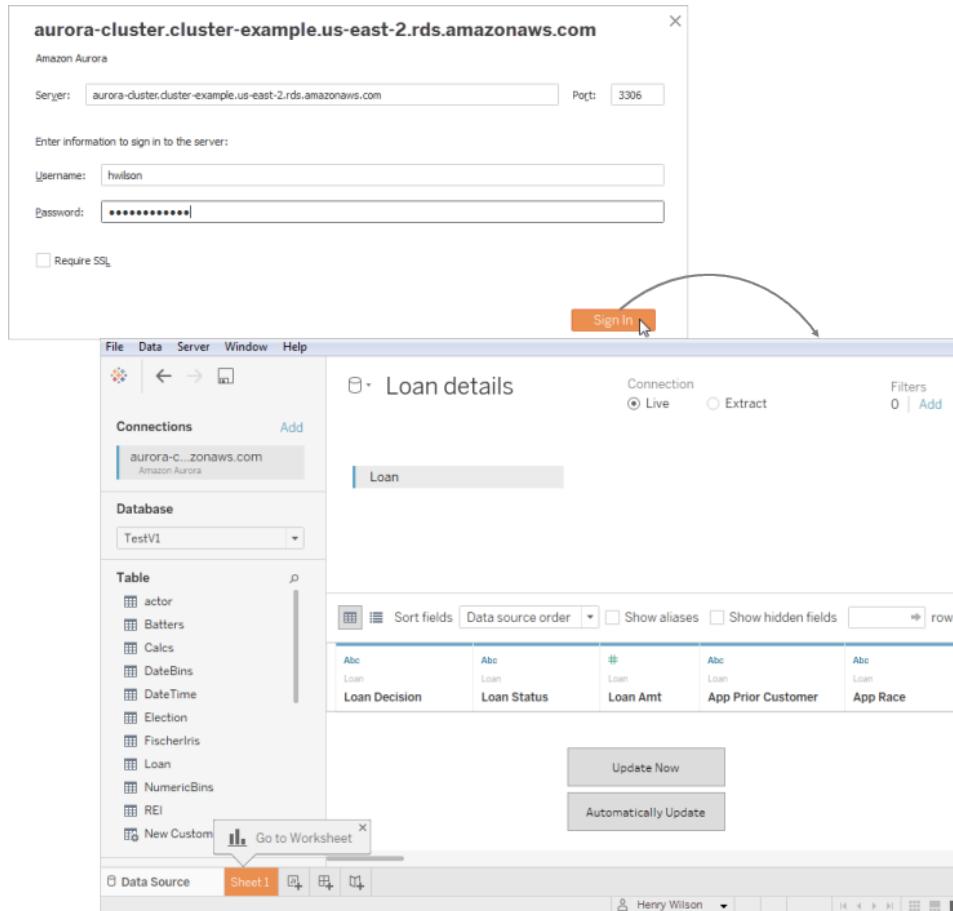
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

[Add data source filters to restrict the visibility and use of the fields in the data source](#)

Click **Add** at the top of the data source page to add data source filters .

Amazon Aurora Data Source Example

Here is an example of an Amazon Aurora data source using Tableau Desktop on a Windows computer:



Amazon EMR

This topic describes how to connect Tableau to an Amazon EMR (Elastic MapReduce) database and set up the data source.

In this topic:

- **Before you begin** on the next page
 - **Make the connection and set up the data source** on the next page
 - **Sign in on a Mac** on page 1334
 - **(Optional) Review and manage the data source** on page 1335
 - **Query limit for Amazon EMR and Impala** on page 1336
 - **Work with Hadoop Hive Data** on page 1336
-

Before you begin

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to and port number
- Type of database: Hive Server, Hive Server 2, Impala
- (Optional) Initial SQL statement to run every time Tableau connects

Note: Make sure you use the latest available drivers. To get the latest drivers, see [Amazon EMR](#) on the Tableau Driver Download page.

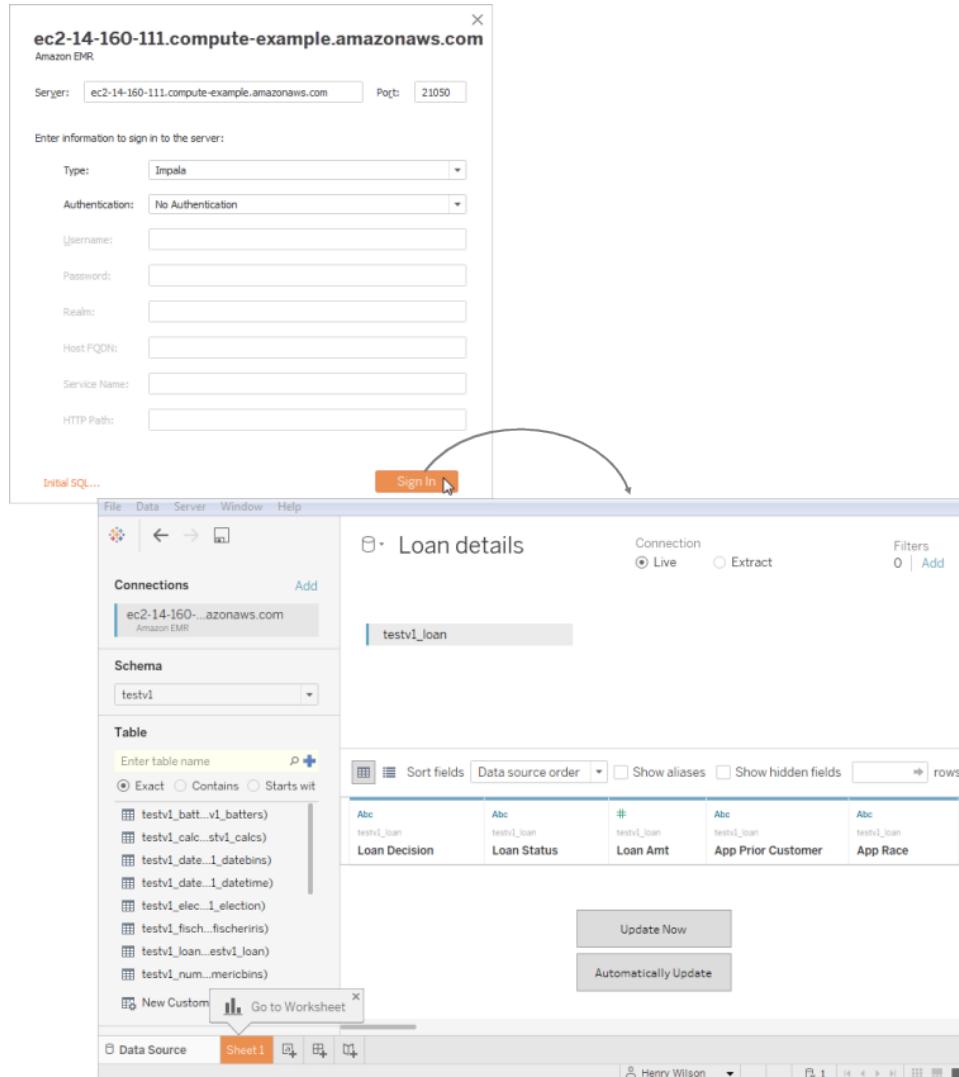
Make the connection and set up the data source

1. On the start page, under **Connect**, click **Amazon EMR**, and then do the following:
 1. Enter the name of the server that hosts the database and the port number to use.
 2. Select how to connect to the database. Depending on the version of Amazon EMR and the drivers you have installed, you can connect using one of the following:
 - Hive Server
 - Hive Server 2
 - Impala
 3. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL](#) on page 349.
 4. Click **Sign In**.If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, click the search icon or enter the schema name in the text box and click the search icon, and then select the schema.
 3. In the **Table** text box, click the search icon or enter the table name and click the search icon, and then select the table.
 4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source.
For more information, see [Connect to a Custom SQL Query](#) on page 404.

Amazon EMR data source example

Here is an example of an Amazon EMR data source connecting to an Impala server using Tableau Desktop on a Windows computer:



Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information,

see [Split a Field into Multiple Fields](#) on page 429.

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

[Connect to live data or use an extract](#)

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

[Add data source filters to restrict the visibility and use of the fields in the data source](#)

Click **Add** at the top of the data source page to add data source filters .

Query limit for Amazon EMR and Impala

When connecting to Amazon EMR using Impala, the queries are limited to 100,000 rows of data when the query includes an ORDER BY clause. This limit can be configured by using a TDC customization such as `<customization name="order-by-limit" value='10' />`.

Important: Tableau does not test or support TDC files. These files should be used as a tool to explore or occasionally address issues with your data connection. Creating and maintaining TDC files requires careful manual editing, and there is no support for sharing these files.

Work with Hadoop Hive Data

[Work with date/time data](#)

Tableau supports TIMESTAMP and DATE types natively. However, if you store date/time data as a string in Hive, be sure to store it in ISO format (YYYY-MM-DD). You can create a calculated field that uses the DATEPARSE or DATE function to convert a string to a date/time format. Use DATEPARSE() when working with an extract, otherwise use DATE(). For more information, see [Date Functions](#) on page 1565.

For more information about Hive data types, see [Dates](#) on the Apache Hive website.

NULL value returned

A NULL value is returned when you open a workbook in Tableau 9.0.1 and later and 8.3.5 and later 8.3.x releases that was created in an earlier version and has date/time data stored as a string in a format that Hive doesn't support. To resolve this issue, change the field type back to **String** and create a calculated field using DATEPARSE() or DATE() to convert the date. Use DATEPARSE() when working with an extract, otherwise use the DATE() function.

High latency limitation

Hive is a batch-oriented system and is not yet capable of answering simple queries with very quick turnaround. This limitation can make it difficult to explore a new data set or experiment with calculated fields. Some of the newer SQL-on-Hadoop technologies (for example, Cloudera's Impala and Hortonworks' Stringer project) are designed to address this limitation.

Amazon Redshift

This topic describes how to connect Tableau to an Amazon Redshift database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Database name
- User name and password
- Are you connecting to an SSL server?

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Amazon Redshift**, and then do the following:

1. Enter the name of the server that hosts the database and the name of the database you want to connect to.
2. Enter the user name and password, and then click **Sign In**.

Select the **Require SSL** check box when connecting to an SSL server.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.

2. From the **Schema** drop-down list, select a schema or use the text box to search for a schema by name.
3. Under **Table**, select a table or use the text box to search for a table by name.
4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query](#) on page 404.

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data](#) on page 354.
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data](#) on page 369.

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.

- In the grid view, sort the row values by clicking the sort button next to the column name.

[Hide a field](#)

Click the column drop-down arrow and select **Hide**.

[Rename a field](#)

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

[Split columns](#)

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields](#) on page 429.

[Create a new calculation based on an existing field in the data source](#)

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

[Connect to live data or use an extract](#)

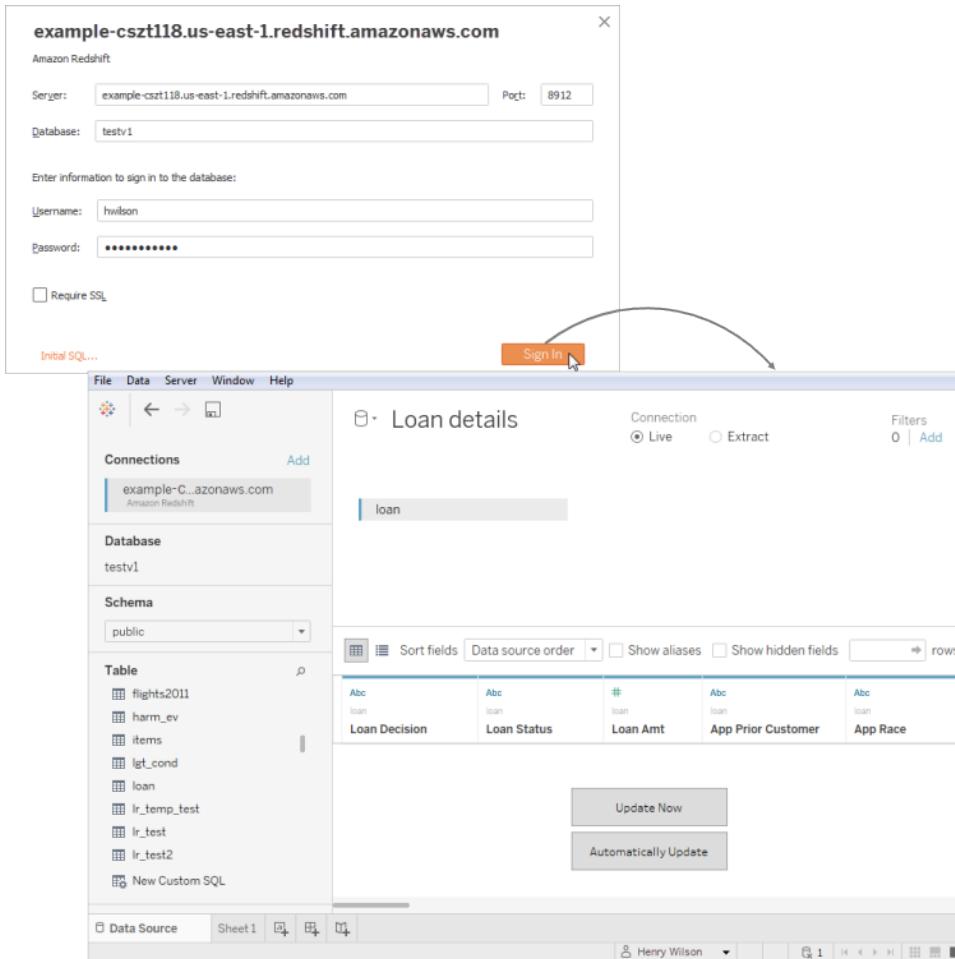
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

[Add data source filters to restrict the visibility and use of the fields in the data source](#)

Click **Add** at the top of the data source page to add data source filters .

Amazon Redshift data source example

Here is an example of an Amazon Redshift data source using Tableau Desktop on a Windows computer:



Anaplan

This topic describes how to connect Tableau to Anaplan data and set up the data source.

In this topic:

- **Before you begin** on the next page
- **Make the connection and set up the data source** on the next page
- **Date range selections can impact performance** on page 1343
- **(Optional) Review and manage the data source** on page 1344
- **Groom your data** on page 1345
- **Refresh your data** on page 1346

Before you begin

Before you begin, gather this information for connecting:

- One of the following Anaplan credentials:
 - An email address and password for your Anaplan account.
 - A certificate that Anaplan generates for you. To create a certificate, see the [Certificates tab](#) on the Anaplan website.
- The Anaplan workspace, model, and any exports (in CSV format) that you want to use. An Anaplan export action must be created in Anaplan before you can use the export in Tableau.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Anaplan**, and then do the following:

1. Select how you want to sign in to Anaplan:
 - **Credentials** – Enter your email address and password.
 - **Certificate** – Click **Upload** to upload your certificate and then follow the prompts to sign in.

For single sign-on (SSO) support, ask your Anaplan Administrator to set up single sign-on. For more information, see [Single Sign-on \(SSO\)](#) on the Anaplan Community website.

2. Select a workspace, model, and one or more exports.

Note: The export data must be in CSV format. Tableau lists only Anaplan exports with .csv extensions. If you want to use an Anaplan export that is not listed in Tableau, follow these steps:

1. Close Tableau Desktop.
2. In Anaplan, rename the export with a .csv extension.
3. Open Tableau Desktop and select the renamed export.

3. Click **Connect**.

If Tableau can't make the connection, verify that your credentials or certificate are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network or Anaplan administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.

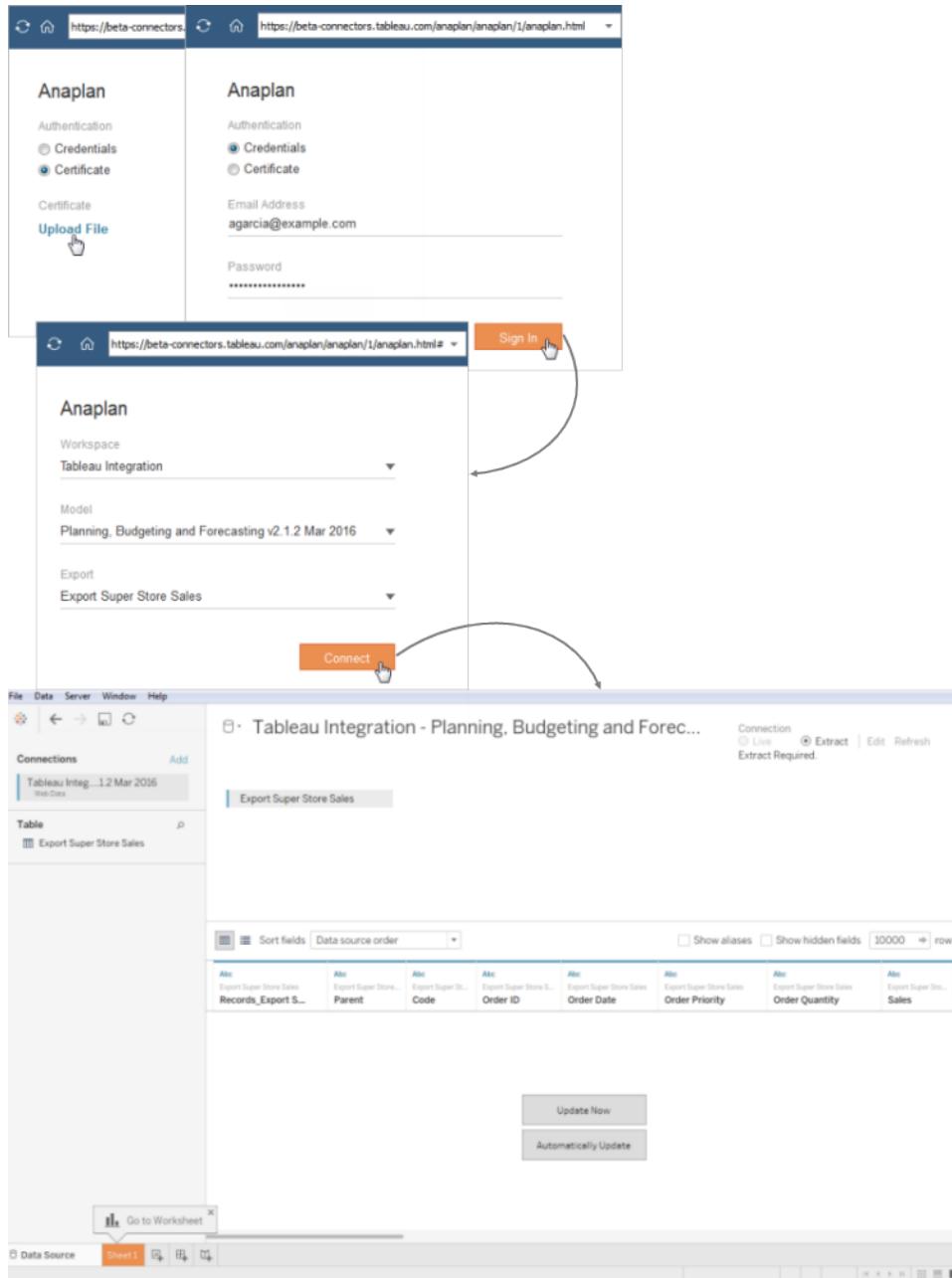
2. Under **Table**, select a table and drag it to the top of the canvas. If you have only one table, it appears on the canvas.
3. Click the sheet tab to go to the worksheet.

After you click the sheet tab, Tableau imports the data by creating an extract.

Creating extracts may take some time depending on the amount of data that is included.

[Anaplan data source example](#)

Here is an example of an Anaplan data source using Tableau Desktop on a Windows computer:



Date range selections can impact performance

It's tempting to gather as much data as possible when you do an analysis, however, retrieving records from Anaplan can be time-consuming. Tableau doesn't know how much data there is in a particular date range until it retrieves the data. For this reason, you should restrict your date range at first, and then expand after you evaluate performance.

To give you a rough idea of how much time it might take to retrieve data from Anaplan, tests were conducted using a high-speed connection. This table shows how long it took in the test environment to retrieve a given number of records.

Number of records	Time to retrieve
100,000	1.7 minutes
250,000	4 minutes
500,000	8.5 minutes

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

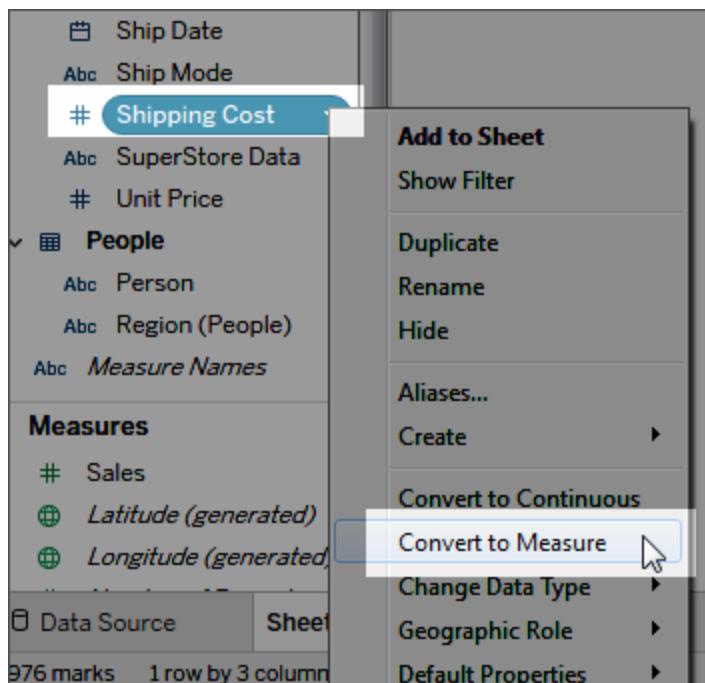
Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Groom your data

All data from Anaplan is returned as type **String**. To groom your data, convert the fields to the proper data type, for example, **Number** or **Date**. You can also change **Geographic Role**, and you can convert a dimension to a measure. For information about changing the data type, see [Data Types on page 262](#).

To convert a dimension to a measure, in the **Data** pane on the worksheet, click the drop-down arrow next to the field name and select **Convert to Measure**.



For information about dimensions and measures, see [Field Types on page 267](#).

Refresh your data

Tableau supports only extract connections for Anaplan, not live connections. You can update the data by refreshing the extract; incremental refreshes are not supported. For more information, see [Refresh Extracts](#) on page 445. For information about refresh schedules, see [Schedule Extract Refreshes as You Publish a Workbook](#) on page 1251.

Note: When there is a scheduled refresh of the Tableau extract on Tableau Server or Tableau Online, an export action is called in Anaplan. The export action creates a blocking operation that locks the model in Anaplan.

Aster Database

This topic describes how to connect Tableau to Aster Database data and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Database name
- User name and password
- (Optional) Initial SQL statement to run every time Tableau connects

In Aster, initial SQL can be used to generate an output table, which can improve performance of subsequent database access. Initial SQL in Aster also supports SQL-MapReduce, a framework created by Teradata Aster to enable developers to write powerful and highly expressive SQL-MapReduce functions. For more information, see [MapReduce, SQL-MapReduce Resources & Learning](#) on the Teradata website.

Use this connector with Tableau Desktop on a Windows computer

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Aster Database**, and then do the following:
 1. Enter the name of the server that hosts the database and the name of the database that you want to connect to.
 2. Enter the user name and password.
 3. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL](#) on page 349.
 4. Click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. From the **Schema** drop-down list, select a schema or use the text box to search for a schema by name.
3. Under **Table**, select a table or use the text box to search for a table by name.
4. Drag a table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query](#) on page 404.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data](#) on page 354.
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data](#) on page 369.

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

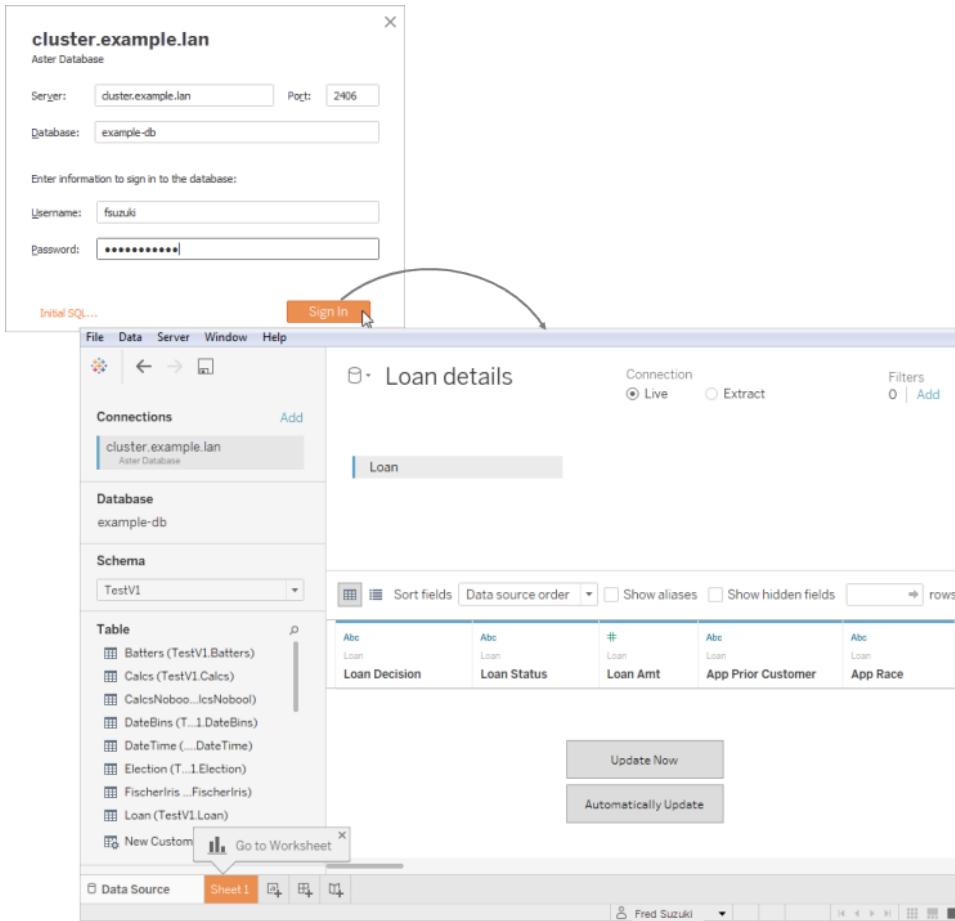
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Aster Database data source example

Here is an example of an Aster Database data source:



Cisco Information Server

This topic describes how to connect Tableau to a Cisco Information Server virtual database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the data you want to connect to
- (Optional) Domain name
- Datasource name
- Authentication method: LDAP or user name and password

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Cisco Information Server**, and then do the following:

1. Enter the name of the server that hosts the data you want to connect to.
2. (Optional) Enter the name of the domain.
3. Enter the name of the **Datasource** you want to connect to.
4. Select how you want to sign in to the server. Specify whether to use **LDAP** or a user name and password.
5. Click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. From the **Catalog** drop-down list, enter the catalog name in the text box, or select the catalog from the list.
3. From the **Schema** drop-down list, enter the schema name in the text box, or select the schema from the list.
4. Under **Table**, enter the table name in the text box, or select the table from the list.
5. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update**

Now again to see your changes.

- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields](#) on page 429.

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Cisco Information Server data source example

Here is an example of a Cisco Information Server data source:

The screenshot shows two windows side-by-side. On the left is the 'cis.example.lan' configuration window for Cisco Information Server. It has fields for Server (cis.example.lan), Domain (composite), Port (9401), and Datasource (MSSQL2008). Below these are fields for signing in: 'LDAP' (radio button), 'Use a specific username and password' (radio button, selected), 'Username' (fsuzuki), and 'Password' (redacted). A 'Sign In' button is highlighted with a red arrow. On the right is the Tableau interface. The top navigation bar includes File, Data, Server, Window, Help. The left sidebar shows Connections (cis.example.lan, Cisco Information Server), Catalog (TestV1), Schema (dbo), and Table (Batters, Calcs, DateBins, DateTime, Election, FischerIris, Loan, NumericBins). A 'New Custom' button is highlighted with a red arrow. The main workspace displays a dashboard titled 'Loan details' with a table view. The table has columns: Loan Decision, Loan Status, Loan Amt, App Prior Customer, and App Race. Buttons for 'Update Now' and 'Automatically Update' are visible at the bottom of the table area.

Cloudera Hadoop

This topic describes how to connect Tableau to a Cloudera Hadoop database and set up the data source.

In this topic:

- [Before you begin](#) below
 - [Make the connection and set up the data source](#) on the next page
 - [Sign in on a Mac](#) on page 1355
 - [\(Optional\) Review and manage the data source](#) on page 1356
 - [Work with Hadoop Hive Data](#) on page 1357
-

Before you begin

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to and port number
- Type of database: Hive Server, Hive Server 2, Impala
- Authentication method choices depend on the type of database, and can include the following:
 - No Authentication
 - Kerberos
 - User Name
 - User Name and Password
 - User Name and Password (SSL)
 - Microsoft Azure HDInsight Emulator
 - Microsoft Azure HDInsight Service
 - HTTP
 - HTTPS
- Sign in credentials depend on the authentication method you choose, and can include the following:
 - User name
 - Password
 - Realm
 - Host FQDN
 - Service Name
 - HTTP Path
- (Optional) Initial SQL statement to run every time Tableau connects

Note: Make sure you use the latest available drivers. To get the latest drivers, see [Cloudera Hadoop](#) on the Tableau Driver Download page.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Cloudera Hadoop**, and then do the following:
 1. Enter the name of the server that hosts the database and the port number to use. If you are connecting using Cloudera Impala, you must use port 21050; this is the default port if you are using the 2.5.x driver (recommended).
 2. In the **Type** drop-down list, select the type of database to connect to. Depending on the version of Hadoop and the drivers you have installed, you can connect using one of the following:
 - **Hive Server**
 - **Hive Server 2**
 - **Impala**
 3. In the **Authentication** drop-down list, select the authentication method to use.
 4. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL on page 349](#).
 5. Click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

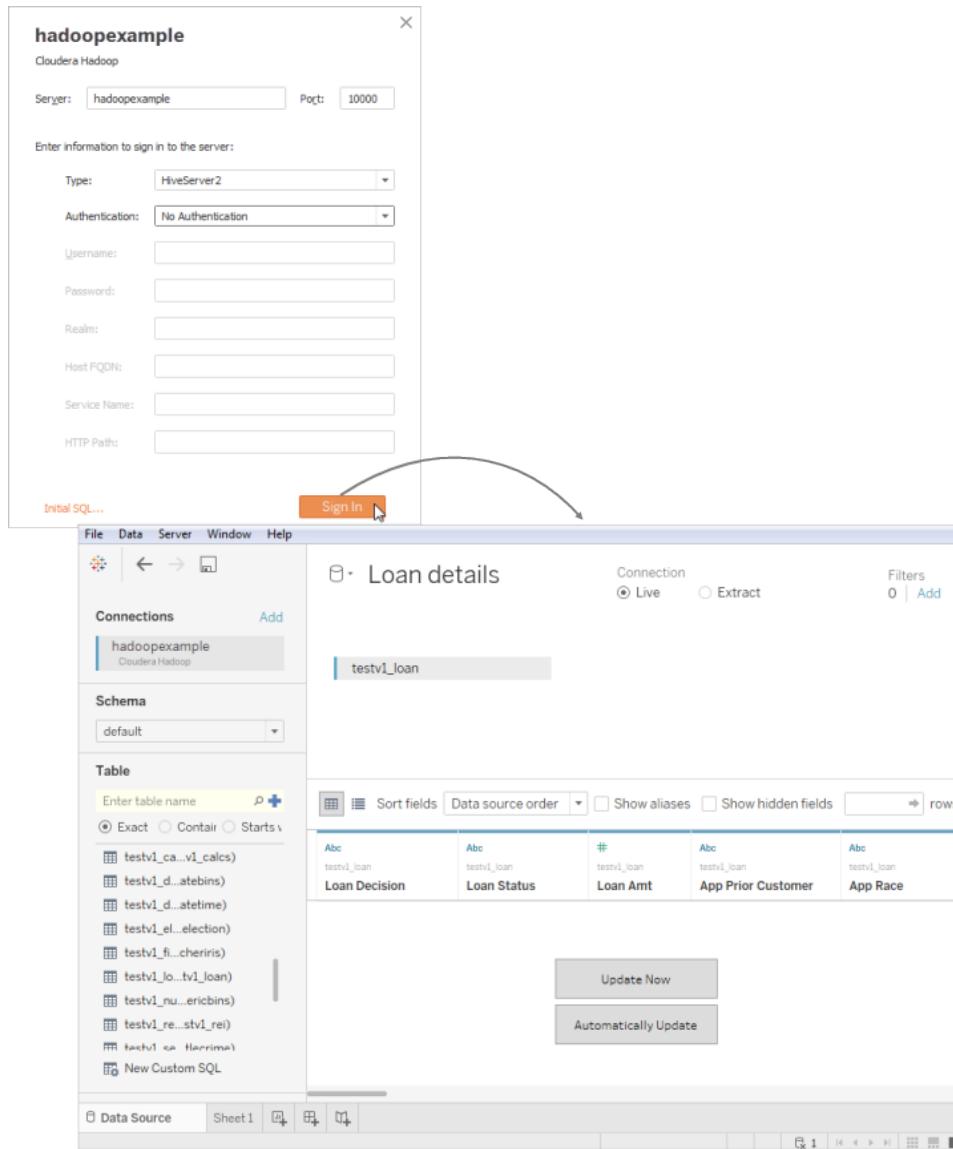
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, click the search icon or enter the schema name in the text box and click the search icon, and then select the schema.
 3. In the **Table** text box, click the search icon or enter the table name and click the search icon, and then select the table.
 4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query](#) on page 404.

Note: This database type only supports equal (=) join operations.

Cloudera Hadoop data source example

Here is an example of a Cloudera Hadoop data source using Tableau Desktop on a Windows computer:



Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name,

such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Work with Hadoop Hive Data

Work with date/time data

Tableau supports **TIMESTAMP** and **DATE** types natively. However, if you store date/time data as a string in Hive, be sure to store it in ISO format (YYYY-MM-DD). You can create a calculated field that uses the **DATEPARSE** or **DATE** function to convert a string to a date/time format. Use **DATEPARSE()** when working with an extract, otherwise use **DATE()**. For more information, see [Date Functions on page 1565](#).

For more information about Hive data types, see [Dates](#) on the Apache Hive website.

NULL value returned

A NULL value is returned when you open a workbook in Tableau 9.0.1 and later and 8.3.5 and later 8.3.x releases that was created in an earlier version and has date/time data stored as a string in a format that Hive doesn't support. To resolve this issue, change the field type back to **String** and create a calculated field using **DATEPARSE()** or **DATE()** to convert the date. Use **DATEPARSE()** when working with an extract, otherwise use the **DATE()** function.

High latency limitation

Hive is a batch-oriented system and is not yet capable of answering simple queries with very quick turnaround. This limitation can make it difficult to explore a new data set or experiment with calculated fields. Some of the newer SQL-on-Hadoop technologies (for example, Cloudera's Impala and Hortonworks' Stringer project) are designed to address this limitation.

DataStax Enterprise

This topic describes discusses how to connect Tableau to DataStax Enterprise database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- (Optional) Initial SQL statement to run every time Tableau connects

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **DataStax Enterprise**, and then do the following:
 1. Enter the name of the server that hosts the database.
 2. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL on page 349](#).
 3. Click **Sign In**.
If the connection is unsuccessful, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, click the search icon or enter the schema name in the text box and click the search icon, and then select the schema.
 3. In the **Table** text box, click the search icon or enter the table name and click the search icon, and then select the table.
 4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query](#) on page 404.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data](#) on page 354.
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data](#) on page 369.

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields](#) on page 429.

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

DataStax Enterprise data source example

Here is an example of a DataStax Enterprise data source:

The screenshot shows the DataStax Enterprise Data Source interface. At the top, there is a connection dialog box for 'datastax.example.lan' with port 10000. Below it, the main interface has a 'Sign In' button highlighted with a red arrow. The left sidebar shows 'Connections' (with 'datastax.example.lan' selected), 'Schema' (with 'TestV1' selected), and 'Table' (with a search bar and dropdown for 'Exact', 'Contains', or 'Starts with'). The main area displays the 'Loan details' table with one row: Key (loan), Column1 (loan), and Value (loan). There are buttons for 'Update Now' and 'Automatically Update' at the bottom of the table view. The bottom navigation bar includes tabs for 'Data Source' and 'Sheet1'.

EXASolution

This topic describes how to connect Tableau to data stored in the EXASolution platform and set up the data source. Tableau can connect to EXASolution version 4.2 and later.

Before you begin, gather this connection information:

- Name of the server you want to connect to
- User name and password

Make the connection and set up the data source

1. On the start page, under **Connect**, click **EXASolution**, and then do the following:
 1. Enter the name of the server that you want to connect to.
 2. Enter the user name and password, and then click **Sign In**.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, select a schema or use the text box to search for a schema by name.
 3. Under **Table**, select a table or use the text box to search for a table by name.
 4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press **Ctrl+C** (**Command-C** on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

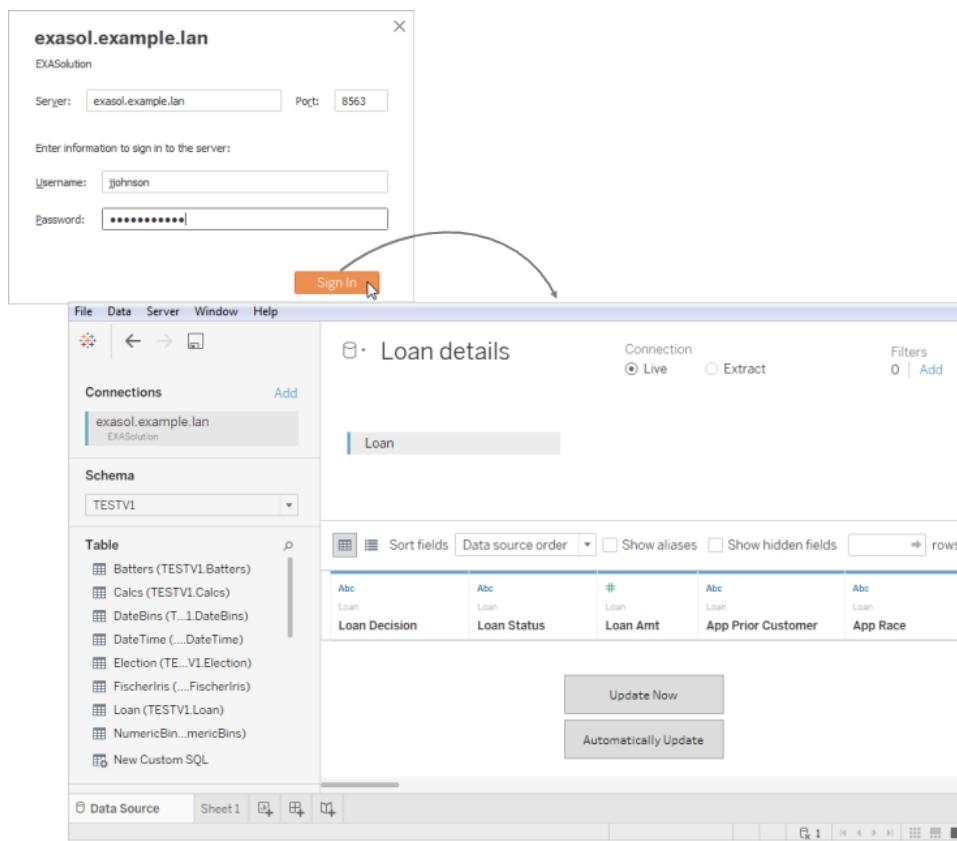
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

EXASolution data source example

Here is an example of an EXASolution data source using Tableau Desktop on a Windows computer:



Firebird

This topic describes how to connect Tableau to a Firebird database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Location of the database
- User name and password

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Firebird**, and then do the following:

1. Enter the name of the server that hosts the database.
2. Enter the database, or browse to the location of the database.
3. Enter the user name and password, and then click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. Select a table, drag it to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query](#) on page 404.

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press **Ctrl+C** (**Command-C** on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

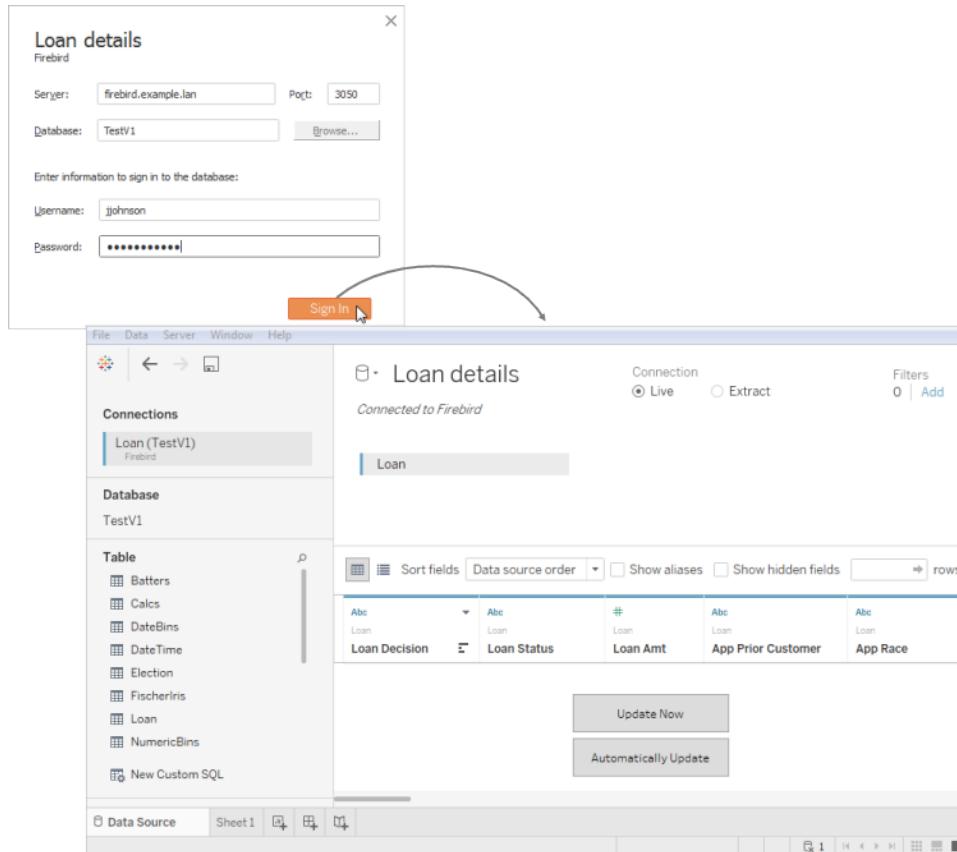
Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Note: In order to take full advantage of the calculation language in Tableau, you need to load some extra functions into your Firebird database. These functions are located in three .dll files in the \bin\udf folder of the install directory. Load the files using DEFINE FUNCTION commands. You can use the .sql script files located with the UDF dlls along with iSQL to load the function definitions into your server. Keep in mind that the scripts load all of the functions in the corresponding udf libraries, so if you have existing functions you may need to cut and paste.

Firebird data source example

Here is an example of a Firebird data source using Tableau Desktop on a Windows computer.



Google Analytics

This topic describes how to connect Tableau to Google Analytics (GA) and set up the data source, and explains when a query returns all data or sampled data.

In this topic:

- **Before you begin below**
- **Make the connection and set up the data source** on the next page
- **All data vs. sampled data** on page 1370

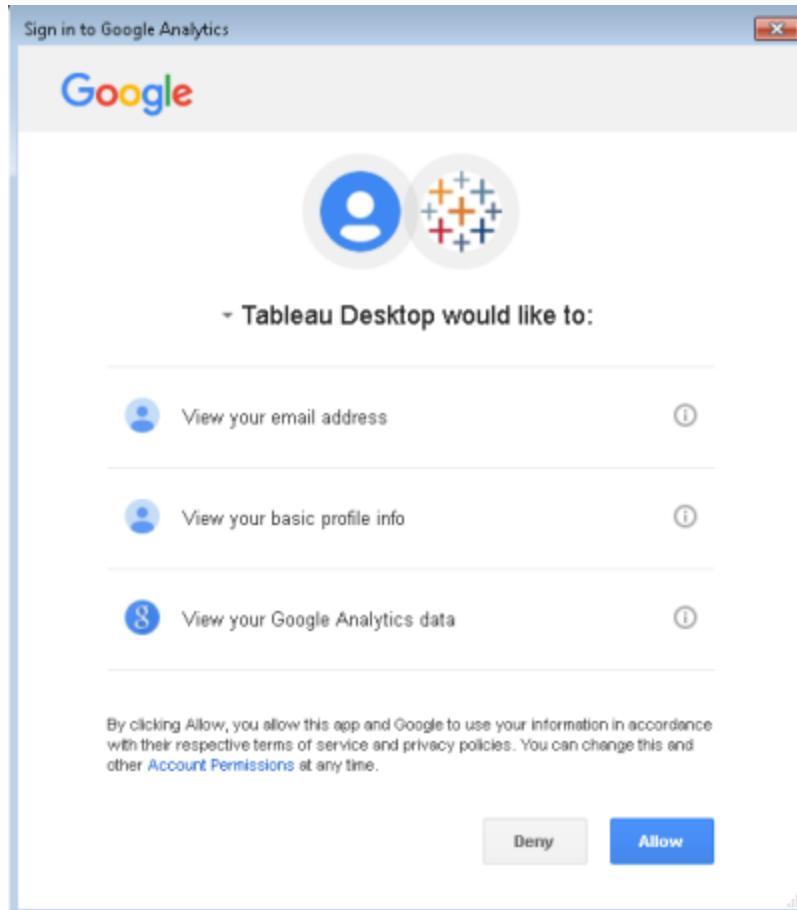
Before you begin

Before you begin, gather this connection information:

- GA email address and password

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Google Analytics**, and then do the following:
 1. Sign in to GA using your email address and password.
 2. Click **Allow** to let Tableau Desktop access your GA data.



2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. Follow the steps at the top of the data source page to complete the connection.

Step 1 – Select an Account, Property, and Profile using the drop-down menus.

Step 2 – Select filters for a date range and a segment.

- For **Date Range**, you can select one of the predefined date ranges or select specific dates. When selecting a date range, GA can provide complete data only up to the previous full day. For example, if you choose Last 30 days, data will be retrieved for the last 30-day period ending yesterday.
- For **Segment**, select a segment to filter your data. Segments are preset filters that you can set for a GA connection. Default Segments are defined by Google, and Custom Segments are defined by the user on the GA website. Segments also help prevent sampling to occur by filtering the data as defined by the segment. For example, with a segment, you can get results for a specific platform, such as tablets, or for a particular search engine, such as Google.

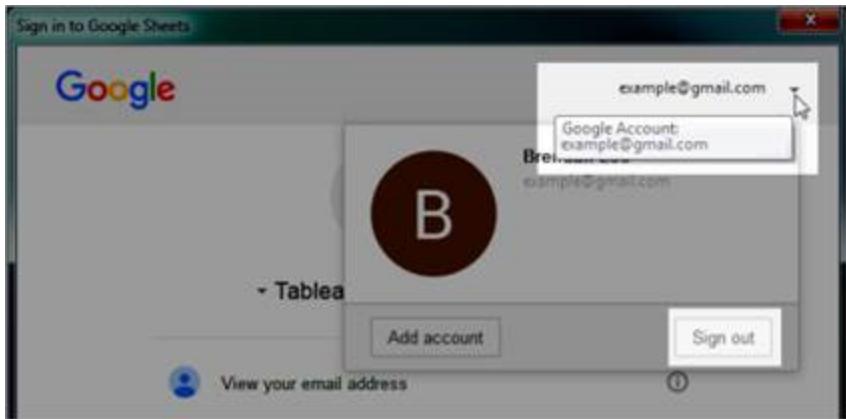
Note: GA restricts the amount of data that it returns in a query. When you attempt to retrieve more data than GA allows in a single query, GA returns sampled data instead. If Tableau detects that your GA query might return sampled data, Tableau attempts to bypass the query restriction to return all data instead. For more information, see [All data vs. sampled data on the next page](#) below.

Step 3 – Add dimensions and measures by using the **Add Dimension** and **Add Measure** drop-down menus, or select a predefined set of measures from the **Choose a Measure Group** drop-down menu. Some dimensions and measures cannot be used together. For more information, see [Dimensions & Metrics Reference Guide](#) on the Google developer website.

3. Click the sheet tab to start your analysis. After you click the sheet tab, Tableau imports the data by creating an extract. Note that Tableau Desktop supports only extracts for Google Analytics. You can update the data by refreshing the extract. For more information, see [Extract Your Data on page 440](#).

Change your Google account

If Google bypasses the sign-in screen when you connect, it means that you are already signed in to a Google account. If you are signed in, make sure that you are using the account that provides access to your Google Analytics data. To sign out of an account, click the drop-down arrow next to the account name, and then click **Sign out**. You can then sign in using the correct credentials.



Google Analytics data source example

Here is an example of a Google Analytics data source connection using Tableau Desktop on a Windows computer:

MyCompany Page Usage
Connected to Google Analytics mycompany@gmail.com

Step 1: Choose an Account, Property, and Profile:

- Account: www.mycompany.com (674216)
- Property: MyCompany All-sites Roll-up (id:07)
- Profile: All Sites - (exclude Public and webpar...

Step 2: Select Filters:

- Date Range: Last 30 Days
1/3/2015 to 2/1/2015
- Segment: All Sessions (-1)

Step 3: Select up to 7 Dimensions and 10 Measures:

Choose a Measure Group: Page Usage

Add Dimension	Add Measure
Date	Avg. Session Duration
	Bounces
	Goal Completions
	New Users
	Pageviews
	% New Sessions
	Session Duration

Data Source Fields:

Field Name	Table	Remote Field Name
ga:date		ga:date
# gasessions		gasessions
# gaglobalCompletionsAll		gaglobalCompletionsAll
# ga:transactionRevenue		ga:transactionRevenue
# ga:pageviews		ga:pageviews

Buttons at the bottom:

- Go to Worksheet
- Data Source (highlighted)
- Sheet 1
- More options

All data vs. sampled data

GA restricts the amount of data that it returns from a query and provides sampled data instead. Sampled data is a random subset of your data. When performing analysis on sampled data, you can miss interesting outliers, and aggregations can be inaccurate. If Tableau detects that your query might return sampled data, by default, Tableau creates multiple queries from your query, and then combines the results from the queries to return all data.

You see the following message when Tableau returns all data.

Query returns: All data. Sample data

Note: If the query stays within the boundaries of the query restrictions, GA doesn't return sampled data and you do not see the above message.

Troubleshoot issues with returning all data

If your query continues to return sampled data, consider the following:

- **Missing date dimension** – You must use the date dimension in your query to return all data.
- **Too much data** – Your query might contain too much data. Reduce the date range.
Note: The minimum date range is one day.
- **Non-aggregatable dimensions and measures** – Some dimensions and measures cannot be separated into multiple queries. If you suspect a problematic dimension or measure in your query, hover over the **All data** button to see the tooltip that shows which dimensions or measures to remove from your query.
- **Legacy workbooks** – Workbooks created in Tableau Desktop 9.1 and earlier cannot return all data. Open the legacy workbook in Tableau Desktop 9.2 and save the workbook.

Return sampled data

In cases when workbook performance is critical or there are specific dimensions and measures you want to use in your query that are not supported by Tableau's default query process, use sampled data instead. To return sampled data, click the **Sample data** button.



Google BigQuery

This topic describes how to connect Tableau to Google BigQuery and set up the data source.

In this topic:

- **Before you begin** below
 - **Make the connection and set up the data source** below
 - **(Optional) Review and manage the data source** on page 1374
 - **Use customization attributes to improve query performance** on page 1376
 - **Switch from standard SQL back to legacy SQL** on page 1378
-

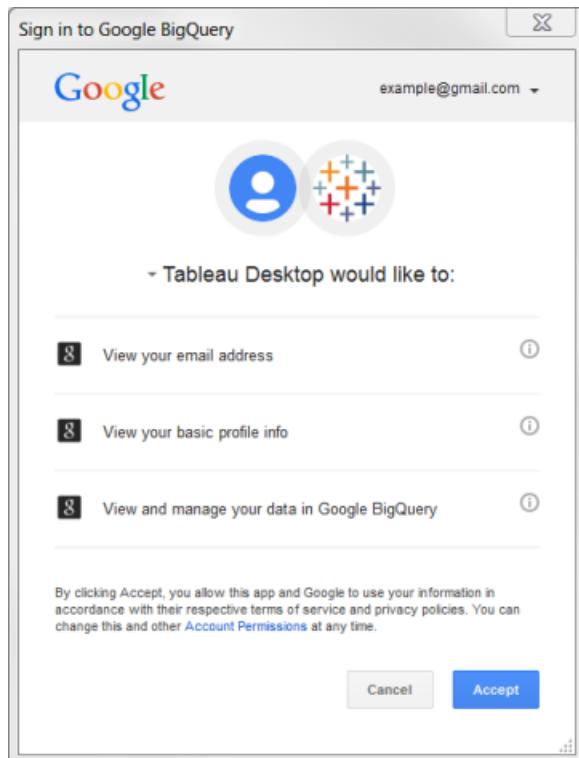
Before you begin

Before you begin, gather this connection information:

- Google BigQuery email address and password

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Google BigQuery**, and then do the following:
 1. Sign in to Google BigQuery using your email address and password.
 2. Click **Accept** to let Tableau Desktop access your Google BigQuery data.



2. On the data source page, do the following:

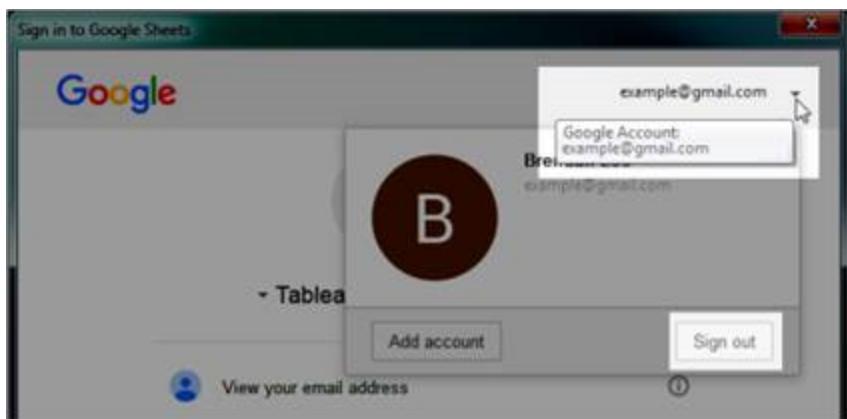
1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. (Optional) From the **Billing Project** drop-down list, select a billing project. If you don't select a billing project, `EmptyProject` appears in the field after you have selected the remaining fields.
3. From the **Project** drop-down list, select a project. Alternatively, select **publicdata** to connect to sample data in BigQuery.
4. From the **Dataset** drop-down list, select a data set.
5. Under **Table**, select a table.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Note: Google BigQuery has changed support from BigQuery legacy SQL (BQL) to standard SQL. Your workbooks will upgrade to support standard SQL when you open them in Tableau.

Change your Google account

If Google bypasses the sign-in screen when you connect, it means that you are already signed in to a Google account. If you are signed in, make sure that you are using the account that provides access to your Google BigQuery data. To sign out of an account, click the drop-down arrow next to the account name, and then click **Sign out**. You can then sign in using the correct credentials.



Google BigQuery data source example

Here is an example of a Google BigQuery data source using Tableau Desktop on a Windows computer:

The screenshot shows the Tableau Data Source Editor interface. On the left, there's a sidebar with 'Connections' (googleexample.com/bigquery, Add), 'Billing Project' (EmptyProject), 'Project' (publicdata), and 'Dataset' (samples). Below that is a 'Table' section with various tables listed. The main workspace is titled 'Shakespeare' and contains a grid view of data. The grid has four columns: 'Word', 'Word Count', 'Corpus', and 'Corpus Date'. At the bottom of the grid, there are two buttons: 'Update Now' and 'Automatically Update'. The top right of the workspace shows 'Connection' (Live) and 'Filters' (0 | Add).

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Note: Because of the large volume of data in BigQuery, Tableau recommends that you connect live.

Use customization attributes to improve query performance

You can use customization attributes to improve the performance of large result sets returned from BigQuery to Tableau Online and Tableau Server, and on Tableau Desktop.

You can have the customization attributes included in your published workbook or data source, as long as you specify the attributes before you publish the workbook or data source to Tableau Online or Tableau Server.

Use Google BigQuery customization attributes

Customization attributes accept integer values and affect both live queries and extract refreshes for the specified connection.

The following attributes help the most to increase performance of large result sets:

bq-fetch-tasks	Number of parallel background tasks to use when fetching data using HTTP. The default is 10.
bq-large-fetch-rows	Number of rows to fetch in each batch for spool queries. The default is 50000.

The following attributes are also available and are primarily used for small queries:

bq-fetch-rows	Number of rows to fetch in each batch for non-spool queries. The default is 10000.
bq-response-rows	Number of rows returned in non-spool non-batched queries. The default is 10000.

This capability setting accepts yes or no values and can be useful when testing:

CAP_BIGQUERY_FORCE_SPOOL_JOB	Force all queries to use the temp table approach. The default value is "no." Change the value to "yes" to turn this attribute on.
------------------------------	---

How Tableau returns rows from Google BigQuery

Tableau uses two approaches to return rows from BigQuery: the default non-spool approach, or the temp table (spool) approach:

- On the first attempt, queries are executed using the default, non-spool query, which uses the bq-fetch-rows setting.
- If the result set is too large, the BigQuery API returns an error and the Tableau BigQuery connector retries the query by saving the results into a BigQuery temp table. The BigQuery connector then reads from that temp table, which is a spool job that uses the bq-large-fetch-rows setting.

How to specify the attributes

You can specify attributes in one of two ways: in a Tableau Datasource Customization .tdc file, or in the workbook or data source XML.

Specify attributes in a .tdc file

To specify customization attributes during a publish workbook or publish data source operation from Tableau Desktop, follow these steps:

1. Create an XML file that contains the customization attributes.
2. Save the file with a .tdc extension, for example, BigQueryCustomization.tdc.
3. Save the file to the My Tableau Repository\Datasources folder.

The customization attributes in the .tdc file are read and included by Tableau Desktop when the data source or workbook is published to Tableau Online or Tableau Server.

Important: Tableau does not test or support TDC files. These files should be used as a tool to explore or occasionally address issues with your data connection. Creating and maintaining TDC files requires careful manual editing, and there is no support for sharing these files.

Example of a .tdc file with recommended settings for large extracts

```
<connection-customization class='bigquery' enabled='true' version='8.0'>

    <vendor name='bigquery' />

    <driver name='bigquery' />

    <customizations>

        <customization name='bq-fetch-tasks' value='10' />
        <customization name='bq-large-fetch-rows' value='10000' />
    </customizations>

</connection-customization>
```

Manually embed attributes in the XML of the workbook or data source file

You can manually embed customization attributes inside the 'connection' tag in the workbook .twb file or the data source .tds file. The BigQuery customization attributes are bold in the following example to make them easier for you to see.

Example of manually embedded attributes

```
<connection CATALOG='publicdata' EXECCATALOG='some-project-123'
REDIRECT_URI='some-url:2.0:oob'
SCOPE='https://www.googleapis.com/auth/bigquery
https://www.googleapis.com/auth/userinfo.profile
https://www.googleapis.com/auth/userinfo.email'
authentication='yes' bq-fetch-tasks='10' bq-large-fetch-
rows='10000' bq_schema='samples' class='bigquery' connection-
dialect='google-bql' connection-protocol='native-api' login_
title='Sign in to Google BigQuery' odbc-connect-string-extras=''
project='publicdata' schema='samples'
server='googleapis.com/bigquery' server-oauth=''
table='wikipedia' username=''>
```

Switch from standard SQL back to legacy SQL

Google BigQuery upgraded their APIs to use standard SQL, in addition to BigQuery SQL (now called legacy SQL), and Tableau upgraded the Google BigQuery connector to support this change to standard SQL. With standard SQL comes benefits for BigQuery users including level of detail expressions, faster metadata validation, and the ability to select a billing project with your connection.

When you open an workbook created in a previous version of Tableau Desktop, Tableau upgrades your workbook from legacy to standard SQL if the workbook doesn't contain custom SQL or raw SQL calculations. If the workbook does contain custom SQL or raw SQL, you're prompted to upgrade when you open the workbook. If you choose to upgrade your workbook, you must replace the custom or raw SQL with standard SQL in order for your workbook to be fully functional.

If you have a workbook that uses database views written in legacy SQL, Tableau doesn't recognize the legacy SQL and updates your workbook without prompting you. In this case, the following error message might appear:

Cannot reference a Legacy SQL view in a SQL query.

If this happens, rewrite your database views to use standard SQL. If you can't, you can use the Tableau Datasource Customization (TDC) below to switch back to legacy SQL until you can rewrite the SQL. This `.tdc` option will work with Tableau Desktop, Tableau Server, and Tableau Online.

```
<connection-customization class='bigquery' enabled='true' ver-
sion='10.1' >

    <vendor name='bigquery' />

    <driver name='bigquery' />
```

```

<customizations>
    <customization name='connection-dialect' value='google-bql' />
</customizations>
</connection-customization>

```

Important: Tableau does not test or support TDC files. These files should be used as a tool to explore or occasionally address issues with your data connection. Creating and maintaining TDC files requires careful manual editing, and there is no support for sharing these files.

For more information about migrating from legacy SQL to standard SQL, see [Migrating from legacy SQL](#) on the Google Cloud Platform website.

Google Cloud SQL

This topic describes how to connect Tableau to a Google Cloud SQL database instance and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- User name and password

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Google Cloud SQL**, and then do the following:
 1. Enter the name of the server that hosts the database.
 2. Enter the user name and password, and then click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Database** drop-down list, select a database or use the text box to search for a database by name.
 3. Under **Table**, select a table or use the text box to search for a table by name.

4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query](#) on page 404.

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data](#) on page 354.
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data](#) on page 369.

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

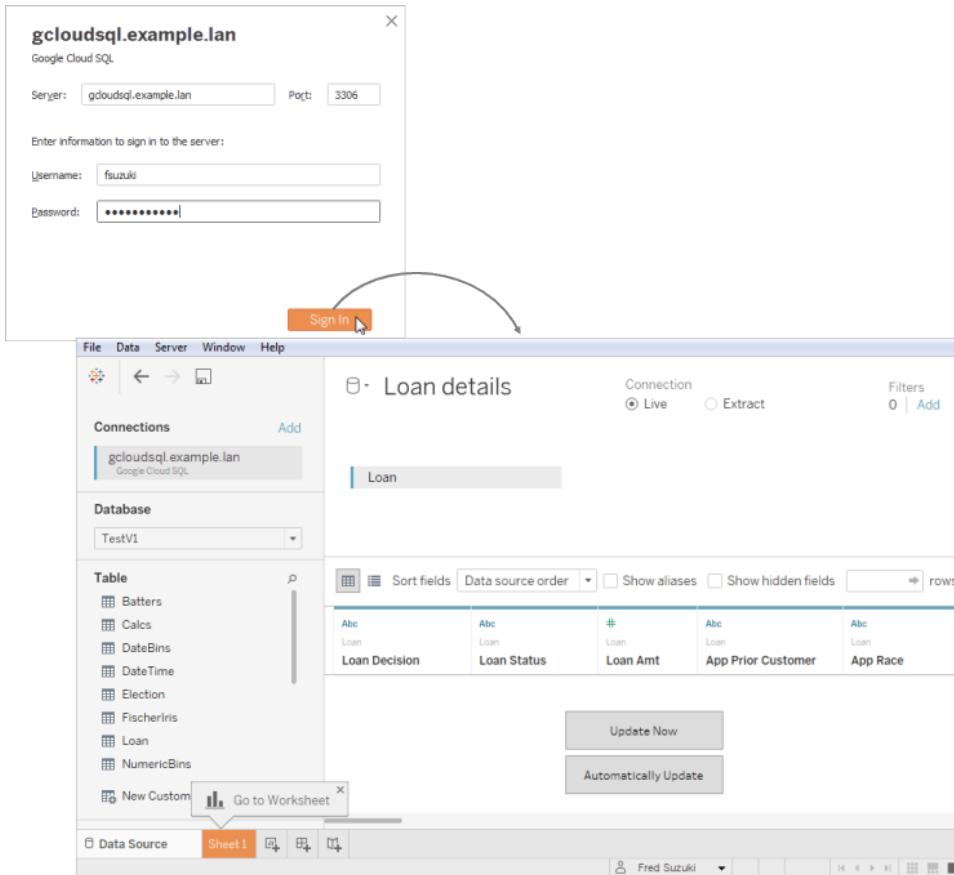
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Google Cloud SQL data source example

Here is an example of a Google Cloud SQL data source using Tableau Desktop on a Windows computer.



Google Sheets

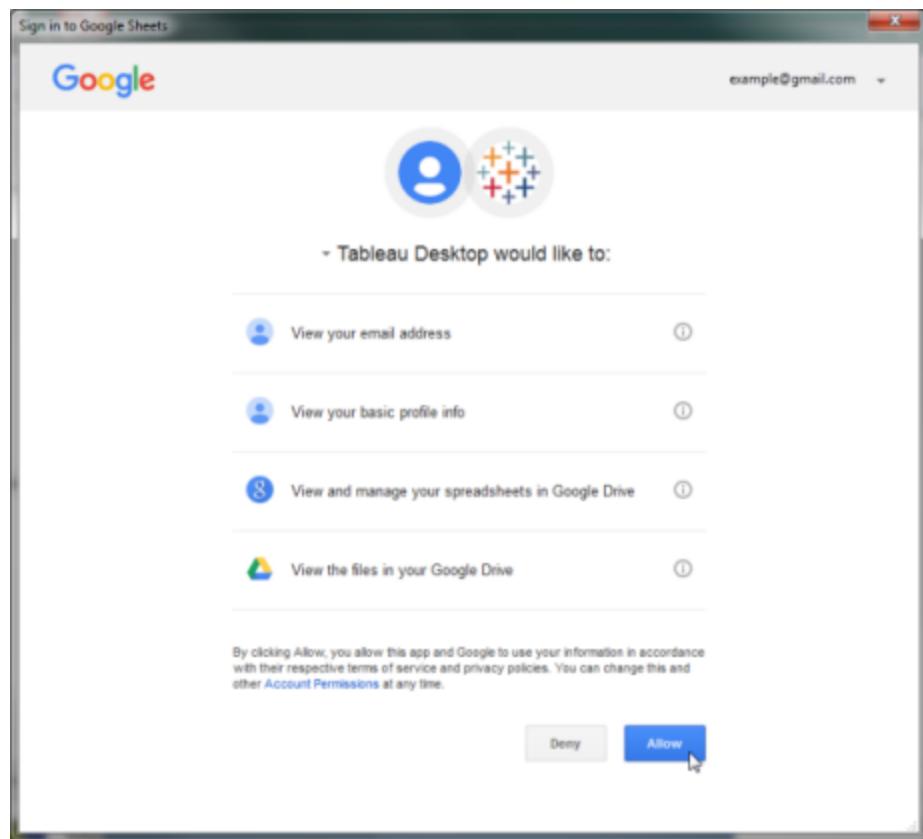
This topic describes how to connect Tableau to Google Sheets and set up the data source, lists optional settings available to you, and covers troubleshooting Google Sheets issues.

Before you begin, gather this connection information:

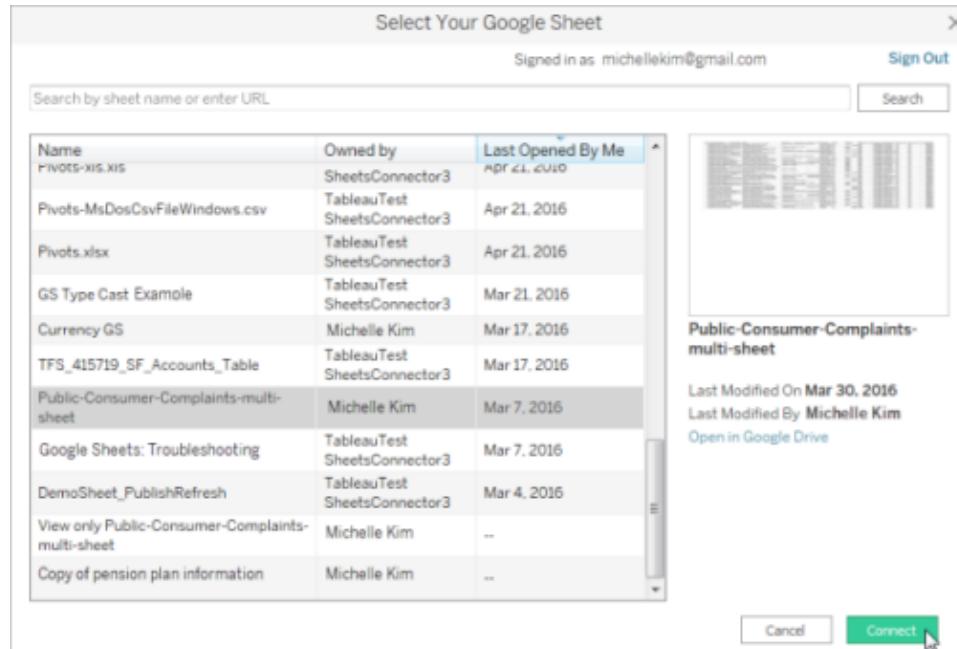
- Google account email and password

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Google Sheets**, and then do the following:
 1. Sign in to Google Sheets using your Google account email address and password, if you're not already signed in.
 2. Click **Allow** to let Tableau Desktop access your Google Sheets data.



3. Select a Google Sheet from the list or use the text box to search for a Google Sheet by name or by URL, and then click **Connect**.



2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data figure out which data source to connect to.
 2. If your Google Sheets file has one table, click the sheet tab to start your analysis.

Note: Tableau doesn't support pivot tables in Google Sheets.

Select Your Google Sheet dialog box functionality

The Select Your Google Sheet dialog box includes the following functionality:

- The list of sheets that you can select from includes your private sheets, sheets shared with you, and the public sheets that you've accessed in the past.
- If you search by URL and the URL doesn't exist or you don't have access to it, an error displays.
- You can click the **Name** and **Last opened by me** column names to sort the Google Sheets, and when you select a sheet you can preview it in the right pane. You cannot sort by **Owned by**.

Connect to more data

You can connect to more than one table by using join. For more information, see **Join Your Data** on page 354.

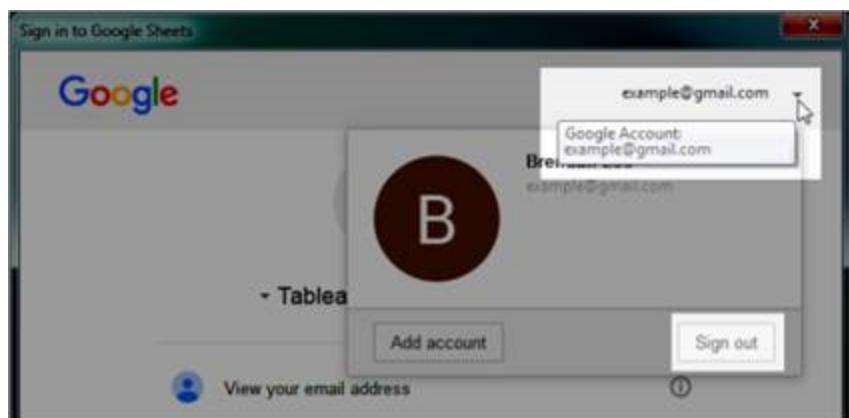
You can also connect to a named range the same way you connect to a worksheet. The named range functions as a table in Tableau.

You create named ranges in Google Sheets by highlighting a range of cells and then selecting **Data > Named ranges**. When you connect to a named range in Tableau, an icon appears next to the sheet in the **Data Source** tab as shown below.



Change your Google account

If Google bypasses the sign-in screen when you connect to Google Sheets and goes directly to the approval screen shown in step 3 above, it means that you are already signed in to a Google account. If you are signed in, make sure that you are using the account that provides access to your Google Sheets data. To sign out of an account, click the drop-down arrow next to the account name, and then click **Sign out**. You can then sign in using the correct credentials.



Optional settings

You can set the following options before you build the view.

Set table options

On the canvas, click the table drop-down arrow and then specify whether the data includes field names in the first row. If so, these names will become the field names in Tableau. If field names are not included, Tableau generates them automatically. You can rename the fields later.

Clean your data with Data Interpreter

If Tableau detects that it can help optimize your data source for analysis, it prompts you to use Data Interpreter. Data Interpreter can detect sub-tables that you can use and remove unique

formatting that might cause problems later on in your analysis. For more information, see [Clean Your Data with Data Interpreter](#) on page 428.

Review and manage the data source

In addition to setting table options and cleaning your data, you can make the following changes to the data source before you start your analysis:

- **View the data in the Tableau data source** – In the grid, click **Update Now** to preview the first 1,000 rows of your live or extract data source. If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes. If you want changes to automatically appear in the grid, click **Automatically Update**.
- **Manage metadata** – Click the metadata grid button to quickly examine the general structure of the Tableau data source and its fields and to perform routine management tasks like sorting fields and rows, hiding multiple fields at once, or quickly renaming or resetting fields.
- **Sort fields and rows** – From the **Sort fields** drop-down list, select how you want to sort the columns in the grid or metadata grid. You can sort the columns by data source or table order. Sort the row values by clicking the sort button next to the column name.
- **Hide fields** – Hide a field by clicking the column drop-down arrow and selecting **Hide**.
- **Rename fields and reset field names** – To rename the field, in the data preview grid, double-click the field name, or in the metadata grid, click the field name. You can also select a column or multiple columns, click the column drop-down, and then select **Reset Name** to revert back to the original name of the field.
- **Pivot columns** – Pivot fields to transform data in a crosstab format into a columnar format. For more information, see [Pivot Data from Columns to Rows](#) on page 423.
- **Split columns** – Depending on the data source, you can split the columns in the data source into new fields. For more information, see [Split a Field into Multiple Fields](#) on page 429.
- **Create calculations** – Create a new calculation based on an existing field in the data source. Click the column drop-down arrow and select **Create Calculated Field**.
- **Copy values** – Copy values in the grid by selecting the values and then pressing Ctrl+C. Alternatively, to copy values in the metadata grid, select the values, right-click, and then select **Copy**.
- **Connect live or use an extract** – At the top of the data source page, select a live or extract connection to the data source. If you choose to take an extract, an **Edit** link displays that you can use to set up filters that define a subset of the data to include in the extract.
- **Add data source filters** – At the top of the data source page, add data source filters to restrict the visibility and use of the fields in the data source.

Troubleshoot Google Sheets issues

Data limit in Google Drive

You can store up to 2 million cells for spreadsheets that are created in or converted to Google Sheets. For more information, see [Files you can store in Google Drive](#) in the Google Drive Help.

Error message: Internal Error - An unexpected error occurred and the operation could not be completed.

If there are errors in your Google Sheet, such as #DIV/0! or #N/A, Tableau is unable to create an extract and an error message will appear. To resolve this issue, wrap the function with `iferror()` and have it return a blank, or any value that's appropriate.

For example, the sheet below includes a #DIV/0! error.

100	40	2.5
100	35	2.857142857
100	30	3.333333333
100	25	4
100	20	5
100	15	6.666666667
100	10	10
100	5	20
100	0	#DIV/0!

Error
Function DIVIDE parameter 2
cannot be zero.

The solution is to wrap the calculation in an `iferror()` calculation.

=iferror(A12/B12,)		
A	B	C
100	10	10
100	5	20
100	0	

Google Sheets data source example

Here is an example of a Google Sheets data source:

Hortonworks Hadoop Hive

This topic describes how to connect Tableau to a Hortonworks Hadoop Hive database and set up the data source.

In this topic:

- [Before you begin below](#)
- [Make the connection and set up the data source on the next page](#)
- [Sign in on a Mac on page 1391](#)
- [\(Optional\) Review and manage the data source on page 1391](#)
- [Work with Hadoop Hive Data on page 1393](#)

Before you begin

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Type of database: HiveServer, HiveServer2
- Authentication method

HiveServer doesn't require authentication. HiveServer2 authentication choices include the following:

- No Authentication
- Kerberos
- User Name
- User Name and Password
- User Name and Password (SSL)
- Microsoft Azure HDInsight Emulator
- Microsoft Azure HDInsight Service
- HTTP
- HTTPS
- Sign-in credentials

Credentials depend on the authentication method you choose, and can include the following:

- User name
- Password
- Realm
- Host FQDN
- Service Name
- HTTP Path
- Are you connecting to an SSL server?
- (Optional) Initial SQL statement to run every time Tableau connects

Note: Make sure you use the latest available drivers. To get the latest drivers, see [Hortonworks Hadoop Hive](#) on the Tableau Driver Download page.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Hortonworks Hadoop Hive**, and then do the following:
 1. Enter the name of server that hosts the data base.
 2. In the **Type** drop-down list, select the type of database to connect to:
 - **HiveServer**
 - **HiveServer2**
 3. In the **Authentication** drop-down list, select the authentication method to use.

4. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL on page 349](#).

5. Click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

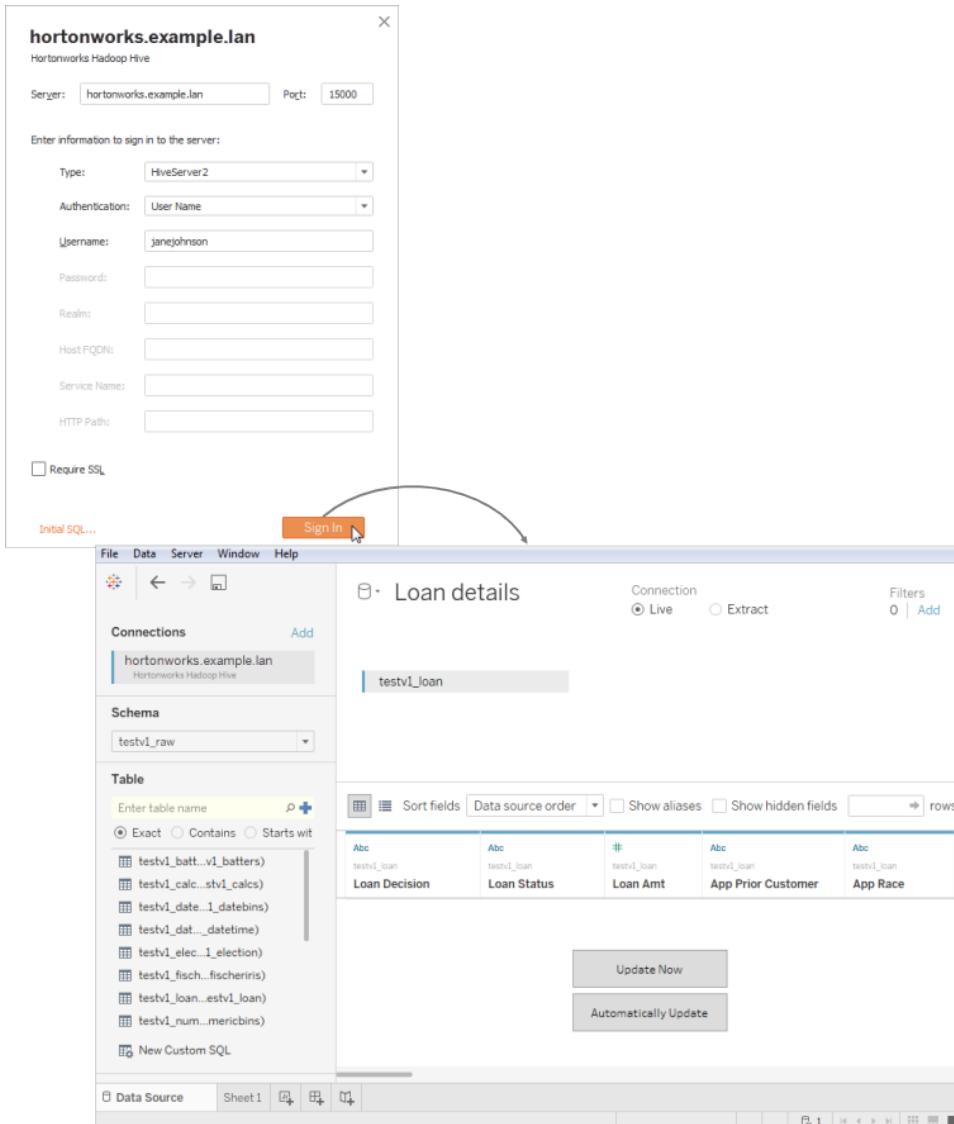
1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. From the **Schema** drop-down list, click the search icon or enter the schema name in the text box and click the search icon, and then select the schema.
3. In the **Table** text box, click the search icon or enter the table name and click the search icon, and then select the table.
4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Note: This database type only support equal (=) join operations.

[Hortonworks Hadoop Hive data source example](#)

Here is an example of a Hortonworks Hadoop Hive data source using Tableau Desktop on a Windows computer:



Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press **Ctrl+C** (**Command-C** on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Work with Hadoop Hive Data

Work with date/time data

Tableau supports **TIMESTAMP** and **DATE** types natively. However, if you store date/time data as a string in Hive, be sure to store it in ISO format (YYYY-MM-DD). You can create a calculated field that uses the **DATEPARSE** or **DATE** function to convert a string to a date/time format. Use **DATEPARSE()** when working with an extract, otherwise use **DATE()**. For more information, see [Date Functions on page 1565](#).

For more information about Hive data types, see [Dates](#) on the Apache Hive website.

NULL value returned

A **NULL** value is returned when you open a workbook in Tableau 9.0.1 and later and 8.3.5 and later 8.3.x releases that was created in an earlier version and has date/time data stored as a string in a format that Hive doesn't support. To resolve this issue, change the field type back to **String** and create a calculated field using **DATEPARSE()** or **DATE()** to convert the date. Use **DATEPARSE()** when working with an extract, otherwise use the **DATE()** function.

High latency limitation

Hive is a batch-oriented system and is not yet capable of answering simple queries with very quick turnaround. This limitation can make it difficult to explore a new data set or experiment with calculated fields. Some of the newer SQL-on-Hadoop technologies (for example, Cloudera's Impala and Hortonworks' Stringer project) are designed to address this limitation.

Truncated columns in Tableau

The default string column length for Hortonworks Hadoop Hive is 255 characters. For more information about Hortonworks Hive ODBC driver configuration options, and specifically about **DefaultStringColumnLength**, see the [Hive ODBC Driver User Guide](#) from Hortonworks.

HP Vertica

This topic describes how to connect Tableau to an HP Vertica database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Database name
- User name and password
- (Optional) Initial SQL statement to run every time Tableau connects

Make the connection and set up the data source

1. On the start page, under **Connect**, click **HP Vertica**, and then do the following:
 1. Enter the name of the server that hosts the database and the name of the database that you want to connect to.
 2. Enter the user name and password.
 3. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL on page 349](#).
 4. Click **Sign In**.If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, select a schema or use the text box to search for a schema by name.
 3. Under **Table**, select a table or use the text box to search for a table by name.
 4. Drag the table to the canvas, and then click the sheet tab to start your analysis.Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click

the column drop-down arrow, and then select **Reset Name**.

[Split columns](#)

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields](#) on page 429.

[Create a new calculation based on an existing field in the data source](#)

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

[Connect to live data or use an extract](#)

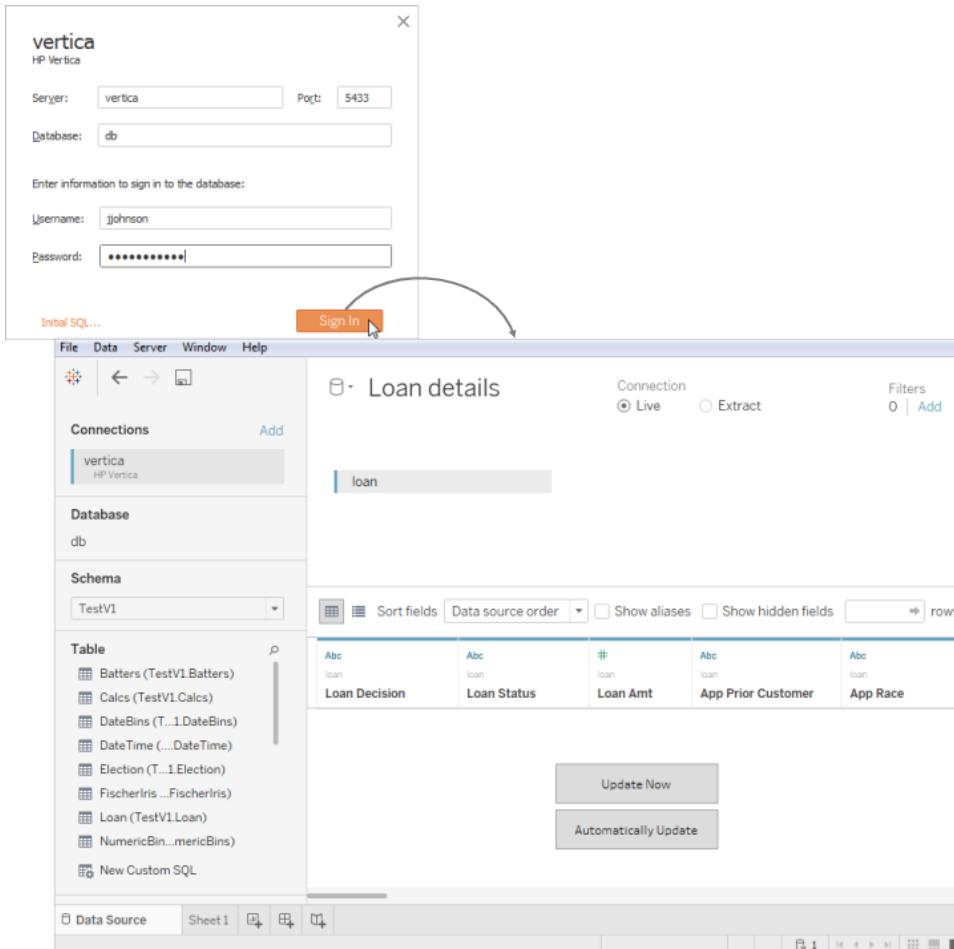
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

[Add data source filters to restrict the visibility and use of the fields in the data source](#)

Click **Add** at the top of the data source page to add data source filters .

HP Vertica data source example

Here is an example of an HP Vertica data source using Tableau Desktop on a Windows computer:



IBM BigInsights

This topic describes how to connect Tableau to an IBM BigInsights database.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to and port number
- Database name
- User name and password
- (Optional) Initial SQL statement to run every time Tableau connects

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **IBM BigInsights**, and then do the following:
 1. Enter the name of the server that hosts the database and the port number to use.
 2. Enter the name of the database you want to connect to.
 3. Enter your user name and password.
 4. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL on page 349](#).
 5. Click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, select a schema or use the text box to search for a schema by name.
 3. Under **Table**, select a table or use the text box to search for a table by name.
 4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

[View the data in the Tableau data source](#)

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

[Manage metadata](#)

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

[Sort fields and rows](#)

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

[Hide a field](#)

Click the column drop-down arrow and select **Hide**.

[Rename a field](#)

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

[Split columns](#)

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

[Create a new calculation based on an existing field in the data source](#)

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

IBM BigInsights data source example

Here is an example of an IBM BigInsights data source:

The screenshot shows the Tableau Data Source setup window for an IBM BigInsights connection named 'ibmbi.example.lan'. The 'Sign In' button is highlighted with a red arrow. The main pane displays the 'Loan details' schema, specifically the 'TESTV1_LOAN' table. The table has columns: TESTV1_LOAN (Loan Decision), TESTV2_LOAN (Loan Status), TESTV1_LOAN (Loan Amt), TESTV1_LOAN (App Prior Customer), and TESTV1_LOAN (App Race). Buttons for 'Update Now' and 'Automatically Update' are visible below the table preview. The left sidebar shows other connections like 'ibmbi.example.lan (IBM BigInsights)', databases ('BIGSQL'), and schemas ('TESTV1'). A list of tables is also present.

IBM DB2

This topic describes how to connect Tableau to an IBM DB2 database and set up the data source. You can also use this connector to connect to an IBM DB2 for z/OS database. Refer to

the [Technical Specifications](#) to confirm which DB2 databases are supported.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to and port number
- Database name
- User name and password
- (Optional) Initial SQL statement to run every time Tableau connects

The port number is dependent on the type of server you are connecting to and whether you are connecting to an encrypted port. Generally, use 50000 for a non-encrypted port and 60000 for an encrypted port. It is possible that your server is configured to use a non-standard port. Contact your administrator if you don't know which port to connect to.

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **IBM DB2**, and then do the following:
 1. Enter the name of the server that hosts the database and the name of the database that you want to connect to.
 2. Enter your user name and password, and then click **Sign In**.
If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
 3. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL on page 349](#).
 2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, select a schema or use the text box to search for a schema by name.
 3. Under **Table**, select a table or use the text box to search for a table by name.
 4. Drag a table to the canvas, and then click the sheet tab to start your analysis.
- Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

[Connect to live data or use an extract](#)

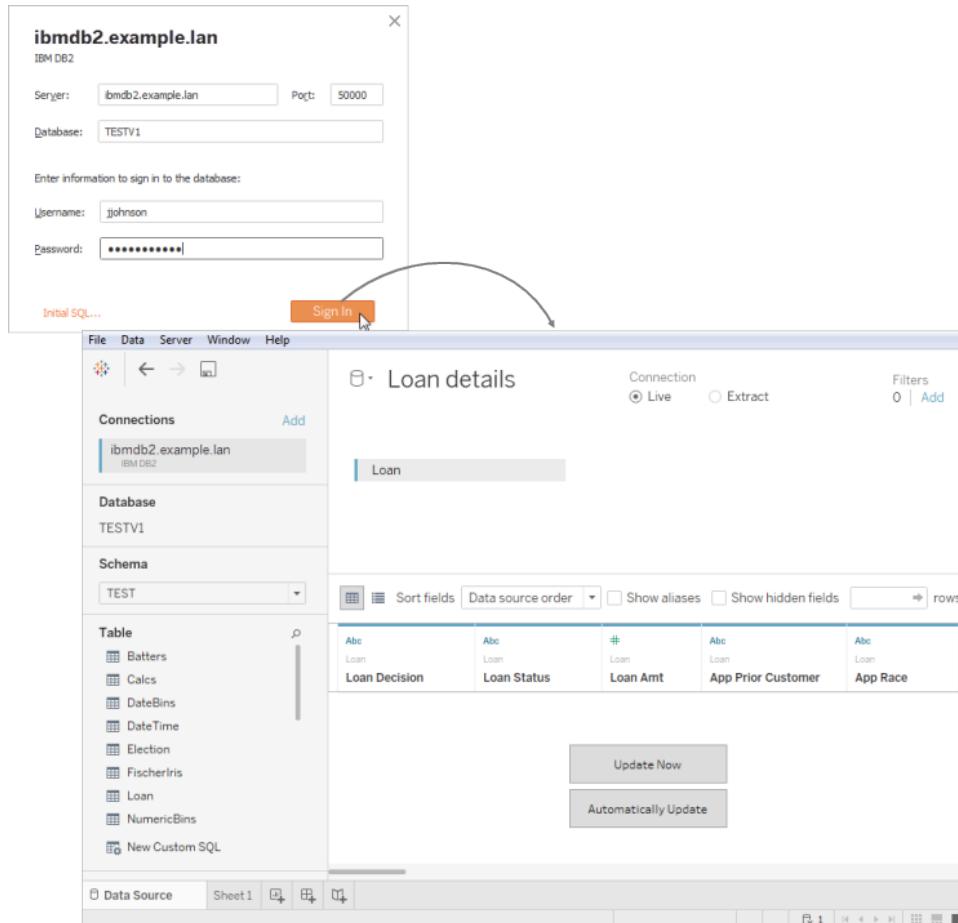
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

[Add data source filters to restrict the visibility and use of the fields in the data source](#)

Click **Add** at the top of the data source page to add data source filters .

IBM DB2 data source example

Here is an example of an IBM DB2 data source:



IBM PDA (Netezza)

This topic describes how to connect Tableau to an IBM PDA (PureData System for Analytics) database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Database name
- User name and password
- Are you connecting to an SSL server?

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **IBM PDA (Netezza)**, and then do the following:
 1. Enter the name of the server that hosts the database.
 2. Enter the name of the database that you want to connect to.
 3. Enter the user name and password, and then click **Sign In**.
Select the **Require SSL** check box when connecting to an SSL server.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. Under **Table**, select a table or use the text box to search for a table by name.
 3. Drag a table to the canvas, and then click the sheet tab to start your analysis.
Use custom SQL to connect to a specific query rather than the entire data source.
For more information, see [Connect to a Custom SQL Query on page 404](#).

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

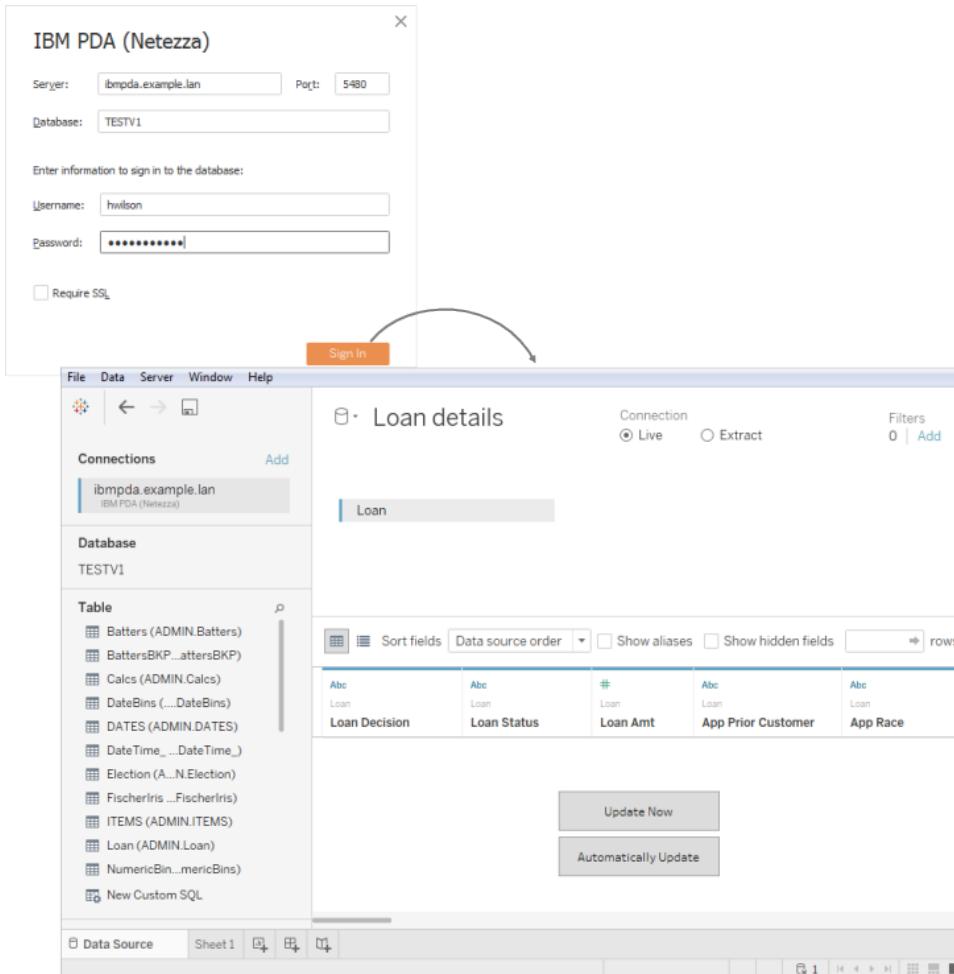
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

IBM PDA (Netezza) data source example

Here is an example of an IBM PDA (Netezza) data source:



Kognitio

This topic describes how to connect Tableau to a Kognitio database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- User name and password
- Are you connecting to an SSL server?

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Kognitio**, and then do the following:
 1. Enter the name of the server that hosts the database you want to connect to.
 2. Enter the user name and password, and then click **Sign In**.

Select the **Require SSL** check box when connecting to an SSL server.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. From the **Schema** drop-down list, select a schema or use the text box to search for a schema by name.
3. Under **Table**, select a table or use the text box to search for a table by name.
4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source.

For more information, see [Connect to a Custom SQL Query on page 404](#).

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

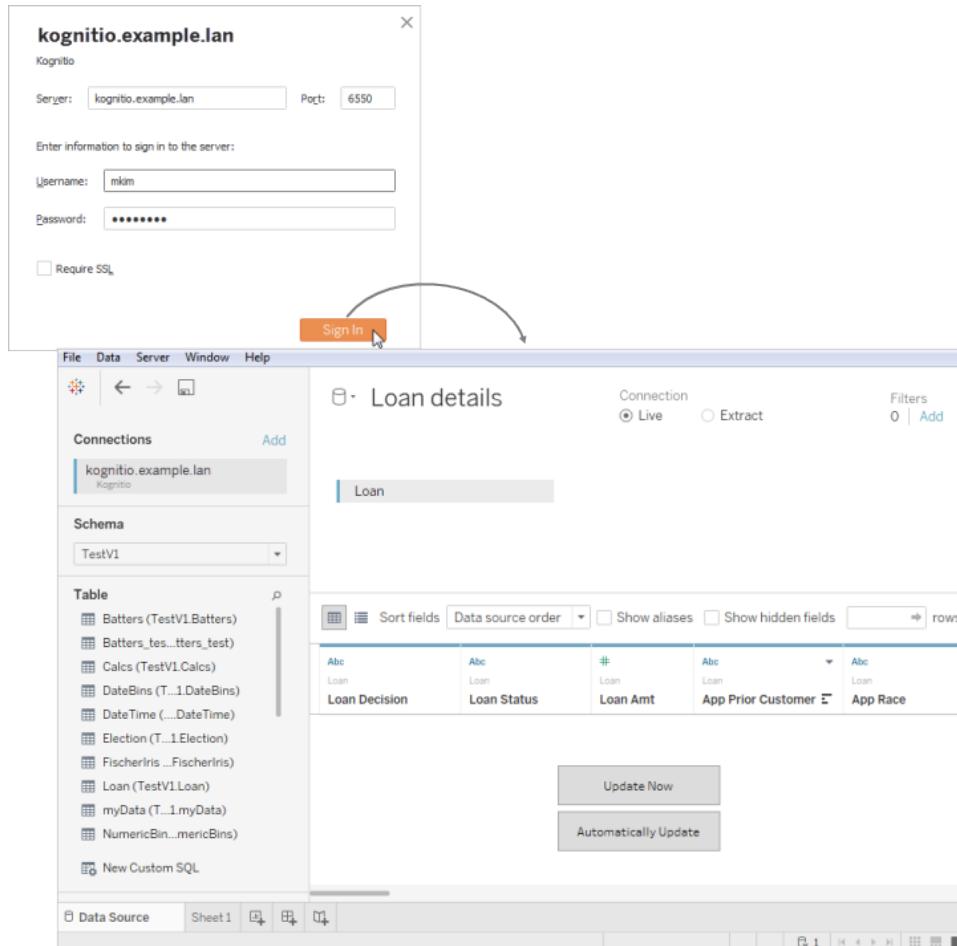
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Kognitio Data Source Example

Here is an example of a Kognitio data source using Tableau Desktop on a Windows computer:



MapR Hadoop Hive

This topic describes how to connect Tableau to a MapR Hadoop Hive database and set up the data source.

In this topic:

- [Before you begin](#) on the next page
- [Make the connection and set up the data source](#) on the next page
- [Sign in on a Mac](#) on page 1413
- [\(Optional\) Review and manage the data source](#) on page 1413
- [Work with Hadoop Hive Data](#) on page 1415

Before you begin

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Type of database: HiveServer, HiveServer2
- Authentication method. HiveServer doesn't require authentication. HiveServer2 authentication choices include the following:
 - No Authentication
 - Kerberos
 - User Name
 - User Name and Password
 - User Name and Password (SSL)
 - Microsoft Azure HDInsight Emulator
 - Microsoft Azure HDInsight Service
 - HTTP
 - HTTPS
- Sign-in credentials. Your choices depend on the authentication method you choose, and can include the following:
 - User name
 - Password
 - Realm
 - Host FQDN
 - Service Name
 - HTTP Path
- (Optional) Initial SQL statement to run every time Tableau connects

Note: Make sure you use the latest available drivers. To get the latest drivers, see [MapR Hadoop Hive](#) on the Tableau Driver Download page.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **MapR Hadoop Hive**, and then do the following:

1. Enter the name of the server that hosts the database.
2. In the **Type** drop-down list, select the type of database to connect to. You can select one of the following:
 - **HiveServer**
 - **HiveServer2**
3. In the **Authentication** drop-down list, select the authentication method to use.
4. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL on page 349](#).
5. Click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

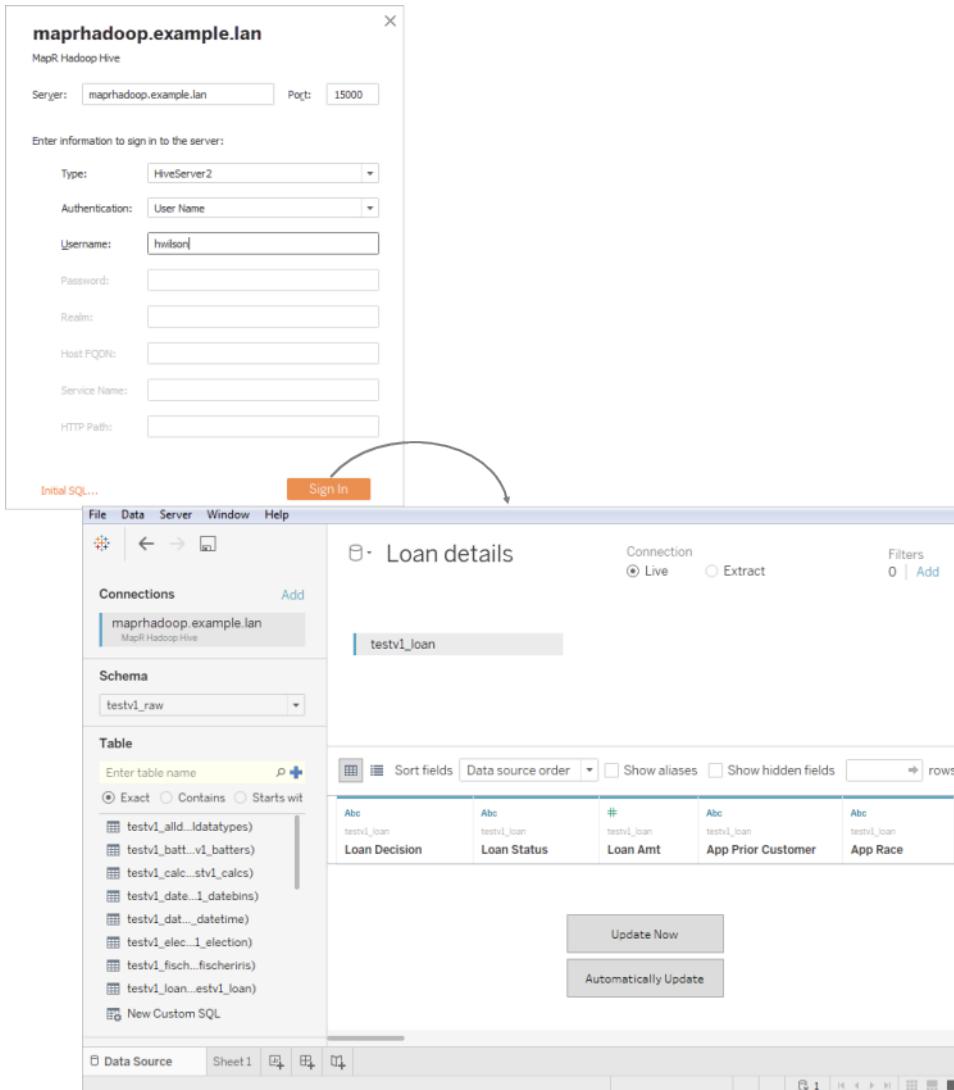
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, click the search icon or enter the schema name in the text box and click the search icon, and then select the schema.
 3. In the **Table** text box, click the search icon or enter the table name and click the search icon.
 4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Note: This database type supports only equal (=) join operations.

[MadR Hadoop data source example](#)

Here is an example of a MapR Hadoop Hive data source using Tableau Desktop on a Windows computer.



Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press **Ctrl+C** (**Command-C** on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Work with Hadoop Hive Data

Work with date/time data

Tableau supports **TIMESTAMP** and **DATE** types natively. However, if you store date/time data as a string in Hive, be sure to store it in ISO format (YYYY-MM-DD). You can create a calculated field that uses the **DATEPARSE** or **DATE** function to convert a string to a date/time format. Use **DATEPARSE()** when working with an extract, otherwise use **DATE()**. For more information, see **Date Functions on page 1565**.

For more information about Hive data types, see **Dates** on the Apache Hive website.

NULL value returned

A **NULL** value is returned when you open a workbook in Tableau 9.0.1 and later and 8.3.5 and later 8.3.x releases that was created in an earlier version and has date/time data stored as a string in a format that Hive doesn't support. To resolve this issue, change the field type back to **String** and create a calculated field using **DATEPARSE()** or **DATE()** to convert the date. Use **DATEPARSE()** when working with an extract, otherwise use the **DATE()** function.

High latency limitation

Hive is a batch-oriented system and is not yet capable of answering simple queries with very quick turnaround. This limitation can make it difficult to explore a new data set or experiment with calculated fields. Some of the newer SQL-on-Hadoop technologies (for example, Cloudera's Impala and Hortonworks' Stringer project) are designed to address this limitation.

Marketo

This topic describes how to connect Tableau to Marketo data and set up the data source. It also describes how date range selections can impact performance.

Before you begin, gather this connection information:

- Custom Service Endpoint
- Client ID
- Client Secret

For details about how to create the custom service information you need from Marketo to connect, see the [Custom Service](#) topic on the Marketo website.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Marketo**, and then do the following:
 1. Enter your Custom Service Endpoint, Client ID, and Client Secret.
 2. Click **Sign In**.
 3. Select **Filter Type: Relative date range** or **Fixed date range**, and then select or specify the range.
 4. Click **Connect** to authorize Marketo to securely share your data with Tableau Desktop.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network or Marketo administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. Under **Table**, select a table and drag it to the top of the canvas.
 3. Click the sheet tab to start your analysis.

After you click the sheet tab, Tableau imports the data by creating an extract. Note that Tableau Desktop supports only extracts for Marketo. You can update the data by refreshing the extract. For more information, see [Extract Your Data on page 440](#).

Creating extracts may take some time depending on the amount of data that is included.

Date range selections can impact performance

It's tempting to gather as much data as possible when you do an analysis, however, retrieving records from Marketo can be time-consuming. Tableau doesn't know how much data there is in a particular date range until it retrieves the data. For this reason, you should restrict your date range at first, and then expand after you evaluate performance.

To give you a rough idea of how much time it might take to retrieve data from Marketo, tests were conducted using a high-speed connection. This table shows how long it took in the test environment to retrieve a given number of records.

Number of records	Time to retrieve
1,000	4.5 seconds
10,000	45 seconds
100,000	8 minutes
1,000,000	75 minutes

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information see, [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button  to perform routine management tasks.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for Split on the Data pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

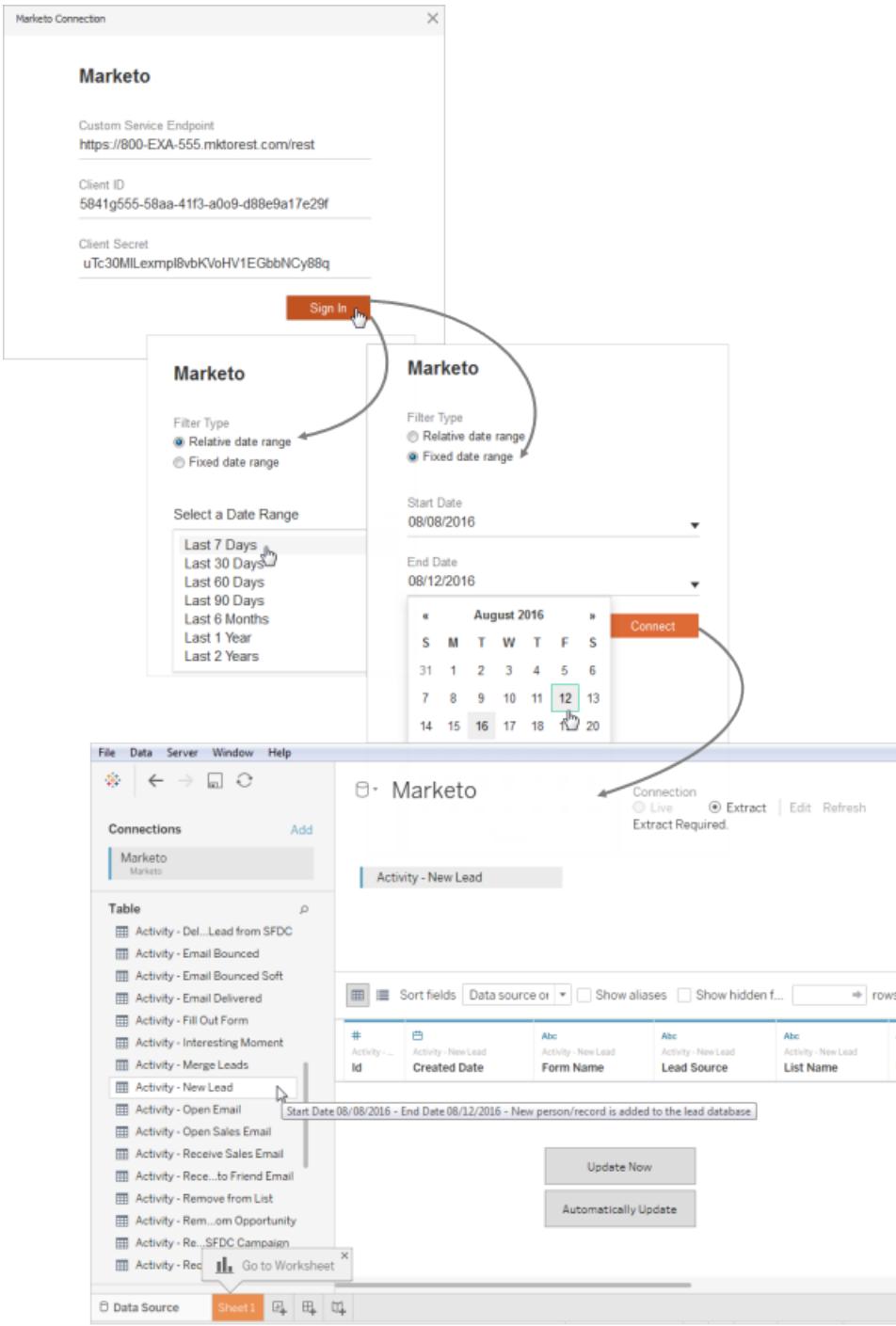
Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Marketo data source example

Here is an example of a Marketo data source using Tableau Desktop on a Windows computer:



MarkLogic

This topic describes how to connect Tableau to a MarkLogic database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Port number for the ODBC server process
- User name and password

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **MarkLogic**, and then do the following:
 1. Enter the name of the server that hosts the database you want to connect to.
 2. Enter the port number for the ODBC server process of the database you want to connect to.
 3. Enter your user name and password, and then click **Sign In**.

If the connection is unsuccessful, verify that your user name and password are correct and that the port number correctly identifies the MarkLogic database configured by your database administrator. If the connection continues to fail, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, select a schema or use the text box to search for a schema by name.
 3. Under **Table**, select a table or use the text box to search for a table by name.
 4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

[Get more data from different databases](#)

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

[View the data in the Tableau data source](#)

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

[Manage metadata](#)

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

[Sort fields and rows](#)

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

[Hide a field](#)

Click the column drop-down arrow and select **Hide**.

[Rename a field](#)

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

[Split columns](#)

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

[Create a new calculation based on an existing field in the data source](#)

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

MarkLogic data source example

Here is an example of a MarkLogic data source:

The screenshot shows the Tableau Data Source setup for a MarkLogic connection. At the top, there is a 'Sign In' dialog box with fields for Server (marklogic.example.com), Port (5432), Username (agarcia), and Password. A red arrow points from the 'Sign In' button in this dialog to the 'Sign In' button in the main Tableau interface. The main interface shows a 'Connections' pane with 'marklogic.example.com' selected. The 'Schema' pane shows 'testv1'. The 'Table' pane lists various views: batters_view, calcs_view, databin_view, datetime_view, election_view, fischeris_view, loan_view, numericbin_view, rei_view, and New Custom. The 'Loans' view is currently selected. The 'Fields' pane displays the fields for the 'loan_view' table, which includes App Employ Status, App Yrs Current Job, App Self Emp, App Tot Month Inc, Coapp Tot Month Inc, App Ext Credit Rating, and App Liquid Assets. The 'Connection' section at the top right indicates 'Live' is selected. The bottom navigation bar shows 'Data Source' and 'Sheet1'.

MemSQL

This topic describes how to connect Tableau to a MemSQL database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- User name and password
- Are you connecting to an SSL server?

Make the connection and set up the data source

1. On the start page, under **Connect**, click **MemSQL**, and then do the following:
 1. Enter the name of the server that hosts the database.
 2. Enter the user name and password, and then click **Sign In**.

Select the **Require SSL** check box when connecting to an SSL server.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Database** drop-down list, select a database or use the text box to search for a database by name.
 3. Under **Table**, select a table or use the text box to search for a table by name.
 4. Drag a table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press **Ctrl+C** (**Command-C** on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

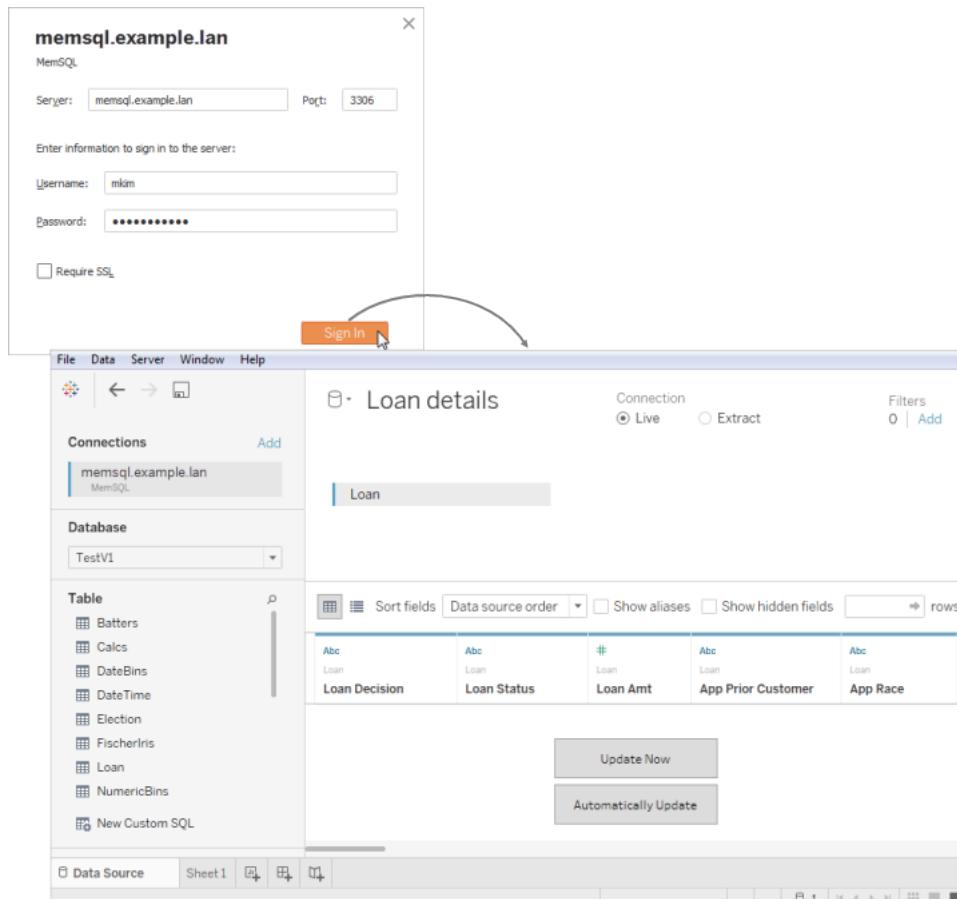
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

MemSQL data source example

Here is an example of a MemSQL data source using Tableau Desktop on a Windows computer:



In order to use your net services definitions in Tableau, you must set either TNS_ADMIN or ORACLE_HOME as an environment variable. To set TNS_ADMIN as the environment variable use the full path of the directory that contains the tnsnames.ora file. To set ORACLE_HOME as an environment variable use the path to the main Oracle directory.

Microsoft Analysis Services

This topic describes how to connect Tableau to a Microsoft Analysis Services database and set up a data source.

Before you begin, gather this connection information:

- For a remote cube file: Server name, or URL if connecting to the server using HTTP
- For a local cube file: File name
- Authentication method: Windows Authentication or user name and password

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Microsoft Analysis Services**, and then do the following:

1. Select whether to connect to a remote cube file on a server or to a local cube file.

To connect to a remote cube file, select **Server** and enter the name of the server in the text box. If you are connecting to the server using HTTP, you can enter the URL as the server name.

To connect to a local cube file, select **Local cube file** and click **Browse** to navigate to the cube file on your computer.

2. Select how you want to sign in to the server. Specify whether to use Windows Authentication or a specific user name and password. If the cube is password protected, and you are not in a Kerberos environment, you must enter your user name and password.

Specify whether to use Windows Authentication or a specific user name and password.

3. Click **Sign In**.

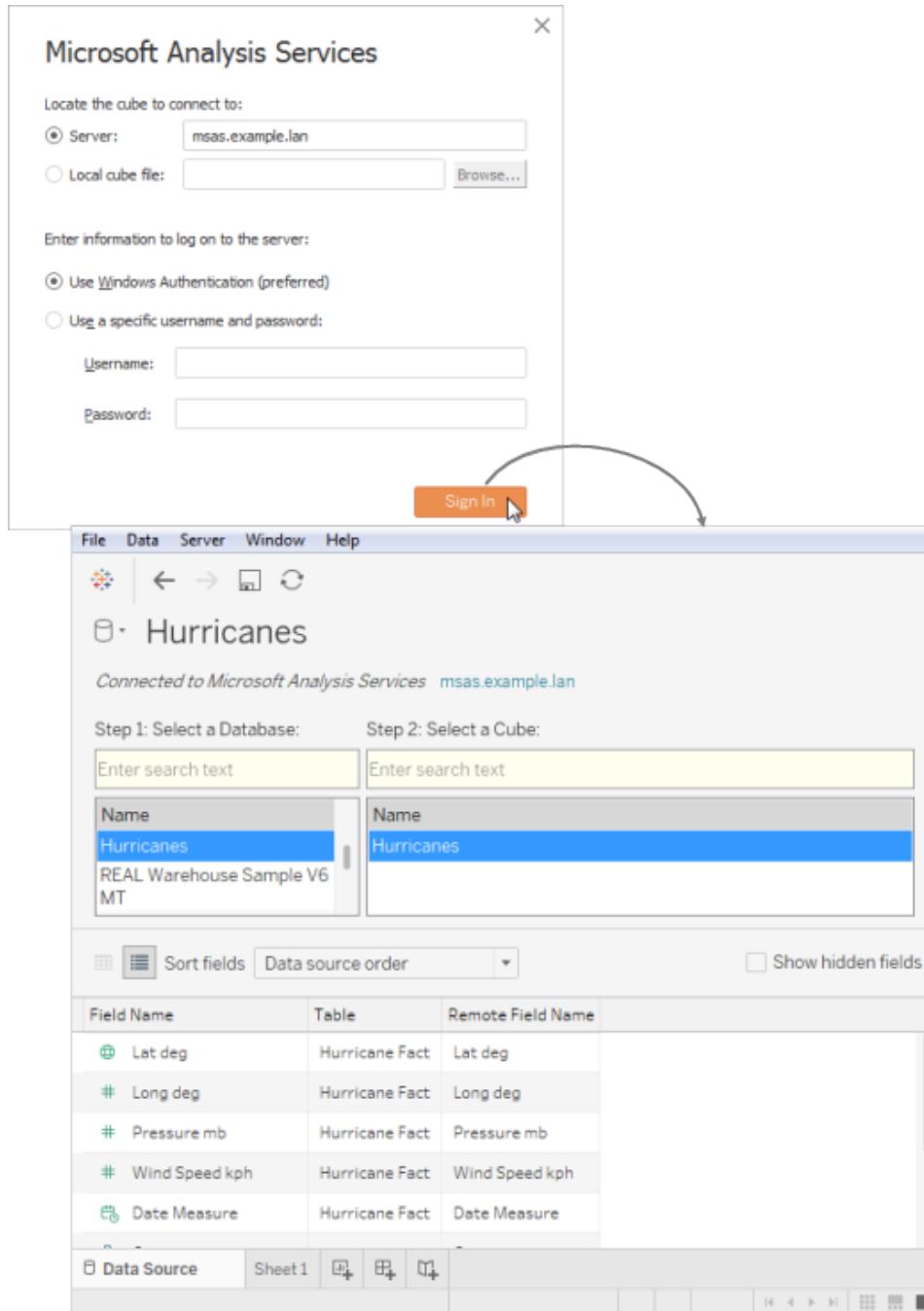
If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. Select a database.
3. Select a cube from the database.
4. Click the sheet tab to start your analysis.

Microsoft Analysis Services data source example

Here is an example of a Microsoft Analysis Services data source:



Microsoft PowerPivot

This topic describes how to connect Tableau to a Microsoft PowerPivot database.

Before you begin, gather this connection information:

- SharePoint URL, or SharePoint, UNC, or local Excel file name

If you plan to publish the workbook to Tableau Server, make sure to connect to a PowerPivot file on SharePoint. Tableau Server does not support connections to local PowerPivot files.

PowerPivot data may contain one or more perspectives. Perspectives are subsets of objects from the model that define sets of data. Typically perspectives are defined for a particular group of users or business scenario. Using perspectives can make it easier to navigate large data sources.

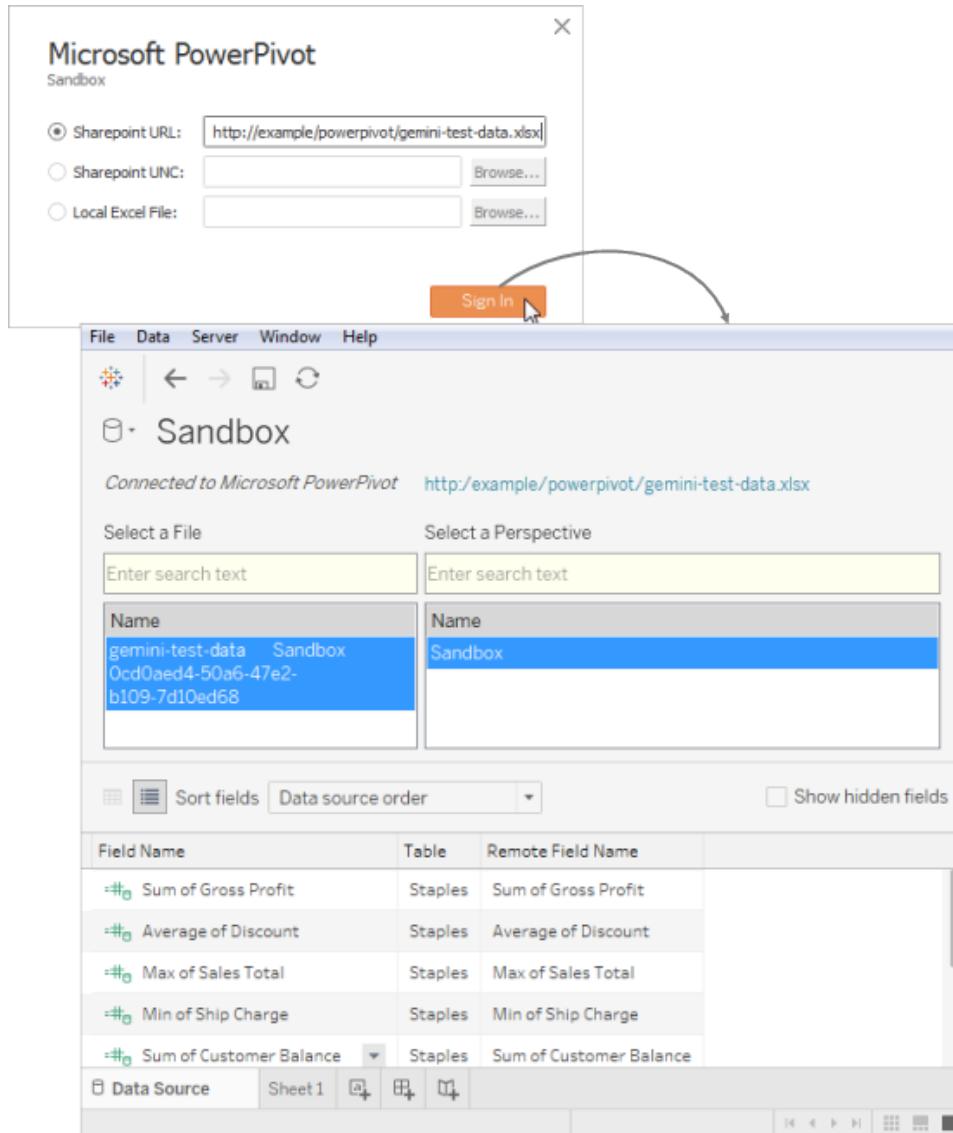
Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Microsoft PowerPivot**, and then do the following:
 1. Select whether to connect to a PowerPivot file using a SharePoint URL, a SharePoint UNC (file path), or a local Excel file.
 2. Click **Sign In**.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. The file name appears under **Select a File**. Note that there is one file per connection. Search for or select a perspective available in that file.
 3. Click the sheet tab to start your analysis.

Microsoft PowerPivot data source example

Here is an example of a Microsoft PowerPivot data source:



Microsoft SQL Server

This topic describes how to connect Tableau to a Microsoft SQL Server database and set up the data source.

Note: Use the Microsoft SQL Server connector to connect to Microsoft SQL Server Parallel Data Warehouse (PDW), Microsoft Azure SQL Data Warehouse, or Microsoft Azure SQL Database.

Before you begin, gather this connection information:

- Name of the server you want to connect to
- (Optional) Port number if you want to connect to a non-default port
- (Optional) Database name if you want to connect to a contained database
- Authentication method: Windows Authentication or user name and password
- Are you connecting to an SSL server?
- Do you want to set the database isolation level to read uncommitted data?
- (Optional) Initial SQL statement to run every time Tableau connects

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Microsoft SQL Server**, and then do the following:
 1. Enter the name of the server you want to connect to.
To connect to a non-default port, use the <server name>, <port number> format when entering the server name. For example, ExampleServer, 8055.
 2. (Optional) Enter a database name if you want to connect to a contained database.
 3. Select how you want to sign in to the server. Specify whether to use Windows Authentication or a specific user name and password. If the server is password protected, and you are not in a Kerberos environment, you must enter the user name and password.
Select the **Require SSL** check box when connecting to an SSL server.
 4. Specify whether to **Read uncommitted data**. This option sets the database isolation level to Read Uncommitted. Long queries from Tableau, including extract refreshes, can lock the database and delay transactions. Select this option to allow queries to read rows that have been modified by other transactions even when they have not been committed yet. When this option is cleared, Tableau uses the default isolation level specified by the database.
 5. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL on page 349](#).
 6. Click **Sign In**.
If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. From the **Database** drop-down list, select a database or use the text box to search for a database by name.
3. Under **Table**, select a table or use the text box to search for a table by name.
You can also specify a stored procedure in the database. For more information about stored procedures, including a list of constraints specific to SQL Server databases, see [Use a Stored Procedure on page 414](#).
4. Drag the table or stored procedure to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Note: Tableau Desktop does not support the Microsoft SQL Server TIME data type. Fields of this data type are not imported and do not appear in Tableau Desktop. If included in stored procedures, TIME data type fields will not appear in Tableau Desktop. For more information, see [Use a Stored Procedure on page 414](#).

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

Resolve "Worksheet Unavailable" error message

When you open a workbook, you may see a "Worksheet Unavailable" error message and then, when you click **Edit Connection**, you are prompted to provide your sign-in credentials. This error typically displays under one of two circumstances: you don't have access to a data connection used in the workbook, or your credentials are invalid, for example, your password has expired.

To resolve, verify that your sign-in credentials are correct. If they are, contact the Tableau administrator to ask about access to the data connection.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

[Connect to live data or use an extract](#)

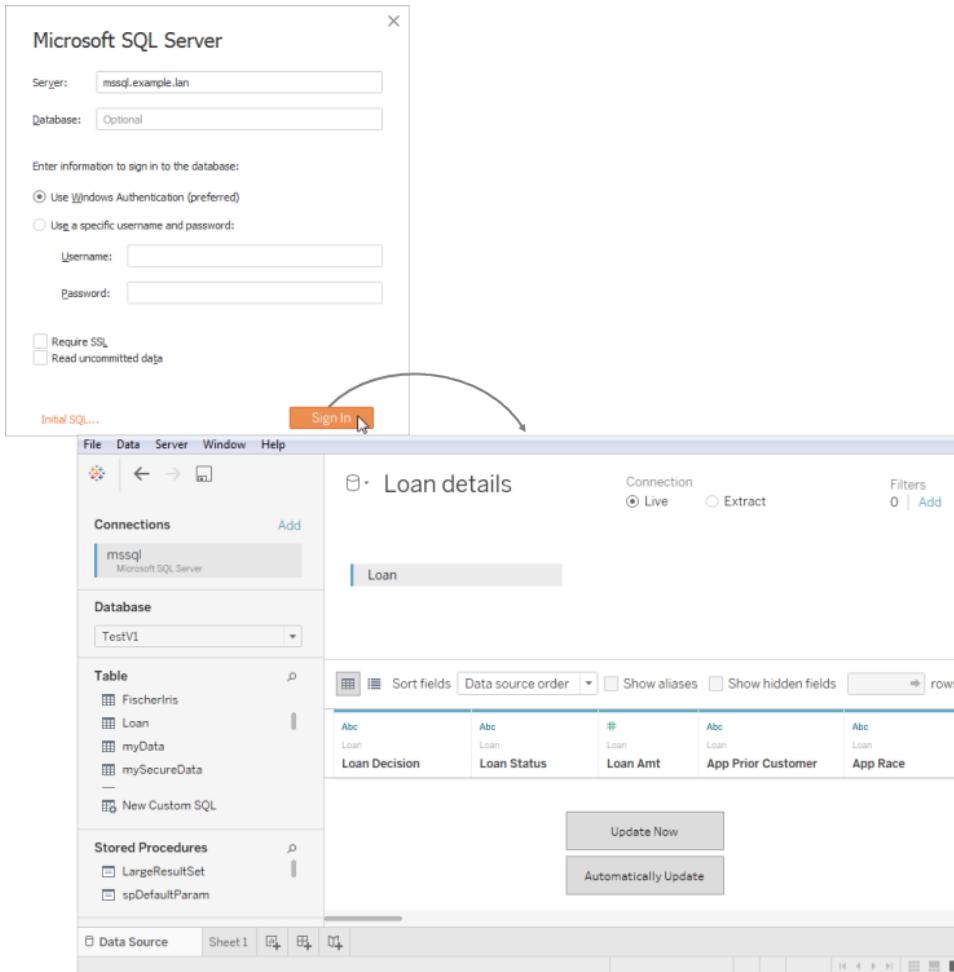
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

[Add data source filters to restrict the visibility and use of the fields in the data source](#)

Click **Add** at the top of the data source page to add data source filters .

Microsoft SQL Server data source example

Here is an example of a Microsoft SQL Server data source using Tableau Desktop on a Windows computer:



MonetDB

This topic describes how to connect Tableau to a MonetDB database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Database name
- User name and password

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **MonetDB**, and then do the following:
 1. Enter the name of the server that hosts the database and the name of the database that you want to connect to.
 2. Enter the user name and password, and then click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, select a schema.
 3. Under **Table**, select a table or use the text box to search for a table by name.
 4. Drag a table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

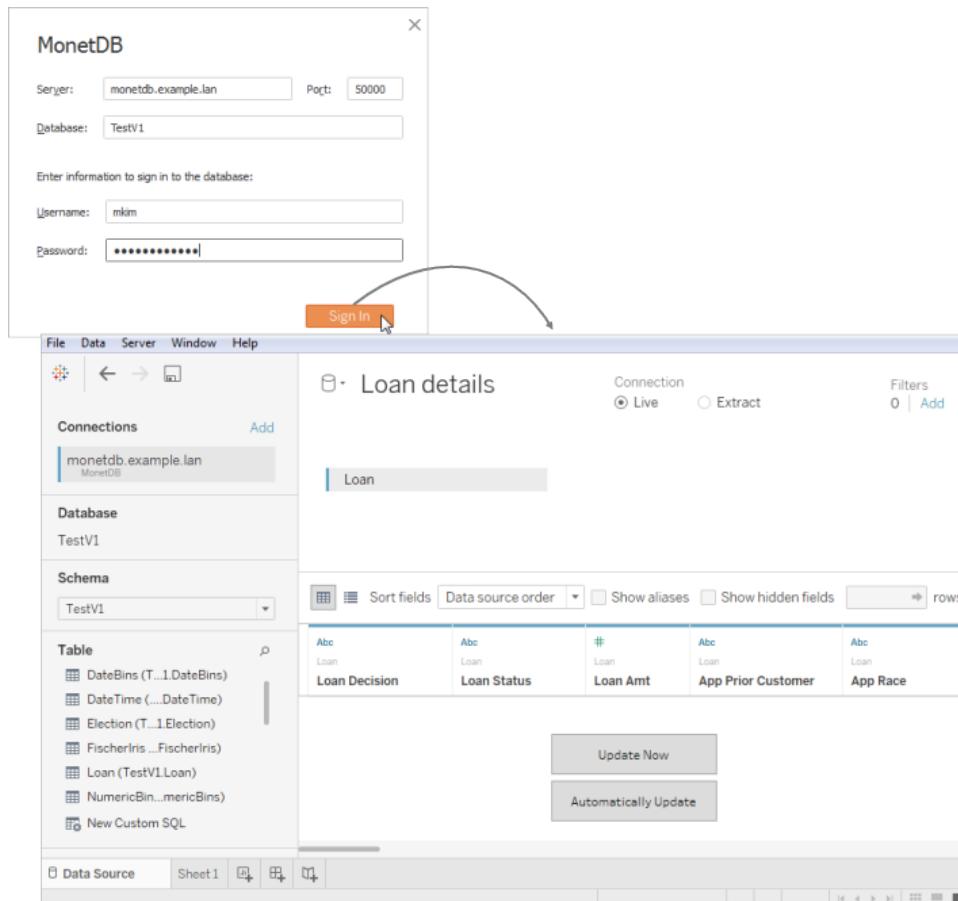
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

MonetDB data source example

Here is an example of a MonetDB data source:



MySQL

This topic describes how to connect Tableau to a MySQL database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- User name and password
- Are you connecting to an SSL server?

Make the connection and set up the data source

1. On the start page, under **Connect**, click **MySQL**, and then do the following:

1. Enter the name of the server that hosts the database.
2. Enter the user name and password, and then click **Sign In**.

Select the **Require SSL** check box when connecting to an SSL server.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. From the **Database** drop-down list, select a database or use the text box to search for a database by name.
3. Under **Table**, select a table or use the text box to search for a table by name.
4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source.

For more information, see [Connect to a Custom SQL Query on page 404](#).

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

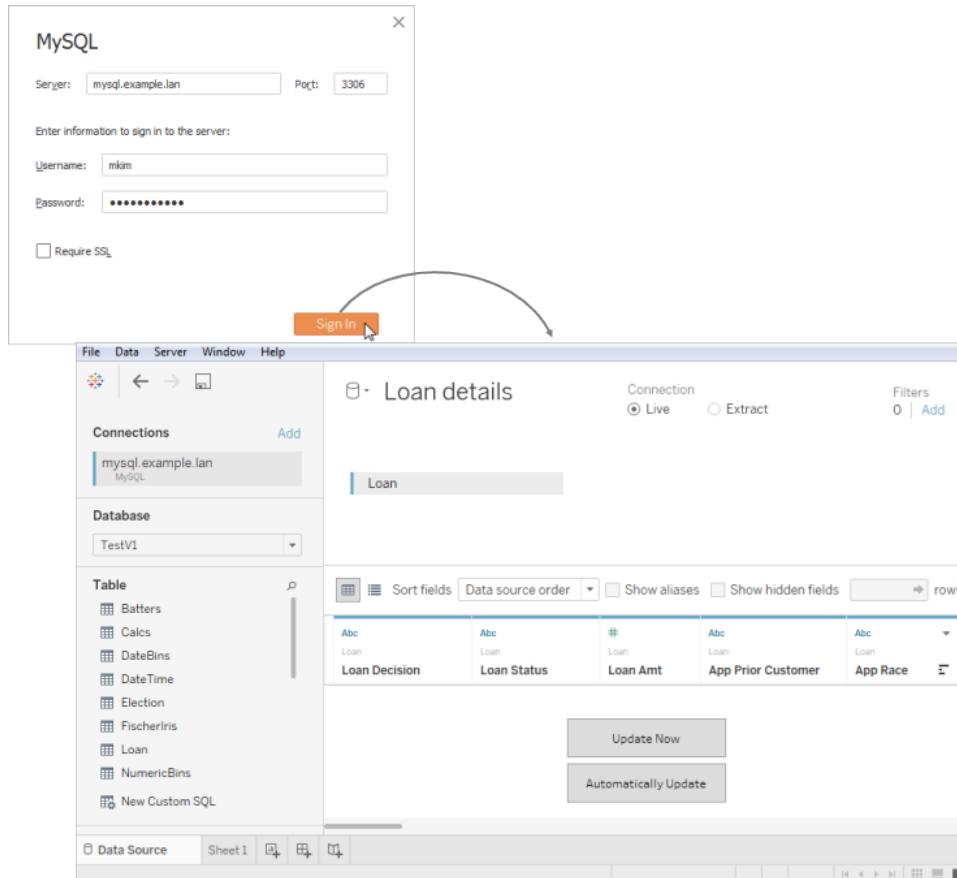
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

MySQL data source example

Here is an example of a MySQL data source using Tableau Desktop on a Windows computer:



OData

This topic describes how to connect Tableau to an OData data source. Tableau connects to OData V2.

Note the following:

- Use the OData connector to connect to your Windows Azure Marketplace data. Workbooks created in earlier versions of Tableau that use the Windows Azure Marketplace DataMarket connector will work as expected.
- The OData connector doesn't support browsing OData service documents.
- The OData connector doesn't support connecting to SharePoint Lists.

Before you begin, gather this connection information:

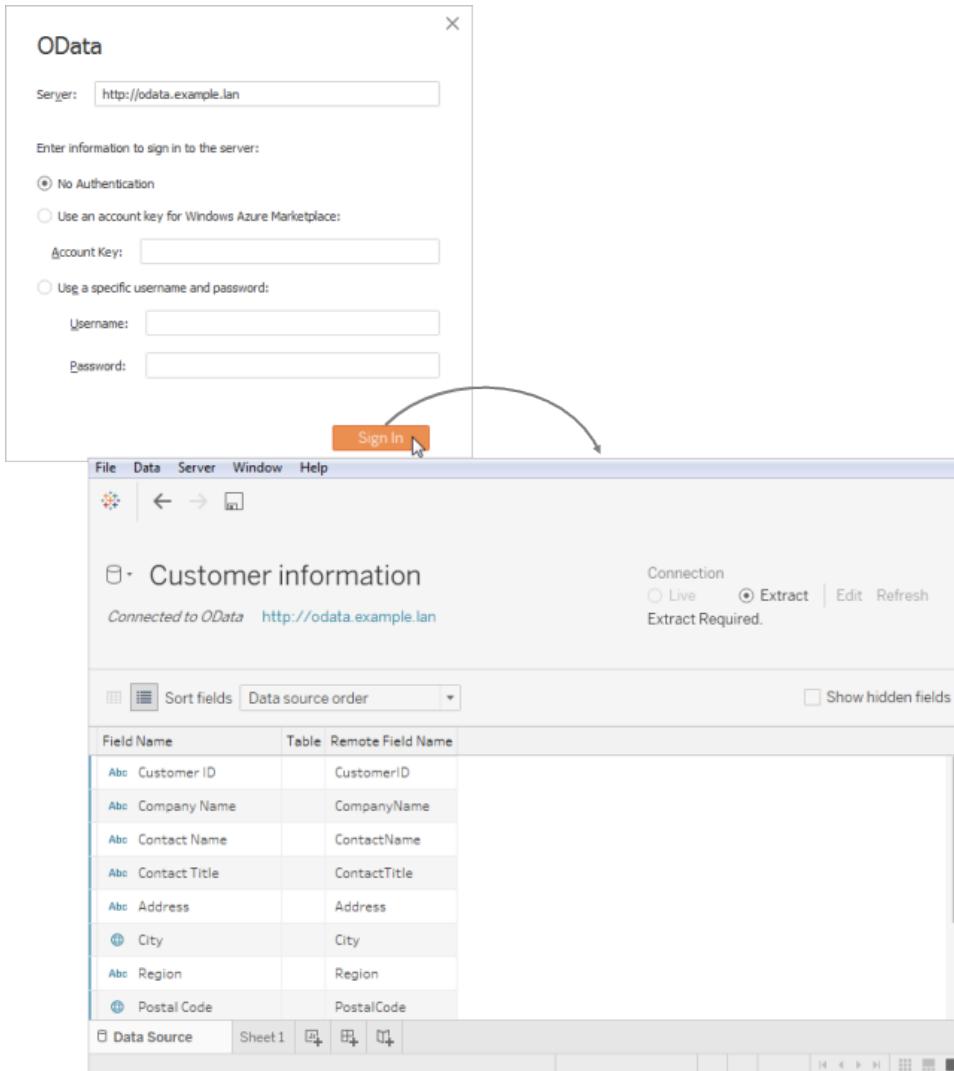
- Server URL for the data you want to connect to
- Sign-in credentials, if required
 - For Windows Azure Marketplace: Account key
 - User name and password

Make the connection and set up the data source

1. On the start page, under **Connect**, click **OData**, and then do the following:
 1. Enter the server URL for the data you are connecting to.
 2. If necessary, enter authentication information.
You can authenticate using your Windows Azure Marketplace (DataMarket) account key, or a user name and password.
 3. Click **Sign In**.
If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
Note: If you publish the data source to Tableau Server or Tableau Online, the data source name is limited to 255 characters. Longer names cause a `RepositoryException` error.
 2. Click the sheet tab to start your analysis.
After you click the sheet tab, Tableau imports the data by creating an extract. For more information about extracts, see [Extract Your Data on page 440](#). Note that Tableau Desktop supports only extracts for OData. Unlike many extract connections, extracts from OData data sources cannot be refreshed. To get the updated data, you must reconnect to the data source. If, however, you publish the extract to Tableau Server, you can refresh the extract.

OData data source example

Here is an example of an OData data source:



Oracle

This topic describes how to connect Tableau to an Oracle database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- (Optional) Oracle service name and port
- Authentication method: Integrated Authentication or user name and password
- (Optional) Initial SQL statement to run every time Tableau connects

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Oracle**, and then do the following:
 1. Enter the server name.
 2. (Optional) Enter the Oracle service name and port number.
 3. Select how you want to sign in to the server. Specify whether to use Integrated Authentication or a specific user name and password.
 4. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL on page 349](#).
 5. Click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Schema** drop-down list, click the search icon or enter the schema name in the text box and click the search icon, and then select the schema. **Note:** Search is case-sensitive.
 3. Under **Table**, click the search icon or enter the table name and click the search icon, and then select the table. **Note:** Search is case-sensitive.
- Tableau Desktop also supports connecting to Oracle table functions, which appear under **Stored Procedures** in the left pane of the Data Source page.
4. Drag a table to the canvas, and then click the sheet tab to start your analysis.
- Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Note: In order to use your net services definitions in Tableau, you must set either TNS_ADMIN or ORACLE_HOME as an environment variable. To set TNS_ADMIN as the environment variable use the full path of the directory that contains the tnsnames.ora file. To set ORACLE_HOME as an environment variable use the path to the main Oracle directory.

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click

the column drop-down arrow, and then select **Reset Name**.

[Split columns](#)

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields](#) on page 429.

[Create a new calculation based on an existing field in the data source](#)

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

[Connect to live data or use an extract](#)

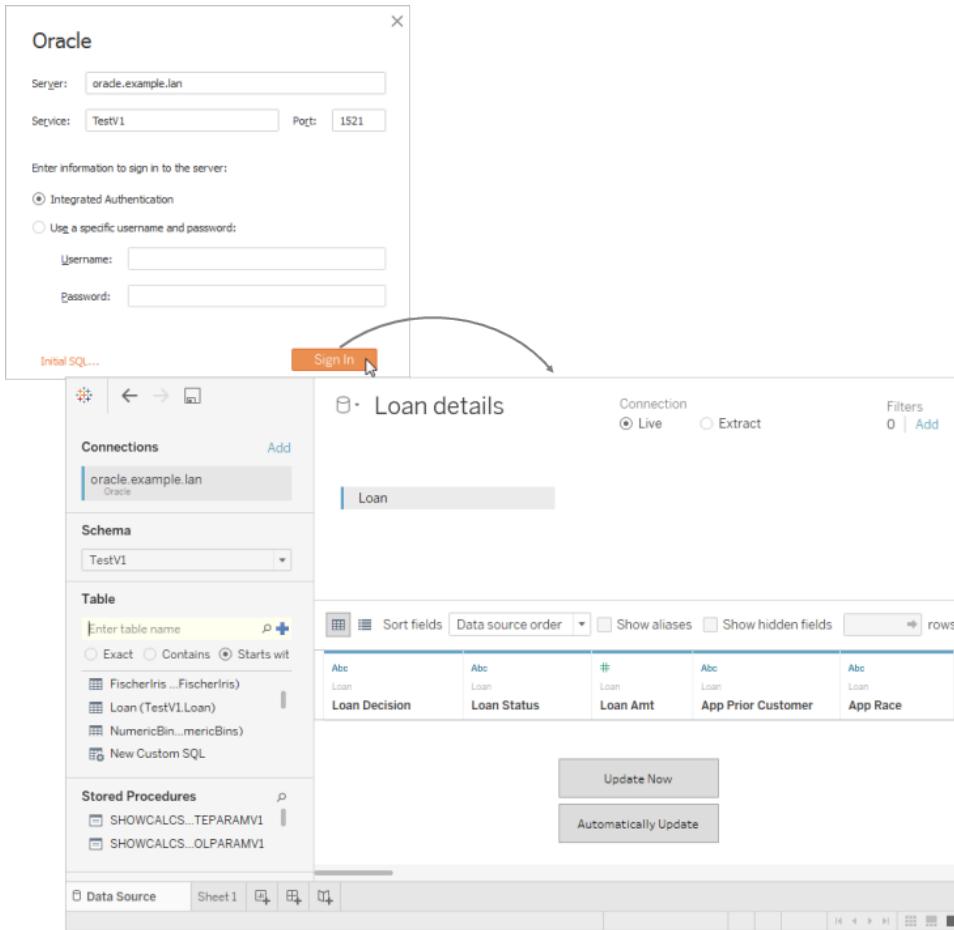
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

[Add data source filters to restrict the visibility and use of the fields in the data source](#)

Click **Add** at the top of the data source page to add data source filters .

Oracle data source example

Here is an example of an Oracle data source using Tableau Desktop on a Windows computer:



Oracle Eloqua

This topic describes how to connect Tableau to Oracle Eloqua data and set up the data source. It also describes how date range selections can impact performance.

Before you begin, gather and confirm this information for connecting:

- You must be a member of the following Eloqua security groups:
 - Advanced Users - Marketing
 - API Users
- Company Name
- Username
- Password

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Oracle Eloqua**, and then do the following:
 1. Click **Login**.
 2. Enter Company Name, Username, and Password, and then click **Sign In**.
 3. Click **Sign In** to grant Tableau access to Eloqua on your behalf.
 4. Click **Accept**.
 5. Select **Filter Type: Relative date range** or **Fixed date range**, and then select or specify the range.
 6. Click **Connect**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network or Eloqua administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. Under **Table**, select a table and drag it to the top of the canvas.
3. Click the sheet tab to start your analysis.

After you click the sheet tab, Tableau imports the data by creating an extract. Note that Tableau Desktop supports only extracts for Eloqua. You can update the data by refreshing the extract. For more information, see [Extract Your Data on page 440](#).

Creating extracts may take some time depending on the amount of data that is included.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information see, [Blend Your Data on page 369](#).

[View the data in the Tableau data source](#)

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

[Manage metadata](#)

Click the metadata grid button  to perform routine management tasks.

[Sort fields and rows](#)

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

[Hide a field](#)

Click the column drop-down arrow and select **Hide**.

[Rename a field](#)

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

[Create a new calculation based on an existing field in the data source](#)

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

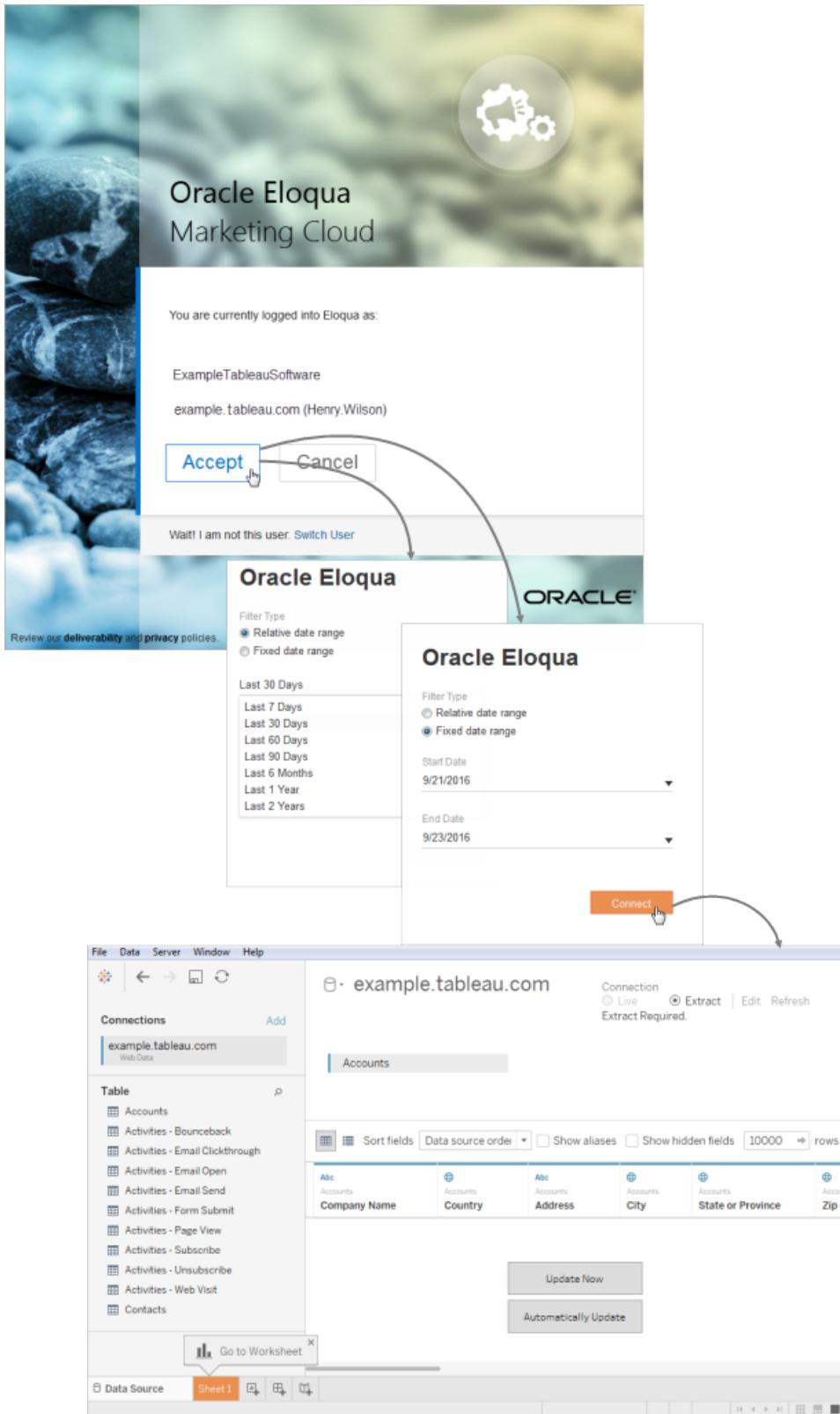
Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

Oracle Eloqua data source example

Here is an example of an Oracle Eloqua data source using Tableau Desktop on a Windows computer:



Date range selections can impact performance

It's tempting to gather as much data as possible when you do an analysis, however, retrieving records from Eloqua can be time-consuming. Tableau doesn't know how much data there is in a particular date range until it retrieves the data. For this reason, you should restrict your date range at first, and then expand after you evaluate performance. You can modify the date range on the Data Source page. Under **Connections**, click the drop-down arrow for the connection and select **Edit connection**.

To give you a rough idea of how much time it might take to retrieve data from Eloqua, tests were conducted using a high-speed connection. This table shows how long it took in the test environment to retrieve a given number of records. The connector retrieves 1000 records per request.

Number of records	Time to retrieve
50,662 records	3 minutes 17 seconds
95,214 rows	6 minutes 14 seconds
194,679 rows	12 minutes 55 seconds

Eloqua Limits

Field limits

Some Eloqua tables can be customized to have more than 250 fields. Tableau currently retrieves the first 250 fields and ignores the rest.

Daily table request limits

Each table requested by Tableau counts against a daily Eloqua request limit of 2000. When the limit is exceeded, Eloqua operations might choose to disable Tableau, in which case, they will contact the account owners. You can monitor your daily usage in Eloqua using the Eloqua Marketing Operations Center.

Oracle Essbase

This topic describes how to connect Tableau to an Oracle Essbase database.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- User name and password

Use this connector with Tableau Desktop on a Windows computer.

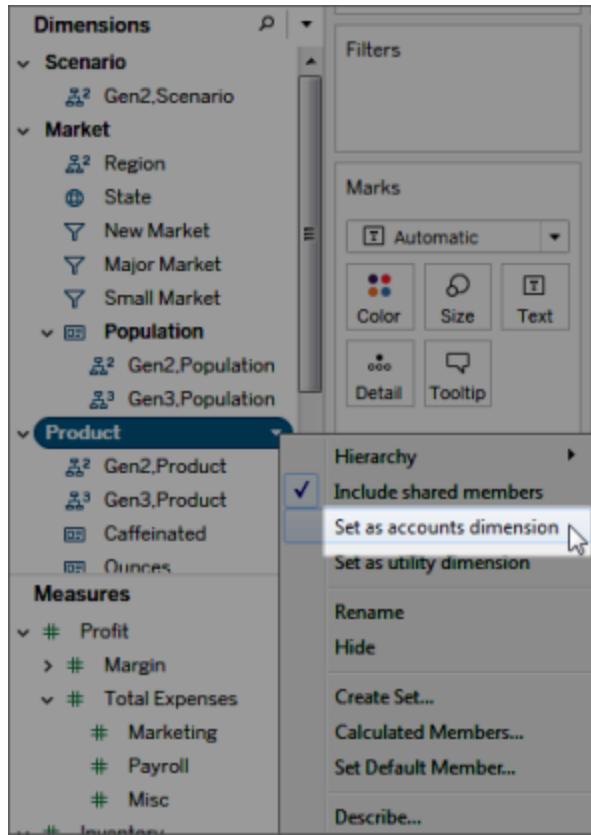
Make the connection and set up the data source

1. On the start page, under **Connect**, click **Oracle Essbase**, and then do the following:
 1. Enter the name of the server that hosts the database.
 2. Enter your user name and password to sign in to the server, and then click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. Search for or select an application.
 3. Search for or select a database from your application.
 4. Click the sheet tab to start your analysis.

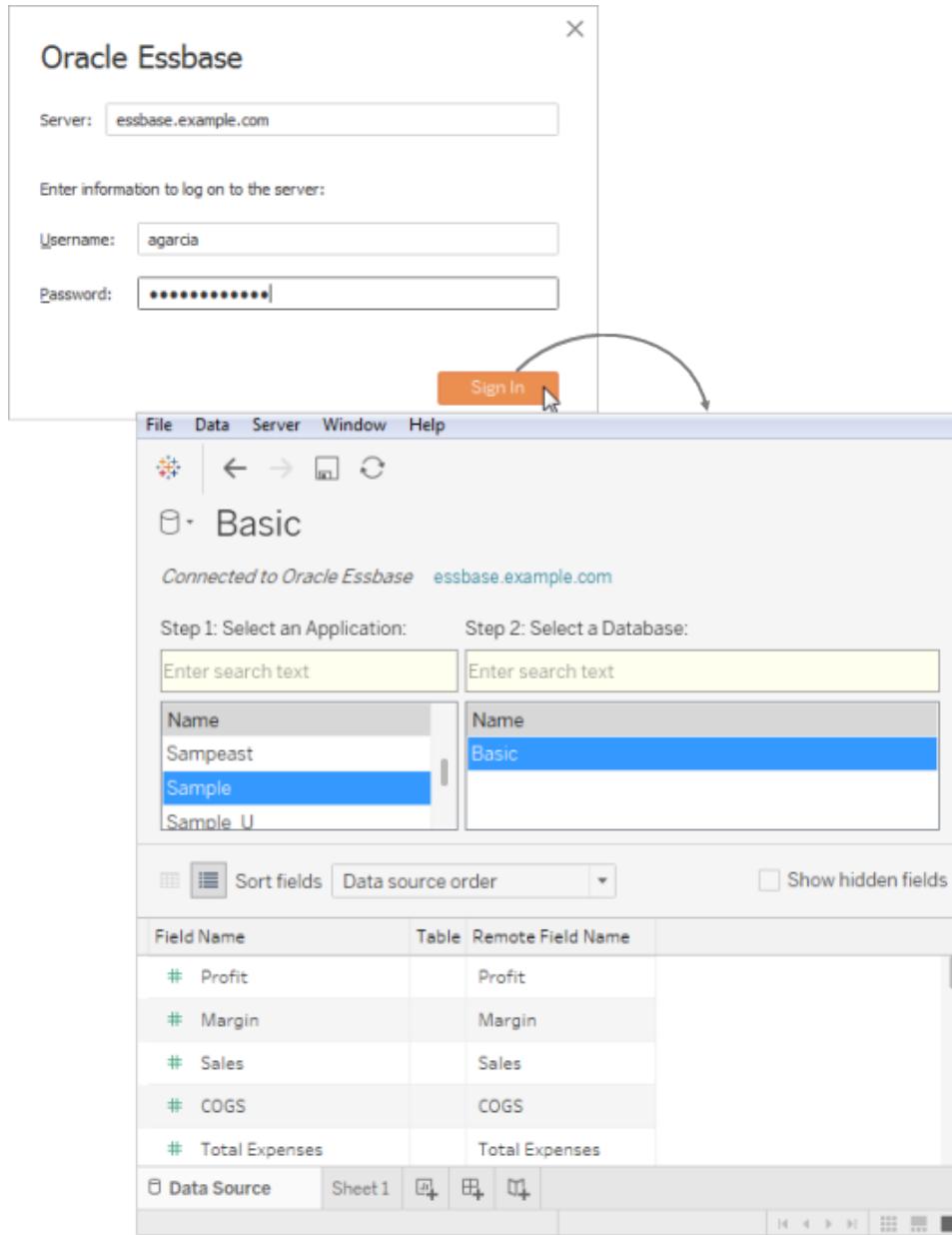
Set an Accounts Dimension

In some case, the accounts dimension for your data source can appear in the Dimensions area of the Data pane. This might occur if there is an error in the cube and another field is identified as the accounts dimension or there is no accounts dimension set at all. The accounts dimension defines the fields that are included as measures. To correct this error, right-click the field and then select **Set as Accounts Dimension** from the context menu.



Oracle Essbase data source example

Here is an example of an Oracle Essbase data source:



Pivotal Greenplum Database

This topic describes how to connect Tableau to a Pivotal Greenplum Database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Database name

- User name and password
- (Optional) Initial SQL statement to run every time Tableau connects

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Pivotal Greenplum Database**, and then do the following:
 1. Enter the name of the server that hosts the database and the name of the database you want to connect to.
 2. Enter the user name and password.
 3. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL on page 349](#).
 4. Click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. Select and drag a table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press **Ctrl+C** (**Command-C** on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

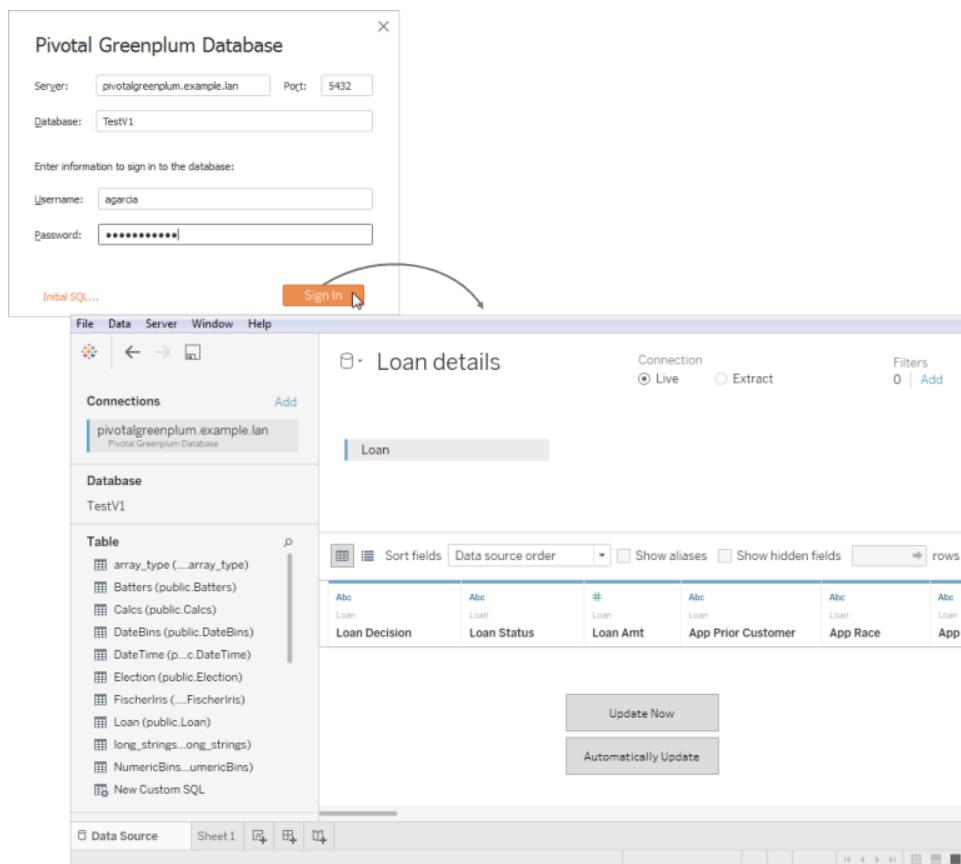
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Pivotal Greenplum Database data source example

Here is an example of a Pivotal Greenplum Database data source using Tableau Desktop on a Windows computer.



PostgreSQL

This topic describes how to connect Tableau to a PostgreSQL database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Database name
- Authentication method: Integrated Authentication or user name and password
- Are you connecting to an SSL server?

Note: If you're using a Mac, and it is not attached to the domain correctly, the Mac won't know that Kerberos is being used in the domain, and the **Authentication** drop-down list won't be available.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **PostgreSQL**, and then do the following:
 1. Enter the name of the server that hosts the database that you want to connect to.
 2. Enter the name of the database.
 3. Select how you want to sign in to the server. Specify whether to use **Integrated Authentication** or **User Name and Password**. If the server is password protected, and you are not in a Kerberos environment, you must enter the user name and password.
 4. Click **Sign In**.

Select the **Require SSL** check box when connecting to an SSL server.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. Under **Table**, select a table or use the text box to search for a table by name.
 3. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query](#) on page 404.

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click

the column drop-down arrow, and then select **Reset Name**.

[Split columns](#)

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields](#) on page 429.

[Create a new calculation based on an existing field in the data source](#)

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

[Connect to live data or use an extract](#)

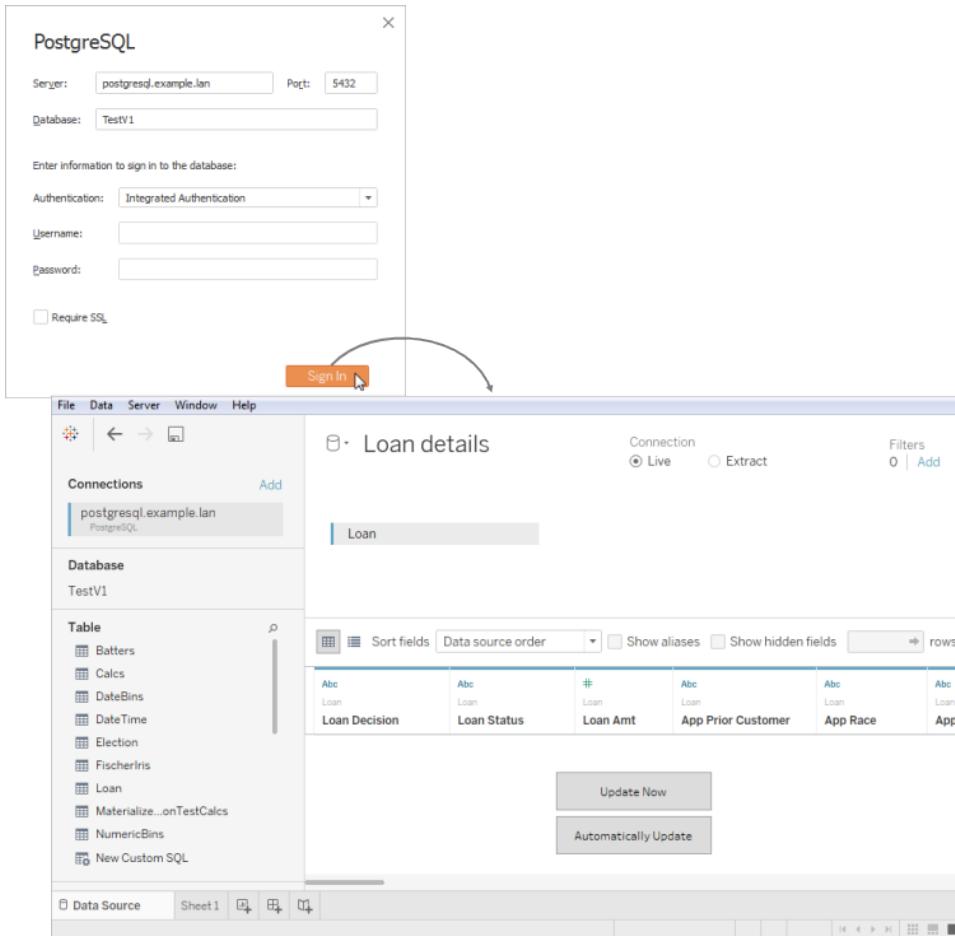
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

[Add data source filters to restrict the visibility and use of the fields in the data source](#)

Click **Add** at the top of the data source page to add data source filters .

PostgreSQL data source example

Here is an example of a PostgreSQL data source using Tableau Desktop on a Windows computer.



Presto

This topic describes how to connect Tableau to a Presto database and set up the data source. Tableau connects to Presto 141t from Teradata.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Database name
- User name

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Presto**, and then do the following:
 1. Enter the name of the server you want to connect to.
 2. Enter the database name.
 3. Enter the user name to sign in to the database.

4. Click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. From the **Schema** drop-down list, click the search icon or enter the schema name in the text box and click the search icon, and then select the schema.
3. In the **Table** text box, click the search icon or enter the table name and click the search icon, and then select the table.
4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.example.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, add the domain to the list of Search Domains for the Mac computer. To update the list of Search Domains, go to **System Preferences > Network > Advanced**; then open the DNS tab. If you do this, then you need only provide the server name when connecting.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update**

Now again to see your changes.

- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields](#) on page 429.

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

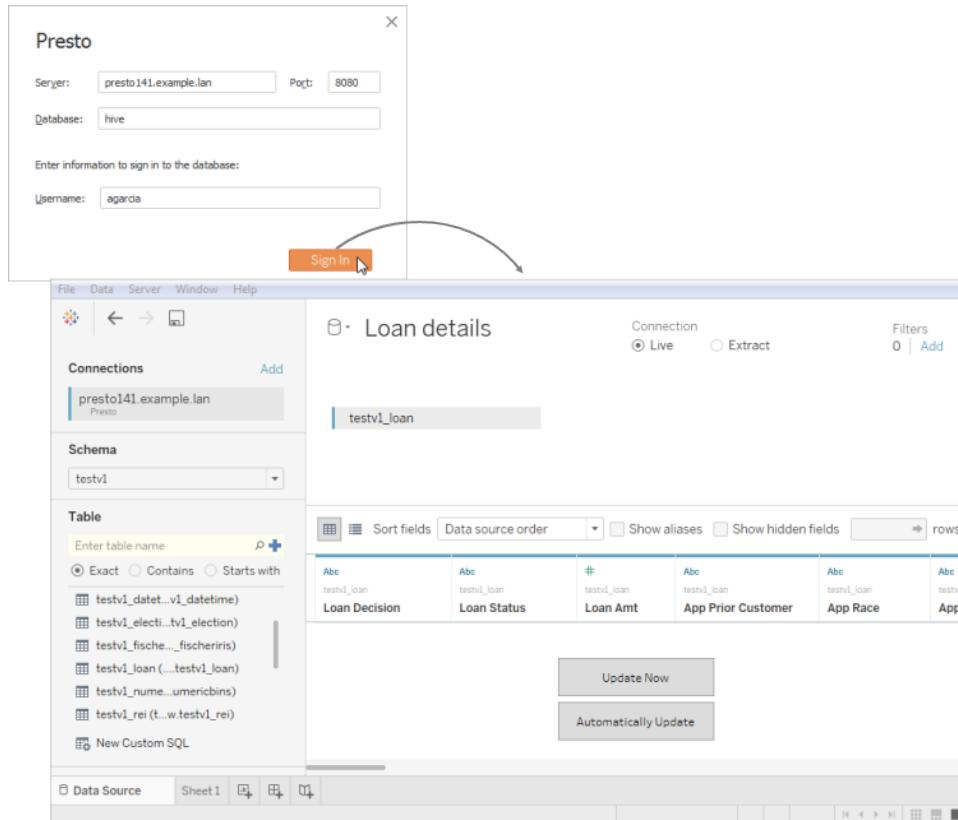
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Presto data source example

Here is an example of a Presto data source using Tableau Desktop on a Windows computer:



Progress OpenEdge

This topic describes how to connect Tableau to a Progress OpenEdge database and set up the data source.

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to
- Database name
- User name and password

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Progress OpenEdge**, and then do the following:
 1. Enter the name of the server that hosts the database you want to connect to.
 2. Enter the name of the database.
 3. Enter the user name and password, and then click **Sign In**.
If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. Under **Table**, select a table or use the text box to search for a table by name.
 3. Drag the table to the canvas, and then click the sheet tab to start your analysis.
Use custom SQL to connect to a specific query rather than the entire data source.
For more information, see [Connect to a Custom SQL Query on page 404](#).

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

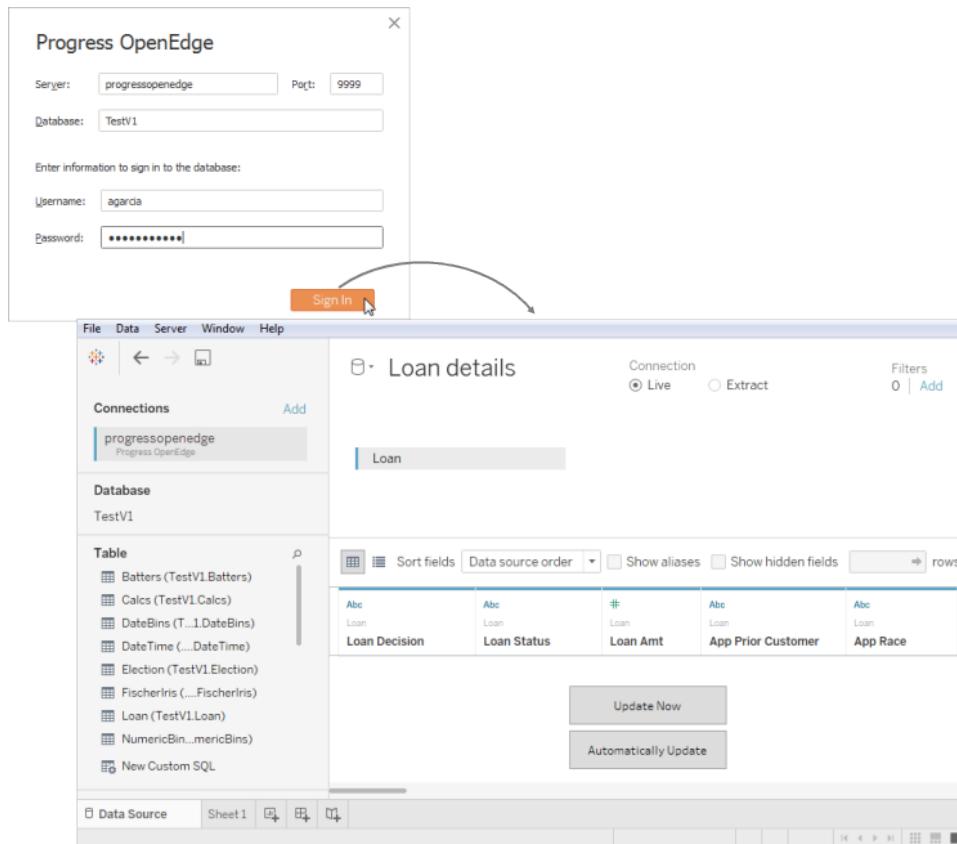
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Progress OpenEdge data source example

Here is an example of a Progress OpenEdge data source:



QuickBooks Online

This topic describes how to connect Tableau to QuickBooks Online data and set up the data source. It also describes an example of how to create a workbook with multiple user accounts and how to refresh the data in the workbook. Finally, you can review troubleshooting suggestions for QuickBooks Online errors.

Before you begin, gather this connection information:

- Your email address or user ID and password for your QuickBooks Online account

Note: To connect Tableau to QuickBooks Online data, you must be a Company Administrator on the QuickBooks Online company. Only one administrator per company can connect Tableau to QuickBooks Online.

Make the connection and set up a data source

1. On the start page, under **Connect**, click **QuickBooks Online**, and then do the following:

1. Enter your email address or user ID and password for your QuickBooks Online account.
2. Click **Sign In**.

QuickBooks Online periodically prompts for two-step verification. If this happens, follow the prompts to get a code to confirm your Intuit account.

3. If more than one company is associated with your account, select the company you want to connect to.
4. Click **Authorize** to authorize Intuit to securely share your data with Tableau Desktop.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. Under **Sheets**, select a sheet and drag it to the top of the canvas.
3. Click the sheet tab to start your analysis.

After you click the sheet tab, Tableau imports the data by creating an extract. Note that Tableau Desktop supports only extracts for QuickBooks Online. You can update the data by refreshing the extract. For more information, see [Extract Your Data on page 440](#).

Creating extracts may take some time depending on the amount of data that is included.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).

- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data](#) on page 369.

[View the data in the Tableau data source](#)

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

[Manage metadata](#)

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

[Sort fields and rows](#)

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

[Hide a field](#)

Click the column drop-down arrow and select **Hide**.

[Rename a field](#)

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

[Split columns](#)

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields](#) on page 429.

[Create a new calculation based on an existing field in the data source](#)

Click the column drop-down arrow and select **Create calculated field**.

[Copy values](#)

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

[Copy values in the metadata grid](#)

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Create a workbook that connects to multiple user accounts

You may want to create a workbook that connects to multiple QuickBooks Online user accounts. When you do this, pay close attention to which account you are signed in to when you refresh the extract for each account.

Here is an example of the general process you follow to create and then refresh the data for a workbook that connects to user account A and user account B.

Create the workbook

1. Follow the steps in "Make a connection and set up a data source" section to connect to user account A.
2. Click the Tableau icon to open the **Connect** pane and repeat the steps to connect to user account B.
Note that you are already signed in to user account A, but you want to sign in to user account B. To do this, on the "Authorize Intuit to securely share..." screen, in the upper-right corner, click "Not you?" to sign in with a different user account.
3. Create your view and then save this example as a packaged workbook (.twbx).

Refresh the data in the workbook

Open the workbook and sign in to each account.

1. When you sign in to user account A, be sure to select the correct company.
2. Refresh the extract.
3. When you sign in to user account B, remember that you are already signed in to user account A, so click the "Not you?" link to sign in with a different user account. Be sure to select the correct company.
4. Refresh the extract.

Note: Tableau refreshes the extract with data from the account you're signed in with and the company you select, so be sure that you've made the correct choices. If the company

name is a dimension in the data, you can check it to confirm that the data is what you expect.

Troubleshoot QuickBooks Online errors

You might see one of the following errors when you try to connect Tableau to your QuickBooks Online data.

[Sorry, only administrators can buy apps...](#)

QuickBooks Online requires that each user who connects an app to QuickBooks Online be a Company Administrator. The following Intuit error appears if you are not an administrator on the company account:

Sorry, only administrators of <email address>'s Company can buy apps in the company.
Please contact administrator in the company or else choose another company.

To resolve this issue, ask your company account owner to make you a Company Administrator on the account.

[Error Code: app_already_purchased](#)

Only one Company Administrator is allowed to connect per application, for example to Tableau Desktop. If someone in your company has already connected Tableau to your QuickBooks Online account, the following Intuit error appears:

Oops! An error has occurred. Please close this window and try again.

[Error Code: app_already_purchased](#)

Message: The application has already been subscribed to by another user for this company. Please contact <email address> to make changes to this subscription.

To resolve this issue, you need to ask that Company Administrator to give up the application privileges for Tableau Desktop and provide them to you. If that's not possible, for example, because the administrator actively uses Tableau, another option is for the administrator to publish the data source to Tableau Server or Tableau Online, so that anyone with permission can access the data.

[Changing a Connection Opens the My Apps Page](#)

If you need to access another QuickBooks Online company, after you sign in (and possibly go through the two-step authentication process), Intuit might show the My Apps page, rather than the "select the company to connect to" page. To resolve this issue, follow these steps:

1. Close the My Apps page.
2. On the Tableau **Connect** pane, select QuickBooks Online.
3. If more than one company is associated with your account, select the company you would like to connect to.
4. Click **Authorize** to open the Tableau data source page.

QuickBooks Online data source example

Here is an example of a QuickBooks Online data source using Tableau Desktop on a Windows computer:



Salesforce

This topic describes how to connect Tableau to Salesforce.com data and set up a data source.

Before you begin, gather this connection information:

- Salesforce user name and password

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under Connect, click **Salesforce**, and then do the following:

1. Enter your user name and password for Salesforce.com.
2. Click **Log In**.
3. In the Allow Access dialog box, click **Allow**.

If the connection is unsuccessful, verify that the authentication information is correct. If the connection continues to fail, your computer is having trouble locating the server or you may not have permission to access the data. Contact your network administrator or Salesforce administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. Select a standard connection or create your own custom connection.

Under **Standard Connection**, you can select from a list of predefined queries, which represent commonly used objects, and drag it to the canvas. Or, under **Table**, you can select a single table, which includes any custom objects your company has created, and drag it to the canvas.

3. Click the sheet tab to start your analysis.

After you click the sheet tab, Tableau imports the data by creating an extract. Note that Tableau Desktop supports only extracts for Salesforce. You can update the data by refreshing the extract. For more information, see [Extract Your Data on page 440](#).

The initial extract may take some time depending on the amount of data that is included. After the initial extract, you can incrementally update objects by refreshing the extract.

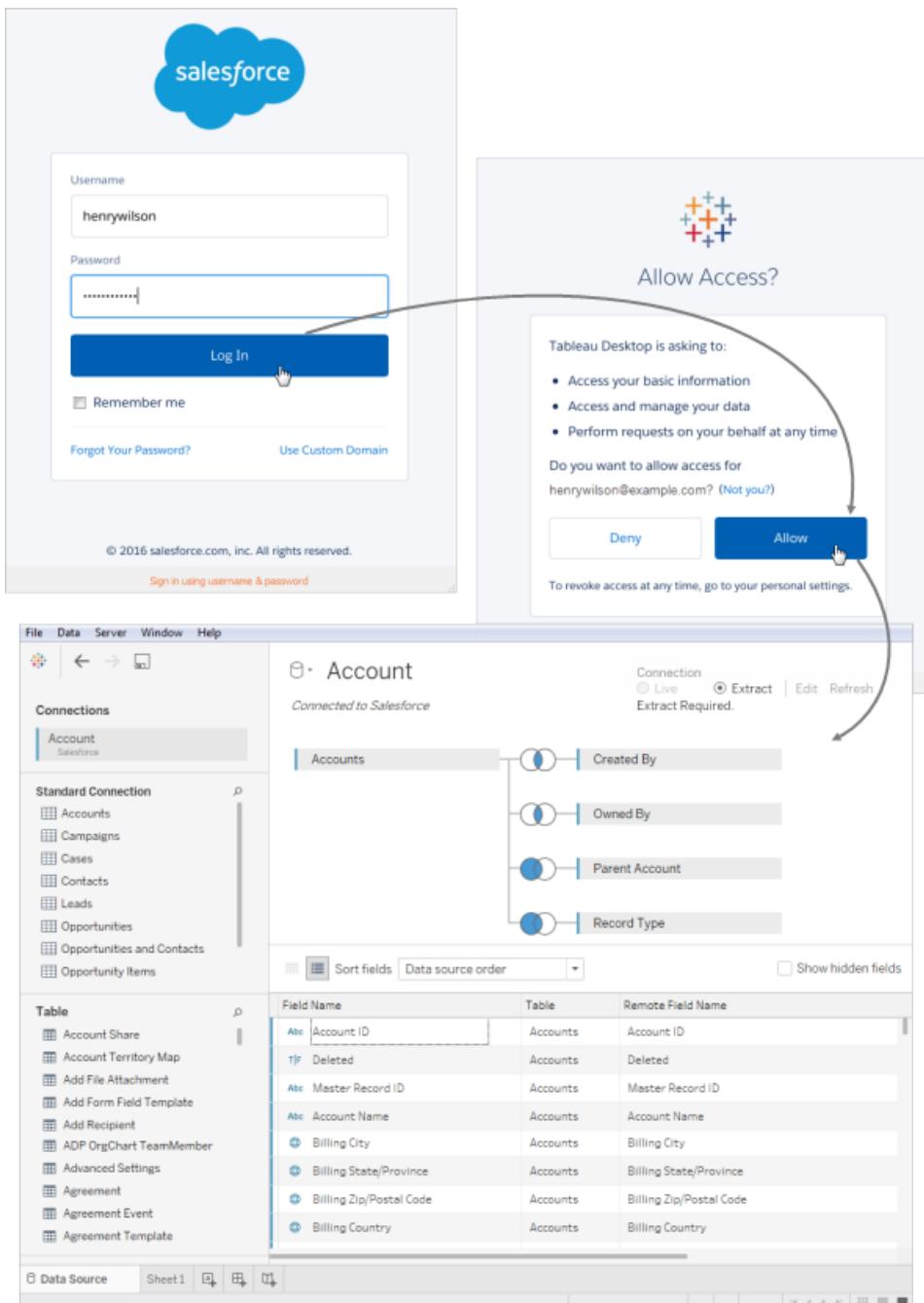
Using Joins in Salesforce

When you set up the data source on the data source page, when you drag a table to the canvas, the list of tables you can select from to join is filtered. You can create joins on any string fields and on fields that are constrained references between tables. (Only left and inner joins

are supported.) You can also add joins later. For more information, see [Join Your Data](#) on page 354.

Salesforce data source example

Here is an example of a Salesforce data source:



After you log in and click **Allow** in step 1.3 above, Tableau imports the data by creating an extract. You can update the data by refreshing the extract. For more information, see [Extract Your Data on page 440](#).

Note: The initial extract may take some time depending on the amount of data that is included. After the initial extract, you can incrementally update objects by refreshing the extract.

Updated Salesforce API

Beginning in Tableau 9.0, Salesforce extracts are created with a newer Salesforce API, version 30.0. When you attempt to do a full or incremental refresh on an extract created prior to Tableau 9.0 (which was created with an older version of the API), Tableau prompts you to either upgrade the extract to use the newer Salesforce API v, or continue to use the older Salesforce API. For more information on the benefits and considerations of upgrading, see [Upgrade to Salesforce API 30.0](#) in the Tableau Knowledge Base.

Troubleshooting Salesforce Connections

Tableau leverages the various force.com APIs to connect to Salesforce.com, Force.com, and Database.com data. These services have some restrictions on the type of data and the amount of data that you can access at a given time. If you are having trouble connecting, the following list of common restrictions may help you find a solution.

Company Account and User Profile Configuration

In order for Tableau to connect to Salesforce data, all of the following must be enabled on both your company's account and your user profile:

- SOAP API for signing in
- REST API for getting meta data
- BULK API for downloading objects
- REST API for downloading objects that the BULK API does not support
- Replication SOAP APIs for retrieving changes in the data

In order to optimize for performance and ensure the APIs are available for all of their customers, Salesforce.com balances the load by limiting the number of concurrent API requests as well as limiting the total number of API requests overall. An error occurs if these limits are reached while connected using Tableau. See the Salesforce.com developer documentation to learn more about these limits.

Some editions of Salesforce may not allow API access at all. Contact your Salesforce administrator to verify that your account has access to the above list of APIs.

Errors During Extract

When you connect to Salesforce using Tableau, the data is automatically extracted into a Tableau Data Extract file. In some cases, certain fields cannot be extracted because of character limits. Specifically, text fields that are greater than 4096 characters and calculated fields will not be included in the extract. If you have calculated fields in your data, you will need to recreate them in Tableau after creating the extract.

In addition, the Force.com API restricts queries to 10,000 total characters.

Splunk

This topic describes how to connect Tableau to Splunk data and set up a data source.

Before you begin, gather this connection information:

- Server URL, including the name of the database, if there are multiple databases
- User name and password

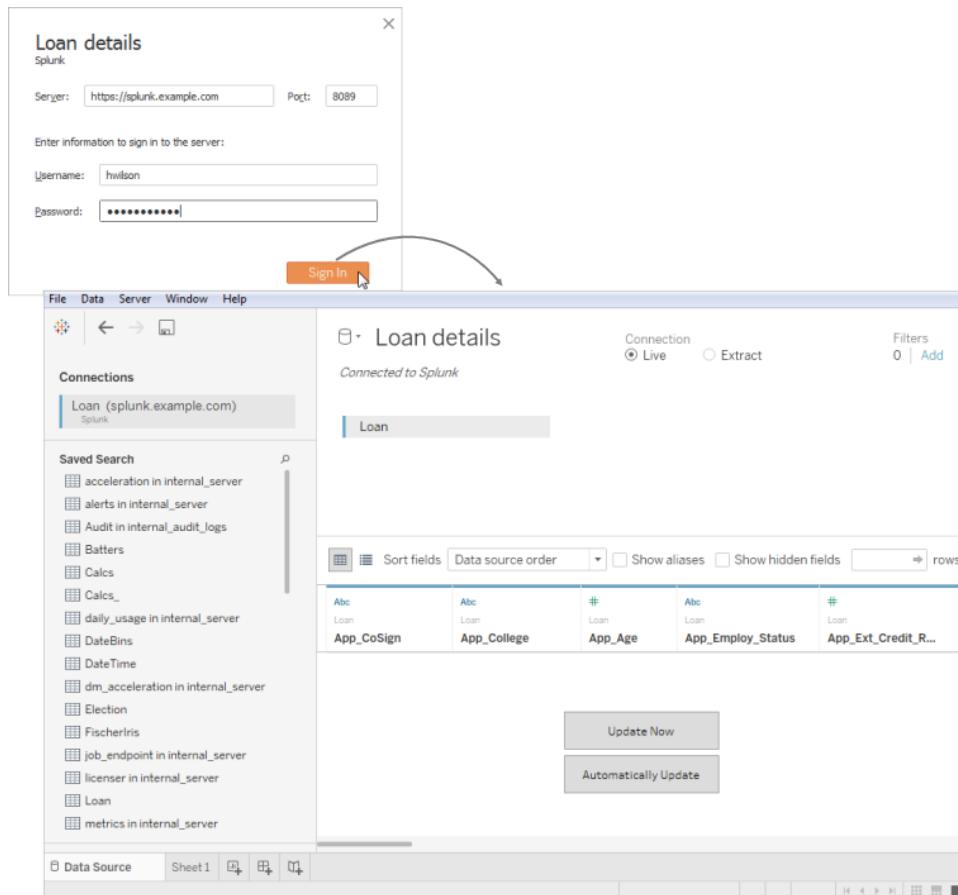
Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Splunk**, and then do the following:
 1. Enter the server URL. You should use HTTPS to connect to Splunk data.
If your server contains multiple databases, enter the name of a specific database on the server.
Note: Database names are case sensitive.
 2. Enter your user name and password, and then click **Sign In**.
If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. Under **Saved Search**, enter a saved search name in the text box or select a saved search from the list, and drag it to the box at the top of the data source page.
 3. Click the sheet tab to start your analysis.

Splunk data source example

Here is an example of a Splunk data source:



SAP HANA

This topic describes how to connect Tableau to an SAP HANA database, pass an option to the Java virtual machine (JVM), install trusted SSL certificates, and select variables and input parameters.

In this topic:

- **Before you begin** on the next page
- **Make the connection and set up the data source** on the next page
- **Sign in on a Mac** on page 1481
- **(Optional) Review and manage the data source** on page 1481
- **Pass an option to the JVM on a Mac** on page 1483

- [Install trusted SSL certificates on a Mac](#) on page 1484
 - [Select variables and input parameters](#) on page 1484
-

Before you begin

Before you begin, gather this connection information:

- Name of server that hosts the database you want to connect to
- Authentication method:
 - Windows: Windows Authentication or user name and password
 - Mac: Kerberos or user name and password
- Are you connecting to an SSL server?
- (Optional) Initial SQL statement to run every time Tableau connects

Make the connection and set up the data source

1. On the start page, under **Connect**, click **SAP HANA**, and then do the following:
 1. Enter the name of the server that hosts the database you want to connect to.
 2. Specify how you want to sign in to the server:

On Windows

Select **Use Windows Authentication** if your environment supports single sign-on (SSO). For more information, see [Quick Start: SAP HANA Single Sign-On on page 1684](#). **Note:** Tableau Desktop requires SAP HANA driver version 1.00.85 and later to support SSO for SAP HANA on Windows.

Or, select **Use a specific username and password**. If the server is password protected and you are not in a Kerberos environment, you must enter the user name and password.

On Mac

Select **Kerberos** or select **Use a specific username and password**.

3. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL](#) on page 349.
4. Click **Sign In**.

Select the **Require SSL** check box when connecting to an SSL server.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. From the **Schema** drop-down list, click the search icon or enter the schema name in the text box and click the search icon, and then select the schema.
3. In the **Table** text box, click the search icon or enter the table name and click the search icon, and then select the table.
4. Drag the table to the canvas, and then click the sheet tab to start your analysis.

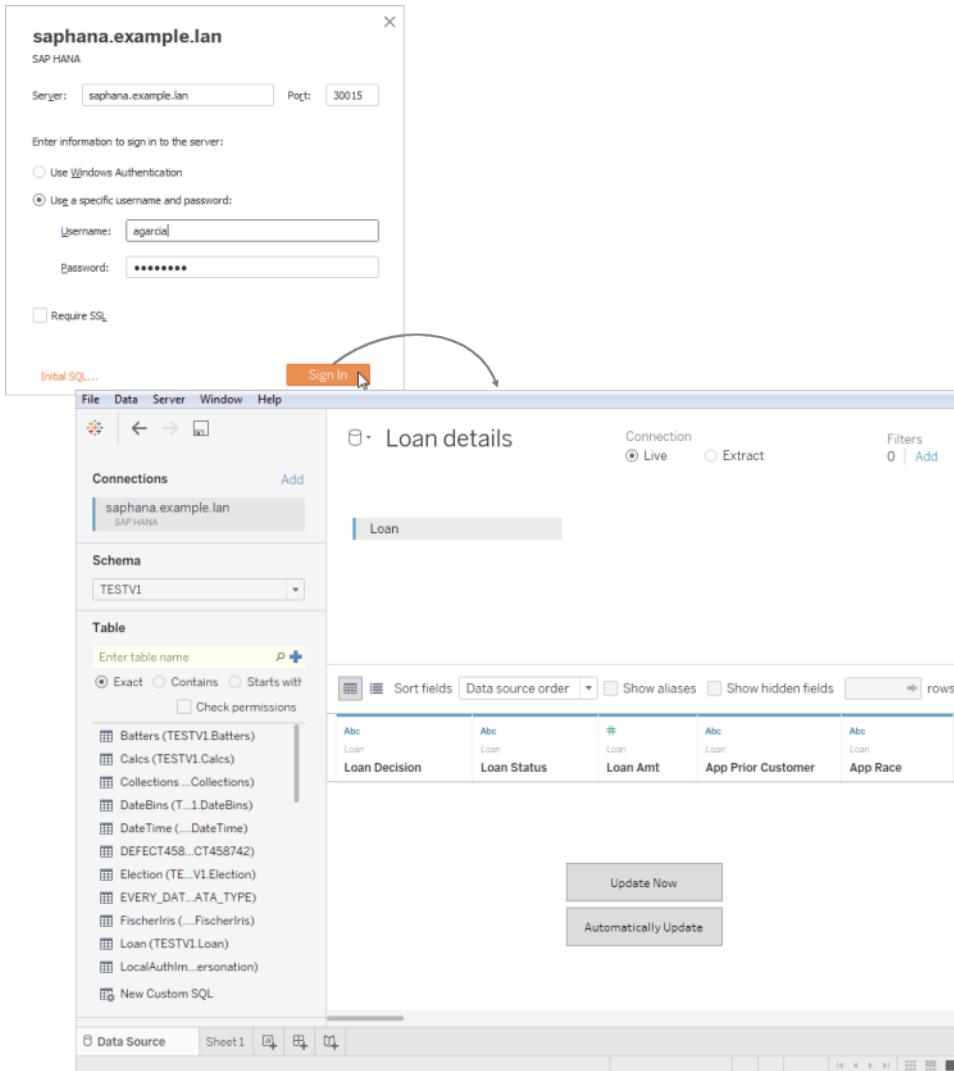
By default, column labels are displayed instead of column names.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Note: When you create a self-join with an analytic view, if one of the tables contains variables, make sure that the table with variables is on the left. Otherwise, the join may not return the expected results.

SAP HANA data source example

Here is an example of an SAP HANA data source using Tableau Desktop on a Windows computer.



Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.example.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, add the domain to the list of Search Domains for the Mac computer. To update the list of Search Domains, go to **System Preferences > Network > Advanced**; then open the DNS tab. If you do this, then you need only provide the server name when connecting.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press **Ctrl+C** (**Command-C** on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Pass an option to the JVM on a Mac

To connect Tableau Desktop on a Mac to SAP HANA, you must have Java installed on your Mac. For more information, see [SAP HANA](#) on the Driver Download page.

If you're familiar with how to pass a system property to the Java virtual machine (JVM), and you are using an Oracle JVM, you can use the environment variable `_JAVA_OPTIONS` to override the default options.

The `_JAVA_OPTIONS` variable takes a string, which is equivalent to command-line JVM arguments. The following example sets the maximum heap size to 256 MB, and specifies the path to the Kerberos configuration file:

```
export _JAVA_OPTIONS="-Xmx256m -Dsun-
.security.krb5.conf=/etc/krb5.conf"
```

Tableau starts the JVM with the following default options:

- `-Xmx512m`
Sets the maximum heap size to 512 MB
- `sun.security.jgss.native=true`
Forces the JVM to delegate Kerberos authentication to the operating system library

If the default options don't work, you can use one of the following options to configure Kerberos authentication:

```
sun.security.krb5.conf
java.security.krb5.realm
java.security.krb5.kdc5.conf
sun.security.jgss.native
```

Install trusted SSL certificates on a Mac

If you want to use SSL/TLS to encrypt communications between Tableau Desktop and your database, you may need to add certificates to the Java certificate trust store. This is required if your database uses self-signed certificates.

1. Start the Terminal application, and navigate to your Java Home directory, which is usually located at /Library/Internet Plug-Ins/JavaAppletPlugin.plugin/Contents/Home, unless you have installed the JDK or set the JAVA_HOME environment variable.
2. Run the following command. (You might need to use the `sudo` Terminal command, which is used by administrators to execute commands as a different user, such as root).

```
bin/keytool -importcert -keystore cacerts -alias <certificate name> -file <path to certificate file>
```

You can now connect Tableau Desktop to your database using SSL/TLS.

For more information on installing trusted certificates, consult the documentation for your Java Runtime Environment.

Select variables and input parameters

If the table you use includes required or optional variables or parameters, the Variables and Input Parameters dialog box opens.

Enter values for Variables and Parameters or add a new Variable

Name	Current Value
Prod_Type	Equals "Coffee"
VAR_MarketSize	
VAR_Market	
VAR_ProductType	Equals "Coffee"
VAR_ProductType	Equals "Espresso"
VAR_ProductType	Equals "Herbal Tea"
VAR_ProductType	Equals "Tea"
Add New	

Edit Variable

Select an item from the list to change its values

- Required variables and parameters display their current value or ***Required**.
- Optional variables and parameters display their current value or are blank.
- Prompt for the variable when the workbook is opened by selecting the check box under **Prompt**.

Select a variable or parameter and type or select a value for it. Repeat for all required values and any optional values that you want to include, and then click **OK**.

Note: You can use SQL query-based SAP HANA prompts.

SAP NetWeaver Business Warehouse

This topic describes how to connect Tableau to SAP NetWeaver Business Warehouse (BW) data and provides information that might help you work with your SAP BW data in Tableau.

In this topic:

- [Install SAP software](#) below
 - [Before you connect](#) below
 - [Make the connection and set up the data source](#) on the next page
 - [Select variables and input parameters](#) on page 1487
 - [Select the alias for a dimension](#) on page 1488
 - [Support for creating an extract of SAP BW data](#) on page 1489
 - [Frequently asked questions](#) on page 1490
-

Install SAP software

Before you can connect Tableau to SAP BW data, you must install SAP software on your computer. For instructions on how to install SAP software, see [Connecting to SAP BW](#) on Tableau Community.

Before you connect

Before you connect, gather this connection information:

- Connection name
- User name and password
- (Optional) Client for BW system
- (Optional) Language

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **SAP NetWeaver Business Warehouse**, and then do the following:

1. Select a connection from the drop-down list.

A connection appears in the drop-down list if a system entry exists for it. You can create a new system entry using the SAP Logon utility, or contact your database administrator.

2. Enter the user name and password to sign in to the server. Optionally, enter the **Client ID** for the BW system and the **Language**.

3. Click **Sign In**.

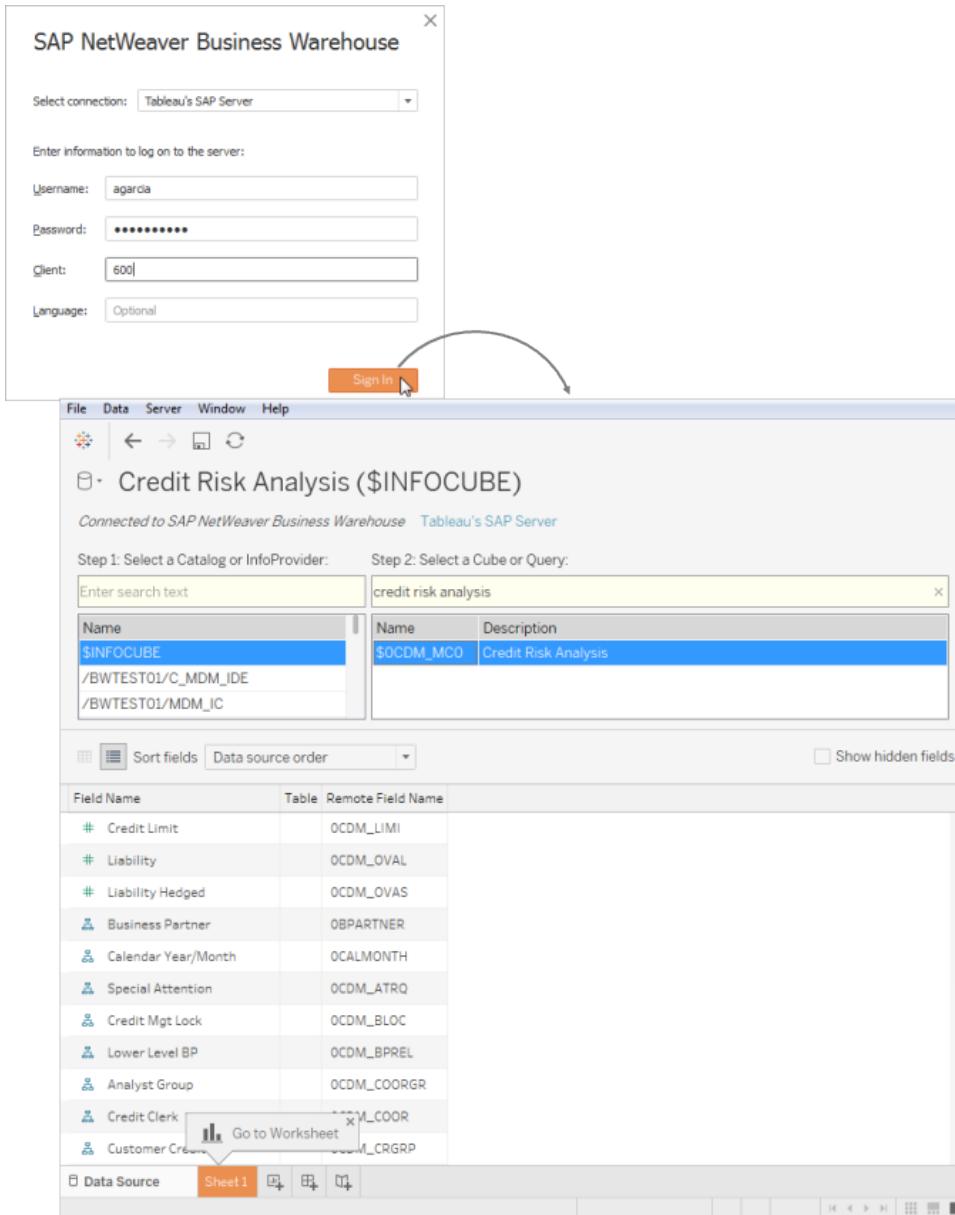
If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. Search for or select a catalog or InfoProvider.
3. Search for or select a cube or query.
4. Click the sheet tab to start your analysis.

SAP BW data source example

Here is an example of an SAP BW data source:



Select variables and input parameters

If the table you use includes required or optional variables or parameters, the Variables and Input Parameters dialog box opens.

Enter values for Variables and Parameters or add a new Variable

Name	Current Value
Prod_Type	Equals "Coffee"
VAR_MarketSize	
VAR_Market	
VAR_ProductType	Equals "Coffee"
VAR_ProductType	Equals "Espresso"
VAR_ProductType	Equals "Herbal Tea"
VAR_ProductType	Equals "Tea"
Add New	

Edit Variable

Select an item from the list to change its values

- Required variables and parameters display their current value or ***Required**.
- Optional variables and parameters display their current value or are blank.
- Prompt for the variable when the workbook is opened by selecting the check box under **Prompt**.

Select a variable or parameter and type or select a value for it. Repeat for all required values and any optional values that you want to include, and then click **OK**.

Select the alias for a dimension

When you connect to SAP NetWeaver Business Warehouse (SAP BW) databases, you have the option to select which alias to use for dimensions. SAP BW offers a number of different options including long name, short name, medium name, and key. Sometimes the alias used by default is not very useful. You can right-click the dimension and change it to another option such as **Short name**, which changes the labels to something more meaningful.

The screenshot shows the Tableau Data Source interface. On the left, the 'Dimensions' section is expanded, with 'Currency' selected. A context menu is open over 'Currency', showing options like 'Rename', 'Hide', 'Create Set...', 'Calculated Members...', 'Set Default Member...', 'Aliases...', and 'Describe...'. The 'Aliases...' option is highlighted. A secondary dropdown menu titled 'Aliases' is open, listing several alias types: 'Key', 'Key - short name', 'Key - medium name', 'Key - long name', 'Short name' (which is checked), 'Short name - key', 'Medium name', 'Medium name - key', 'Long name', and 'Long name - key'. The 'Short name' option is also highlighted.

Support for creating an extract of SAP BW data

You can create an extract of SAP BW data. To do this, you need a special license key for Tableau. You can get this free license key from the [Tableau Community SAP BW Extract](#) site. For access to the SAP BW Extract site, contact your Tableau sales representative.

For information about SAP BW extract limitations, see [SAP BW Extract Limitations](#) on Tableau Community.

Important: Creating extracts of SAP BW data is considered Beta functionality and is not supported by Tableau Support.

Frequently asked questions

Requirements

What versions do I need?

- For the client machine where Tableau Desktop or Tableau Server is installed: SAP GUI 7.30 or later for Windows.
- For the server machine: SAP BW 7.40 or later.
- Tableau Desktop or Tableau Server, any supported version. For information about which versions are supported, see [Support Versions](#) on the Tableau website.

Do I need any special licenses like OpenHub to use Tableau with SAP BW?

Tableau uses the SAP OLE DB for OLAP provider (part of the Open Analysis Interfaces) to interact with SAP BW. This provider is a mechanism to do 'reporting' and retrieving 'reasonable' amounts of data. Unlike OpenHub, the interface is not designed for moving large amounts of data. You should verify with SAP that you have licensed SAP BW correctly for your environment and that your licensing terms permit the use of the OLE DB for OLAP provider for tools like Tableau.

Does the SAP BW connector require HANA?

No. The Tableau SAP BW connector does not require HANA. The connector will work with any of the supported SAP BW backend databases such as Oracle and SQL Server. However, you may use HANA if you prefer.

Connection

Is the connection between Tableau and SAP BW a live connection or is it a Tableau extract?

Tableau supports live and extract connections to SAP BW data.

- With a live connection you will always see the latest data in your workbooks and dashboards.
- Extract connections have limitations and require a special license key (which is free) to activate the functionality. For more information, see [Support for creating an extract of SAP BW data](#) on the previous page

Does Tableau support connecting directly to the underlying database tables that support the BW instance?

No. Tableau does not support this method for connecting to the SAP BW system.

Performance

How does performance compare to existing tools we use?

System performance should be comparable to existing tools. If performance is an issue in your environment, you might want to consider adopting SAP HANA in your environment. For more

information, see [SAP HANA](#) on the SAP website.

Security

How does Tableau handle user security?

Tableau leverages all the security policies already implemented in the SAP BW server. There is no need for you to duplicate this in Tableau. All user authentication and authorization is performed by SAP BW. Tableau passes your credentials to the SAP BW system to be authenticated. Once authenticated, you will see only the objects and data that your profile or role on the SAP BW system has been enabled to access.

How does user security work with Tableau Server when I publish a dashboard?

By default, worksheets and dashboards connected to SAP BW data require you to log into SAP BW with your SAP user name and password. However, Tableau Server can be configured to embed credentials so that you are taken directly to the published views and dashboards. In addition, Tableau Server offers an additional layer of security if desired. Tableau Server administrators can configure user authentication and authorization, independent and in addition to the SAP BW authentication and authorization. For more information about Tableau Server security, refer to the [Security](#) section of the Tableau Server Online Help.

SAP Sybase ASE

This topic describes how to connect Tableau to an SAP Sybase ASE database and set up the data source. Tableau connects to Sybase ASE version 15.2 and later.

Before you begin, gather this connection information:

- Name of the server you want to connect to and port number
- User name and password
- (Optional) Initial SQL statement to run every time Tableau connects

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **SAP Sybase ASE**, and then do the following:
 1. Enter the name of the server you want to connect to, and specify the port to use.
 2. Enter the user name and password.
 3. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL on page 349](#).
 4. Click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. From the **Database** drop-down list, select a database or use the text box to search for a database by name.
3. From the **Schema** drop-down list, select a schema or use the text box to search for a schema by name.
4. Under **Table**, select a table or use the text box to search for a table by name.

You can also specify a stored procedure in the database. To access stored procedures, the database must be appropriately configured with a linked server, which could be a remote server or a pointer to the database itself (known as *loopback*). For more information, see [Use a Stored Procedure on page 414](#).

5. Drag a table or stored procedure to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

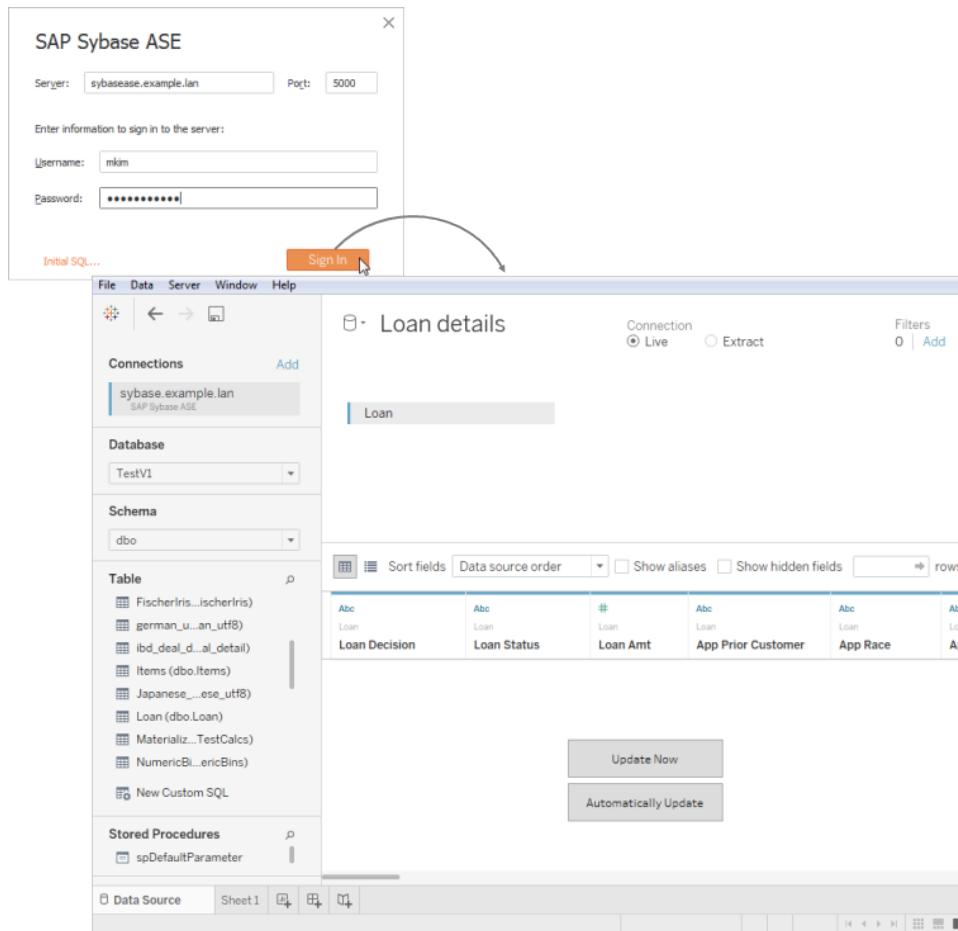
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

SAP Sybase ASE data source example

Here is an example of an SAP Sybase ASE data source:



SAP Sybase IQ

This topic describes how to connect Tableau to a SAP Sybase IQ database and set up the data source.

Before you begin, gather this connection information:

- Name of the host that hosts the database you want to connect to
- Server name
- Authentication method: Windows Authentication or user name and password

Use this connector with Tableau Desktop on a Windows computer.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **SAP Sybase IQ**, and then do the following:
 1. Enter the host name that hosts the database that you want to connect to.
 2. Enter the server name.
 3. Select how you want to sign in to the server. Specify whether to use Windows Authentication or a specific user name and password. If the server is password protected, you must enter the user name and password.
 4. Click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Database** drop-down list, select a database or use the text box to search for a database by name.
 3. Under **Table**, select a table or use the text box to search for a table by name.
 4. Drag a table to the top area of the data source page, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update**

Now again to see your changes.

- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields](#) on page 429.

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

SAP Sybase IQ data source example

Here is an example of an SAP Sybase IQ data source:

The screenshot shows the Tableau Data Source setup for an SAP Sybase IQ connection named 'sybaseiq'. The connection details are as follows:

- Host: sybaseiq
- Port: 2640
- Server: TestV1

For sign-in information:

- Use Windows Authentication
- Use a specific username and password:
Username: hwilson
Password: [REDACTED]

A 'Sign In' button is highlighted with a red arrow pointing from the text above.

The main interface shows the 'Loan details' sheet. The connection is set to 'Live'. The data preview shows a table with columns: Loan Decision, Loan Status, #, Loan Amt, App Prior Customer, App Race, and App. Buttons for 'Update Now' and 'Automatically Update' are visible.

The sidebar on the left lists the connection, server, database, and tables available in the 'test' database.

Snowflake

This topic describes how to connect Tableau to a Snowflake data warehouse and set up the data source.

Before you begin, gather this connection information:

- Name of the server name that you want to connect to
- User name and password
- (Optional) Initial SQL statement to run every time Tableau connects

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Snowflake**, and then do the following:
 1. Enter the name of the server that you want to connect to.
 2. Enter your user name and password, and then click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
 3. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run Initial SQL on page 349](#).
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Warehouse** drop-down list, select the warehouse or use the text box to search for a warehouse by name.
 3. From the **Database** drop-down list, select a database or use the text box to search for a database by name.
 4. From the **Schema** drop-down list, select a schema or use the text box to search for a schema by name.
 5. Under **Table**, select a table or use the text box to search for a table by name.
 6. Drag a table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query on page 404](#).

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data on page 354](#).
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data on page 369](#).

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press **Ctrl+C** (**Command-C** on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

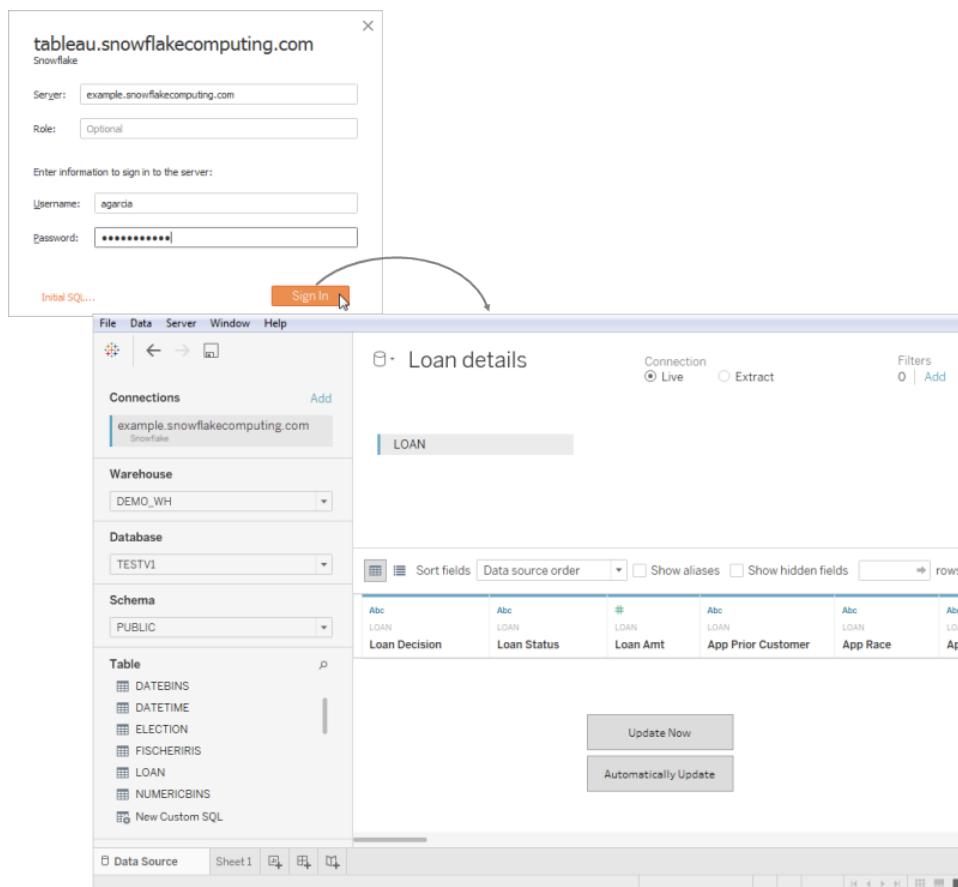
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Snowflake data source example

Here is an example of a Snowflake data source using Tableau Desktop on a Windows computer.



Spark SQL

This topic describes how to connect Tableau to a Spark SQL database and set up the data source. Tableau can connect to Spark version 1.2.1 and later.

Note: On Windows, you can use the Spark SQL connector to connect to a Spark cluster on Azure HDInsight. To do this, install the [Spark on Azure HDInsight drivers](#).

Before you begin, gather this connection information:

- Name of the server that hosts the database you want to connect to and port number
- Authentication method:
 - No Authentication
 - User Name
 - User Name and Password
 - User Name and Password (SSL)
 - Microsoft Azure HDInsight Emulator
 - Microsoft Azure HDInsight Service
 - HTTP
 - HTTPS
- Sign-in credentials. Your choices depend on the authentication method you choose, and can include the following:
 - User name
 - Password
 - HTTP Path
- (Optional) Initial SQL statement to run every time Tableau connects

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Spark SQL**, and then do the following:
 1. Enter the name of the server that hosts the database and the port number to use.
 2. Connect to the database using SparkThriftServer. Note that the legacy SharkServer and SharkServer2 connections are provided for your use, but are not supported by Tableau.
 3. (Optional) Click **Initial SQL** to specify a SQL command to run at the beginning of every connection, such as when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. For more information, see [Run](#)

[Initial SQL](#) on page 349.

4. Click **Sign In**.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. From the **Schema** drop-down list, click the search icon or enter the schema name in the text box and click the search icon, and then select the schema.
3. In the **Table** text box, click the search icon or enter the table name and click the search icon, drag the table to the canvas, and then click the sheet tab to start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query](#) on page 404.

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data](#) on page 354.
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data](#) on page 369.

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update**

Now again to see your changes.

- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields](#) on page 429.

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

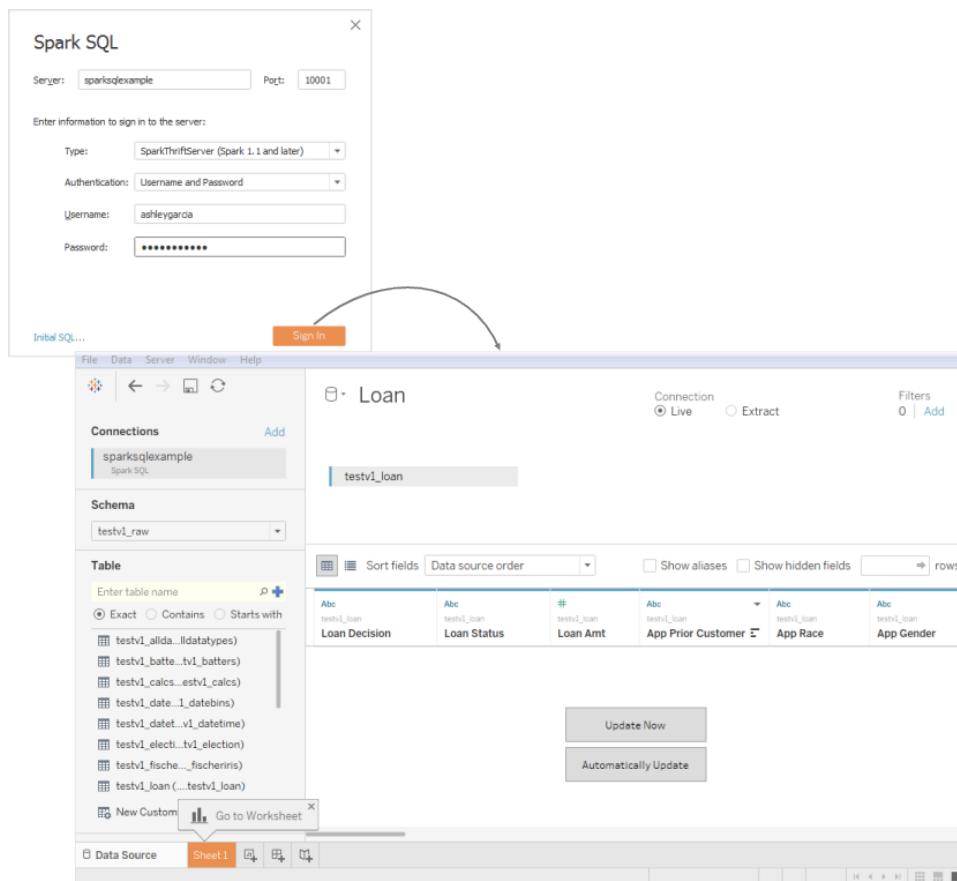
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Spark SQL data source example

Here is an example of a Spark SQL data source using Tableau Desktop on a Windows computer.



Teradata

This topic describes how to connect Tableau to a Teradata database or a Teradata Unity server and set up the data source.

Before you begin, gather this connection information:

- For Teradata database: Name of the server you want to connect to
- For Teradata Unity server: URL of the server you want to connect to
- Authentication method: Teradata Database, LDAP, or Integrated Authentication

- Depending on the environment, the user name and password
- Do you need an encrypted connection?

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Teradata**, and then do the following:
 1. Enter the name of the server that hosts the database you want to connect to.
Note: You can connect to a Teradata Unity server by entering the appropriate URL in the **Server** text box.
 2. Select how you want to sign in to the server. Specify whether to use the built-in **Teradata Database** authentication, **LDAP**, or **Integrated Authentication**.
If the server is password protected, and you are not in a Kerberos environment, you must select **Teradata Database** or **LDAP** and enter the user name and password.
 3. Select the **Require Encryption** check box if you need an encrypted connection.
 4. (Optional) Click **Query Banding and Initial SQL**.
Query banding enables you to pass parameters into the Teradata environment. For more information, see [Advanced Teradata Options on page 1508](#). Additionally, you can specify a SQL command that will be run once upon connection. For more information, see [Run Initial SQL on page 349](#).
 5. Click **Sign In**.
If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.
2. On the data source page, do the following:
 1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
 2. From the **Database** drop-down list, click the search icon or enter the schema name in the text box and click the search icon, and then select the database.
 3. In the **Table** text box, click the search icon or enter the table name and click the search icon, and then select the table.
You can also specify a stored procedure in the database. For more information and for a list of constraints specific to Teradata databases, see [Use a Stored Procedure on page 414](#).
 4. Drag the table or stored procedure to the canvas, and then click the sheet tab to

start your analysis.

Use custom SQL to connect to a specific query rather than the entire data source. For more information, see [Connect to a Custom SQL Query](#) on page 404.

Sign in on a Mac

If you use Tableau Desktop on a Mac, when you enter the server name to connect, use a fully qualified domain name, such as mydb.test.ourdomain.lan, instead of a relative domain name, such as mydb or mydb.test.

Alternatively, you can add the domain to the list of Search Domains for the Mac computer so that when you connect, you need to provide only the server name. To update the list of Search Domains, go to **System Preferences > Network > Advanced**, and then open the **DNS** tab.

(Optional) Review and manage the data source

You can make the following changes before you start your analysis:

Get more data from different databases

- Click **Add** next to **Connections**. For more information, see [Join Your Data](#) on page 354.
- If a connection you want is not listed, select **Data > New Data Source** to add a new data source. For more information, see [Blend Your Data](#) on page 369.

View the data in the Tableau data source

- In the grid, click **Update Now** to preview the first 1,000 rows of your data.
- If you add tables, remove tables, or make changes to the join conditions, click **Update Now** again to see your changes.
- To have changes automatically reflect in the grid, click **Automatically Update**.

Manage metadata

Click the metadata grid button () to perform routine management tasks, such as hiding or renaming fields.

Sort fields and rows

- Select how you want to sort the columns in the grid or metadata grid from the **Sort fields** drop-down list. You can sort the columns by data source or table order.
- In the grid view, sort the row values by clicking the sort button next to the column name.

Hide a field

Click the column drop-down arrow and select **Hide**.

Rename a field

- Double-click the field name.
- To revert back to the original name of the field, select a column or multiple columns, click the column drop-down arrow, and then select **Reset Name**.

Split columns

If your data supports split, click the column drop-down arrow, and then select **Split**. In some cases, you need to check for **Split** on the **Data** pane in the worksheet. For more information, see [Split a Field into Multiple Fields on page 429](#).

Create a new calculation based on an existing field in the data source

Click the column drop-down arrow and select **Create calculated field**.

Copy values

Select the values in the grid and then press Ctrl+C (Command-C on a Mac).

Copy values in the metadata grid

Select the values, right-click, and then select **Copy**.

Connect to live data or use an extract

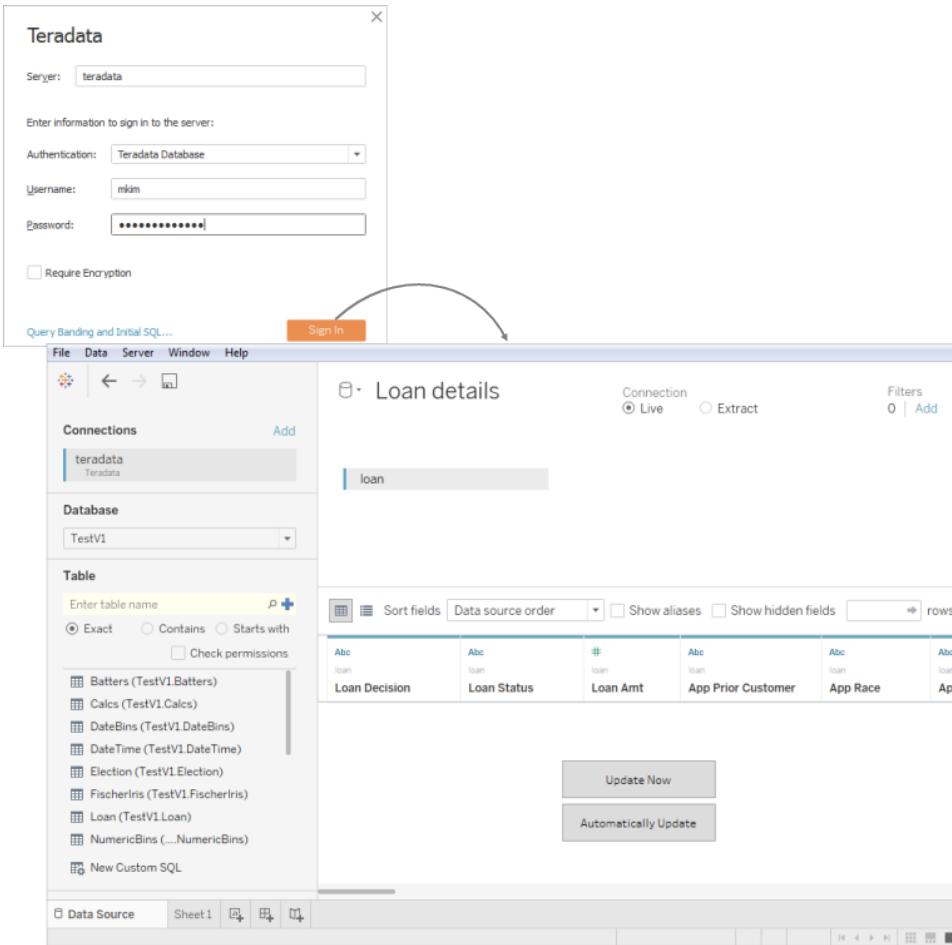
- If your data supports it, select **Live** or **Extract** under **Connection** at the top of the data source page.
- If you choose to create an extract, click the **Edit** link that appears to set up filters that define a subset of the data to include in the extract.

Add data source filters to restrict the visibility and use of the fields in the data source

Click **Add** at the top of the data source page to add data source filters .

Teradata data source example

Here is an example of a Teradata data source using Tableau Desktop on a Windows computer.



Advanced Teradata Options

When connecting to Teradata databases you can optionally set up query bands and initial sql. These advanced options are used to increase performance and take advantage of the built in security rules of the database.

Query Bands

When connecting to a Teradata database, you can optionally define query band statements that run during connection. Query banding allows you to pass parameters into the Teradata environment. Use these to set up a workbook to filter the data based on security rules that exist in the database. For example, you can pass in the Tableau Server user name for the current user so when the view is loaded it only shows the data specific to that user. Query bands can also be used to improve performance. When connecting to Teradata, you can define a map between the name of the attributes passed into the query band and the corresponding values from Tableau.

To set up query banding:

1. On the data source page, select **Data > Query Banding and Initial SQL**.
2. In the subsequent dialog box, specify name/value pairs in the top text box labeled Query Banding. You can use the **Insert** drop-down menu to add Tableau values. The Tableau values are described in the table below.

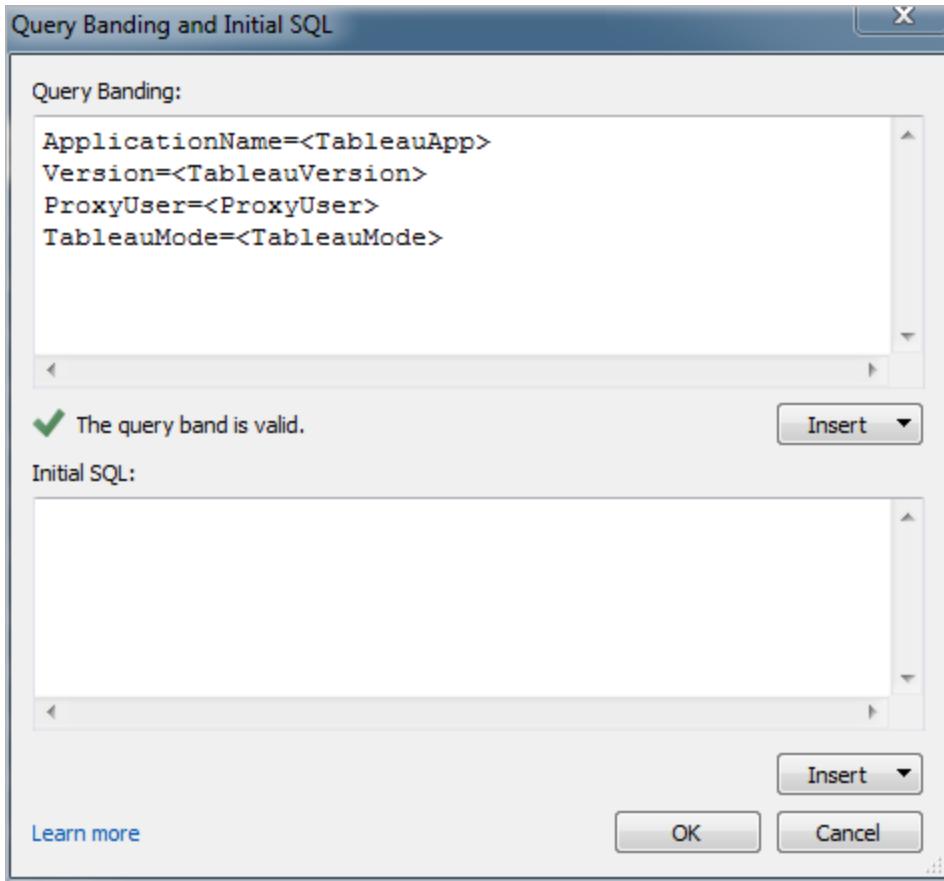
Value	Description	Example
<TableauMode>	The mode Tableau is operating in when generating queries. This value will either be “Connect” when retrieving metadata or “Analytical” when retrieving actual data.	Connect or Analytical
<LoginUser>	The user name of the person signed in to the database.	jsmith
<ServerUser>	<p>The signed-in Server user. Does not include domain name.</p> <p>Note: <code>ProxyUser</code> returns the same value as <code>ServerUser</code>. However, <code>ProxyUser</code> sets up impersonation and stores the Tableau Server user in the query band parameter. If you are using query banding for security purposes, you should use <code>ProxyUser</code> instead. This will ensure that query results are not shared between different users. <code>ServerUser</code> should only be used for auditing purposes.</p>	jsmith
<ServerUserFull>	<p>The signed-in server user including the domain name (if the server is using Active Directory).</p> <p>Note: <code>ProxyUserFull</code> returns the same value as <code>ServerUserFull</code>. However,</p>	domain.lan\jsmith

Value	Description	Example
	<p>ProxyUserFull sets up impersonation and stores the Tableau Server user in the query band parameter. If you are using query banding for security purposes, you should use ProxyUserFull instead. This will ensure that query results are not shared between different users.</p> <p>ServerUserFull should only be used for auditing purposes.</p>	
<ProxyUser>	Used when setting up impersonation on the server. Provides the username of the current server user.	jsmith
<ProxyUserFull>	Used when setting up impersonation on the server. Provides the username and domain name of the current server user.	domain.lan\jsmith
<TableauApp>	The name of the Tableau application.	Tableau Desktop Professional or Tableau Server
<TableauVersion>	The version of the Tableau application	6100.11.0428.0300
<WorkbookName>	The name of the workbook.	Financial-Analysis

An example query band statement is shown below. This example passes the user name for the current server user.

```
ApplicationName=<TableauApp>Version=<TableauVersion>ProxyUser=<ProxyUser>TableauMode=<TableauMode>
```

Tableau checks the statement for errors as you type. When it is valid a green check mark displays at the bottom of the text box.



Initial SQL

When connecting to Teradata databases, you can optionally specify a SQL command that will be run once upon connection. See [Run Initial SQL on page 349](#) to learn more about adding these commands to your connection.

Note: If Tableau detects that the Initial SQL payload cannot be delivered in one query, it attempts to split the query into separate statements, where each statement must end with a semicolon followed by a newline character, with no additional characters (such as an extra space) in between. This requirement is important because if a semicolon exists within the interior of a statement, for example within a string literal, it may not be safe to split the statement at that point. If the Initial SQL code that you specify in Tableau returns an error, and the same Initial SQL is deemed valid by the Teradata databases, it may be that Tableau has split the query incorrectly. If this happens, try reformatting the SQL code in Tableau manually.

Teradata OLAP Connector

This topic describes how to establish a Teradata OLAP connection.

Before you begin, gather this connection information:

- Connection name
- User name and password

Use this connector with Tableau Desktop on a Windows computer.

When you connect to Teradata OLAP data, you select from a list of connections, each representing a system data source name (DSN). A connection appears in the drop-down list if a system DSN exists for that connection. You can use the Windows ODBC Data Source Administrator utility to create a system DSN, or contact your database administrator.

Make the connection and set up the data source

1. On the start page, under **Connect**, click **Teradata OLAP Connector**, and then do the following:

1. Select a connection from the drop-down list.
2. Enter your user name and password to log on to the server.

If Tableau can't make the connection, verify that your credentials are correct. If you still can't connect, your computer is having trouble locating the server. Contact your network administrator or database administrator.

3. Click **Sign In**.

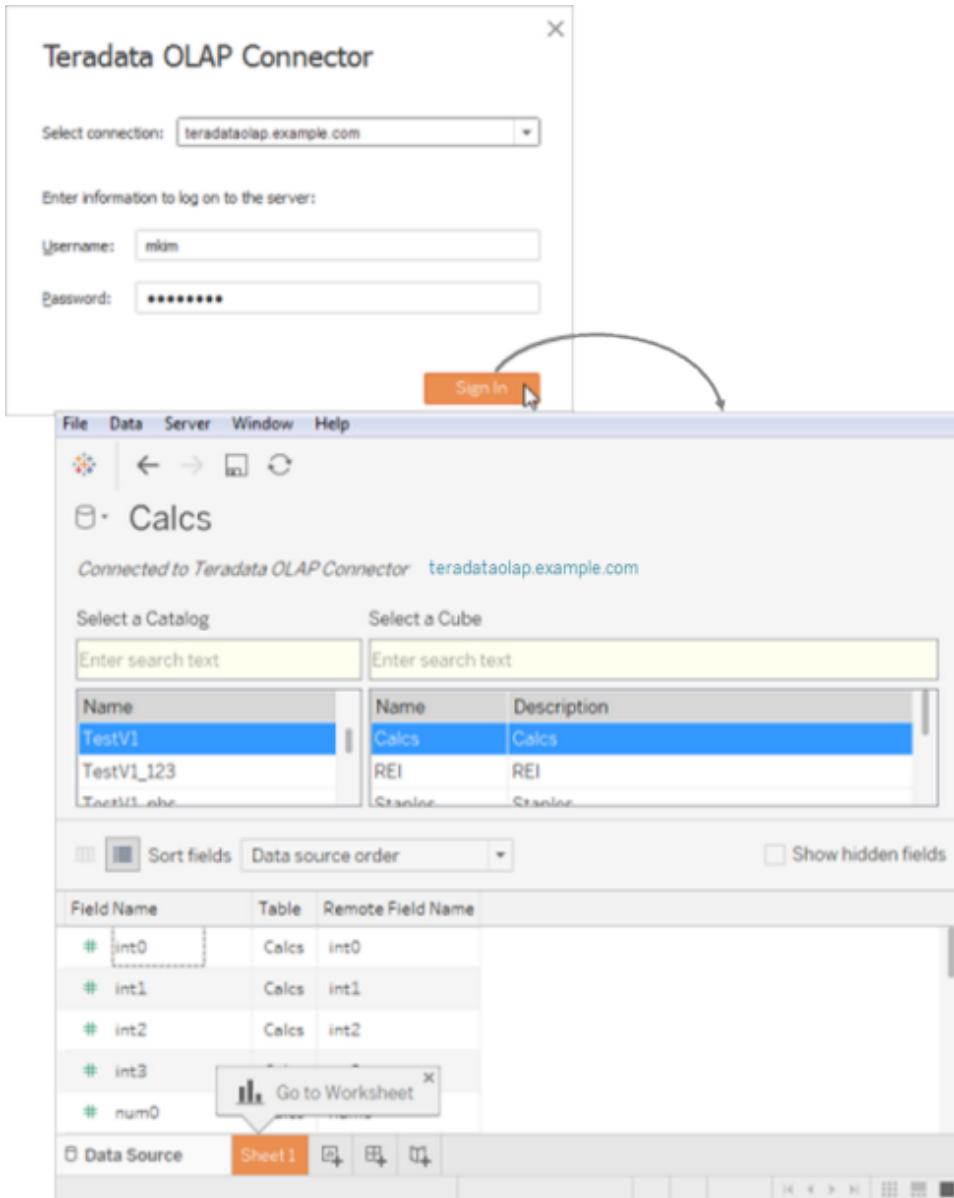
2. On the data source page, do the following:

1. (Optional) Click the default data source name at the top of the page, and then enter a unique data source name for use in Tableau. For example, use a data source naming convention that helps other users of the data source figure out which data source to connect to.
2. Search for or select a catalog.
3. Search for or select a cube from the catalog.
4. Click the sheet tab to start your analysis.

Named sets from a Teradata OLAP data source are displayed in the **Sets** area of the **Data** pane in Tableau. You can interact with these named sets in the same way you interact with other custom sets in Tableau. For more information, see [Sets on page 512](#). You can view underlying data for Teradata OLAP data sources, provided the database administrator has enabled this functionality. For more information, see [View Data on page 573](#).

Teradata OLAP connection example

Here is an example of a Teradata OLAP connection:



Web Data Connector

This topic describes web data connectors, what to do before you use a connector, and how to connect Tableau to a web data connector.

You can use a web data connector to connect to data that is accessible over HTTP and that doesn't already have a connector. A web data connector is an HTML file that includes JavaScript code. You can create your own web data connector or use one that has been

created by someone else. The web data connector must be hosted on a web server running locally on your computer, on a third-party web server, or on Tableau Server.

Test and vet the web data connector

You should use a web data connector that you trust. If you are unfamiliar with the web data connector, you or your Tableau Server administrator should test and vet the web data connector before you use it. For more information, see [Testing and Vetting Web Data Connectors](#) in the Tableau Server documentation.

When to use an imported web data connector

When you use a web data connector, Tableau creates an extract of the data that the connector accesses. You can refresh the extract in Tableau Desktop.

When you publish the data source or workbook to Tableau Server, you cannot refresh the extract on Tableau Server unless the web data connector was imported to Tableau Server before you used it. For more information, see [Web Data Connectors in Tableau Server](#) in the Tableau Server Help.

The import process results in a URL where the web data connector is available on Tableau Server. The server administrator should give this URL to you and to anyone else who wants to use the connector in a workbook.

If you open a workbook on Tableau Server that was created using a web data connector, but the connector has not been imported to Tableau Server, and you want to be able to refresh the extract on Tableau Server, follow the process for testing, vetting, and importing the connector. Then, in your workbook, edit the connection and change the URL of the connector to the URL provided by Tableau Server. You can then refresh your data on Tableau Server.

When you publish to Tableau Online, as a security measure, Tableau Online can't connect to or refresh an extract created by a web data connector. To refresh a web data connector extract, you use the sync client. For more information, see [Schedule Refreshes Using the Sync Client](#) in the Tableau Online Help.

Connect to the data source

1. On the start page, under **Connect**, click **Web Data Connector**.
2. Enter the URL of the web data connector that you want to connect to, and then press Enter.

Note: If you enter a URL to a website, rather than to a web data connector, an error message will appear.



3. If the connector displays a webpage, enter any information that you're prompted for, and then submit the page.
4. Wait while the connector retrieves your data and imports it into Tableau as an extract.
5. Click the sheet tab to start your analysis.

Optional settings

After you connect, you can make the following configuration changes to the data source before you start your analysis.

- **Sort fields and rows** – From the **Sort fields** drop-down list, select how you want to sort the columns in the grid or metadata grid. Sort the row values by clicking the sort button next to the column name.
- **Rename or hide columns** – Click the column header drop-down arrow and select the option you want.
- **Split columns** – Depending on how the data is structured, you can split the columns in your data into new fields. For more information, see [Split a Field into Multiple Fields on page 429](#).
- **Create calculations** – Create a new calculation based on an existing field in the data source.
- **Copy values** – Copy values in the grid by selecting the values and then pressing Ctrl+C. Alternatively, to copy values in the metadata grid, select the values, right-click, and then select **Copy**.
- **Add new or edit other data sources** – Click the drop-down arrow next to the data source.

Web data connector data source example

An example of a web data connector data source is shown below. The first time you connect to a web data connector, the connection page outlines how to connect to the web data connector.

After you connect to a web data connector, on the connection page, you can do the following:

- View a history of all the web data connectors that you've connected to, or clear the history, in the drop-down list in the address bar.
- View your five most recently used connectors under Recent Connectors.
- Click links at the bottom on the window to learn more about web data connectors:

- **What's this?** links to an overview of this feature.
- **Use a connector** links to this topic.
- **Build a connector** links to the **Web Data Connector SDK**, which provides tools and documentation for building a web data connector.

The screenshot illustrates the Tableau Web Data Connector interface. At the top, there is a search bar labeled "Enter your web data connector URL here". Below it, a list of "Recent Connectors" includes entries such as "Weather Web Data Connector", "USGS Earthquake Feed", "Google Spreadsheet Web Data Connector", "Facebook Page Feed Web Data Connector", and "Facebook Page Feed Web Data Connector". At the bottom of this section are three links: "What's this? >", "Use a connector >", and "Build a connector >".

A large arrow points from the "Use a connector >" link down to a second window titled "http://weatherserver.example.com/weathersearch/weatherwebconnect.html". This window displays a "Weather Web Data Connector" dashboard featuring a map of the world with various cities marked. A callout bubble highlights the "Get Weather Forecast" button. Below the map, a section titled "5 Day Forecast for Top 300 Cities (by population)" is visible. The main content area shows a "Weather Forecast Data" table with columns: COUNTRY, CITY ID, CITY, DAYTEMP, MINTEMP, MAXTEMP, and NIGHTT. Buttons for "Update Now" and "Automatically Update" are present at the bottom of the table view.

Other Databases (ODBC)

You can see the file and database types that are supported by Tableau Desktop listed on the start page, under **Connect**. For supported files and databases, Tableau provides built-in connectors that are built for and optimized for those data sources.

If your file or database type is not listed on the start page, you might be able to connect to it using ODBC. ODBC (Open Database Connectivity) is a standard way to connect to a database. If the database you want to connect to implements the ODBC standard, you can connect Tableau to your data using the ODBC driver for your database and the Tableau ODBC connector.

Note: This data source option is supported for Windows computers only.

To connect to your data using the ODBC driver, do the following:

1. On the start page, under **Connect**, click **Other Databases (ODBC)**.
2. Enter the information required to make the connection.

If table or schema names are not listed, click the search icon or enter the name in the text box and click the search icon, and then select the name.

Important: When you use an ODBC driver to connect to a database, the outcome may vary and compatibility with Tableau Desktop features is not guaranteed.

If your connection works as you need it to, you can begin your analysis. If, however, you need to customize your ODBC connection to improve functionality and performance, refer to the following articles.

See also

[Tableau and ODBC on the next page](#) – Provides background information about ODBC, describes how Tableau determines the functionality of an ODBC driver, and lists frequently asked questions.

[Customize and Tune ODBC Connections on page 1522](#) – Describes how to fine tune the ODBC connection information for improved functionality and performance.

[Tableau Capability Customizations Reference on page 1530](#) – Lists customizations you can use to define which Tableau capabilities are supported by the data source.

[ODBC/SQL Customizations Reference on page 1543](#) – Lists customizations that represent the parts of the ODBC and SQL standards that the ODBC driver reports supporting.

Tableau and ODBC

Tableau includes built-in connectors for many databases. These connectors take advantage of capabilities and optimizations specific to these data sources, and they're developed and tested to ensure they are robust and provide good performance.

But you might be working with a database that Tableau doesn't have a connection for. In that case, you still might be able to connect Tableau to the database. Tableau has a connector that uses the ODBC standard. By using ODBC, you can access any data source that supports the SQL standard and implements the ODBC API.

In this article

ODBC basics below

[How Tableau determines the capabilities of the ODBC driver](#) on the next page

[Tune ODBC connector performance](#) on page 1520

[Tableau support for ODBC connections](#) on page 1520

[ODBC FAQ](#) on page 1521

ODBC basics

ODBC (Open Database Connectivity) is an industry standard that enables a wide variety of software to access data. The basis of ODBC is a standard syntax for SQL queries that is used by software applications to connect to and request data from databases. The ODBC driver, usually provided by the database vendor, accepts requests in this standard syntax and converts the request to the native format that the target database prefers. In effect, the ODBC driver is a translation layer to convert from a general-purpose request to the database-specific request.

For example, the ODBC specification details that dates should be specified in SQL statements by using the syntax: {d 'yyyy-mm-dd'}. This format is then translated by the driver to the proper date syntax of the actual database. If you use the ODBC connector to connect to an Oracle database, the ODBC connector sends a request to the Oracle ODBC driver in this format:

```
select name from emp where birthdate > {d '1987-12-29'}
```

The driver converts your request to the actual format the Oracle database requires, which is this:

```
select name from emp where birthdate > '29-DEC-87'
```

The ODBC specification includes syntax for making function requests, join syntax, data types, and data type conversions. The SQL language itself supports complex concepts such as nested queries, correlated subqueries, temporary tables, and a variety of functions that can be used in select, where, group by, join clauses, and more. The ODBC driver is responsible for the proper conversion of all these requests to the syntax of the target database.

How Tableau determines the capabilities of the ODBC driver

There are differences in how each database vendor implements capabilities of the ODBC standard. Tableau uses a feature-discovery API in ODBC to question a database driver about the functionality it supports. One example of how Tableau changes behavior based on what the driver reports is the list of functions available when creating a calculated field. Fewer driver limitations means that more functions are available.

In some cases Tableau will require you to create a data extract from the data returned by the ODBC connector. And there are some ODBC drivers and databases that Tableau can't connect to.

Note: The ODBC connector is supported for Windows computers only.

ODBC discovery

During the initial connection, Tableau performs requests to the driver and SQL discovery queries in order to determine the capabilities of the driver.

Some examples of the capabilities discovery that Tableau performs include:

- What scalar and aggregate functions are available.
- What date and time manipulation functions are available.
- Can temporary tables be created and can if select into statements can be used.
- Are subqueries supported.
- Are top and limit queries supported.
- What join styles are supported (outer, inner, full).
- What data types are supported.

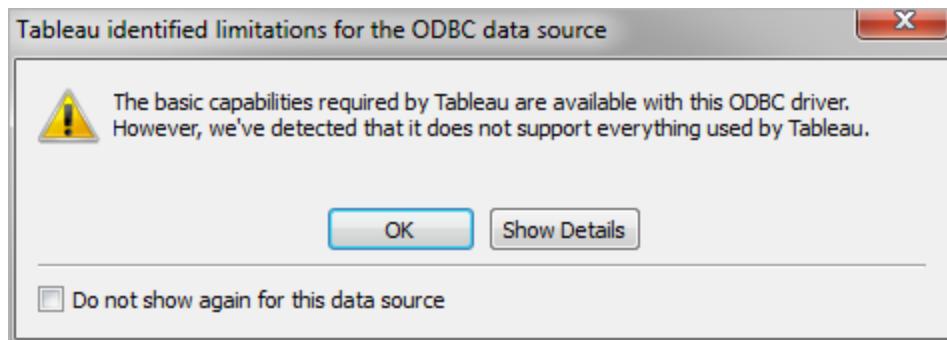
Based on the outcome of the capabilities discovery, Tableau will classify the current connection as being in one of four categories:

- **Fully Functional.** This is a driver that supports all the functions and capabilities that Tableau uses.
- **Minor limitations.** This is a driver that has a small number of non-critical limitations. An example would be a driver that doesn't support the full set of numeric, string, or date functions that Tableau normally makes available. Tableau will modify its behavior to account for these limitations and proceed.
- **Major limitations.** This is a driver that has a large number of limitations or critical capabilities that Tableau relies upon but that are not supported. However, even with these major restrictions, Tableau might be able to connect well enough to create a Tableau extract to retrieve the data into Tableau for further work offline from the database. In this case, Tableau will recommend that you create an extract. When you

create the extract, you might not be able to create a filter on the data source to reduce the amount of data in the extract. (For more information, see [Filter Data from Data Sources](#) on page 434.) After the data has been extracted, full Tableau functionality is available when working with the extract.

- **Fatal limitations.** This is a driver that does not support even the most minimal set of capabilities that Tableau needs in order to connect and run the basic queries to create extract files. Therefore, Tableau is unable to proceed any further with this driver.

After the connection is established, if Tableau has determined that the capabilities available classify this connection as anything less than fully functional, a message is displayed to inform you of the limitations that have been detected. For example, a connection to an ODBC driver with minor limitations displays this message:



For more critical limitations, the warning dialog box will advise you to create a Tableau extract in order to proceed.

[Tune ODBC connector performance](#)

Tableau supports the ability to customize your ODBC data connection, which can improve the connection experience. For more information, see [Customize and Tune ODBC Connections](#) on page 1522.

[Tableau support for ODBC connections](#)

Tableau provides no guarantee or warranty that using Tableau with any particular ODBC driver or database will be able to successfully connect and query data. Some ODBC drivers will support the full Tableau interactive experience, while others may work only for creating extracts. Some ODBC drivers might not work with Tableau.

Note: Tableau will provide reasonable levels of customer support to assist in troubleshooting connections with ODBC drivers, but can't create or customize a connector to work with a specific ODBC driver.

ODBC FAQ

What's a typical use case for the ODBC connector?

The most common use case for an ODBC connection is to access data specifically to retrieve it into a Tableau extract. Bringing this data into a Tableau extract then allows the full Tableau capabilities to work with the data. Many ODBC drivers support the needed functionality to connect and perform the simpler queries required to perform an extract. Whether you extract all of the data or just select a few columns and set some filters to retrieve a relevant subset, this use of ODBC is a good scenario to keep in mind.

Where do I get ODBC drivers for my database?

Most database vendors distribute ODBC drivers for use with their databases. Contact your database vendor to obtain the drivers. Additionally there are a number of third party manufacturers of ODBC drivers that can provide drivers for a variety of common databases.

What ODBC version driver do I need?

Be sure your ODBC driver is a version 3 or higher driver, meaning it implements the ODBC specification version 3. Each driver provider will have their own version numbering system for their driver versions, which will likely be different from the ODBC version number it implements. The ODBC version 3 specification was introduced in 1995, so it is likely your database provider has a driver that implements this level of compliance. We have seen a number of Tableau customers with drivers that are not version 3 compliant (which fail to connect), who then upgraded to a newer driver and were able to proceed.

Has Tableau tested database [x]?

Tableau has tested ODBC connections with many data sources, but due to the volume of ODBC drivers available on the market we haven't tested against every possible driver, nor can Tableau perform this breadth of testing. The best recommendation for any particular driver is to give it a try, and let us know how it works.

What do I do if it doesn't work?

First check the driver version. If you look at the Tableau logs we will echo back the ODBC version level the driver reports. Search the logs for an entry like the following:

```
ODBCProtocol: driver ODBC version: 03.52
```

The number at the end indicates the ODBC version level. If it is less than 03.00, the driver needs to be upgraded.

If you get warnings when you connect about capabilities not supported, check with your database vendor to see if they have an updated driver that supports more capabilities. Not all drivers will work with Tableau.

Should I use the named database connector or the ODBC connector?

If you are connecting to a database for which Tableau has a named connection option, then use the named connector. The named connectors are optimized for the particular database.

See also

[Other Databases \(ODBC\) on page 1517](#) – Describes how to connect to your data using the ODBC connector.

[Customize and Tune ODBC Connections below](#) – Describes how to fine tune the ODBC connection information for improved functionality and performance.

[Tableau Capability Customizations Reference on page 1530](#) – Lists customizations you can use to define which Tableau capabilities are supported by the data source.

[ODBC/SQL Customizations Reference on page 1543](#) – Lists customizations that represent the parts of the ODBC and SQL standards that the ODBC driver reports supporting.

Customize and Tune ODBC Connections

When you use a connector for a database that supports SQL, Tableau generates SQL statements that are tuned for that database. Because Tableau has no representation of the SQL dialect used by an ODBC data source, it must infer the supported syntax through a variety of tests. If a driver reports incorrect or incomplete information about the SQL dialect it supports, you can use Tableau customizations to fine tune this connection information in order to improve functionality and performance.

In this article, you will make an ODBC connection, examine the resulting .tds file and use part of it to create a Tableau Datasource Customization (.tdc) file that you can then use to customize your ODBC connection. Before you begin, you should be familiar with the content in [Tableau and ODBC on page 1518](#).

Note: Tableau will provide reasonable levels of customer support to assist in troubleshooting connections with ODBC drivers, but can't create or customize a connector to work with a specific ODBC driver.

In this article

[Make an ODBC connection](#) on the next page

[Review the XML structure of a TDS file](#) on page 1524

[Make customizations global with a .tdc file](#) on page 1525

[Customize your ODBC connection](#) on page 1528

[ODBC documentation reference](#) on page 1529

Make an ODBC connection

This section shows you how to create an ODBC connection using an example. In the example, you connect to a SQLite database using the SQLite ODBC driver.

Prerequisites

The ODBC connection in this article is based on SQLite (<http://www.sqlite.org/>), an open-source database.

You need to download the following two items:

Disclaimer: This information refers to a third-party product. This example is not an endorsement of this product over any other competing products.

- The SQLite 32-bit ODBC driver (required for 32-bit or 64-bit Windows). Download and install the following (select the SQLite 2 drivers):
<http://www.ch-werner.de/sqliteodbc/sqliteodbc.exe>
- A sample SQLite database created from the Microsoft example Northwind database. Download and expand the following:
<http://download.vive.net/Northwind.zip>

Create a connection

To create an ODBC connection, you connect to the Northwind database using the SQLite3 ODBC driver, and then save the connection as a Tableau Data Source (.tds) file.

1. Open Tableau Desktop.
2. On the start page under **Connect**, click **Other Databases (ODBC)**.
3. Under **Connect Using**, select **Driver**, and then from the drop-down list, select **SQLite3 ODBC Driver**.
4. Click **Connect**.
5. Next to the **Database Name** text box, click **Browse**, navigate to the location of your **Northwind.sl3** file, and then click **Open**.
6. Click **OK** to close the dialog box, and then click **Sign In**.
7. On the data source page, in the **Table** text box, enter **Orders**.
8. Drag the **Orders** table to the canvas, and then click the sheet tab.

A "Tableau identified limitations for the ODBC data source" dialog box opens. You can review the details or dismiss the dialog box and continue. For more information, see [How Tableau determines the capabilities of the ODBC driver](#).

9. Select **Data > Orders [path to Northwind.sl3] > Add to Saved Data Sources**.

10. Click **Save**.

11. Close the workbook. When asked to save changes, click **No**.

Now you can open the `.tds` file to examine the SQLite connection.

Review the XML structure of a TDS file

Open the saved data source (`.tds` file) in a text editor to view the XML structure of the `.tds` file. By default, the file you created above is named `Northwind.s13.tds` and is in the following path:

`Users\[your name]\Documents\My Tableau Repository\Datasources`

or

`Users\[your name]\Documents\My Tableau Repository (Beta)\Data-sources`

This is an XML document describing the SQLite connection to the Northwind Orders table. Within the `<connection>` section, find the `<connection-customization>` element, which contains `<customization>` elements that you can edit.

Vendor and driver name

The connection customization section begins with the names Tableau detected for the ODBC driver and the database vendor. Tableau uses this information to associate a specific connection customization with a single type of ODBC data source. The section looks like this:

```
<connection-customization class='genericodbc' enabled='false' version='10.1'>

<vendor name='SQLite' />

<driver name='SQLite3 ODBC Driver' />

<customizations>
```

Types of customizations

Tableau permits two types of customizations: Tableau-specific capabilities, and ODBC API calls to `SQLGetInfo`. These customizations are made of name/value pairs, with the names following a convention of `CAP_` for Tableau capabilities and `SQL_` for the `SQLGetInfo` API calls.

The saved data source file contains examples of both types of customizations. These customizations represent the values that Tableau was able to detect by querying the driver when you connected. The list of customizations might be incomplete or incorrect. You can use customization to modify and shape Tableau's behavior when connecting to an ODBC data source.

The following articles contain a complete reference for the customizations. You should have an idea of how you need to adjust your ODBC connection to perform as you expect in your environment. Review the lists to get an idea of the customizations you can use to tune your ODBC connection.

- [Tableau Capability Customizations Reference](#) on page 1530
- [ODBC/SQL Customizations Reference](#) on page 1543

Format of customization values

- The customization values are all represented as strings for each name/value pair.
- Tableau capabilities are all Boolean values represented by yes or no.
- The SQLGetInfo values can be long-integer, short-integer, or string data, depending on the expected return value of SQLGetInfo for the given identifier.
- Many of the integer fields are bitmasks that represent a collection of capabilities.

Make customizations global with a .tdc file

To make your customization changes apply to all connections for a given ODBC data source, you need to create a Tableau Datasource Customization (.tdc) file. This file contains only the <connection-customization> section and is applied to any new Tableau connection that matches the database vendor name and driver name described in the .tdc file (as noted in [Vendor and driver name on the previous page](#)). Any existing workbook or data source file that already has an enabled customization section will use only the customizations that it supplies, not the .tdc file.

Important: Tableau does not test or support TDC files. These files should be used as a tool to explore or occasionally address issues with your data connection. Creating and maintaining TDC files requires careful manual editing, and there is no support for sharing these files.

Create a .tdc file

When you create a .tdc file, you need to save it to the right location so that it can be used by your connection. If you create a workbook that you want to publish to Tableau Server, then you need to save the .tdc file to the server as well. For more information, see [Using a .tdc File with Tableau Server](#) in Tableau Knowledge Base.

Tableau Desktop

1. Using a text editor, copy and paste the entire <connection-customization> section of your saved data source file. For reference, see the [Sample SQLite TDC file](#).
2. Name the file odbc-sqlite.tdc and save it to Documents\My Tableau Repository\Datasources.

Note: The file must be saved using a `.tdc` extension, but the name does not matter.

3. Restart Tableau Desktop to apply the change.
4. Create a new connection to SQLite as described in [Create a connection](#).
5. Go to and open the Tableau Desktop log file, and look for a record similar to the example below to verify that this customization file was applied to your new connection.

Log File Location	Example of Record
By default, <code>C:\User- s\<user>\Documents\My Tableau Repository\Logs</code>	Found matching TDC ' <code>C:\\User- s\\<name>\\Documents\\My Tableau Repository\\Datasources\\odbc-sql- ite.tdc</code> ' for class='genericodbc', vendor='SQLite', and driver='SQLite3 ODBC Driver'

Tableau Server

1. Using a text editor, copy and paste the entire `<connection-customization>` section of your saved data source file. For reference, see the [Sample SQLite TDC file](#).
 2. Name the file `odbc-sqlite.tdc` and save it to `Program
Files\Tableau\Tableau Server\<version>\bin`.
- To save the file, you must be a Windows Administrator on the server computer.

Note: The file must be saved using a `.tdc` extension, but the name does not matter.

3. Restart Tableau Server to apply the change.
4. In Tableau Desktop, create a new connection to SQLite as described in [Create a connection](#).
5. Go to and open the Tableau Server log file, and look for a record similar to the example below to verify that this customization file was applied to your new connection.

Log File Location	Example of Record
<code>C:\\Pro- gramData\\Tableau\\Tableau Server- \\data\\t-</code>	Found matching TDC ' <code>C:\\\\Pro- gramData\\\\Tableau\\\\Tableau Server- \\\\data\\\\tabsvc\\\\vizqlserver\\\\Datasources\\\\odbc- sqlite.tdc</code> ' for class='genericodbc',

Log File Location	Example of Record
absvc\vizqlserver\Logs	vendor='SQLite', and driver='SQLite3 ODBC Driver'

Sample SQLite .tdc file

```

<?xml version='1.0' encoding='utf-8' ?>

<connection-customization class='genericodbc' enabled='true' version='7.8'>

<vendor name='SQLite' />

<driver name='SQLite3 ODBC Driver' />

<customizations>

    <customization name='CAP_CREATE_TEMP_TABLES' value='yes' />

    <customization name='CAP_QUERY_BOOLEXPR_TO_INTEXPR' value='yes' />
/>

    <customization name='CAP_QUERY_GROUP_BY_ALIAS' value='no' />

    <customization name='CAP_QUERY_GROUP_BY_DEGREE' value='yes' />

    <customization name='CAP_QUERY_JOIN_ACROSS_SCHEMAS' value='no' />
/>

    <customization name='CAP_QUERY_JOINQUIRES_SCOPE' value='no' />
/>

    <customization name='CAP_QUERY_SUBQUERIES' value='yes' />

    <customization name='CAP_QUERY_SUBQUERIES_WITH_TOP' value='yes' />
/>

    <customization name='CAP_SELECT_INTO' value='no' />

    <customization name='CAP_SELECT_TOP_INTO' value='yes' />

    <customization name='SQL_AGGREGATE_FUNCTIONS' value='127' />

    <customization name='SQL_SQL_CONFORMANCE' value='4' />

</customizations>

</connection-customization>

```

Customize your ODBC connection

Let's assume that you've made an ODBC connection to your data source and verified that you can get the metadata and data you need. Now you need to determine if creating an extract or if using the live connection as is gives you the functionality you need. If it does, the `.tdc` file you created is sufficient, and there is no need to customize your connection.

If the `.tdc` file is not performing the way you want it to, you can customize the connection. You should have an idea of how you need to adjust your ODBC connection to perform as you expect in your environment. Review the lists to see the customizations you can use to tune your ODBC connection.

- [Tableau Capability Customizations Reference](#) on page 1530
- [ODBC/SQL Customizations Reference](#) on page 1543

Common customizations for improving functionality

The following customizations can help improve functionality for data sources with default capabilities that are not optimal.

- `CAP_SUPPRESS_DISCOVERY_QUERIES` – Setting this value to `true` prevents Tableau from performing any SQL queries at the beginning of the connection to determine the driver's capabilities. You will need to consider which other capabilities should be explicitly enabled, since Tableau will not be able to determine those automatically.
- `SQL_SQL_CONFORMANCE` – This setting indicates which level of the SQL standard is fully supported by the data source. Tableau works best with at least entry-level conformance, so this value should be at least 1 if the data source supports it.
- `SQL_AGGREGATE_FUNCTIONS` – This setting indicates which aggregate functions are supported, such as `MIN`, `MAX`, `SUM`. A value of '`127`' indicates support for all standard SQL aggregate functions.

Handle severely limited data sources

Some data sources are so severely limited that Tableau is unable to complete the steps of creating a connection. Occasionally this is due to crashes within the driver, which cause Tableau to cease working. You can use a global TDC file to prevent Tableau from issuing queries or checking for capabilities that might be associated with the instability.

To create this file, you need to know the database vendor name and ODBC driver name.

1. Create a new connection in Tableau, open your log file, and then look for a line like the following:

```
GenericODBCProtocol::Connect: Detected vendor: 'SQLite' and  
driver: 'SQLite3 ODBC Driver'
```

2. Create a .tdc file with the listed vendor name and driver name. See [Vendor and driver name on page 1524](#) for an example.
3. Use customizations, such as CAP_SUPPRESS_DISCOVERY_QUERIES, to constrain Tableau's interaction with a limited data source.

[Configure ODBC driver settings](#)

ODBC drivers provide dialog boxes for supplying connection details such as the server, user name, and password. Many offer advanced options for controlling the connection behavior. You should consider these options when exploring ways to improve the functionality or performance of your ODBC connection. In particular, look for settings that control the items listed below, as these have been the cause of past issues with ODBC connections in Tableau:

- Transaction Isolation - Prefer READ, COMMITTED, or SERIALIZABLE to ensure that queries do not include data from pending transactions in their results.
- Cursors - Prefer Holdable or Scrollable over Streaming cursors. Since Tableau closes a transaction after every read-only analytical query, streaming cursors may become truncated and lead to incomplete results displayed in Tableau.
- Row/Resultset Buffers - Prefer larger buffer sizes to improve the performance of fetching numerous rows, which can greatly improve the speed of creating extracts. This is sometimes called the cache size or response size.
- Character Set - In decreasing order of preference: UTF-16, UTF-8, ASCII.

[Advanced SQLite customizations](#)

Tableau has special customizations for SQLite that are built in to the product. These customizations take precedence over any connection customizations for SQLite in the workbook file, data source file, or global .tds file. In order to make advanced changes to SQLite connection customizations, you need to trick Tableau into ignoring its own SQLite dialect. You can do so by changing the reported name of the database vendor (`<customization name='SQL_DBMS_NAME' value='SQLite' />`) to a different value such as SQLite-Tableau.

[ODBC documentation reference](#)

Microsoft's MSDN has documentation for the ODBC standard. The [ODBC Appendixes](#) are the most applicable as resources for the article information, specifically:

- Appendix C: [SQL Grammar](#) - especially the [SQL Minimum Grammar](#) and [ODBC Escape Sequences](#)
- Appendix D: [Data Types](#)
- Appendix E: [Scalar Functions](#)

See also

[Other Databases \(ODBC\) on page 1517](#) – Describes how to connect to your data using the ODBC connector.

[Tableau and ODBC on page 1518](#) – Provides background information about ODBC, describes how Tableau determines the functionality of an ODBC driver, and lists frequently asked questions.

[Tableau Capability Customizations Reference below](#) – Lists customizations you can use to define which Tableau capabilities are supported by the data source.

[ODBC/SQL Customizations Reference on page 1543](#) – Lists customizations that represent the parts of the ODBC and SQL standards that the ODBC driver reports supporting.

Tableau Capability Customizations Reference

You can set the following capability customizations in the Tableau Datasource Customization (TDC) file to define which Tableau capabilities are supported by the ODBC connection. For more information, see [Customize and Tune ODBC Connections on page 1522](#).

Many of these customizations influence the type of SQL queries that Tableau issues. For settings that are not defined, Tableau attempts to determine the proper values for each capability by issuing various forms of SQL queries to experimentally verify which forms are supported, as described in [How Tableau determines the capabilities of the ODBC driver](#).

CAP_CREATE_TEMP_TABLES	Set to 'yes' if Tableau can create temporary tables needed for certain complex or optimized queries. See also: CAP_SELECT_INTO.
CAP_CONNECT_STORED_PROCEDURE	Set to 'yes' to allow support for connecting to a stored procedure.
CAP_FAST_METADATA	Set to 'yes' if you have small to moderate size schemas. This capability controls whether Tableau should enumerate all of the objects immediately when you connect. Set the value to "yes" to enable this capability for better performance when creating new connections to your Oracle data. Disable this capability to allow search for specific schemas or tables instead of retrieving all objects. You can search for all objects by using an empty string. This capability is available in 9.0 and later.
CAP_ISOLATION_LEVEL_READ_COMMITTED	Set to 'yes' to force the transaction isolation level to Read Committed if the data source supports it. Only one of the four transaction isolation levels should be set to 'yes'. See also: CAP_SET_

	ISOLATION_LEVEL_VIA_SQL, CAP_SET_ISOLATION_LEVEL_VIA_ODBC_API.
CAP_ISOLATION_LEVEL_READ_UNCOMMITTED	Set to 'yes' to force the transaction isolation level to Read Uncommitted if the data source supports it. Only one of the four transaction isolation levels should be set to 'yes'. This capability can improve speed by reducing lock contention, but may result in partial or inconsistent data in query results. See also: CAP_SET_ISOLATION_LEVEL_VIA_SQL, CAP_SET_ISOLATION_LEVEL_VIA_ODBC_API.
CAP_ISOLATION_LEVEL_REPEATABLE_READS	Set to 'yes' to force the transaction isolation level to Repeatable Reads if the data source supports it. Only one of the four transaction isolation levels should be set to 'yes'. See also: CAP_SET_ISOLATION_LEVEL_VIA_SQL, CAP_SET_ISOLATION_LEVEL_VIA_ODBC_API.
CAP_ISOLATION_LEVEL_SERIALIZABLE	Set to 'yes' to force the transaction isolation level to Serializable if the data source supports it. Only one of the four transaction isolation levels should be set to 'yes'. This is a very conservative setting that may improve stability at the expense of performance. See also: CAP_SET_ISOLATION_LEVEL_VIA_SQL, CAP_SET_ISOLATION_LEVEL_VIA_ODBC_API.
CAP_SET_ISOLATION_LEVEL_VIA_ODBC_API	Set to 'yes' to force Tableau to set the transaction isolation level for the data source using the ODBC API. CAP_SET_ISOLATION_LEVEL_VIA_ODBC_API must be set to 'yes' when any one of the four CAP_ISOLATION_LEVEL capabilities has been set to 'yes'.
CAP_SET_ISOLATION_LEVEL_VIA_SQL	Set to 'yes' to force Tableau to set the transaction isolation level for the data source using a SQL query. CAP_SET_ISOLATION_LEVEL_VIA_SQL must be set to 'yes' when any one of the four CAP_ISOLATION_LEVEL capabilities has been set to 'yes'.
CAP_MULTIPLE_CONNECTIONS_FROM_SAME_IP	Set to 'no' to prevent Tableau from creating more than one active connection to the database. This is a conservative setting that may increase stability at

	the expense of performance.
CAP_ODBC_BIND_DETECT_ALIAS_CASE_FOLDING	Set to 'yes' to allow Tableau to detect and recover from an ODBC data source that reports the field names in a result set using only upper-case or lower-case characters, instead of the expected field names.
CAP_ODBC_BIND_BOOL_AS_WCHAR_01LITERAL	Set to 'yes' to bind a Boolean data type as a WCHAR containing values '0' or '1'.
CAP_ODBC_BIND_BOOL_AS_WCHAR_TFLITERAL	Set to 'yes' to bind a Boolean data type as WCHAR containing values 't' or 'f'.
CAP_ODBC_BIND_FORCE_DATE_AS_CHAR	Set to 'yes' to force the Tableau native ODBC protocol to bind date values as CHAR.
CAP_ODBC_BIND_FORCE_DATETIME_AS_CHAR	Set to 'yes' to force the Tableau native ODBC protocol to bind datetime values as CHAR.
CAP_ODBC_BIND_FORCE_MAX_STRING_BUFFERS	Set to 'yes' to force the Tableau native ODBC protocol to use maximum-sized buffers (1MB) for strings instead of the size described by metadata.
CAP_ODBC_BIND_FORCE_MEDIUM_STRING_BUFFERS	Set to 'yes' to force the Tableau native ODBC protocol to use medium-sized buffers (1K) for strings instead of the size described by metadata.
CAP_ODBC_BIND_FORCE_SMALL_STRING_BUFFERS	Set to 'yes' to force the Tableau native ODBC protocol to use small buffers for strings instead of the size described by metadata.
CAP_ODBC_BIND_FORCE_SIGNED	Set to 'yes' to force binding integers as signed.
CAP_ODBC_BIND_PRESERVE_BOM	Set to 'yes' to preserve BOM when present in strings. Hive will return BOM and treat strings containing it as distinct entities.
CAP_ODBC_BIND_SUPPRESS_COERCE_TO_STRING	Set to 'yes' to prevent the Tableau native ODBC protocol from binding non-string data as strings (i.e. requesting driver conversion).
CAP_ODBC_BIND_SUPPRESS_INT64	Set to 'yes' to prevent the Tableau native ODBC protocol from using 64-bit integers for large numeric data.
CAP_ODBC_BIND_SUPPRESS_PREFERRED_CHAR	Set to 'yes' to prevent the Tableau native ODBC protocol from preferring a character type that differs from the driver default.

CAP_ODBC_BIND_SUPPRESS_PREFERRED_TYPES	Set to 'yes' to prevent the Tableau native ODBC protocol from binding any data according to its preferred wire types. With this capability set, Tableau will only bind according to the data types described by the ODBC driver via metadata.
CAP_ODBC_BIND_SUPPRESS_WIDE_CHAR	Set to 'yes' to prevent the Tableau native ODBC protocol from binding strings a WCHAR. Instead they will be bound as single-byte CHAR arrays, and processed locally for any UTF-8 characters contained within.
CAP_ODBC_CONNECTION_STATE_VERIFY_FAST	Set to 'yes' to check if a connection is broken with a fast ODBC API call.
CAP_ODBC_CONNECTION_STATE_VERIFY_PROBE	Set to 'yes' to check if a connection is broken with a forced probe.
CAP_ODBC_CONNECTION_STATE_VERIFY_PROBE_IF_STALE	Set to 'yes' to check if a connection is broken with a forced probe only if it is "stale" (i.e., unused for about 30 minutes).
CAP_ODBC_CONNECTION_STATE_VERIFY_PROBE_PREPARED_QUERY	Set to 'yes' to check if a connection is broken using a prepared query.
CAP_ODBC_CURSOR_DYNAMIC	Set to 'yes' to force the Tableau native ODBC protocol to set the cursor type for all statements to Dynamic (scrollable, detects added/removed/modified rows).
CAP_ODBC_CURSOR_FORWARD_ONLY	Set to 'yes' to force the Tableau native ODBC protocol to set the cursor type for all statements to Forward-only (non-scrollable).
CAP_ODBC_CURSOR_KEYSET_DRIVEN	Set to 'yes' to force the Tableau native ODBC protocol to set the cursor type for all statements to Keyset-driven (scrollable, detects changes to values within a row).
CAP_ODBC_CURSOR_STATIC	Set to 'yes' to force Tableau to set the cursor type for all statements to Static (scrollable, does not detect changes).
CAP_ODBC_ERROR_IGNORE_FALSE_ALARM	Set to 'yes' to allow the Tableau native ODBC protocol to ignore SQL_ERROR conditions where SQLSTATE is '00000' (meaning "no error").
CAP_ODBC_EXPORT_ALLOW_	Set to 'yes' to allow the use of single-byte char data

CHAR_UTF8	type for binding Unicode strings as UTF-8.
CAP_ODBC_EXPORT_BIND_FORCE_TARGET_METADATA	Set to 'yes' to force binding for export based on all of the metadata from the target table instead of the ODBC metadata for the parameterized insert statement.
CAP_ODBC_EXPORT_BIND_PREFER_TARGET_METADATA	Set to 'yes' to prefer binding for export based on specific types of metadata from the target table instead of the ODBC metadata for the parameterized insert statement.
CAP_ODBC_EXPORT_BUFFERS_RESIZABLE	Set to 'yes' to allow export buffers to be reallocated after the first batch to improve performance.
CAP_ODBC_EXPORT_BUFFERS_SIZE_FIXED	Set to 'yes' to ignore the width of a single row when computing the total rows to insert at a time.
CAP_ODBC_EXPORT_BUFFERS_SIZE_LIMIT_512KB	Set to 'yes' to limit export buffers to 512 KB. This is an uncommon setting.
CAP_ODBC_EXPORT_BUFFERS_SIZE_MASSIVE	Set to 'yes' to force the use of large buffers for insert. If CAP_ODBC_EXPORT_BUFFERS_RESIZABLE is not set or disabled, a fixed row count is used.
CAP_ODBC_EXPORT_BUFFERS_SIZE_MEDIUM	Set to 'yes' to force the use of medium-sized buffers for insert. If CAP_ODBC_EXPORT_BUFFERS_RESIZABLE is not set or disabled, a fixed row count is used.
CAP_ODBC_EXPORT_BUFFERS_SIZE_SMALL	Set to 'yes' to force the use of small buffers for insert. If CAP_ODBC_EXPORT_BUFFERS_RESIZABLE is not set or disabled, a fixed row count is used.
CAP_ODBC_EXPORT_CONTINUE_ON_ERROR	Set to 'yes' to continue data insert despite errors. Some data sources report warnings as errors.
CAP_ODBC_EXPORT_DATA_BULK	Set to 'yes' to allow the use of ODBC bulk operations for data insert.
CAP_ODBC_EXPORT_DATA_BULK_VIA_INSERT	Set to 'yes' to allow the use of ODBC bulk operations based on 'INSERT INTO' parameterized queries.
CAP_ODBC_EXPORT_DATA_BULK_VIA_ROWSET	Set to 'yes' to allow the use of ODBC bulk operations based on a rowset cursor.

CAP_ODBC_EXPORT_FORCE_INDICATE_NTS	Set to 'yes' to force the use of indicator buffers for identifying null-terminated strings (NTS).
CAP_ODBC_EXPORT_FORCE_SINGLE_ROW_BINDING	Set to 'yes' to force the use of a single row for binding export buffers to insert data.
CAP_ODBC_EXPORT_FORCE_SINGLE_ROW_BINDING_WITH_TIMESTAMPS	Set to 'yes' to force the use of a single row for binding export buffers when dealing with timestamp data. This is required for some versions of Teradata.
CAP_ODBC_EXPORT_FORCE_STRING_WIDTH_FROM_SOURCE	Set to 'yes' to force the use of the source string width (from Tableau metadata), overriding the destination string width (from insert parameter metadata).
CAP_ODBC_EXPORT_FORCE_STRING_WIDTH_USING_OCTET_LENGTH	Set to 'yes' to force the use of the source string width from the octet length.
CAP_ODBC_EXPORT_SUPPRESS_STRING_WIDTH_VALIDATION	Set to 'yes' to suppress validating that the target string width can accommodate the widest source strings.
CAP_ODBC_EXPORT_TRANSACTIONS_COMMIT_BATCH_MASSIVE	Set to 'yes' to commit in massive batches of INSERT statements (~100,000). This may be useful with single-row export binding.
CAP_ODBC_EXPORT_TRANSACTIONS_COMMIT_BATCH_MEDIUM	Set to 'yes' to commit in medium-sized batches of INSERT statements (~50). A single statement may be bound to multiple records.
CAP_ODBC_EXPORT_TRANSACTIONS_COMMIT_BATCH_SMALL	Set to 'yes' to commit in small batches of INSERT statements (~5). A single statement may be bound to multiple records.
CAP_ODBC_EXPORT_TRANSACTIONS_COMMIT_BYTES_MASSIVE	Set to 'yes' to commit in massive batches of data (~100 MB).
CAP_ODBC_EXPORT_TRANSACTIONS_COMMIT_BYTES_MEDIUM	Set to 'yes' to commit in medium batches of data (~10 MB).
CAP_ODBC_EXPORT_TRANSACTIONS_COMMIT_BYTES_SMALL	Set to 'yes' to commit in small batches of data (~1 MB).
CAP_ODBC_EXPORT_	Set to 'yes' to commit after executing each INSERT

TRANSACTIONS_COMMIT_EACH_STATEMENT	statement. A single statement may be bound to multiple records.
CAP_ODBC_EXPORT_TRANSACTIONS_COMMIT_INTERVAL_LONG	Set to 'yes' to commit in long intervals of elapsed time (~100 seconds).
CAP_ODBC_EXPORT_TRANSACTIONS_COMMIT_INTERVAL_MEDIUM	Set to 'yes' to commit in medium intervals of elapsed time (~10 seconds).
CAP_ODBC_EXPORT_TRANSACTIONS_COMMIT_INTERVAL_SHORT	Set to 'yes' to commit in short intervals of elapsed time (~1 seconds).
CAP_ODBC_EXPORT_TRANSACTIONS_COMMIT_ONCE_WHEN_COMPLETE	Set to 'yes' to commit only once at the end after the export is complete.
CAP_ODBC_EXPORT_TRANSLATE_DATA_PARALLEL	Set to 'yes' to use parallel loops to translate Tableau DataValues to wire buffers on export.
CAP_ODBC_FETCH_ABORT_FORCE_CANCEL_STATEMENT	Set to 'yes' to cancel the statement handle upon interrupting SQLFetch with a cancel exception.
CAP_ODBC_FETCH_BUFFERS_RESIZABLE	Set to 'yes' to allow buffers to be reallocated after fetch to improve performance or handle data truncation.
CAP_ODBC_FETCH_BUFFERS_SIZE_FIXED	Set to 'yes' to ignore the width of a single row when computing the total rows to fetch.
CAP_ODBC_FETCH_BUFFERS_SIZE_MASSIVE	Set to 'yes' to force the use of large buffers. If CAP_ODBC_FETCH_BUFFERS_SIZE_FIXED is enabled, a fixed row count is used.
CAP_ODBC_FETCH_BUFFERS_SIZE_MEDIUM	Set to 'yes' to force the use of medium-sized buffers. If CAP_ODBC_FETCH_BUFFERS_SIZE_FIXED is enabled, a fixed row count is used.
CAP_ODBC_FETCH_BUFFERS_SIZE_SMALL	Set to 'yes' to force the use of small buffers. If CAP_ODBC_FETCH_BUFFERS_SIZE_FIXED is enabled, a fixed row count is used.
CAP_ODBC_FETCH_CONTINUE_ON_ERROR	Set to 'yes' to allow the Tableau native ODBC protocol to continue resultset fetch despite errors (some data sources report warnings as errors).
CAP_ODBC_FETCH_IGNORE_FRACTIONAL_SECONDS	Set to 'yes' to allow the Tableau native ODBC

	protocol to ignore the fractional seconds component of a time value when fetching query result set data. This is useful when working with data sources that do not follow the ODBC specification for fractional seconds, which must be represented as billionths of a second.
CAP_ODBC_FETCH_RESIZE_BUFFERS	Set to 'yes' to allow the Tableau native ODBC protocol to automatically resize buffers and fetch again if data truncation occurred.
CAP_ODBC_FORCE_SINGLE_ROW_BINDING	Set to 'yes' to force the Tableau native ODBC protocol to use a single row for result set transfers instead of the more efficient bulk-fetch.
CAP_ODBC_IMPORT_ERASE_BUFFERS	Set to 'yes' to reset the contents of data buffers before fetching each block.
CAP_ODBC_IMPORT_TRUST_METADATA_CONTAINSNULL	Set to 'yes' to use efficient import if Tableau can trust the result set metadata representing the absence of NULLs.
CAP_ODBC_METADATA_FORCE_LENGTH_AS_PRECISION	Set to 'yes' to force the Tableau native ODBC protocol to use the column "length" as the numeric precision. This is an uncommon setting.
CAP_ODBC_METADATA_FORCE_NUM_PREC_RADIX_10	Set to 'yes' to force the Tableau native ODBC protocol to assume the numeric precision is reported in base-10 digits. This is an uncommon setting.
CAP_ODBC_METADATA_FORCE_UNKNOWN_AS_STRING	Set to 'yes' to force the Native ODBC Protocol to treat unknown data types as string instead of ignoring the associated column.
CAP_ODBC_METADATA_FORCE_UTF8_IDENTIFIERS	Set to 'yes' to force the protocol to treat identifiers as UTF-8 when communicating with the driver.
CAP_ODBC_METADATA_SKIP_DESC_TYPE_NAME	Set to 'yes' to remove the check for the SQL_DESC_TYPE_NAME attribute with the SQLColAttribute API.
CAP_ODBC_METADATA_STRING_LENGTH_UNKNOWN	Set to 'yes' to prevent Tableau from allocating memory based on the driver-reported string length, which may not be known or reported properly. Instead, Tableau will use a fixed-sized string length, and will reallocate as needed to handle string data that is too large for the fixed-size

	buffer.
CAP_ODBC_METADATA_STRING_TRUST_OCTET_LENGTH	Set to 'yes' to use the octet length reported by the driver for strings instead of computing it from the number of characters.
CAP_ODBC_METADATA_SUPPRESS_EXECUTED_QUERY	Set to 'yes' to prevent Tableau from executing a query as a means of reading metadata. While Tableau typically includes a row-limiting clause in such metadata queries (e.g., 'LIMIT', or 'WHERE 1=0'), this may not help when used with a Custom SQL connection for database systems with poor query optimizers. Note that this capability may prevent Tableau from determining the connection metadata properly.
CAP_ODBC_METADATA_SUPPRESS_PREPARED_QUERY	Set to 'yes' to prevent Tableau from using a prepared query as a means of reading metadata. A prepared query is often the fastest way to accurately read metadata. However, not all database systems are capable of reporting metadata for a prepared query without actually executing the query. Note that certain metadata -- for example from connections using Custom SQL-- cannot be retrieved if this capability and CAP_ODBC_METADATA_SUPPRESS_EXECUTED_QUERY are both set.
CAP_ODBC_METADATA_SUPPRESS_SELECT_STAR	Set to 'yes' to prevent reading metadata using a 'select *' query.
CAP_ODBC_METADATA_SUPPRESS_SQLCOLUMNS_API	Set to 'yes' to prevent Tableau from using older, less accurate API for reading metadata from ODBC data sources. Setting this capability allows Tableau to read metadata by issuing a full 'select *' query, which is expensive but may enable connectivity for extremely limited or unstable data sources.
CAP_ODBC_METADATA_SUPPRESS_SQLFOREIGNKEYS_API	Set to 'yes' to prevent Tableau from attempting to read metadata describing foreign key constraints. Despite the simple nature of this ODBC API, some drivers may have unstable behavior or produce inaccurate results. Setting this capability may force Tableau to generate less efficient queries involving

	multi-table joins.
CAP_ODBC_METADATA_SUPPRESS_SQLPRIMARYKEYS_API	Set to 'yes' to prevent Tableau from reading primary key metadata using the SQPrimaryKeys API or an equivalent query. This capability is available in Tableau 9.1 and later.
CAP_ODBC_METADATA_SUPPRESS_SQLSTATISTICS_API	Set to 'yes' to prevent reading unique constraints and table cardinality estimates using the SQLStatistics API or an equivalent query. This capability is available in Tableau 9.0 and later.
CAP_ODBC_REBIND_SKIP_UNBIND	Set to 'yes' to force the Tableau native ODBC protocol to rebinding a column directly and skip unbinding, which reduces ODBC API calls when resizing buffers to refetch truncated data.
CAP_ODBC_SUPPRESS_INFO_SCHEMA_STORED_PROCS	Set to 'yes' to prevent the INFORMATION.SCHEMA schema from being queried when enumerating stored procedures.
CAP_ODBC_SUPPRESS_PREPARED_QUERY_FOR_ALL_COMMAND_QUERIES	Set to 'yes' to execute all commands directly (i.e., no prepared statement).
CAP_ODBC_SUPPRESS_PREPARED_QUERY_FOR_DDL_COMMAND_QUERIES	Set to 'yes' to execute DDL commands (e.g. CREATE TABLE) directly (i.e., no prepared statement).
CAP_ODBC_SUPPRESS_PREPARED_QUERY_FOR_DML_COMMAND_QUERIES	Set to 'yes' to execute DML commands (e.g. INSERT INTO) directly (i.e., no prepared statement).
CAP_ODBC_SUPPRESS_SYS_SCHEMA_STORED_PROCS	Set to 'yes' to explicitly add the "SYS" schema to the schema exclusions when enumerating stored procedures.
CAP_ODBC_TRANSACTIONS_COMMIT_INVALIDATES_PREPARED_QUERY	Set to 'yes' to indicate that a transaction will invalidate all prepared statements and close any open cursors.
CAP_ODBC_TRANSACTIONS_SUPPRESS_AUTO_COMMIT	Set to 'yes' to prevent the Native ODBC Protocol from using default auto-committing transaction behavior in ODBC. This capability cannot be used with CAP_ODBC_TRANSACTIONS_SUPPRESS_EXPLICIT_COMMIT.
CAP_ODBC_TRANSACTIONS_	Set to 'yes' to prevent the Native ODBC Protocol

SUPPRESS_EXPLICIT_COMMIT	from explicitly managing transactions. This capability cannot be used with CAP_ODBC_TRANSACTIONS_SUPPRESS_AUTO_COMMIT.
CAP_ODBC_TRIM_VARCHAR_PADDING	Set to 'yes' to force the Tableau native ODBC protocol to trim trailing whitespace from VARCHAR columns which the driver has erroneously padded.
CAP_ODBC_UNBIND_AUTO	Set to 'yes' to force the Tableau native ODBC protocol to unbind and deallocate columns automatically, which can reduce ODBC API calls.
CAP_ODBC_UNBIND_BATCH	Set to 'yes' to force the Tableau native ODBC protocol to unbind and deallocate columns in a single batch operation, which can reduce ODBC API calls.
CAP_ODBC_UNBIND_EACH	Set to 'yes' to force the Tableau native ODBC protocol to unbind and deallocate columns individually, which may improve stability.
CAP_ODBC_UNBIND_PARAMETERS_BATCH	Set to 'yes' to unbind all parameters in a single batch operation.
CAP_ORACLE_SHOW_ALL_SYNONYM OWNERS	Set to 'yes' to list all the owners in the all_synonyms view for Oracle. This capability is available in 9.0 and later.
CAP_QUERY_BOOLEXPR_TO_INTEXPR	Set to 'yes' if Tableau must coerce any Boolean expressions to an integer value in order include in a result set.
CAP_QUERY_FROM_REQUIRES_ALIAS	Set to 'yes' if the FROM clause must provide an alias for the given table.
CAP_QUERY_GROUP_ALLOW_DUPLICATES	Set to 'no' if SQL queries cannot contain duplicate expressions in the GROUP BY clause (this is uncommon).
CAP_QUERY_GROUP_BY_ALIAS	Set to 'yes' if SQL queries with aggregations can reference the grouping columns by their corresponding alias in the SELECT list, e.g. GROUP BY "none_ShipCountry_nk".
CAP_QUERY_GROUP_BY_DEGREE	Set to 'yes' if SQL queries with aggregations can reference the grouping columns by the ordinal position of each column, e.g. GROUP BY 2, 5. See also: CAP_QUERY_SORT_BY_DEGREE

CAP_QUERY_HAVING_REQUIRES_GROUP_BY	Set to 'yes' if Tableau must use an artificial grouping field for any query which has a HAVING clause but no grouping columns.
CAP_QUERY_HAVING_UNSUPPORTED	Set to 'yes' if the SQL syntax for HAVING is unsupported. Tableau may be able to work around this using subqueries. See also: CAP_QUERY_SUBQUERIES.
CAP_QUERY_INCLUDE_GROUP_BY_COLUMNS_IN_SELECT	Set to 'yes' to require all GROUP BY expressions to also appear in the SELECT expression list.
CAP_QUERY_JOIN_ACROSS_SCHEMAS	Set to 'yes' if SQL queries can express joins between tables located in different schemas.
CAP_QUERY_JOIN_ASSUME_CONSTRAINED	Set to 'yes' to cull inner joins even if the database tables does do not have FK-PK relationships.
CAP_QUERY_JOIN_PUSH_DOWN_CONDITION_EXPRESSIONS	Set to 'yes' to rewrite joins to simplify the ON clause conditions to simple identifier comparisons.
CAP_QUERY_JOINQUIRES_SCOPE	Set to 'yes' if SQL queries must scope each join clause within parentheses to ensure a proper order of evaluation.
CAP_QUERY_JOINQUIRES_SUBQUERY	Set to 'yes' to force join expressions involving more than two tables to be composed with subqueries.
CAP_QUERY_NULLQUIRES_CAST	Set to 'yes' if the data source requires that all NULL literals are cast to an explicit data type.
CAP_QUERY_SELECT_ALIASES_SORTED	Set to 'yes' if Tableau must impose a deterministic order on the SELECT expressions (sorted by alias) to ensure that query results can be properly matched with each field in the Tableau visualization. This is only required for data sources which do not preserve the aliases of the SELECT expressions when returning metadata with the query results.
CAP_QUERY_SORT_BY_DEGREE	Set to 'yes' if SQL queries can reference the sorting columns by the ordinal position of each column, e.g. ORDER BY 2, 5. See also: CAP_QUERY_GROUP_BY_DEGREE.
CAP_QUERY_SUBQUERIES	Set to 'yes' if the data source supports subqueries.
CAP_QUERY_SUBQUERIES_WITH_TOP	Set to 'yes' if the data source supports a TOP or LIMIT row-limiting clause within a subquery.

CAP_QUERY_SUBQUERY_DATASOURCE_CONTEXT	Set to 'yes' to use subquery filtered query context to implement data source filters. This capability is available in Tableau 8.0 through Tableau 9.3 only.
CAP_QUERY_SUBQUERY_QUERY_CONTEXT	Set to 'yes' to force Tableau to use a subquery for context filters instead of a temporary table or locally cached results.
CAP_QUERY_TOP_N	Set to 'yes' if the data source supports any form of row-limiting clause. The exact forms supported are described below.
CAP_QUERY_TOPSTYLE_LIMIT	Set to 'yes' if the data source uses LIMIT as the row-limiting clause.
CAP_QUERY_TOPSTYLE_ROWNUM	Set to 'yes' if the data source supports an Oracle-style filter on ROWNUM as the row-limiting clause.
CAP_QUERY_TOPSTYLE_TOP	Set to 'yes' if the data source uses TOP as the row-limiting clause.
CAP_QUERY_USE_QUERY_FUSION	Set to 'no' to prevent Tableau from combining multiple individual queries into a single combined query. Turn off this capability for performance tuning or if the database is unable to process large queries. This capability is enabled by default and is available in Tableau 9.0 and later for all data sources except Tableau data extracts. Support for this capability in Tableau data extracts is available in Tableau 9.0.6.
CAP_SELECT_INTO	Set to 'yes' if Tableau can create a table on the fly from the resultset of another query. See also: CAP_CREATE_TEMP_TABLES.
CAP_SELECT_TOP_INTO	Set to 'yes' if Tableau can use a TOP or LIMIT row-limiting clause when creating a table from a query resultset.
CAP_STORED_PROCEDURE_PREFER_TEMP_TABLE	Set to 'yes' to use a temporary table to support remote queries over the stored procedure result set.
CAP_STORED_PROCEDURE_REPAIR_TEMP_TABLE_STRINGS	Set to 'yes' to attempt to compute actual string widths if metadata indicates no width or non-positive width.
CAP_STORED_PROCEDURE_TEMP_	Set to 'yes' to populate the temporary table from a

TABLE_FROM_BUFFER	result set buffered in entirety.
CAP_STORED_PROCEDURE_TEMP_TABLE_FROM_NEW_PROTOCOL	Set to 'yes' to populate the temporary table from a separate protocol created for just this operation.
CAP_SUPPRESS_DISCOVERY_QUESTIONS	Set to 'yes' to prevent Tableau from detecting the supported SQL syntax for a variety of clauses.
CAP_SUPPRESS_DISPLAY_LIMITATIONS	Set to 'yes' to suppress displaying any warnings about limitations for this data source.

See also

[Other Databases \(ODBC\) on page 1517](#) – Describes how to connect to your data using the ODBC connector.

[Tableau and ODBC on page 1518](#) – Provides background information about ODBC, describes how Tableau determines the functionality of an ODBC driver, and lists frequently asked questions.

[Customize and Tune ODBC Connections on page 1522](#) – Describes how to fine tune the ODBC connection information for improved functionality and performance.

[ODBC/SQL Customizations Reference below](#) – Lists customizations that represent the parts of the ODBC and SQL standards that the ODBC driver reports supporting.

ODBC/SQL Customizations Reference

You can set the following customizations in the Tableau Datasource Customization (TDC) file to define parts of the ODBC and SQL standards that the ODBC driver supports.

In this article

[Documentation resources for SQLGetInfo below](#)

[SQLGetInfo long-integer values on page 1545](#)

[SQLGetInfo short-integer values on page 1547](#)

[SQLGetInfo string values on page 1547](#)

[Documentation resources for SQLGetInfo](#)

The names of these customizations come from the identifiers used as parameters to SQLGetInfo.

For more information, see the MSDN documentation and the source code header file `sqlext.h` for the numeric and bit-mask values associated with each customization.

- The Microsoft documentation for the ODBC API function SQLGetInfo:

<http://msdn.microsoft.com/en-us/library/ms711681%28VS.85%29.aspx>

- A sample header file, `sqlext.h`, for the ODBC API identifier values used with `SQLGetInfo`:

<http://cpansearch.perl.org/src/MIMER/DBD-Mimer-1.00/sqlext.h>

SQLGetInfo long-integer values

SQL_ODBC_INTERFACE_CONFORMANCE	Currently unused.
SQL_SQL_CONFORMANCE	Integer bitmask. Defines the level which the data source conforms to the SQL standard: '1' for entry-level SQL-92 conformance, '2' for FIPS 127-2 transitional, '4' for Intermediate and '8' for Full conformance.
SQL_CATALOG_USAGE	Integer bitmask. Defines the SQL statements in which a catalog identifier can be used.
SQL_SCHEMA_USAGE	Integer bitmask. Defines the SQL statements in which a schema identifier can be used.
SQL_AGGREGATE_FUNCTIONS	Integer bitmask. Defines which standard SQL aggregation forms are supported.
SQL_NUMERIC_FUNCTIONS	Integer bitmask. Defines which SQL scalar numeric functions are supported.
SQL_STRING_FUNCTIONS	Integer bitmask. Defines which SQL scalar string functions are supported.
SQL_TIMEDATE_FUNCTIONS	Integer bitmask. Defines which SQL scalar date / time functions are supported.
SQL_TIMEDATE_ADD_INTERVALS	Integer bitmask. Defines which date / time intervals are supported with the TIMESTAMPADD scalar function.
SQL_TIMEDATE_DIFF_INTERVALS	Integer bitmask. Defines which date / time intervals are supported with the TIMESTAMPDIF scalar function.
SQL_DATETIME_LITERALS	Integer bitmask. Defines which SQL-92 literals are supported for representing DATE / TIME constants and INTERVALs.
SQL_SYSTEM_FUNCTIONS	Integer bitmask. Defines support for special SQL system scalar functions: IFNULL, DBNAME and USERNAME.
SQL_SQL92_VALUE_EXPRESSIONS	Integer bitmask. Defines which logical functions are supported for testing and manipulating values: CASE, CAST and NULLIF.
SQL_SQL92_NUMERIC_VALUE_FUNCTIONS	Integer bitmask. Defines which functions can produce a numeric value from non-numeric data, including: EXTRACT (for date / time part extraction), CHAR_LENGTH, CHARACTER_LENGTH and POSITION(.. IN ..).
SQL_SQL92_STRING_	Integer bitmask. Defines which string manipulation

FUNCTIONS	functions are supported.
SQL_SQL92_DATETIME_FUNCTIONS	Integer bitmask. Defines which date / time manipulation functions are supported for determining the current date, time or timestamp.
SQL_OJ_CAPABILITIES	Integer bitmask. Defines which type of outer joins are supported.
SQL_SQL92_RELATIONAL_JOIN_OPERATORS	Integer bitmask. Defines which types of JOIN operators are supported, e.g. INNER, OUTER.
SQL_SQL92_PREDICATES	Integer bitmask. Defines which predicates are supported for logical tests of values, e.g. IS NULL, LIKE, IN.
SQL_CONVERT_FUNCTIONS	Integer bitmask. Defines which ODBC scalar functions are supported for CASTing or CONVERTing one data type to another.
SQL_CONVERT_TINYINT	Integer bitmask. Determines which other data types that this named type can be converted to using the ODBC scalar function CONVERT.
SQL_CONVERT_SMALLINT	Integer bitmask. Same as above.
SQL_CONVERT_INTEGER	Integer bitmask. Same as above.
SQL_CONVERT_BIGINT	Integer bitmask. Same as above.
SQL_CONVERT_REAL	Integer bitmask. Same as above.
SQL_CONVERT_FLOAT	Integer bitmask. Same as above.
SQL_CONVERT_DOUBLE	Integer bitmask. Same as above.
SQL_CONVERT_CHAR	Integer bitmask. Same as above.
SQL_CONVERT_VARCHAR	Integer bitmask. Same as above.
SQL_CONVERT_LONGVARCHAR	Integer bitmask. Same as above.
SQL_CONVERT_DECIMAL	Integer bitmask. Same as above.
SQL_CONVERT_NUMERIC	Integer bitmask. Same as above.
SQL_CONVERT_BIT	Integer bitmask. Same as above.
SQL_CONVERT_GUID	Integer bitmask. Same as above.
SQL_CONVERT_BINARY	Integer bitmask. Same as above.
SQL_CONVERT_VARBINARY	Integer bitmask. Same as above.
SQL_CONVERT_	Integer bitmask. Same as above.

LONGVARBINARY	
SQL_CONVERT_DATE	Integer bitmask. Same as above.
SQL_CONVERT_TIME	Integer bitmask. Same as above.
SQL_CONVERT_TIMESTAMP	Integer bitmask. Same as above.
SQL_CONVERT_INTERVAL_DAY_TIME	Integer bitmask. Same as above.

SQLGetInfo short-integer values

SQL_CURSOR_COMMIT_BEHAVIOR	Short integer value. "0" if the prepared statements are closed upon commit for the data source.
SQL_MAX_IDENTIFIER_LEN	Integer value. Defines the maximum number of characters that can be used in an identifier. Tableau leaves room for one extra character as the string terminator.
SQL_TXN_CAPABLE	Short integer value. "0" if the data source does not support the transaction.
SQL_QUOTED_IDENTIFIER_CASE	Integer bitmask.

SQLGetInfo string values

SQL_COLUMN_ALIAS	Boolean value. "Y" if the data source supports using aliases for columns listed in the SELECT clause.
SQL_IDENTIFIER_QUOTE_CHAR	String value. Indicates the character which can be used for quoting identifiers. Because the connection customization is an XML document, any entities must properly be encoded. For example, double-quotes will be """. Additionally this character is assumed to work as the opening and closing character around identifiers, so some data sources which require '[' to enclose identifiers will not be supported.
SQL_CATALOG_NAME_SEPARATOR	Character value. Indicates the separator character to use between identifiers when qualifying them with a catalog, schema or table name. This is typically the period character.
SQL_SPECIAL_CHARACTERS	String value. Indicates the special characters which are allowed in identifier strings.
SQL_CATALOG_TERM	String value. This is the descriptive term for a database

	catalog, which appears in the Tableau connection dialog for this ODBC data source.
SQL_SCHEMA_TERM	String value. This is the descriptive term for a database schema, which appears in the Tableau connection dialog for this ODBC data source.
SQL_TABLE_TERM	String value. This is the descriptive term for a database table, which appears in the Tableau connection dialog for this ODBC data source.
SQL_DRIVER_NAME	String value. This is the name of the database ODBC driver.
SQL_DRIVER_VER	String value. This is the version number of the ODBC driver.
SQL_DRIVER_ODBC_VER	String value. This is the version of the ODBC API which the driver supports.
SQL_ODBC_VER	String value. This is the version of ODBC which the Windows ODBC Driver Manager supports. This should not need to be customized.
SQL_DBMS_NAME	String value. This is the name of the database vendor.
SQL_DBMS_VER	String value. This is the version of the database system.
SQL_SERVER_NAME	String value. This is the named network address of the database server.
SQL_USER_NAME	String value. This is the name of the currently authenticated user.

See also

[Other Databases \(ODBC\) on page 1517](#) – Describes how to connect to your data using the ODBC connector.

[Tableau and ODBC on page 1518](#) – Provides background information about ODBC, describes how Tableau determines the functionality of an ODBC driver, and lists frequently asked questions.

[Customize and Tune ODBC Connections on page 1522](#) – Describes how to fine tune the ODBC connection information for improved functionality and performance.

[Tableau Capability Customizations Reference on page 1530](#) – Lists customizations you can use to define which Tableau capabilities are supported by the data source.

Manage Licenses

Tableau supports the following product licenses:

- **Perpetual License** - This type of license permits the continued use of the licensed software for as long as you comply with all of the terms of the license agreement. There is no set expiration to these rights. You can elect to purchase Support and Maintenance Services to have ongoing access to product updates and technical support. If you choose not to renew the Support and Maintenance Services for a perpetual license, you can continue using the existing licensed software version in perpetuity.
- **Term License** - Often referred to as a subscription license, this type of license grants users the rights to access and use the licensed software only for a specific period of time. When the term expires, you must either purchase a new license to use the software, or stop using the software. Term licenses generally include Support and Maintenance Services in the purchase price rather than as a separate line item.

Renewing a license

Subscription license types, such as Term Licenses, must be renewed to continue providing uninterrupted service. As the license nears its expiration period, a warning is triggered to tell you how much time is left on your term. Warnings are triggered 15 days before the expiration date, one day before the expiration date, and then finally on the expiration date.

Term licenses can be renewed continuously as each specified period expires. If you don't renew your term license and the term expires, Tableau will stop working and you will no longer have access to the software.

Note: Tableau trial licenses expire at the end of the trial period.

For information about renewing your license, contact your Tableau Account Manager or Partner.

Functions, Operators, & Data Types

This section explains how to use and combine the various data types supported by Tableau. In addition, this section discusses how to format and use the building blocks of formulas in Tableau. These parts include literal expressions, functions, and operators.

All of these features are important to understand when you create custom fields such as calculations.

Data Types

Tableau supports string, date/datetime, number, and boolean data types. These data types are automatically handled in the proper fashion. However, if you create calculated fields of your

own, you need to be aware of how to use and combine the different data types in formulas. For example, you cannot add a string to a number. Also, many functions that are available to you when you define a calculation only work when they are applied to specific data types. For example, the DATEPART() function can accept only a date/datetime data type as an argument. So, you can write DATEPART('year', #April 15, 2004#) and expect a valid result: 2004. You cannot write DATEPART('year', "Tom Sawyer") and expect a valid result. In fact, this example returns an error because "Tom Sawyer" is a string, not a date/datetime.

Although Tableau will attempt to fully validate all calculations, some data type errors cannot be found until the query is run against the database. These issues appear as error dialogs at the time of the query rather than in the calculation dialog box.

The data types supported by Tableau are described below. Refer to [Type Conversion](#) on page 1569 to learn about converting from one data type to another.

STRING

A sequence of zero or more characters. For example, "Wisconsin", "ID-44400", and "Tom Sawyer" are all strings. Strings are recognized by single or double quotes. The quote character itself can be included in a string by repeating it. For example, 'O"Hanrahan'.

DATE/DATETIME

A date or a datetime. For example "January 23, 1972" or "January 23, 1972 12:32:00 AM". If you would like a date written in long-hand style to be interpreted as a date/datetime, place the # sign on either side of it. For instance, "January 23, 1972" is treated as a string data type but #January 23, 1972# is treated as a date/datetime data type.

NUMBER

Numerical values in Tableau can be either integers or floating-point numbers.

With floating-point numbers, results of some aggregations may not always be exactly as expected. For example, you may find that the SUM function returns a value such as -1.42e-14 for a column of numbers that you know should sum to exactly 0. This happens because the Institute of Electrical and Electronics Engineers (IEEE) 754 floating-point standard requires that numbers be stored in binary format, which means that numbers are sometimes rounded at extremely fine levels of precision. You can eliminate this potential distraction by using the ROUND function (see [Number Functions](#) on page 1552 or by formatting the number to show fewer decimal places).

Operations that test floating point values for equality can behave unpredictably for the same reason. Such comparisons can occur when using level of detail expressions as dimensions, in categorical filtering, creating ad-hoc groups, creating IN/OUT sets, and with data blending.

Note: The largest signed 64-bit integer is 9,223,372,036,854,775,807. When connecting to a new data source, any column with data type set to **Number (Whole)**, can accommodate values up to this limit; for larger values, Tableau will use floating point.

BOOLEAN

A field that contains the values `TRUE` or `FALSE`. An unknown value arises when the result of a comparison is unknown. For example, the expression `7 > Null` yields `unknown`. Unknown booleans are automatically converted to `Null`.

Formatting Literals

When you are using functions you will sometimes want to use literal expressions to represent numbers, strings, dates, and more. A literal expression signifies a constant value that is represented “as is.” For example, you may have a function where your input is a date. Rather than just type “May 1, 2005”, which would be interpreted as a string, you would type `#May 1, 2005#`, which is equivalent to using a date function to convert the argument from a string to a date (refer to [Date Functions on page 1565](#)). You can use numeric, string, date, boolean, and `Null` literals. The way to format each of these literals is described below.

Numeric Literals

A numeric literal is written exactly like you usually write numbers. If you want to input the number one as a numeric literal you would type `1`. Subsequently, if you want to input the number `3.1415` as a numeric literal you would type `3.1415`.

String Literals

A string literal can be written either using single quotations or double quotations. If your string has a single or double quotation within it, simply type the symbol twice. For example, if you wanted to input the string “cat” as a string literal you could type `'cat'` or `"cat"`. Additionally, if you want to type the string “She’s my friend.” as a string literal you could type `'She''s my friend.'` or `"She's my friend."`

Date Literals

Date literals are signified by the pound symbol (`#`). If you wanted to input the date “August 22, 2005” as a literal date you would type `#August 22, 2005#`.

Boolean Literals

Boolean literals are written as either true or false. If you wanted to input “true” as a boolean literal you would type `true`.

Null Literals

Null literals are written simply as Null. If you wanted to input “Null” as a Null literal you would type Null.

Functions

The calculation functions are grouped into categories. These are the same categories used in the calculation editor. The aggregate functions such as sum, average, and so on are described in [Aggregations on page 329](#).

For information on calculations, see [Calculated Fields on page 822](#).

Number Functions

ABS(number)

Returns the absolute value of the given number.

Examples

ABS (-7) = 7

ABS ([Budget Variance])

The second example returns the absolute value for all the numbers contained in the Budget Variance field.

ACOS(number)

Returns the arc cosine of the given number. The result is in radians.

Example

ACOS (-1) = 3.14159265358979

ASIN(number)

Returns the arc sine of a given number. The result is in radians.

Example

ASIN (1) = 1.5707963267949

ATAN(number)

Returns the arc tangent of a given number. The result is in radians.

Example

ATAN (180) = 1.5652408283942

ATAN2(y number, x number)

Returns the arc tangent of two given numbers (x and y). The result is in radians.

Example

ATAN2(2, 1) = 1.10714871779409

CEILING(number)

Rounds a number to the nearest integer of equal or greater value.

Example

CEILING(3.1415) = 4

Availability by data source

Data Source	Support
Microsoft Access	Not supported
Microsoft Excel	Supported
Text File	Supported
Statistical File	Supported
Tableau Server	Supported
Actian Vectorwise	Not supported
Amazon Aurora	Not supported
Amazon EMR	Supported
Amazon Redshift	Not supported
Aster Database	Not supported
Cloudera Hadoop	Supported
DataStax Enterprise	Supported
EXASolution	Not supported
Firebird	Not supported
Google Analytics	Supported
Google BigQuery	Supported

Google Cloud SQL	Not supported
Hortonworks Hadoop Hive	Supported
HP Vertica	Not supported
IBM BigInsights	Not supported
IBM DB2	Not supported
IBM Netezza	Not supported
MapR Hadoop Hive	Supported
MarkLogic	Not supported
Microsoft Analysis Services	Not supported
Microsoft PowerPivot	Not supported
Microsoft SQL Server	Not supported
MySQL	Not supported
Oracle	Not supported
Oracle Essbase	Not supported
ParAccel	Not supported
Pivotal Greenplum	Not supported
PostgreSQL	Not supported
Progress OpenEdge	Not supported
Salesforce	Supported
SAP HANA	Not supported
SAP Sybase ASE	Not supported
SAP Sybase IQ	Not supported
Spark SQL	Supported

Splunk	Not supported
Teradata	Not supported
Teradata OLAP Connector	Not supported

COS(number)

Returns the cosine of an angle. Specify the angle in radians.

Example

```
COS(PI() / 4) = 0.707106781186548
```

COT(number)

Returns the cotangent of an angle. Specify the angle in radians.

Example

```
COT(PI() / 4) = 1
```

DEGREES(number)

Converts a given number in radians to degrees.

Example

```
DEGREES(PI() / 4) = 45.0
```

DIV(integer1, integer2)

Returns the integer part of a division operation, in which integer1 is divided by integer2.

Example

```
DIV(11,2) = 5
```

EXP(number)

Returns e raised to the power of the given number.

Examples

```
EXP(2) = 7.389
```

```
EXP(-[Growth Rate]*[Time])
```

FLOOR(number)

Rounds a number to the nearest integer of equal or lesser value.

Example

FLOOR (3.1415) = 3

Availability by data source

Data Source	Support
Microsoft Access	Not supported
Microsoft Excel	Supported
Text File	Supported
Statistical File	Supported
Tableau Server	Supported
Actian Vectorwise	Not supported
Amazon Aurora	Not supported
Amazon EMR	Supported
Amazon Redshift	Not supported
Aster Database	Not supported
Cloudera Hadoop	Supported
DataStax Enterprise	Supported
EXASolution	Not supported
Firebird	Not supported
Google Analytics	Supported
Google BigQuery	Supported
Google Cloud SQL	Not supported
Hortonworks Hadoop Hive	Supported
HP Vertica	Not supported

IBM BigInsights	Not supported
IBM DB2	Not supported
IBM Netezza	Not supported
MapR Hadoop Hive	Supported
MarkLogic	Not supported
Microsoft Analysis Services	Not supported
Microsoft PowerPivot	Not supported
Microsoft SQL Server	Not supported
MySQL	Not supported
Oracle	Not supported
Oracle Essbase	Not supported
ParAccel	Not supported
Pivotal Greenplum	Not supported
PostgreSQL	Not supported
Progress OpenEdge	Not supported
Salesforce	Supported
SAP HANA	Not supported
SAP Sybase ASE	Not supported
SAP Sybase IQ	Not supported
Spark SQL	Supported
Splunk	Not supported
Teradata	Not supported
Teradata OLAP Connector	Not supported

HEXBINX(number, number)

Maps an x, y coordinate to the x-coordinate of the nearest hexagonal bin. The bins have side length 1, so the inputs may need to be scaled appropriately.

HEXBINX and HEXBINY are binning and plotting functions for hexagonal bins. Hexagonal bins are an efficient and elegant option for visualizing data in an x/y plane such as a map. Because the bins are hexagonal, each bin closely approximates a circle and minimizes variation in the distance from the data point to the center of the bin. This makes the clustering both more accurate and informative.

Example

```
HEXBINX([Longitude], [Latitude])
```

HEXBINY(number, number)

Maps an x, y coordinate to the y-coordinate of the nearest hexagonal bin. The bins have side length 1, so the inputs may need to be scaled appropriately.

Example

```
HEXBINY([Longitude], [Latitude])
```

LN(number)

Returns the natural logarithm of a number. Returns `Null` if number is less than or equal to 0.

LOG(number [, base])

Returns the logarithm of a number for the given base. If the base value is omitted, base 10 is used.

MAX(number, number)

Returns the maximum of the two arguments, which must be of the same type. Returns `Null` if either argument is `Null`. MAX can also be applied to a single field in an aggregate calculation.

Examples

```
MAX(4, 7)
```

```
MAX(Sales, Profit)
```

```
MAX([First Name], [Last Name])
```

MIN(number, number)

Returns the minimum of the two arguments, which must be of the same type. Returns `Null` if either argument is `Null`. MIN can also be applied to a single field in an aggregate calculation.

Examples

```
MIN(4, 7)
```

`MIN(Sales, Profit)`
`MIN([First Name], [Last Name])`

`PI()`

Returns the numeric constant pi: 3.14159.

`POWER(number, power)`

Raises the number to the specified power.

Examples

`POWER(5, 2) = 5^2 = 25`
`POWER(Temperature, 2)`

You can also use the `^` symbol:

`5^2 = POWER(5, 2) = 25`

`Radians(number)`

Converts the given number from degrees to radians.

Example

`RADIANS(180) = 3.14159`

`ROUND(number, [decimals])`

Rounds numbers to a specified number of digits. The `decimals` argument specifies how many decimal points of precision to include in the final result. If `decimals` is omitted, number is rounded to the nearest integer.

Example

This example rounds every Sales value to an integer:

`ROUND(Sales)`

Some databases, such as SQL Server, allow specification of a negative `length`, where -1 rounds number to 10's, -2 rounds to 100's, and so on. This is not true of all databases. For example, it is not true of Excel or Access.

`SIGN(number)`

Returns the sign of a number: The possible return values are -1 if the number is negative, 0 if the number is zero, or 1 if the number is positive.

Example

If the average of the profit field is negative, then

`SIGN(AVG(Profit)) = -1`

SIN(number)

Returns the sine of an angle. Specify the angle in radians.

Example

SIN(0) = 1.0

SIN(PI()/4) = 0.707106781186548

SQRT(number)

Returns the square root of a number.

Example

SQRT(25) = 5

SQUARE(number)

Returns the square of a number.

Example

SQUARE(5) = 25

TAN(number)

Returns the tangent of an angle. Specify the angle in radians..

Example

TAN(PI()/4) = 1.0

ZN(expression)

Returns the expression if it is not null, otherwise returns zero. Use this function to use zero values instead of null values.

Example

ZN([Profit]) = [Profit]

String Functions

ASCII(string)

Return the ASCII code for the first character of string.

Example

ASCII('A') = 65

CHAR(number)

Returns the character encoded by the ASCII code number.

Example

CHAR(65) = 'A'

Contains(string, substring)

Returns true if the given string contains the specified substring.

Example

CONTAINS("Calculation", "alcu") = true

ENDSWITH(string, substring)

Returns true if the given string ends with the specified substring. Trailing white spaces are ignored.

Example

ENDSWITH("Tableau", "leau") = true

FIND(string, substring, [start])

Returns the index position of substring in string, or 0 if the substring isn't found. If the optional argument start is added, the function ignores any instances of substring that appear before the index position start. The first character in the string is position 1.

Examples

FIND("Calculation", "alcu") = 2

FIND("Calculation", "Computer") = 0

FIND("Calculation", "a", 3) = 7

FIND("Calculation", "a", 2) = 2

FIND("Calculation", "a", 8) = 0

FINDNTH(string, substring, occurrence)

Returns the position of the nth occurrence of substring within the specified string, where n is defined by the occurrence argument.

Example

FINDNTH("Calculation", "a", 2) = 7

LEFT(string, number)

Returns the left-most number of characters in the string.

Example

LEFT("Matador", 4) = "Mata"

LEN(string)

Returns the length of the string.

Example

```
LEN ("Matador") = 7
```

LOWER(string)

Returns `string`, with all characters lowercase.

Example

```
LOWER ("ProductVersion") = "productversion"
```

LTRIM(string)

Returns the string with any leading spaces removed.

Example

```
LTRIM(" Matador ") = "Matador "
```

MAX(a, b)

Returns the maximum of `a` and `b` (which must be of the same type). This function is usually used to compare numbers, but also works on strings. With strings, `MAX` finds the value that is highest in the sort sequence defined by the database for that column. It returns `Null` if either argument is `Null`.

Example

```
MAX ("Apple", "Banana") = "Banana"
```

MID(string, start, [length])

Returns the string starting at index position `start`. The first character in the string is position 1. If the optional argument `length` is added, the returned string includes only that number of characters.

Examples

```
MID("Calculation", 2) = "alculation"
```

```
MID("Calculation", 2, 5) ="alcul"
```

MIN(a, b)

Returns the minimum of `a` and `b` (which must be of the same type). This function is usually used to compare numbers, but also works on strings. With strings, `MIN` finds the value that is lowest in the sort sequence. It returns `Null` if either argument is `Null`.

Example

```
MIN ("Apple", "Banana") = "Apple"
```

REPLACE(string, substring, replacement)

Searches string for substring and replaces it with replacement. If substring is not found, the string is not changed.

Example

```
REPLACE("Version8.5", "8.5", "9.0") = "Version9.0"
```

RIGHT(string, number)

Returns the right-most number of characters in string.

Example

```
RIGHT("Calculation", 4) = "tion"
```

RTRIM(string)

Returns string with any trailing spaces removed.

Example

```
RTRIM(" Calculation ") = " Calculation"
```

SPACE(number)

Returns a string that is composed of the specified number of repeated spaces.

Example

```
SPACE(1) = " "
```

SPLIT(string, delimiter, token number)

Returns a substring from a string, using a delimiter character to divide the string into a sequence of tokens.

The string is interpreted as an alternating sequence of delimiters and tokens. So for the string abc-defgh-i-jkl, where the delimiter character is ‘-‘, the tokens are abc, defgh, i, and jlk. Think of these as tokens 1 through 4. SPLIT returns the token corresponding to the token number. When the token number is positive, tokens are counted starting from the left end of the string; when the token number is negative, tokens are counted starting from the right.

Examples

```
SPLIT ('a-b-c-d', '- ', 2) = 'b'
```

```
SPLIT ('a|b|c|d', '| ', -2) = 'c'
```

Some data sources impose limits on splitting string. The following table shows which data sources support negative token numbers (splitting from the right) and whether there is a limit on the number of splits allow per data source. A SPLIT function that specifies a negative token number and would be legal with other data sources will return this error with these data sources: "Splitting from right is not support by the data source."

Data Source	Left/Right Constraints	Maximum Number of Splits	Version Limitations
Tableau Data Extract	Both	Infinite	
Microsoft Excel	Both	Infinite	
Text file	Both	Infinite	
Salesforce	Both	Infinite	
OData	Both	Infinite	
Google Analytics	Both	Infinite	
Tableau Data Server	Both	Infinite	Supported in version 9.0.
HP Vertica	Left only	10	
Oracle	Left only	10	
MySQL	Both	10	
PostgreSQL	Left only prior to version 9.0; both for version 9.0 and above	10	
Teradata	Left only	10	Version 14 and later
Amazon Redshift	Left only	10	
Aster Data-	Left only	10	

base			
Google BigQuery	Left only	10	
Hortonworks Hadoop Hive	Left only	10	
Cloudera Hadoop	Left only	10	Impala supported starting in version 2.3.0.
Microsoft SQL Server	Both	10	2008 and later

STARTSWITH(string, substring)

Returns true if string starts with substring. Leading white spaces are ignored.

Example

```
STARTSWITH("Joker", "Jo") = true
```

TRIM(string)

Returns the string with leading and trailing spaces removed. For example, TRIM (" Calculation ") = "Calculation"

UPPER(string)

Returns string, with all characters uppercase.

Example

```
UPPER("Calculation") = "CALCULATION"
```

Date Functions

Tableau provides a variety of date functions. Many of the examples use the # symbol with date expressions. See [Formatting Literals on page 1551](#) for an explanation of this symbol.

Additionally, many date functions use date_part, which is a constant string argument. The valid date_part values that you can use are:

date_part	Values
'year'	Four-digit year
'quarter'	1-4

date_part	Values
'month'	1-12 or "January", "February", and so on
'dayofyear'	Day of the year; Jan 1 is 1, Feb 1 is 32, and so on
'day'	1-31
'weekday'	1-7 or "Sunday", "Monday", and so on
'week'	1-52
'hour'	0-23
'minute'	0-59
'second'	0-60

Note: Date functions do not take account of the configured fiscal year start. See [.Fiscal Dates](#) on page 623.

[DATEADD\(date_part, interval, date\)](#)

Returns the specified date with the specified number `interval` added to the specified `date_part` of that date.

Example

```
DATEADD('month', 3, #2004-04-15#) = 2004-07-15 12:00:00 AM
```

This expression adds three months to the date #2004-04-15#.

[DATEDIFF\(date_part, date1, date2, \[start_of_week\]\)](#)

Returns the difference between `date1` and `date2` expressed in units of `date_part`.

The `start_of_week` parameter is optional. If it is omitted, the start of week is determined by the data source. See [Date Properties for a Data Source](#) on page 614.

Example

```
DATEDIFF('week', #2013-09-22#, #2013-09-24#, 'monday')= 1
```

```
DATEDIFF('week', #2013-09-22#, #2013-09-24#, 'sunday')= 0
```

The first expression returns 1 because when `start_of_week` is 'monday', then 22 September (a Sunday) and 24 September (a Tuesday) are in different weeks. The second expression returns 0 because when `start_of_week` is 'sunday' then 22 September (a Sunday) and 24 September (a Tuesday) are in the same week.

DATENAME(date_part, date, [start_of_week])

Returns `date_part` of `date` as a string. The `start_of_week` parameter is optional.

If it is omitted, the start of week is determined by the data source. See [Date Properties for a Data Source](#) on page 614.

Examples

```
DATENAME ('year', #2004-04-15#) = "2004"
```

```
DATENAME ('month', #2004-04-15#) = "April"
```

DATEPART(date_part, date, [start_of_week])

Returns `date_part` of `date` as an integer.

The `start_of_week` parameter is optional. If it is omitted, the start of week is determined by the data source. See [Date Properties for a Data Source](#) on page 614.

Note: When the `date_part` is weekday, the `start_of_week` parameter is ignored. This is because Tableau relies on a fixed weekday ordering to apply offsets.

Examples

```
DATEPART ('year', #2004-04-15#) = 2004
```

```
DATEPART ('month', #2004-04-15#) = 4
```

DATETRUNC(date_part, date, [start_of_week])

Truncates the specified date to the accuracy specified by the `date_part`. This function returns a new date. For example, when you truncate a date that is in the middle of the month at the month level, this function returns the first day of the month. The `start_of_week` parameter is optional. If it is omitted, the start of week is determined by the data source. See [Date Properties for a Data Source](#) on page 614.

Examples

```
DATETRUNC ('quarter', #2004-08-15#) = 2004-07-01 12:00:00 AM
```

```
DATETRUNC ('month', #2004-04-15#) = 2004-04-01 12:00:00 AM
```

DAY(date)

Returns the day of the given date as an integer.

Example

```
DAY (#2004-04-12#) = 12
```

ISDATE(string)

Returns true if a given string is a valid date.

Example

```
ISDATE("April 15, 2004") = true
```

MAKEDATE(year, month, day)

Returns a date value constructed from the specified year, month, and date.

Available for Tableau Data Extracts. Check for availability in other data sources.

Example

```
MAKEDATE(2004, 4, 15) = #April 15, 2004#
```

MAKEDATETIME(date, time)

Returns a datetime that combines a date and a time. The date can be a date, datetime, or a string type. The time must be a datetime. This function is available only for MySQL connections.

Examples

```
MAKEDATETIME("1899-12-30", #07:59:00#) = #12/30/1899 7:59:00 AM#
```

```
MAKEDATETIME([Date], [Time]) = #1/1/2001 6:00:00 AM#
```

MAKETIME(hour, minute, second)

Returns a date value constructed from the specified hour, minute, and second.

Available for Tableau Data Extracts. Check for availability in other data sources.

Example

```
MAKETIME(14, 52, 40) = #14:52:40#
```

MAX(expression) or MAX(expr1, expr2)

Usually applied to numbers but also works on dates. Returns the maximum of *a* and *b* (*a* and *b* must be of the same type). Returns Null if either argument is Null.

Examples

```
MAX(#2004-01-01#, #2004-03-01#) = 2004-03-01 12:00:00 AM
```

```
MAX([ShipDate1], [ShipDate2])
```

MIN(expression) or MIN(expr1, expr2)

Usually applied to numbers but also works on dates. Returns the minimum of *a* and *b* (*a* and *b* must be of the same type). Returns Null if either argument is Null.

Examples

```
MIN(#2004-01-01#, #2004-03-01#) = 2004-01-01 12:00:00 AM
```

```
MIN([ShipDate1], [ShipDate2])
```

MONTH(date)

Returns the month of the given date as an integer.

Example

```
MONTH (#2004-04-15#) = 4
```

NOW()

Returns the current date and time.

The return varies depending on the nature of the connection:

- For a live, unpublished connection, NOW returns the data source server time.
- For a live, published connection, NOW returns the data source server time.
- For an unpublished extract, NOW returns the local system time.
- For a published extract, NOW returns the local time of the Tableau Server Data Engine.
When there are multiple worker machines in different time zones, this can produce inconsistent results.

Example

```
NOW( ) = 2004-04-15 1:08:21 PM
```

TODAY()

Returns the current date.

Example

```
TODAY( ) = 2004-04-15
```

YEAR (date)

Returns the year of the given date as an integer.

Example

```
YEAR (#2004-15#) = 2004
```

Type Conversion

The result of any expression in a calculation can be converted to a specific data type. The conversion functions are STR(), DATE(), DATETIME(), INT(), and FLOAT(). For example, if you want to cast a floating point number like 3.14 as an integer, you could write INT(3.14). The result would be 3, which is an integer. The casting functions are described below.

A boolean can be cast to an integer, float, or string. It cannot be cast to a date. True is 1, 1.0, or "1", while False is 0, 0.0 or "0". Unknown maps to Null.

DATE(expression)

Returns a date given a number, string, or date expression.

Examples

```
DATE([Employee Start Date])  
DATE("April 15, 2004") = #April 15, 2004#  
DATE("4/15/2004")  
DATE(#2006-06-15 14:52#) = #2006-06-15#
```

Quotation marks are required in the second and third examples.

DATETIME(expression)

Returns a datetime given a number, string, or date expression.

Example

```
DATETIME("April 15, 2005 07:59:00") = April 15, 2005 07:59:00
```

DATEPARSE(format, string)

Converts a string to a datetime in the specified format. Support for some locale-specific formats is determined by the computer's system settings. Letters that appear in the data and do not need to be parsed should be surrounded by single quotes (''). For formats that do not have delimiters between values (e.g., MMddyy), verify that they are parsed as expected. The format must be a constant string, not a field value. This function returns Null if the data does not match the format.

This function is available for non-legacy Microsoft Excel and text file connections, MySQL, Oracle, PostgreSQL, and Tableau data extract data sources. Some formats may not be available for all data sources.

Examples

```
DATEPARSE ("dd.MMMM.yyyy", "15.April.2004") = #April 15, 2004#  
DATEPARSE ("h'h' m'm' s's'", "10h 5m 3s") = #10:05:03#
```

For more examples, see [Convert Strings to Date Fields](#) on page 418.

FLOAT(expression)

Casts its argument as a floating point number.

Examples

```
FLOAT(3) = 3.000  
FLOAT([Age]) converts every value in the Age field to a floating point number.
```

[INT\(expression\)](#)

Casts its argument as an integer. For expressions, this function truncates results to the closest integer toward zero.

[Examples](#)

`INT(8.0/3.0) = 2`

`INT(4.0/1.5) = 2`

`INT(0.50/1.0) = 0`

`INT(-9.7) = -9`

When a string is converted to an integer it is first converted to a float and then rounded.

[STR\(expression\)](#)

Casts its argument as a string.

[Example](#)

`STR([Age])` takes all of the values in the measure called `Age` and converts them to strings.

[Logical Functions](#)

[Function expressions documented in this topic:](#)

- [CASE expression WHEN value1 THEN return1 WHEN value2 THEN return2...ELSE default return END below](#)
- [IIF\(test, then, else, \[unknown\]\)](#) on the next page
- [IF test THEN value END / IF test THEN value ELSE else END](#) on the next page
- [IF test1 THEN value1 ELSEIF test2 THEN value2 ELSE else END](#) on page 1573
- [IFNULL\(expression1, expression2\)](#) on page 1575
- [ISNULL\(expression\)](#) on page 1576

For documentation of the MIN, MAX, and ZN functions, see [Number Functions](#) on page 1552.

[Also:](#)

- [Additional use cases for logical functions](#) on page 1576

`CASE expression WHEN value1 THEN return1 WHEN value2 THEN return2...ELSE default return END`

Use the CASE function to perform logical tests and return appropriate values. CASE is often easier to use than IIF or IF THEN ELSE. The CASE function evaluates `expression`, compares it to a sequence of values, `value1, value2, etc.`, and returns a result. When a

value that matches `expression` is encountered, CASE returns the corresponding return value. If no match is found, the default return expression is used. If there is no default return and no values match, then Null is returned.

Typically, you use an IF function to perform a sequence of arbitrary tests, and you use a CASE function to search for a match to an expression. But a CASE function can always be rewritten as an IF function , although the CASE function will generally be more concise.

Many times you can use a group to get the same results as a complicated case function.

Examples

```
CASE [Region] WHEN "West" THEN 1 WHEN "East" THEN 2 ELSE 3 END  
CASE LEFT(DATENAME('weekday',[Order Date]),3) WHEN "Sun" THEN 0  
WHEN "Mon" THEN 1 WHEN "Tue" THEN 2 WHEN "Wed" THEN 3 WHEN "Thu"  
THEN 4 WHEN "Fri" THEN 5 WHEN "Sat" THEN 6 END
```

IIF(test, then, else, [unknown])

Use the IIF function to perform logical tests and return appropriate values. The first argument, `test`, must be a boolean: either a boolean field in the data source, or the result of a logical expression using operators (or a logical comparison of AND, OR, or NOT). If `test` evaluates to TRUE, then IIF returns the `then` value. If `test` evaluates to FALSE, then IIF returns the `else` value.

A boolean comparison may also yield the value UNKNOWN (neither TRUE nor FALSE), usually due to the presence of Null values in `test`. The final argument to IIF is returned in the event of an UNKNOWN result for the comparison. If this argument is left out, Null is returned.

Examples

```
IIF(7>5, "Seven is greater than five", "Seven is less than five")  
IIF([Cost]>[Budget Cost], "Over Budget", "Under Budget")  
IIF([Budget Sales]!=0, [Sales]/[Budget Sales],0)  
IIF(Sales>=[Budget Sales], "Over Cost Budget and Over Sales  
Budget", "Over Cost Budget and Under Sales Budget","Under Cost  
Budget")
```

IF test THEN value END / IF test THEN value ELSE else END

Use the IF THEN ELSE function to perform logical tests and return appropriate values. The IF THEN ELSE function evaluates a sequence of test conditions and returns the value for the first condition that is true. If no condition is true, the ELSE value is returned. Each test must be a boolean: either be a boolean field in the data source or the result of a logical expression. The final ELSE is optional, but if it is not provided and there is no true test expression, then the function returns Null. All of the value expressions must be of the same type.

Examples

```
IF [Cost]>[Budget Cost] THEN "Over Budget" ELSE "Under Budget"  
END  
  
IF [Budget Sales]!=0 THEN [Sales]/[Budget Sales] END  
  
IF test1 THEN value1 ELSEIF test2 THEN value2 ELSE else END
```

Use this version of the IF function to perform logical tests recursively. There is no built-in limit to the number of ELSEIFvalues you can use with an IF function, though individual databases may impose a limit on IF function complexity. While an IF function can be rewritten as a series of nested IIF statements, there are differences in how the expressions will be evaluated. In particular, an IIF statement distinguishes TRUE, FALSE and UNKNOWN, whereas an IF statement only worries about TRUE and not true (which includes both FALSE and UNKNOWN).

Example

When you create bins from a measure, Tableau creates bins of equal size by default. For example, say you have a measure that represents age. When you create bins from that measure, Tableau makes all the bins of equal size. You can specify how big you want the bins to be, but you cannot specify a separate range of values for each bin. A way around this constraint is to create a calculated field to define bins. Then you can create one bin for ages 0 - 20, another for ages 21 - 32, and so on. The following procedure shows how you could do that.

1. Create a new calculated field by choosing **Analysis > Create Calculated Field**.
2. Name the field **Age Groups** and type the following in the definition area

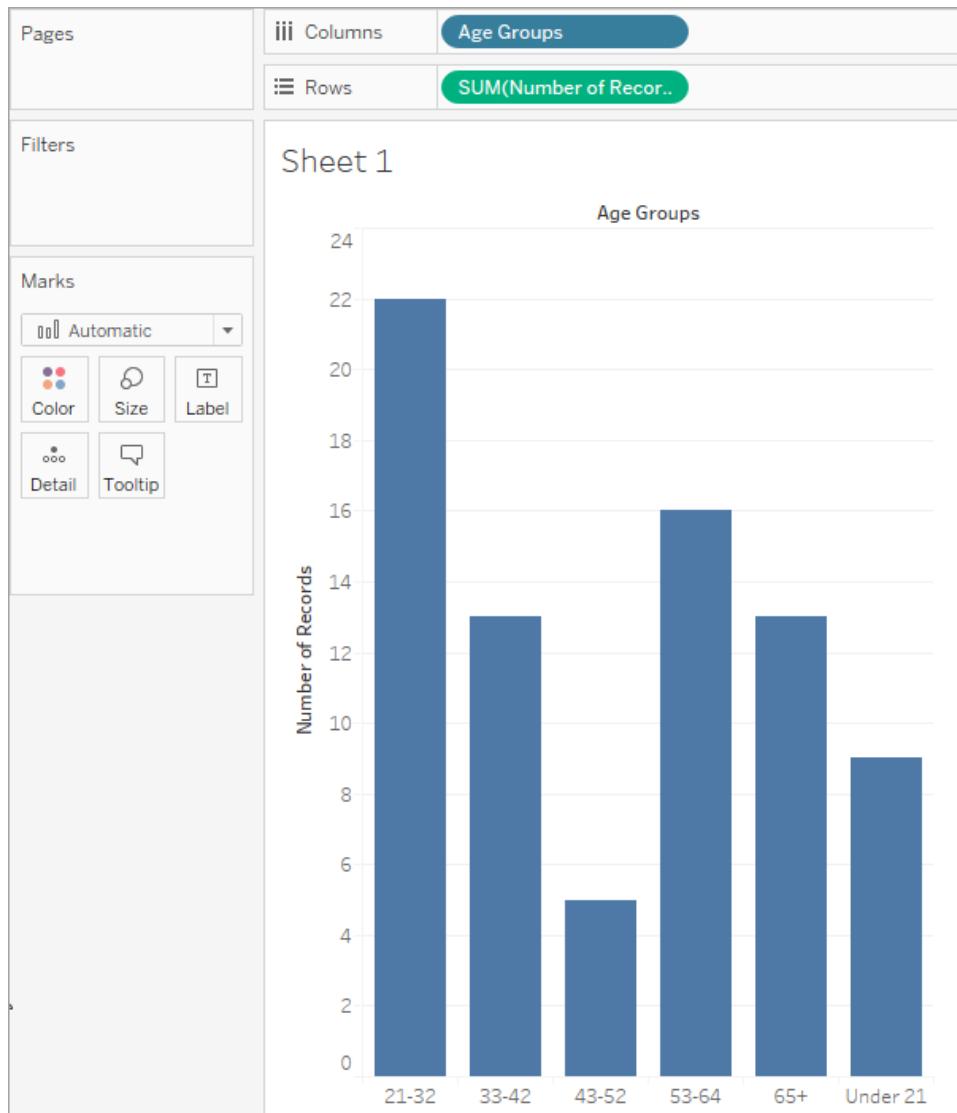
```
IF  
  
[Age] < 21 THEN "Under 21"  
  
ELSEIF  
  
[Age] <= 32 THEN "21-32"  
  
ELSEIF  
  
[Age] <= 42 THEN "33-42"  
  
ELSEIF  
  
[Age] <= 52 THEN "43-52"  
  
ELSEIF  
  
[Age] <= 64 THEN "53-64"  
  
ELSE "65+"
```

END

Confirm that the status message indicates that the formula is valid, and then click **OK**.

3. From the Measures area of the **Data** pane, drag **Number of Records** to **Rows**.
4. From the Dimensions area of the **Data** pane, drag **Age Groups** to **Columns**.

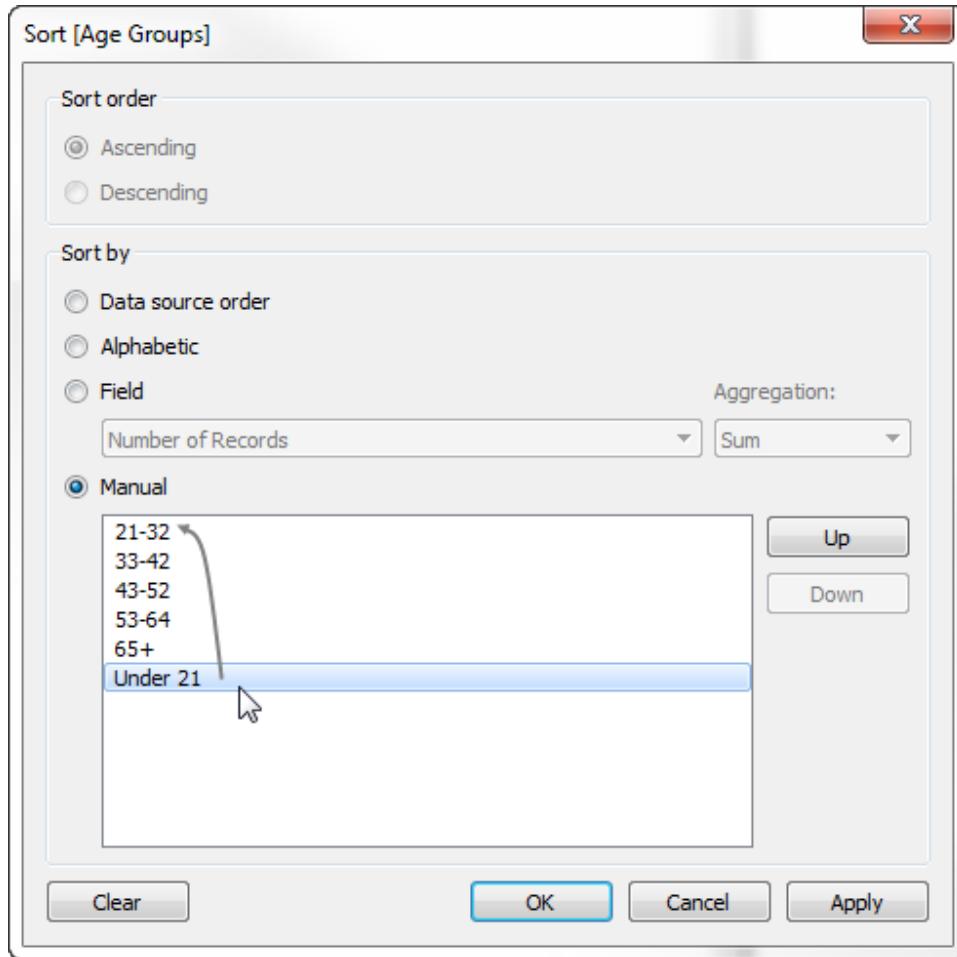
The records are now divided among the six bins that you defined:



Unfortunately, the **Under 21** bin is at far right, when you would expect it to be at far left. Tableau's smart enough to put the bins with entirely numerical names in the right order,

but it can't guess that the bin name beginning with "Under" belongs at the left. Fix the problem with a manual sort.

5. Click the down arrow at the right side of the Age Groups field on Columns and then click Sort. Choose Manual and then move the Under 21 bin up to the top of the list:



Your view is now complete.

IFNULL(expression1, expression2)

The IFNULL function returns the first expression if the result is not null, and returns the second expression if it is null.

Example

IFNULL([Profit], 0) = [Profit]

ISDATE(string)

The ISDATE function returns TRUE if the string argument can be converted to a date and FALSE if it cannot.

Examples

```
ISDATE("January 1, 2003") = TRUE  
ISDATE("Jan 1 2003") = TRUE  
ISDATE("1/1/03") = TRUE  
ISDATE("Janxx 1 2003") = FALSE
```

ISNULL(expression)

The ISNULL function returns TRUE if the expression is Null and FALSE if it is not.

Example

The following example uses ISNULL in combination with IIF to replace null values with 0's.

```
IIF(ISNULL([Sales]), 0, [Sales] )
```

Additional use cases for logical functions

Use logical functions to create specific views from a data source without changing the source itself. For example, combine different members of a dimension, or filter a segment out of a view. Below are a few basic examples, using the IF statement with different types of logical operator. For more information about logical operators, see [Operators on page 1609](#).

Learn how to use logical functions to group members of a field, compare values, exclude values, and create virtual bins:

Group members of a field

In an OR statement, only one of the conditions in the statement needs to be true for the entire calculation to evaluate to true.

In this example, an **OR** operator is used to create a new region called East Coast, made up of members of the East and South regions:

```
IF [Region] = "East" OR [Region] = "South" THEN "East Coast" ELSE  
[Region] END
```

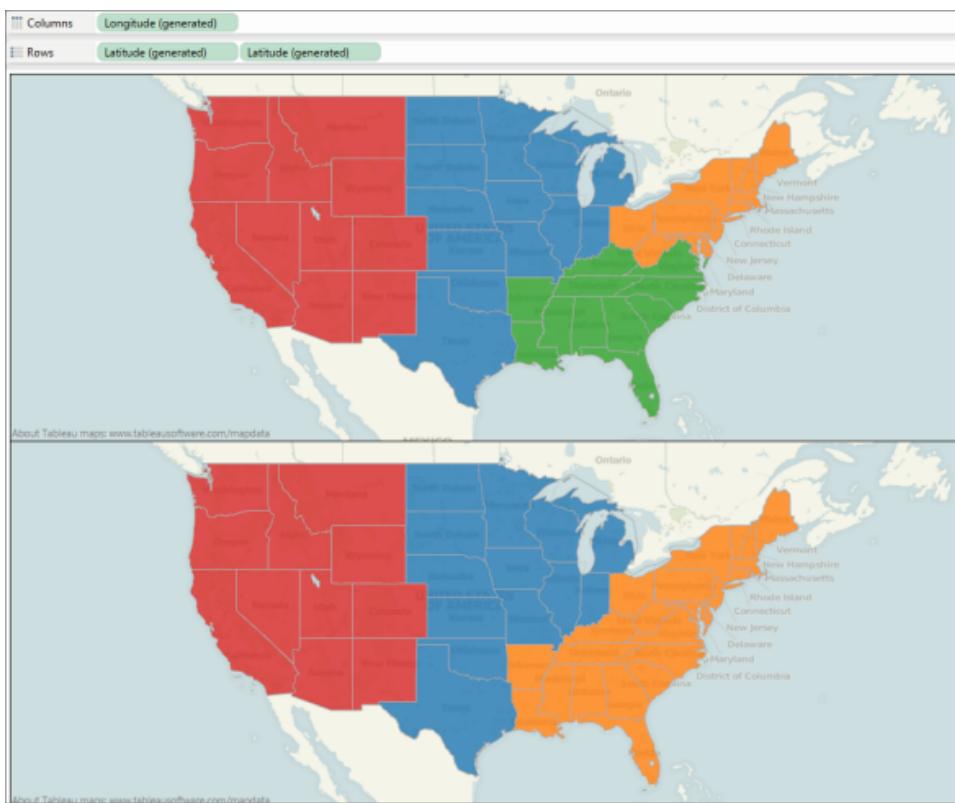
New East Coast Region Sample - Superstore X

```
If [Region] = "East" OR [Region] = "South"
THEN "East Coast"
ELSE [Region] END
```

The calculation is valid.

Apply OK

Using this calculation, your view will update to look like this:



Compare values

Logical statements that use the **AND** operator are useful when more than one field needs to be involved. In an AND statement, all conditions must evaluate to true for the overall calculation to be true.

The following calculation finds furniture sales for the Central region:

```
IF [Region] = "Central" AND [Product Category] = "Furniture" THEN
[Sales] END
```

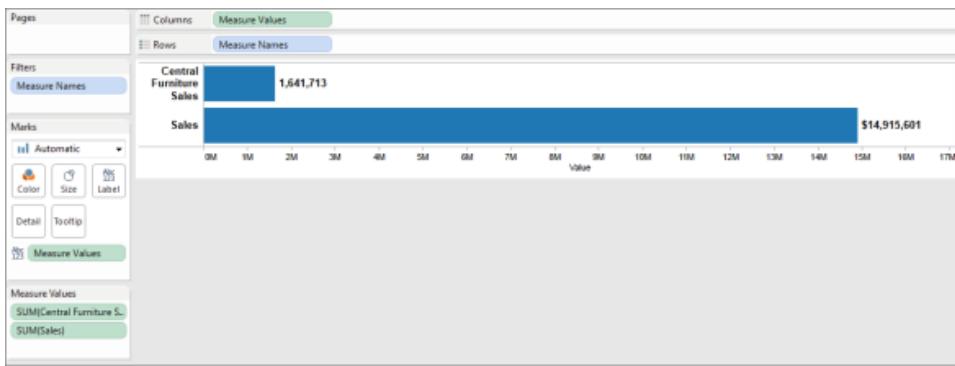
Central Furniture Sales Sample - Superstore X

```
IF [Region] = "Central"
AND [Category] = "Furniture"
THEN [Sales] End
```

The calculation is valid.

Apply OK

Use this calculation in a view to easily show how furniture sales in the Central region compare to total sales.



Exclude values

The calculation in this example uses the **not equal to** operator (\neq). This operator is often used to exclude values or filter a member of a dimension out of a view.

The following formula calculates the total sales for all regions except East:

```
IF Region <> "East" THEN [Sales] END
```

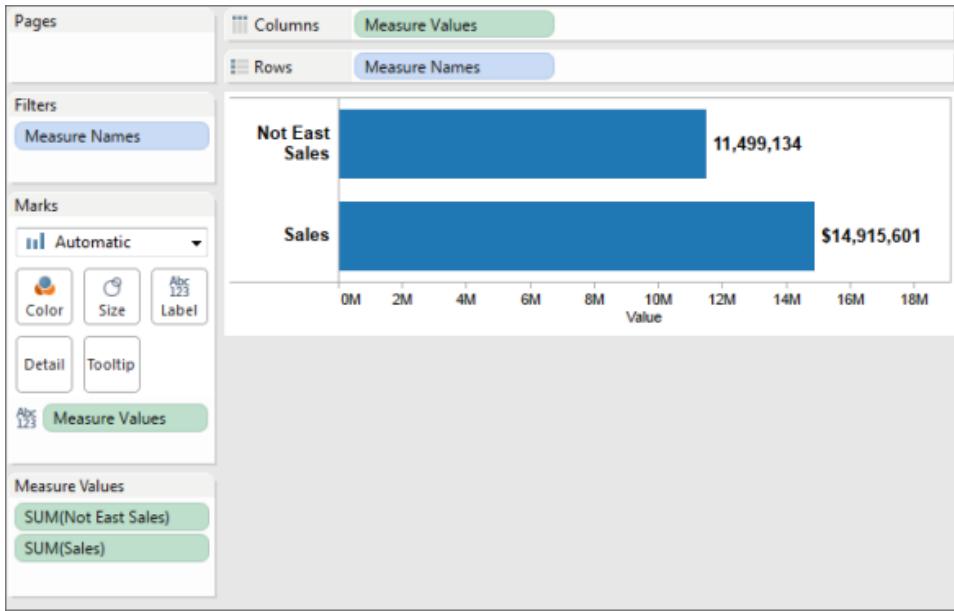
Not East Sales Sample - Superstore X

```
IF [Region] <> "East" THEN [Sales] END
```

The calculation is valid.

Apply OK

Using this calculation, your view will update to look like this:



Create virtual bins

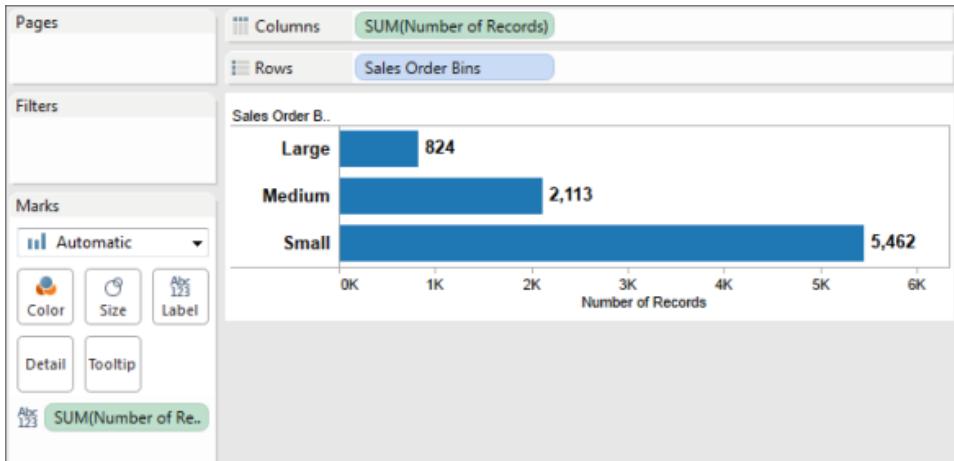
The **greater than or equal to** operator (\geq) can be used to create virtual bins to categorize data in different ways. This type of calculation is very useful and is similar to how KPIs (Key Performance Indicators) are evaluated. In this calculation, the \geq operator is used to group sales into large, medium, and small orders based on the dollar amount.

```
IF [Sales] >= 5000 THEN "Large Order"
ELSEIF [Sales] >= 1000 THEN "Medium Order"
ELSE "Small Order" END
```

```
Sales Order Bins | Sample - Superstore | X
IF [Sales] >= 5000 THEN "Large"
ELSEIF [Sales] >= 1000 THEN "Medium"
ELSE "Small"
END

The calculation is valid. | Apply | OK
```

Using this calculation, your view will update to look like this:



Aggregate Functions

Aggregations and floating-point arithmetic: The results of some aggregations may not always be exactly as expected. For example, you may find that the Sum function returns a value such as -1.42e-14 for a column of numbers that you know should sum to exactly 0. This happens because the Institute of Electrical and Electronics Engineers (IEEE) 754 floating-point standard requires that numbers be stored in binary format, which means that numbers are sometimes rounded at extremely fine levels of precision. You can eliminate this potential distraction by using the ROUND function (see [Number Functions](#) on page 1552) or by formatting the number to show fewer decimal places.

ATTR(expression)

Returns the value of the expression if it has a single value for all rows. Otherwise returns an asterisk. Null values are ignored.

AVG(expression)

Returns the average of all the values in the expression. AVG can be used with numeric fields only. Null values are ignored.

COUNT(expression)

Returns the number of items in a group. Null values are not counted.

COUNTD(expression)

Returns the number of distinct items in a group. Null values are not counted. This function is not available in the following cases: workbooks created before Tableau Desktop 8.2 that use Microsoft Excel or text file data sources, workbooks that use the legacy connection, and workbooks that use Microsoft Access data sources. Extract your data into an extract file to use this function. See [Extract Your Data](#) on page 440.

[MAX\(expression\)](#)

Returns the maximum of an expression across all records. If the expression is a string value, this function returns the last value where last is defined by alphabetical order.

[MEDIAN\(expression\)](#)

Returns the median of an expression across all records. Median can only be used with numeric fields. Null values are ignored. This function is not available for workbooks created before Tableau Desktop 8.2 or that use legacy connections. It is also not available for connections using any of the following data sources:

- Access
- Amazon Redshift
- Cloudera Hadoop
- HP Vertica
- IBM DB2
- IBM Netezza
- Microsoft Excel
- Microsoft SQL Server
- MySQL
- Teradata
- Text files

For other data source types, you can extract your data into an extract file to use this function.

See [Extract Your Data](#) on page 440.

[MIN\(expression\)](#)

Returns the minimum of an expression across all records. If the expression is a string value, this function returns the first value where first is defined by alphabetical order.

[PERCENTILE\(expression, number\)](#)

Returns the percentile value from the given expression corresponding to the specified number. The number must be between 0 and 1 (inclusive)—for example, 0.66, and must be a numeric constant.

This function is available for the following data sources.

- Non-legacy Microsoft Excel and Text File connections.
- Extracts and extract-only data source types (for example, Google Analytics, OData, or Salesforce).
- Sybase IQ 15.1 and later data sources.
- Oracle 10 and later data sources.
- Cloudera Hive and Hortonworks Hadoop Hive data sources.
- EXASolution 4.2 and later data sources.

For other data source types, you can extract your data into an extract file to use this function. See [Extract Your Data](#) on page 440.

[STDEV\(expression\)](#)

Returns the statistical standard deviation of all values in the given expression based on a sample of the population.

[STDEVP\(expression\)](#)

Returns the statistical standard deviation of all values in the given expression based on a biased population.

[SUM\(expression\)](#)

Returns the sum of all values in the expression. SUM can be used with numeric fields only. Null values are ignored.

[VAR\(expression\)](#)

Returns the statistical variance of all values in the given expression based on a sample of the population.

[VARP\(expression\)](#)

Returns the statistical variance of all values in the given expression on the entire population.

Pass-Through Functions (RAWSQL)

These RAWSQL pass-through functions can be used to send SQL expressions directly to the database, without first being interpreted by Tableau. If you have custom database functions that Tableau doesn't know about, you can use these pass-through functions to call these custom functions.

Your database usually will not understand the field names that are shown in Tableau. Because Tableau does not interpret the SQL expressions you include in the pass-through functions, using the Tableau field names in your expression may cause errors. You can use a substitution syntax to insert the correct field name or expression for a Tableau calculation into pass-through SQL. For example, if you had a function that computed the median of a set of values, you could call that function on the Tableau column [Sales] like this:

```
RAWSQLLAGG_REAL("MEDIAN(%1)", [Sales])
```

Because Tableau does not interpret the expression, you must define the aggregation. You can use the RAWSQLLAGG functions described below when you are using aggregated expressions.

RAWSQL pass-through functions will not work with published data sources.

These functions may return different results starting in Tableau Desktop 8.2 than they did in earlier versions of Tableau Desktop. This is because Tableau now uses ODBC for pass-through functions instead of OLE DB. ODBC truncates when returning real values as integer; OLE DB rounds when returning real values as integer.

RAWSQL Functions

The following RAWSQL functions are available in Tableau.

RAWSQL_BOOL("sql_expr", [arg1], ...[argN])

Returns a Boolean result from a given SQL expression. The SQL expression is passed directly to the underlying database. Use %n in the SQL expression as a substitution syntax for database values.

Example

In the example, %1 is equal to [Sales] and %2 is equal to [Profit].

```
RAWSQL_BOOL("IIF( %1 > %2, True, False)", [Sales], [Profit])
```

RAWSQL_DATE("sql_expr", [arg1], ...[argN])

Returns a Date result from a given SQL expression. The SQL expression is passed directly to the underlying database. Use %n in the SQL expression as a substitution syntax for database values.

Example

In this example, %1 is equal to [Order Date].

```
RAWSQL_DATE ("%1", [Order Date])
```

RAWSQL_DATETIME("sql_expr", [arg1], ...[argN])

Returns a Date and Time result from a given SQL expression. The SQL expression is passed directly to the underlying database. Use %n in the SQL expression as a substitution syntax for database values. In this example, %1 is equal to [Delivery Date].

Example

```
RAWSQL_DATETIME("MIN(%1)", [Delivery Date])
```

RAWSQL_INT("sql_expr", [arg1], ...[argN])

Returns an integer result from a given SQL expression. The SQL expression is passed directly to the underlying database. Use %n in the SQL expression as a substitution syntax for database values. In this example, %1 is equal to [Sales].

Example

RAWSQL_INT("500 + %1", [Sales])

RAWSQL_REAL("sql_expr", [arg1], ...[argN])

Returns a numeric result from a given SQL expression that is passed directly to the underlying database. Use %n in the SQL expression as a substitution syntax for database values. In this example, %1 is equal to [Sales]

Example

RAWSQL_REAL("-123.98 * %1", [Sales])

RAWSQL_STR("sql_expr", [arg1], ...[argN])

Returns a string from a given SQL expression that is passed directly to the underlying database. Use %n in the SQL expression as a substitution syntax for database values. In this example, %1 is equal to [Customer Name].

Example

RAWSQL_STR("%1", [Customer Name])

RAWSQLLAGG_BOOL("sql_expr", [arg1], ...[argN])

Returns a Boolean result from a given aggregate SQL expression. The SQL expression is passed directly to the underlying database. Use %n in the SQL expression as a substitution syntax for database values.

Example

In the example, %1 is equal to [Sales] and %2 is equal to [Profit].

RAWSQLLAGG_BOOL("SUM(%1) > SUM(%2)", [Sales], [Profit])

RAWSQLLAGG_DATE("sql_expr", [arg1], ...[argN])

Returns a Date result from a given aggregate SQL expression. The SQL expression is passed directly to the underlying database. Use %n in the SQL expression as a substitution syntax for database values. In this example, %1 is equal to [Order Date].

Example

RAWSQLLAGG_DATE("MAX(%1)", [Order Date])

RAWSQLLAGG_DATETIME("sql_expr", [arg1], ...[argN])

Returns a Date and Time result from a given aggregate SQL expression. The SQL expression is passed directly to the underlying database. Use %n in the SQL expression as a substitution syntax for database values. In this example, %1 is equal to [Delivery Date].

Example

`RAWSQLAGG_DATETIME("MIN(%1)", [Delivery Date])`

`RAWSQLAGG_INT("sql_expr", [arg1], ..., [argN])`

Returns an integer result from a given aggregate SQL expression. The SQL expression is passed directly to the underlying database. Use %n in the SQL expression as a substitution syntax for database values. In this example, %1 is equal to [Sales].

Example

`RAWSQLAGG_INT("500 + SUM(%1)", [Sales])`

`RAWSQLAGG_REAL("sql_expr", [arg1], ..., [argN])`

Returns a numeric result from a given aggregate SQL expression that is passed directly to the underlying database. Use %n in the SQL expression as a substitution syntax for database values. In this example, %1 is equal to [Sales]

Example

`RAWSQLAGG_REAL("SUM(%1)", [Sales])`

`RAWSQLAGG_STR("sql_expr", [arg1], ..., [argN])`

Returns a string from a given aggregate SQL expression that is passed directly to the underlying database. Use %n in the SQL expression as a substitution syntax for database values. In this example, %1 is equal to [Discount].

Example

`RAWSQLAGG_STR("AVG(%1)", [Discount])`

User Functions

Use these user functions to create user filters based on user lists in your data source. For example, assume you have a view that shows the sales performance for each employee. When you publish that view, you may want to only allow employees to see their own sales numbers. You can use the function CURRENTUSER to create a field that returns True if the username of the person signed in to the server is the same as the employee name in the view. Then when you filter the view using this calculated field, only the data for the user who is currently signed in is shown.

FULLNAME()

Returns the full name for the current user. This is the Tableau Server or Tableau Online full name when the user is signed in; otherwise the local or network full name for the Tableau Desktop user.

Example

```
[Manager]=FULLNAME( )
```

If manager Dave Hallsten is signed in, this example returns True only if the Manager field in the view contained Dave Hallsten. When used as a filter, this calculated field can be used to create a user filter that only shows data that is relevant to the person signed in to the server.

ISFULLNAME(string)

Returns true if the current user's full name matches the specified full name, or false if it does not match. This function uses the Tableau Server or Online full name when the user is signed in; otherwise it uses the local or network full name for the Tableau Desktop user.

Example

```
ISFULLNAME ("Dave Hallsten")
```

This example returns true if Dave Hallsten is the current user, otherwise it returns false.

ISMEMBEROF(string)

Returns true if the person currently using Tableau is a member of a group that matches the given string. If the person currently using Tableau is signed in, the group membership is determined by groups on Tableau Server or Tableau Online. If the person is not signed in, this function returns false.

Example

```
IF ISMEMBEROF("Sales") THEN "Sales" ELSE "Other" END
```

ISUSERNAME(string)

Returns true if the current user's username matches the specified username, or false if it does not match. This function uses the Tableau Server or Online username when the user is signed in; otherwise it uses the local or network username for the Tableau Desktop user.

Example

```
ISUSERNAME ("dhallsten")
```

This example returns true if dhallsten is the current user; otherwise it returns false.

USERDOMAIN()

Returns the domain for the current user when the user is signed on to Tableau Server. Returns the Windows domain if the Tableau Desktop user is on a domain. Otherwise this function returns a null string.

Example

```
[Manager]=USERNAME () AND [Domain]=USERDOMAIN ()
```

USERNAME()

Returns the username for the current user. This is the Tableau Server or Tableau Online username when the user is signed in; otherwise it is the local or network username for the Tableau Desktop user.

Example

[Manager] = USERNAME()

If the manager dhallsten was signed in, this function would only return True when the Manager field in the view is dhallsten. When used as a filter this calculated field can be used to create a user filter that only shows data that is relevant to the person signed in to the server.

Table Calculation Functions

Use table calculation functions to customize table calculations. Table Calculations are computations that are applied to the values in the entire table and are often dependent on the table structure itself. See [Customize Table Calculations](#) on page 897 for information about creating and customizing table calculations.

FIRST()

Returns the number of rows from the current row to the first row in the partition. For example, the view below shows quarterly sales. When FIRST() is computed within the Date partition, the offset of the first row from the second row is -1.

		Region				First()
Year of Order Date	Quarter of Order Date	Central	East	South	West	
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961	\$160,877 0
	Q2	\$197,213	\$204,914	\$337,813	\$213,507	\$197,213 -1
	Q3	\$302,678	\$165,201	\$283,806	\$206,512	\$302,678 -2
	Q4	\$297,208	\$226,983	\$214,845	\$230,291	\$297,208 -3
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145	\$180,609 -4
	Q2	\$195,785	\$224,882	\$251,391	\$195,976	\$195,785 -5
	Q3	\$116,613	\$50,363	\$194,601	\$102,731	\$116,613 -6

Example

When the current row index is 3, FIRST() = -2.

INDEX()

Returns the index of the current row in the partition, without any sorting with regard to value. The first row index starts at 1. For example, the table below shows quarterly sales. When INDEX() is computed within the Date partition, the index of each row is 1, 2, 3, 4..., etc.

		Region			
Year of Order Date	Quarter of Order Date	Central	East	South	West
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961
	Q2	\$197,213	\$204,914	\$337,813	\$213,507
	Q3	\$302,678	\$165,201	\$283,806	\$206,512
	Q4	\$297,208	\$226,983	\$214,845	\$230,291
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145
	Q2	\$195,785	\$224,882	\$251,391	\$195,976
	Q3	\$116,613	\$50,363	\$194,601	\$102,731

INDEX()

\$160,877	1
\$197,213	2
\$302,678	3
\$297,208	4
\$180,609	5
\$195,785	6
\$116,613	7

Example

For the third row in the partition, INDEX() = 3.

LAST()

Returns the number of rows from the current row to the last row in the partition. For example, the table below shows quarterly sales. When LAST() is computed within the Date partition, the offset of the last row from the second row is 5.

		Region			
Year of Order Date	Quarter of Order Date	Central	East	South	West
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961
	Q2	\$197,213	\$204,914	\$337,813	\$213,507
	Q3	\$302,678	\$165,201	\$283,806	\$206,512
	Q4	\$297,208	\$226,983	\$214,845	\$230,291
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145
	Q2	\$195,785	\$224,882	\$251,391	\$195,976
	Q3	\$116,613	\$50,363	\$194,601	\$102,731

LAST()

\$160,877	6
\$197,213	5
\$302,678	4
\$297,208	3
\$180,609	2
\$195,785	1
\$116,613	0

Example

When the current row index is 3 of 7, LAST() = 4.

LOOKUP(expression, [offset])

Returns the value of the expression in a target row, specified as a relative offset from the current row. Use FIRST() + n and LAST() - n as part of your offset definition for a target relative to the first/last rows in the partition. If offset is omitted, the row to compare to can be set on the field menu. This function returns NULL if the target row cannot be determined.

The view below shows quarterly sales. When LOOKUP(SUM(Sales), 2) is computed within the Date partition, each row shows the sales value from 2 quarters into the future.

Year of Order Date	Quarter of Order Date	Region			
		Central	East	South	West
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961
	Q2	\$197,213	\$204,914	\$337,813	\$213,507
	Q3	+2 \$302,678	\$165,201	\$283,806	\$206,512
	Q4	\$297,208	\$226,983	\$214,845	\$230,291
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145
	Q2	\$195,785	\$224,882	\$251,391	\$195,976
	Q3	\$116,613	\$50,363	\$194,601	\$102,731

Year of Order Date	Quarter of Order Date	Region			
		Central	East	South	West
2009	Q1	\$302,678	\$165,201	\$283,806	\$206,512
	Q2	\$297,208	\$226,983	\$214,845	\$230,291
	Q3	\$180,609	\$180,123	\$273,943	\$251,145
	Q4	\$195,785	\$224,882	\$251,391	\$195,976
2010	Q1	\$116,613	\$50,363	\$194,601	\$102,731
	Q2				
	Q3				

Example

`LOOKUP(SUM([Profit]), FIRST() + 2)` computes the SUM(Profit) in the third row of the partition.

`PREVIOUS_VALUE(expression)`

Returns the value of this calculation in the previous row. Returns the given expression if the current row is the first row of the partition.

Example

`SUM([Profit]) * PREVIOUS_VALUE(1)` computes the running product of SUM(Profit).

`RANK(expression, ['asc' | 'desc'])`

Returns the standard competition rank for the current row in the partition. Identical values are assigned an identical rank. Use the optional '`'asc'` | '`'desc'`' argument to specify ascending or descending order. The default is descending.

With this function, the set of values (6, 9, 9, 14) would be ranked (4, 2, 2, 1).

Nulls are ignored in ranking functions. They are not numbered and they do not count against the total number of records in percentile rank calculations.

For information on different ranking options, see [Table Calculation Type: Rank](#) on page 880.

Example

The following image shows the effect of the various ranking functions (RANK, RANK_DENSE, RANK_MODIFIED, RANK_PERCENTILE, and RANK_UNIQUE) on a set of values. The data set contains information on 14 students (StudentA through StudentN); the **Age** column shows the current age of each student (all students are between 17 and 20 years of age). The remaining columns show the effect of each rank function on the set of age values, always assuming the default order (ascending or descending) for the function.

Student	Age	RANKofAge	RANK_DENSEofAge	RANK_MODIFIEDofAge	RANK_PERCENTILEofAge	RANK_UNIQUEofAge
StudentA	19	4	2	7	79%	4
StudentB	18	8	3	12	50%	8
StudentC	19	4	2	7	79%	5
StudentD	18	8	3	12	50%	9
StudentE	17	13	4	14	14%	13
StudentF	18	8	3	12	50%	10
StudentG	19	4	2	7	79%	6
StudentH	20	1	1	3	100%	1
StudentI	19	4	2	7	79%	7
StudentJ	20	1	1	3	100%	2
StudentK	20	1	1	3	100%	3
StudentL	17	13	4	14	14%	14
StudentM	18	8	3	12	50%	11
StudentN	18	8	3	12	50%	12

RANK_DENSE(expression, [‘asc’ | ‘desc’])

Returns the dense rank for the current row in the partition. Identical values are assigned an identical rank, but no gaps are inserted into the number sequence. Use the optional ‘asc’ | ‘desc’ argument to specify ascending or descending order. The default is descending.

With this function, the set of values (6, 9, 9, 14) would be ranked (3, 2, 2, 1).

Nulls are ignored in ranking functions. They are not numbered and they do not count against the total number of records in percentile rank calculations.

For information on different ranking options, see [Table Calculation Type: Rank](#) on page 880.

RANK_MODIFIED(expression, [‘asc’ | ‘desc’])

Returns the modified competition rank for the current row in the partition. Identical values are assigned an identical rank. Use the optional ‘asc’ | ‘desc’ argument to specify ascending or descending order. The default is descending.

With this function, the set of values (6, 9, 9, 14) would be ranked (4, 3, 3, 1).

Nulls are ignored in ranking functions. They are not numbered and they do not count against the total number of records in percentile rank calculations.

For information on different ranking options, see [Table Calculation Type: Rank](#) on page 880.

RANK_PERCENTILE(expression, ['asc' | 'desc'])

Returns the percentile rank for the current row in the partition. Use the optional 'asc' | 'desc' argument to specify ascending or descending order. The default is ascending.

With this function, the set of values (6, 9, 9, 14) would be ranked (0.25, 0.75, 0.75, 1.00).

Nulls are ignored in ranking functions. They are not numbered and they do not count against the total number of records in percentile rank calculations.

For information on different ranking options, see [Table Calculation Type: Rank on page 880](#).

RANK_UNIQUE(expression, ['asc' | 'desc'])

Returns the unique rank for the current row in the partition. Identical values are assigned different ranks. Use the optional 'asc' | 'desc' argument to specify ascending or descending order. The default is descending.

With this function, the set of values (6, 9, 9, 14) would be ranked (4, 2, 3, 1).

Nulls are ignored in ranking functions. They are not numbered and they do not count against the total number of records in percentile rank calculations.

For information on different ranking options, see [Table Calculation Type: Rank on page 880](#).

RUNNING_AVG(expression)

Returns the running average of the given expression, from the first row in the partition to the current row.

The view below shows quarterly sales. When RUNNING_AVG (SUM([Sales]) is computed within the Date partition, the result is a running average of the sales values for each quarter.

Region

Year of Order Date	Quarter of Order Date	Central	East	South	West
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961
	Q2	\$197,213	\$204,914	\$337,813	\$213,507
	Q3	\$302,678	\$165,201	\$283,806	\$206,512
	Q4	\$297,208	\$226,983	\$214,845	\$230,291
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145
	Q2	\$195,785	\$224,882	\$251,391	\$195,976
	Q3	\$116,613	\$80,363	\$194,601	\$102,731

Region

Year of Order Date	Quarter of Order Date	Central	East	South	West
2009	Q1	160,877	231,411	133,934	185,961
	Q2	179,045	18,162	235,873	199,734
	Q3	220,256	200,509	251,851	201,993
	Q4	239,494	207,127	242,599	209,068
2010	Q1	227,717	201,726	248,868	217,483
	Q2	222,395	205,586	249,289	213,899
	Q3	207,283	183,411	241,476	198,018

Example

RUNNING_AVG(SUM([Profit])) computes the running average of SUM(Profit).

RUNNING_COUNT(expression)

Returns the running count of the given expression, from the first row in the partition to the current row.

Example

RUNNING_COUNT(SUM([Profit])) computes the running count of SUM(Profit).

RUNNING_MAX(expression)

Returns the running maximum of the given expression, from the first row in the partition to the current row.

Region						
Year of Order Date	Quarter of Order Date	Central	East	South	West	
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961	
	Q2	\$197,213	\$204,914	\$337,813	\$213,507	
	Q3	\$302,678	\$165,201	\$283,806	\$206,512	
	Q4	\$297,208	\$226,983	\$214,845	\$230,291	
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145	
	Q2	\$195,785	\$224,882	\$251,391	\$195,976	
	Q3	\$116,613	\$50,363	\$194,601	\$102,731	

Region						
Year of Order Date	Quarter of Order Date	Central	East	South	West	
2009	Q1	160,877	231,411	133,934	185,961	
	Q2	197,213	231,411	337,813	213,507	
	Q3	302,678	165,201	283,806	206,512	
	Q4	302,678	231,411	337,813	230,291	
2010	Q1	302,678	231,411	337,813	251,145	
	Q2	302,678	231,411	337,813	251,145	
	Q3	302,678	231,411	337,813	251,145	

Example

RUNNING_MAX(SUM([Profit])) computes the running maximum of SUM(Profit).

RUNNING_MIN(expression)

Returns the running minimum of the given expression, from the first row in the partition to the current row.

Region						
Year of Order Date	Quarter of Order Date	Central	East	South	West	
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961	
	Q2	\$197,213	\$204,914	\$337,813	\$213,507	
	Q3	\$302,678	\$165,201	\$283,806	\$206,512	
	Q4	\$297,208	\$226,983	\$214,845	\$230,291	
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145	
	Q2	\$195,785	\$224,882	\$251,391	\$195,976	
	Q3	\$116,613	\$50,363	\$194,601	\$102,731	

Region						
Year of Order Date	Quarter of Order Date	Central	East	South	West	
2009	Q1	160,877	231,411	133,934	185,961	
	Q2	160,877	204,914	133,934	185,961	
	Q3	160,877	165,201	133,934	185,961	
	Q4	160,877	165,201	133,934	185,961	
2010	Q1	160,877	165,201	133,934	185,961	
	Q2	160,877	165,201	133,934	185,961	
	Q3	116,613	50,363	133,934	102,731	

Example

RUNNING_MIN(SUM([Profit])) computes the running minimum of SUM(Profit).

RUNNING_SUM(expression)

Returns the running sum of the given expression, from the first row in the partition to the current row.

The diagram illustrates the calculation of the running sum for the Q1 row of the 2010 partition. An arrow points from the value \$226,983 in the Q1 row of the main table to the corresponding row in the vertical stack of values on the right. The stack shows the cumulative sum as it grows from \$160,877 to \$660,768.

Region						
Year of Order Date	Quarter of Order Date	Central	East	South	West	
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961	
	Q2	\$197,213	\$204,914	\$337,813	\$213,507	
	Q3	\$302,678	\$165,201	\$283,806	\$206,512	
	Q4	\$297,208	\$226,983	\$214,845	\$230,291	
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145	
	Q2	\$195,785	\$224,882	\$251,391	\$195,976	
	Q3	\$116,613	\$50,363	\$194,601	\$102,731	

Region						
Year of Order Date	Quarter of Order Date	Central	East	South	West	
2009	Q1	160,877	231,411	133,934	185,961	
	Q2	358,090	436,325	471,747	399,469	
	Q3	660,768	41,500	755,550	885,000	
	Q4	957,976	828,508	970,398	836,272	
2010	Q1	1,138,585	1,008,631	1,244,341	1,087,417	
	Q2	1,334,369	1,233,513	1,495,732	1,283,392	
	Q3	1,450,982	1,283,877	1,690,333	1,386,123	

Example

`RUNNING_SUM (SUM ([Profit]))` computes the running sum of `SUM(Profit)`

SIZE()

Returns the number of rows in the partition. For example, the view below shows quarterly sales. Within the Date partition, there are seven rows so the `Size()` of the Date partition is 7.

The diagram illustrates the calculation of the size of the 2010 partition. An arrow points from the value \$116,613 in the Q3 row of the main table to the corresponding row in the vertical stack of values on the right. The stack shows the size as 7, indicating the total number of rows in the partition.

Region						
Year of Order Date	Quarter of Order Date	Central	East	South	West	
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961	
	Q2	\$197,213	\$204,914	\$337,813	\$213,507	
	Q3	\$302,678	\$165,201	\$283,806	\$206,512	
	Q4	\$297,208	\$226,983	\$214,845	\$230,291	
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145	
	Q2	\$195,785	\$224,882	\$251,391	\$195,976	
	Q3	\$116,613	\$50,363	\$194,601	\$102,731	

Example

`SIZE () = 5` when the current partition contains five rows.

SCRIPT_BOOL

Returns a Boolean result from the specified expression. The expression is passed directly to a running external service instance. In R expressions, use `.argn` (with a leading period) to reference parameters (`.arg1`, `.arg2`, etc.).

Examples

In this R example, .arg1 is equal to SUM([Profit]):

```
SCRIPT_BOOL("is.finite(.arg1)", SUM([Profit]))
```

The next example returns True for store IDs in Washington state, and False otherwise. This example could be the definition for a calculated field titled IsStoreInWA.

```
SCRIPT_BOOL('grepl(".*_WA", .arg1, perl=TRUE)', ATTR([Store ID]))
```

SCRIPT_INT

Returns an integer result from the specified expression. The expression is passed directly to a running external service instance. In R expressions, use .argn (with a leading period) to reference parameters (.arg1, .arg2, etc.)

Examples

In this R example, .arg1 is equal to SUM([Profit]):

```
SCRIPT_INT("is.finite(.arg1)", SUM([Profit]))
```

In the next example, k-means clustering is used to create three clusters:

```
SCRIPT_INT('result <- kmeans(data.frame(.arg1,.arg2,.arg3,.arg4),  
3);result$cluster;', SUM([Petal length]), SUM([Petal width]),SUM  
([Sepal length]),SUM([Sepal width]))
```

SCRIPT_REAL

Returns a real result from the specified expression. The expression is passed directly to a running external service instance. In R expressions, use .argn (with a leading period) to reference parameters (.arg1, .arg2, etc.)

Examples

In this R example, .arg1 is equal to SUM([Profit]):

```
SCRIPT_REAL("is.finite(.arg1)", SUM([Profit]))
```

The next example converts temperature values from Celsius to Fahrenheit.

```
SCRIPT_REAL('library(udunits2);ud.convert(.arg1, "celsius",  
"degree_fahrenheit")', AVG([Temperature]))
```

SCRIPT_STR

Returns a string result from the specified expression. The expression is passed directly to a running external service instance. In R expressions, use .argn (with a leading period) to reference parameters (.arg1, .arg2, etc.)

Examples

In this R example, .arg1 is equal to SUM([Profit]):

```
SCRIPT_STR("is.finite(.arg1)", SUM([Profit]))
```

The next example extracts a state abbreviation from a more complicated string (in the original form 13XSL_CA, A13_WA):

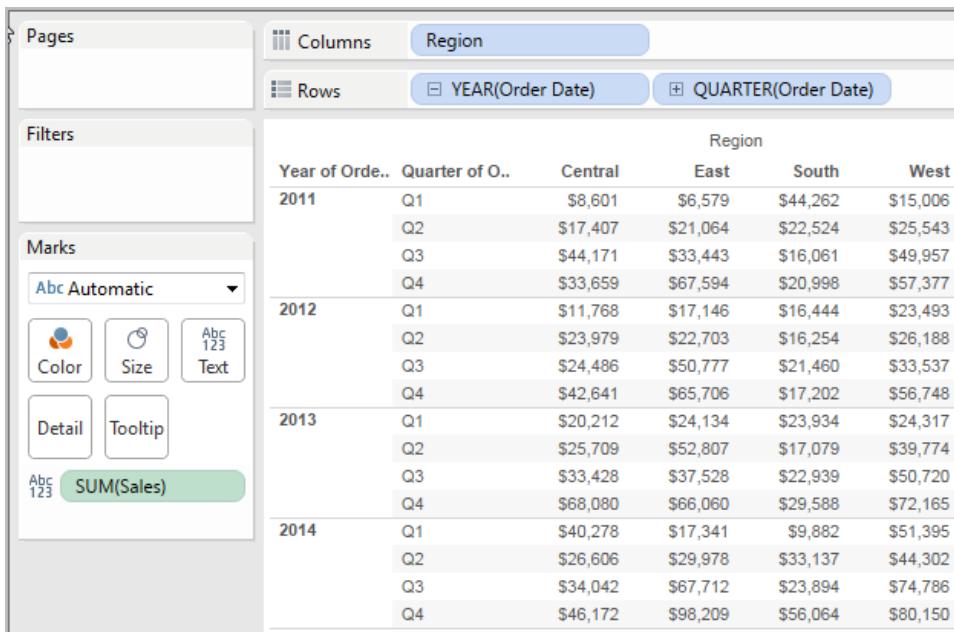
```
SCRIPT_STR('gsub('.*_', "", .arg1)', ATTR([Store ID]))
```

TOTAL(expression)

Returns the total for the given expression in a table calculation partition.

Example

Assume you are starting with this view:



You open the calculation editor and create a new field which you name **Totality**:

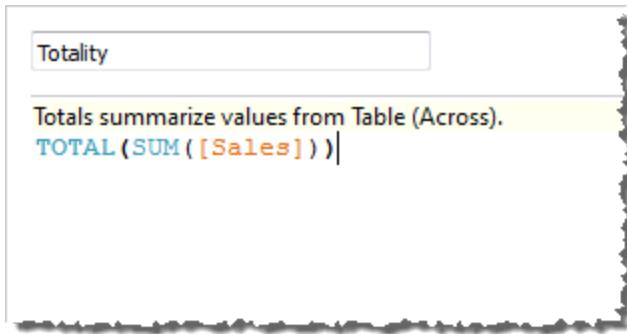


You then drop **Totality** on Text, to replace **SUM(Sales)**. Your view changes such that it sums values based on the default **Compute Using** value:

The screenshot displays a Tableau view with a heatmap. The view includes a Pages shelf, a Filters shelf, and a Marks shelf. The Marks shelf has "Totality" selected. The main area shows a heatmap with columns for "Year of Order Date" (2011, 2012, 2013, 2014) and "Quarter of Order Date" (Q1, Q2, Q3, Q4). The rows represent "Region" (Central, East, South, West). The data values are numerical sales figures. A legend on the right side of the heatmap indicates color mapping for these values.

Year of Order Date	Quarter of Order Date	Region			
		Central	East	South	West
2011	Q1	74,448	74,448	74,448	74,448
	Q2	86,539	86,539	86,539	86,539
	Q3	143,633	143,633	143,633	143,633
	Q4	179,628	179,628	179,628	179,628
2012	Q1	68,852	68,852	68,852	68,852
	Q2	89,124	89,124	89,124	89,124
	Q3	130,260	130,260	130,260	130,260
	Q4	182,297	182,297	182,297	182,297
2013	Q1	92,596	92,596	92,596	92,596
	Q2	135,370	135,370	135,370	135,370
	Q3	144,614	144,614	144,614	144,614
	Q4	235,893	235,893	235,893	235,893
2014	Q1	118,896	118,896	118,896	118,896
	Q2	134,023	134,023	134,023	134,023
	Q3	200,433	200,433	200,433	200,433
	Q4	280,595	280,595	280,595	280,595

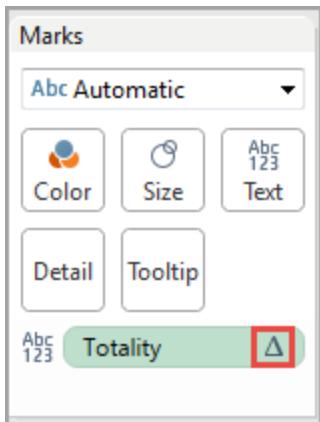
This raises the question, What is the default **Compute Using** value? If you right-click (Control-click on a Mac) **Totality** in the Data pane and choose **Edit**, there is now an additional bit of information available:



The default **Compute Using** value is **Table (Across)**. The result is that **Totality** is summing the values across each row of your table. Thus, the value that you see across each row is the sum of the values from the original version of the table.

The values in the 2011/Q1 row in the original table were \$8601, \$6579, \$44262, and \$15006. The values in the table after **Totality** replaces **SUM(Sales)** are all \$74,448, which is the sum of the four original values.

Notice the triangle next to **Totality** after you drop it on **Text**:



This indicates that this field is using a table calculation. You can right-click the field and choose **Edit Table Calculation** to redirect your function to a different **Compute Using** value. For example, you could set it to **Table (Down)**. In that case, your table would look like this:

Region

Year of Order Date	Quarter of Order Date	Central	East	South	West
2011	Q1	501,240	678,781	391,722	725,458
	Q2	501,240	678,781	391,722	725,458
	Q3	501,240	678,781	391,722	725,458
	Q4	501,240	678,781	391,722	725,458
2012	Q1	501,240	678,781	391,722	725,458
	Q2	501,240	678,781	391,722	725,458
	Q3	501,240	678,781	391,722	725,458
	Q4	501,240	678,781	391,722	725,458
2013	Q1	501,240	678,781	391,722	725,458
	Q2	501,240	678,781	391,722	725,458
	Q3	501,240	678,781	391,722	725,458
	Q4	501,240	678,781	391,722	725,458
2014	Q1	501,240	678,781	391,722	725,458
	Q2	501,240	678,781	391,722	725,458
	Q3	501,240	678,781	391,722	725,458
	Q4	501,240	678,781	391,722	725,458

WINDOW_AVG(expression, [start, end])

Returns the average of the expression within the window. The window is defined by means of offsets from the current row. Use FIRST() +n and LAST() -n for offsets from the first or last row in the partition. If the start and end are omitted, the entire partition is used.

For example, the view below shows quarterly sales. A window average within the Date partition returns the average sales across all dates.

Region

Year of Order Date	Quarter of Order Date	Central	East	South	West
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961
	Q2	\$197,213	\$204,914	\$337,813	\$213,507
	Q3	\$302,678	\$165,201	\$283,806	\$206,512
	Q4	\$297,208	\$226,983	\$214,845	\$230,291
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145
	Q2	\$195,785	\$224,882	\$251,391	\$195,976
	Q3	\$116,613	\$50,363	\$194,601	\$102,731

Region

Year of Order Date	Quarter of Order Date	Central	East	South	West
2009	Q1	207,283	183,411	241,476	198,018
	Q2	207,283	183,411	241,476	198,018
	Q3	207,283	183,411	241,476	198,018
	Q4	207,283	183,411	241,476	198,018
2010	Q1	207,283	183,411	241,476	198,018
	Q2	207,283	183,411	241,476	198,018
	Q3	207,283	183,411	241,476	198,018

WINDOW_AVG(SUM([Sales]), FIRST(), LAST())

FIRST() LAST()

AVG = \$207,283

Example

WINDOW_AVG(SUM([Profit]), FIRST() +1, 0) computes the average of SUM(Profit) from the second row to the current row.

WINDOW_COUNT(expression, [start, end])

Returns the count of the expression within the window. The window is defined by means of offsets from the current row. Use FIRST() + n and LAST() - n for offsets from the first or last row in the partition. If the start and end are omitted, the entire partition is used.

Example

`WINDOW_COUNT(SUM([Profit]), FIRST() + 1, 0)` computes the count of SUM(Profit) from the second row to the current row

WINDOW_MEDIAN(expression, [start, end])

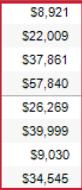
Returns the median of the expression within the window. The window is defined by means of offsets from the current row. Use FIRST() + n and LAST() - n for offsets from the first or last row in the partition. If the start and end are omitted, the entire partition is used.

For example, the view below shows quarterly profit. A window median within the Date partition returns the median profit across all dates.

		Region			
Year of Order Date	Quarter of Order Date	Central	East	South	West
2011	Q1	\$8,921	\$20,575	\$29,654	\$22,647
	Q2	\$22,009	\$11,477	\$14,893	\$30,791
	Q3	\$37,861	\$258	\$31,257	\$25,006
	Q4	\$57,840	\$13,313	\$23,784	\$31,771
2012	Q1	\$26,269	\$30,699	\$30,278	\$18,861
	Q2	\$39,999	\$28,438	\$23,672	(5922)
	Q3	\$9,030	\$22,096	\$20,973	\$22,535
	Q4	\$34,545	\$12,001	\$20,074	\$3,363

		Region			
Year of Order Date	Quarter of Order Date	Central	East	South	West
2011	Q1	30,407	16,944	23,728	22,591
	Q2	30,407	16,944	23,728	22,591
	Q3	30,407	16,944	23,728	22,591
	Q4	30,407	16,944	23,728	22,591
2012	Q1	30,407	16,944	23,728	22,591
	Q2	30,407	16,944	23,728	22,591
	Q3	30,407	16,944	23,728	22,591
	Q4	30,407	16,944	23,728	22,591

`WINDOW_MEDIAN(SUM([Profit]), FIRST(), LAST())`

FIRST() → 

LAST() → 

MEDIAN= **\$30,407**

Example

`WINDOW_MEDIAN(SUM([Profit]), FIRST() + 1, 0)` computes the median of SUM(Profit) from the second row to the current row.

WINDOW_MAX(expression, [start, end])

Returns the maximum of the expression within the window. The window is defined by means of offsets from the current row. Use FIRST() + n and LAST() - n for offsets from the first or last row in the partition. If the start and end are omitted, the entire partition is used.

For example, the view below shows quarterly sales. A window maximum within the Date partition returns the maximum sales across all dates.

Region

Year of Order Date	Quarter of Order Date	Central	East	South	West
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961
	Q2	\$197,213	\$204,914	\$337,813	\$213,507
	Q3	\$302,678	\$165,201	\$283,806	\$206,512
	Q4	\$297,208	\$226,983	\$214,845	\$230,291
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145
	Q2	\$195,785	\$224,882	\$251,391	\$195,976
	Q3	\$116,613	\$50,363	\$194,601	\$102,731

Year of Order Date	Quarter of Order Date	Central	East	South	West
2009	Q1	302,678	231,411	337,813	251,145
	Q2	302,678	231,411	337,813	251,145
	Q3	302,678	231,411	337,813	251,145
	Q4	302,678	231,411	337,813	251,145
2010	Q1	302,678	231,411	337,813	251,145
	Q2	302,678	231,411	337,813	251,145
	Q3	302,678	231,411	337,813	251,145

Example

`WINDOW_MAX(SUM([Sales]), FIRST(), LAST())` computes the maximum of SUM(Sales) from the second row to the current row.

`WINDOW_MIN(expression, [start, end])`

Returns the minimum of the expression within the window. The window is defined by means of offsets from the current row. Use `FIRST() + n` and `LAST() - n` for offsets from the first or last row in the partition. If the start and end are omitted, the entire partition is used.

For example, the view below shows quarterly sales. A window minimum within the Date partition returns the minimum sales across all dates.

Region

Year of Order Date	Quarter of Order Date	Central	East	South	West
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961
	Q2	\$197,213	\$204,914	\$337,813	\$213,507
	Q3	\$302,678	\$165,201	\$283,806	\$206,512
	Q4	\$297,208	\$226,983	\$214,845	\$230,291
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145
	Q2	\$195,785	\$224,882	\$251,391	\$195,976
	Q3	\$116,613	\$50,363	\$194,601	\$102,731

Year of Order Date	Quarter of Order Date	Central	East	South	West
2009	Q1	116,613	50,363	133,934	102,731
	Q2	116,613	50,363	133,934	102,731
	Q3	116,613	50,363	133,934	102,731
	Q4	116,613	50,363	133,934	102,731
2010	Q1	116,613	50,363	133,934	102,731
	Q2	116,613	50,363	133,934	102,731
	Q3	116,613	50,363	133,934	102,731

Example

`WINDOW_MIN(SUM([Sales]), FIRST() + 1, 0)` computes the minimum of SUM(Sales) from the second row to the current row.

WINDOW_PERCENTILE(expression, number, [start, end])

Returns the value corresponding to the specified percentile within the window. The window is defined by means of offsets from the current row. Use FIRST() +n and LAST() -n for offsets from the first or last row in the partition. If the start and end are omitted, the entire partition is used.

Example

`WINDOW_PERCENTILE(SUM([Profit]), 0.75, -2, 0)` returns the 75th percentile for SUM(Profit) from the two previous rows to the current row.

WINDOW_STDEV(expression, [start, end])

Returns the sample standard deviation of the expression within the window. The window is defined by means of offsets from the current row. Use FIRST() +n and LAST() -n for offsets from the first or last row in the partition. If the start and end are omitted, the entire partition is used.

Example

`WINDOW_STDEV(SUM([Profit]), FIRST() +1, 0)` computes the standard deviation of SUM(Profit) from the second row to the current row.

WINDOW_STDEVP(expression, [start, end])

Returns the biased standard deviation of the expression within the window. The window is defined by means of offsets from the current row. Use FIRST() +n and LAST() -n for offsets from the first or last row in the partition. If the start and end are omitted, the entire partition is used.

Example

`WINDOW_STDEVP(SUM([Profit]), FIRST() +1, 0)` computes the standard deviation of SUM(Profit) from the second row to the current row.

WINDOW_SUM(expression, [start, end])

Returns the sum of the expression within the window. The window is defined by means of offsets from the current row. Use FIRST() +n and LAST() -n for offsets from the first or last row in the partition. If the start and end are omitted, the entire partition is used.

For example, the view below shows quarterly sales. A window sum computed within the Date partition returns the summation of sales across all quarters.

Region

Year of Order Date	Quarter of Order Date	Central	East	South	West
2009	Q1	\$160,877	\$231,411	\$133,934	\$185,961
	Q2	\$197,213	\$204,914	\$337,813	\$213,507
	Q3	\$302,678	\$165,201	\$283,806	\$206,512
	Q4	\$297,208	\$226,983	\$214,845	\$230,291
2010	Q1	\$180,609	\$180,123	\$273,943	\$251,145
	Q2	\$195,785	\$224,882	\$251,391	\$195,976
	Q3	\$116,613	\$50,363	\$194,601	\$102,731

Year of Order Date	Quarter of Order Date	Central	East	South	West
2009	Q1	1,450,982	1,283,877	1,690,333	1,386,123
	Q2	1,450,982	1,283,877	1,690,333	1,386,123
	Q3	1,450,982	1,283,877	1,690,333	1,386,123
	Q4	1,450,982	1,283,877	1,690,333	1,386,123
2010	Q1	1,450,982	1,283,877	1,690,333	1,386,123
	Q2	1,450,982	1,283,877	1,690,333	1,386,123
	Q3	1,450,982	1,283,877	1,690,333	1,386,123

Example

`WINDOW_SUM(SUM([Sales]), FIRST(), LAST())` computes the sum of `SUM(Sales)` from the second row to the current row.

WINDOW_VAR(expression, [start, end])

Returns the sample variance of the expression within the window. The window is defined by means of offsets from the current row. Use `FIRST() + n` and `LAST() - n` for offsets from the first or last row in the partition. If the start and end are omitted, the entire partition is used.

Example

`WINDOW_VAR((SUM([Profit])), FIRST() + 1, 0)` computes the variance of `SUM(Profit)` from the second row to the current row.

WINDOW_VARP(expression, [start, end])

Returns the biased variance of the expression within the window. The window is defined by means of offsets from the current row. Use `FIRST() + n` and `LAST() - n` for offsets from the first or last row in the partition. If the start and end are omitted, the entire partition is used.

Example

`WINDOW_VARP(SUM([Profit]), FIRST() + 1, 0)` computes the variance of `SUM(Profit)` from the second row to the current row.

Additional Functions

REGEXP_REPLACE(string, pattern, replacement)

Returns a copy of the given string where the regular expression pattern is replaced by the replacement string. This function is available for Text File, Hadoop Hive, Google BigQuery, PostgreSQL, Tableau Data Extract, Microsoft Excel, Salesforce, HP Vertica, Pivotal Greenplum, Teradata (version 14.1 and above), and Oracle data sources.

For Tableau data extracts, the pattern and the replacement must be constants.

For information on regular expression syntax, see your data source's documentation. For Tableau extracts, regular expression syntax conforms to the standards of the ICU (International Components for Unicode), an open source project of mature C/C++ and Java libraries for Unicode support, software internationalization, and software globalization. See the [Regular Expressions](#) page in the online ICU User Guide.

Example

```
REGEXP_REPLACE('abc 123', '\s', '-') = 'abc-123'
```

REGEXP_MATCH(string, pattern)

Returns true if a substring of the specified string matches the regular expression pattern. This function is available for Text File, Google BigQuery, PostgreSQL, Tableau Data Extract, Microsoft Excel, Salesforce, HP Vertica, Pivotal Greenplum, Teradata (version 14.1 and above), Impala 2.3.0 (through Cloudera Hadoop data sources), and Oracle data sources.

For Tableau data extracts, the pattern must be a constant.

For information on regular expression syntax, see your data source's documentation. For Tableau extracts, regular expression syntax conforms to the standards of the ICU (International Components for Unicode), an open source project of mature C/C++ and Java libraries for Unicode support, software internationalization, and software globalization. See the [Regular Expressions](#) page in the online ICU User Guide.

Example

```
REGEXP_MATCH('-([1234].[The.Market])-','\[\s*(\w*\.).(\w*\s*)\]')=true
```

REGEXP_EXTRACT(string, pattern)

Returns the portion of the string that matches the regular expression pattern. This function is available for Text File, Hadoop Hive, Google BigQuery, PostgreSQL, Tableau Data Extract, Microsoft Excel, Salesforce, HP Vertica, Pivotal Greenplum, Teradata (version 14.1 and above), and Oracle data sources.

For Tableau data extracts, the pattern must be a constant.

For information on regular expression syntax, see your data source's documentation. For Tableau extracts, regular expression syntax conforms to the standards of the ICU (International Components for Unicode), an open source project of mature C/C++ and Java libraries for Unicode support, software internationalization, and software globalization. See the [Regular Expressions](#) page in the online ICU User Guide.

Example

```
REGEXP_EXTRACT('abc 123', '[a-z]+\s+(\d+)') = '123'
```

[REGEXP_EXTRACT_NTH\(string, pattern, index\)](#)

Returns the portion of the string that matches the regular expression pattern. The substring is matched to the nth capturing group, where n is the given index. If index is 0, the entire string is returned. This function is available for Text File, PostgreSQL, Tableau Data Extract, Microsoft Excel, Salesforce, HP Vertica, Pivotal Greenplum, Teradata (version 14.1 and above), and Oracle data sources.

For Tableau data extracts, the pattern must be a constant.

For information on regular expression syntax, see your data source's documentation. For Tableau extracts, regular expression syntax conforms to the standards of the ICU (International Components for Unicode), an open source project of mature C/C++ and Java libraries for Unicode support, software internationalization, and software globalization. See the [Regular Expressions](#) page in the online ICU User Guide.

Example

`REGEXP_EXTRACT_NTH('abc 123', '([a-z]+)\s+(\d+)', 2) = '123'`

[Hadoop Hive Specific Functions](#)

[GET_JSON_OBJECT\(JSON string, JSON path\)](#)

Returns the JSON object within the JSON string based on the JSON path.

[PARSE_URL\(string, url_part\)](#)

Returns a component of the given URL string where the component is defined by url_part. Valid url_part values include: 'HOST', 'PATH', 'QUERY', 'REF', 'PROTOCOL', 'AUTHORITY', 'FILE' and 'USERINFO'.

Example

`PARSE_URL('http://www.tableau.com', 'HOST') = 'www.tableau.com'`

[PARSE_URL_QUERY\(string, key\)](#)

Returns the value of the specified query parameter in the given URL string. The query parameter is defined by the key.

Example

`PARSE_URL_QUERY('http://www.tableau.com?page=1&cat=4', 'page') = '1'`

[XPATH_BOOLEAN\(XML string, XPath expression string\)](#)

Returns true if the XPath expression matches a node or evaluates to true.

Example

```
XPATH_BOOLEAN('<values> <value id="0">1</value><value id="1">5</value>', 'values/value  
[@id="1"] = 5') = true
```

[XPATH_DOUBLE\(XML string, XPath expression string\)](#)

Returns the floating-point value of the XPath expression.

Example

```
XPATH_DOUBLE('<values><value>1.0</value><value>5.5</value> </values>', 'sum(value/*)')  
= 6.5
```

[XPATH_FLOAT\(XML string, XPath expression string\)](#)

Returns the floating-point value of the XPath expression.

Example

```
XPATH_FLOAT('<values><value>1.0</value><value>5.5</value> </values>','sum(value/*)') =  
6.5
```

[XPATH_INT\(XML string, XPath expression string\)](#)

Returns the numerical value of the XPath expression, or zero if the XPath expression cannot evaluate to a number.

Example

```
XPATH_INT('<values><value>1</value><value>5</value> </values>','sum(value/*)') = 6
```

[XPATH_LONG\(XML string, XPath expression string\)](#)

Returns the numerical value of the XPath expression, or zero if the XPath expression cannot evaluate to a number.

Example

```
XPATH_LONG('<values><value>1</value><value>5</value> </values>','sum(value/*)') = 6
```

XPATH_SHORT(XML string, XPath expression string)

Returns the numerical value of the XPath expression, or zero if the XPath expression cannot evaluate to a number.

Example

```
XPATH_SHORT('<values><value>1</value><value>5</value> </values>','sum(value/*)') = 6
```

XPATH_STRING(XML string, XPath expression string)

Returns the text of the first matching node.

Example

```
XPATH_STRING('<sites ><url domain="org">http://www.w3.org</url> <url domain="com">http://www.tableau.com</url></sites>','sites/url[@domain="com"]') = 'http://www.tableau.com'
```

Google BigQuery Specific Functions

DOMAIN(string_url)

Given a URL string, returns the domain as a string.

Example

```
DOMAIN('http://www.google.com:80/index.html') = 'google.com'
```

GROUP_CONCAT(expression)

Concatenates values from each record into a single comma-delimited string. This function acts like a SUM() for strings.

Example

```
GROUP_CONCAT(Region) = "Central,East,West"
```

HOST(string_url)

Given a URL string, returns the host name as a string.

Example

```
HOST('http://www.google.com:80/index.html') = 'www.google.com:80'
```

LOG2(number)

Returns the logarithm base 2 of a number.

Example

LOG2(16) = '4.00'

[**LTRIM_THIS\(string, string\)**](#)

Returns the first string with any leading occurrence of the second string removed.

Example

LTRIM_THIS('[-Sales-]', '-') = 'Sales-'

[**RTRIM_THIS\(string, string\)**](#)

Returns the first string with any trailing occurrence of the second string removed.

Example

RTRIM_THIS('[-Market-]', '-') = '[-Market'

[**TIMESTAMP_TO_USEC\(expression\)**](#)

Converts a TIMESTAMP data type to a UNIX timestamp in microseconds.

Example

TIMESTAMP_TO_USEC(#2012-10-01 01:02:03#)=1349053323000000

[**USEC_TO_TIMESTAMP\(expression\)**](#)

Converts a UNIX timestamp in microseconds to a TIMESTAMP data type.

Example

USEC_TO_TIMESTAMP(1349053323000000) = #2012-10-01 01:02:03#

[**TLD\(string_url\)**](#)

Given a URL string, returns the top level domain plus any country domain in the URL.

Example

TLD('http://www.google.com:80/index.html') = '.com'

TLD('http://www.google.co.uk:80/index.html') = '.co.uk'

Operators

To create calculated fields and formulas, you need to understand the operators supported by Tableau. This section discusses the basic operators that are available and the order (precedence) of operations.

+ (addition)

This means addition when applied to numbers and concatenation when applied to strings. When applied to dates, it can be used to add a number of days to a date. For example,

```
7 + 3  
Profit + Sales  
'abc' + 'def' = 'abcdef'  
#April 15, 2004# + 15 = #April 30, 2004#
```

- (subtraction)

This means subtraction when applied to numbers and negation if applied to an expression. When applied to dates, it can be used to subtract a number of days from a date. Hence it can also be used to calculate the difference in days between two dates. For example,

```
7 - 3  
Profit - Sales  
-(7+3) = -10  
#April 16, 2004# - 15 = #April 1, 2004#  
#April 15, 2004# - #April 8, 2004# = 7
```

*** (multiplication)**

This means numeric multiplication. For example, $5 * 4 = 20$.

/ (division)

This means numeric division. For example, $20 / 4 = 5$.

% (modulo)

Returns the remainder of a division operation. For example, $9 \% 2$ returns 1 because 2 goes into 9 four times with a remainder of 1. Modulo can only operate on integers.

$= =, =, >, <, >=, <=, !=, <>$ (comparisons)

These are the basic comparison operators that can be used in expressions. Their meanings are as follows: $= =$ or $=$ (equal to), $>$ (greater than), $<$ (less than), \geq (greater than or equal to), \leq (less than or equal to), $!=$ and \neq (not equal to).

Each operator compares two numbers, dates, or strings and returns a boolean (TRUE or FALSE).

^ (power)

This symbol is equivalent to the POWER function. It raises a number to the specified power.

For example:

$6^3 = 216$

AND

This is a logical operator. An expression or a boolean must appear on either side of it. For example,

`IIF(Profit =100 AND Sales =1000, "High", "Low")`

If both expressions are TRUE (i.e., not FALSE and not UNKNOWN), then the result is TRUE. If either expression is UNKNOWN, then the result is UNKNOWN. In all other cases, the result is FALSE.

If you create a calculation in which the result of an AND comparison is displayed on a worksheet, Tableau displays TRUE and FALSE. If you would like to change this, use the Format area in the format dialog.

OR

This is a logical operator. An expression or a boolean must appear on either side of it. For example,

`IIF(Profit =100 OR Sales =1000, "High", "Low")`

If either expression is TRUE, then the result is TRUE. If both expressions are FALSE, then the result is FALSE. If both expressions are UNKNOWN, then the result is UNKNOWN.

If you create a calculation in which the result of an OR comparison is displayed on a worksheet, Tableau displays TRUE and FALSE. If you would like to change this, use the Format area in the format dialog. The OR operator employs "short circuit evaluation." This means that if the first expression is evaluated to be TRUE, then the second expression is not evaluated at all. This can be helpful if the second expression results in an error when the first expression is TRUE, because the second expression in this case is never evaluated.

NOT

This is a logical operator. It can be used to negate another boolean or an expression. For example,

`IIF(NOT(Sales = Profit),"Not Equal","Equal")`

Precedence

All operators are evaluated in a specific order. For example, $2*1+2$ is equal to 4 and not equal to 6. The reason is that the * operator is always evaluated before the + operator.

The following table shows the order in which operators are evaluated. The first line has the highest precedence. Operators on the same line have the same precedence. If two operators have the same precedence they are evaluated from left to right in the formula.

Precedence	Operator
1	- (negate)
2	\wedge (power)
3	$*$, $/$, $\%$
4	$+$, $-$
5	$==$, $>$, $<$, \geq , \leq , \neq
6	NOT
7	AND
8	OR

Parentheses can be used as needed. Operators that appear within parentheses are evaluated before those outside of parentheses, starting from the innermost parentheses and moving outward. For example, $(1+ (2*2+1)*(3*6/3)) = 31$.

Keyboard Shortcuts

General Keyboard Shortcuts

Tableau keyboard shortcuts are listed below.

Description	Keyboard Shortcut	
	Windows	Mac
Select all data	Ctrl+A	Command-A
Use Rectangular Selection tool	A	A
Smaller cell size	Ctrl+B	Command-B

Bigger cell size	Ctrl+Shift+B	Command-Shift-B
Copy selected data	Ctrl+C	Command-C
Place selected field on Columns shelf	Alt+Shift+C	Option-Shift-C
Connect to data source	Ctrl+D	Command-D
Use Lasso Selection Tool	D	D
Describe sheet	Ctrl+E	Command-E
Makes the find command in the Data pane active	Ctrl+F	Command-F
Place selected field on Filters shelf	Alt+Shift+F	Option-Shift-F
Enter/Exit Full Screen		Control-Command-F
Switch in and out of Presentation Mode	F7 , Ctrl+H	Option-Return
Place selected field on Size	Alt+Shift+I	Option-Shift-I
Flip orientation of column labels at bottom of view	Ctrl+L	
Place selected field on Detail	Alt+Shift+L	Option-Shift-L
New worksheet	Ctrl+M	Command-T
New workbook	Ctrl+N	Command-N
Open file	Ctrl+O	Command-O
Place selected field on Color	Alt+Shift+O	Option-Shift-O
Print	Ctrl+P	Command-P
Place selected field on Pages shelf	Alt+Shift+P	Option-Shift-P
Place selected field on Rows shelf	Alt+Shift+R	Option-Shift-R
Save file	Ctrl+S	Command-S
Use Radial Selection Tool	S	S

Place selected field on Shape	Alt+Shift+S	Option-Shift-S
Place selected field on Text/Label	Alt+Shift+T	Option-Shift-T
Paste clipboard	Ctrl+V	Command-V
Swap rows and columns	Ctrl+W	Control-Command-W
Cut text selection (e.g., in captions, titles, formulas, etc.)	Ctrl+X	Command-X
Place selected field on Rows shelf	Alt+Shift+X	Option-Shift-X
Redo	Ctrl+Y	Command-Shift-Z
Place selected field on Columns shelf	Alt+Shift+Y	Option-Shift-Y
Undo	Ctrl+Z	Command-Z
Clear the current worksheet	Alt+Shift+Backspace	Option-Shift-Delete
Make rows narrower	Ctrl+Left Arrow	Control-Command-Left Arrow
Make rows wider	Ctrl+Right Arrow	Control-Command-Right Arrow
Make columns shorter	Ctrl+Down Arrow	Control-Command-Down Arrow
Make columns taller	Ctrl+Up Arrow	Control-Command-Up Arrow
Show Me!	Ctrl+1 , Ctrl+Shift+1	Command-1
Add the selected field to the sheet. Only works with a single field	Enter	Return
Opens the Help	F1	Shift-Command-Question Mark
Deletes the selected sheet (on a dashboard)	Ctrl+F4	
Closes the current workbook	Alt+F4	Command-W
Starts and stops forward playback	F4	

on the pages shelf		
Starts and stops backward playback on the pages shelf	Shift+F4	
Refreshes the data source	F5	Command-R
Skip forward one page	Ctrl+Period	Command-Period
Skip backward one page	Ctrl+Comma	Command-Comma
Cycle forward through open worksheets	Ctrl+Tab , Ctrl+F6	Shift-Command-Close Brace
Cycle backward through open worksheets	Ctrl+Shift+Tab , Ctrl+Shift+F6	Shift-Command-Open Brace
Run update	F9	Shift-Command-0
Toggles Automatic Updates on and off	F10	Option-Command-0
Reverts workbook to last saved state	F12	Option-Command-E
Clears the selection (Desktop and Reader only)	Esc	Esc

Navigation and Selection Shortcuts

In addition to the standard keyboard shortcuts there are several key combinations that can make navigating and selecting marks fast and easy.

Description	Keyboard/Mouse Action	
	Windows	Mac
Selects the mark	Click	Click
Selects a group of marks	Drag	Drag
Adds individual marks to the selection	Ctrl+Click	Command-Click
Adds a group of marks to the selection	Ctrl+Drag	Command-Drag

Pans around the view	Shift+Drag	Shift-Drag
Zooms in to a point in the view (requires zoom mode if not map)	Double-click , Ctrl+Shift+Click	Double-click , Shift-Command- Click
Zooms out from a point on a map (requires zoom mode if not map)	Ctrl+Shift+Alt+Click	Shift-Option-Com- mand-Click
Zooms out	Shift+Double-click	Shift–Double-click
Zooms in to an area in the view (requires zoom mode if not map)	Ctrl+Shift+Drag	Shift-Command- Drag
Zooms in and out on a map	Scroll	Scroll
Drags a row and scrolls through a long list simultaneously	Click+Drag to bottom of pane+Hold	Click-Scroll, Com- mand-Hold

Field Selection Shortcuts

There are several key and mouse combinations that can make creating a view and selecting fields fast and easy.

Description	Keyboard/Mouse Action	
	Windows	Mac
Opens the Drop Field menu	Right-click+Drag to shelf	Option-Drag to shelf
Copies a field in the view to be placed on another shelf or card	Ctrl+Drag	Command-Drag
Adds a field to the view	Double-click	Double-click

Toolbar

Hide or display the Tableau toolbar by selecting **Window > Show Toolbar**.

The table below explains the functions of each toolbar button.

Toolbar Button	Description
	Tableau icon: navigates to the start page. For more information, see Start Page on page 125 .
	Undo: reverses the most recent action in the workbook. You can undo an unlimited number of times, back to the last time you opened the workbook, even after you have saved. For more information, see Undo and Redo on page 570 .
	Redo: repeats the last action you reversed with the Undo button. You can redo an unlimited number of times.
	Save: saves the changes made to the workbook. For more information, see Saving Your Work on page 1266 .
	New Data Source: opens the Connect pane, where you can create a new connection or open a saved connection. For more information, see Connect to Your Data on page 346 .
	Pause Auto Updates: controls whether Tableau updates the view when changes are made. Use the drop-down menu to automatically update the entire sheet or just use filters. For more information, see Turn off Automatic Updates to Boost Performance on page 1832 .
	Run Update: runs a manual query of the data to update the view with changes when automatic updates are turned off. Use the drop-down menu to update the entire worksheet or just use filters.
	New Worksheet: creates a new blank worksheet. Use the drop-down menu to create a new worksheet, dashboard, or story. For more information, see Creating New Worksheets, Dashboards, and Stories on page 242 .
	Duplicate Sheet: creates a new worksheet containing the same view as the current sheet. For more information, see Duplicating Sheets on page 244 .
	Clear: clears the current worksheet. Use the drop-down menu to clear specific parts of the view such as filters, formatting, sizing, and axis ranges.

Toolbar Button	Description
	Swap : moves the fields on the Rows shelf to the Columns shelf and vice versa. The Hide Empty Rows and Hide Empty Columns settings are always swapped with this button.
	Sort Ascending : applies a sort in ascending order of a selected field based on the measures in the view. For more information, see Sorting on page 480 .
	Sort Descending : applies a sort in descending order of a selected field based on the measures in the view. For more information, see Sorting on page 480 .
	Highlight : turns on highlighting for the selected sheet. Use the options on the drop-down menu to define how values are highlighted. For more information, see Highlight Toolbar Button on page 1128 .
	Group Members : creates a group by combining selected values. When multiple dimensions are selected, use the drop-down menu to specify whether to group on a specific dimension or across all dimensions. For more information, see Groups on page 497 .
	Show Mark Labels : switches between showing and hiding mark labels for the current sheet. For more information, see Mark Labels on page 548 .
	Fix Axes : switches between a locked axis that only shows a specific range and a dynamic axis that adjusts the range based on the minimum and maximum values in the view. For more information, see Edit Axes on page 1047 .
	Fit specifies how the view should be sized within the window. Select Normal fit, Fit Width, Fit Height, or Entire View.
	Show/Hide Cards : shows and hides specific cards in a worksheet. Select each card that you want to hide or show on the drop-down menu.
	Presentation Mode : switches between showing and hiding everything except the view (i.e., shelves, toolbar, Data pane). For more information, see Reorganizing the Workspace on page 253 .

Toolbar Button	Description
 Show Me	Show Me: helps you choose a view type by highlighting view types that work best with the field types in your data. For more information, see Show Me on page 595 .

Pass Expressions to R

R is an open source software programming language and a software environment for statistical computing and graphics. In Tableau Desktop, you can use a set of four functions to pass R expressions to an Rserve server and obtain a result. The functions are

```
SCRIPT_BOOL
SCRIPT_INT
SCRIPT_REAL
SCRIPT_STR
```

See [Table Calculation Functions on page 1587](#) for details and examples. Because these SCRIPT_ functions are table calculations functions, addressing and partitioning concepts apply. (For an explanation of these concepts, see [Table Calculations: Addressing and Partitioning on page 855](#).) Tableau makes one call to Rserve per partition. Because connecting to Rserve involves some overhead, try to pass values as vectors rather than as individual values whenever possible. For example if you set addressing to Cell (for example, by clicking the field in the view and choosing **Compute Along > Cell**), Tableau will make a separate call per row to Rserve; depending on the size of the data, this can result in a very large number of individual Rserve calls. If you instead use a column that identifies each row that you would use in the level of detail, you could compute along that column so that Tableau could pass those values in a single call.

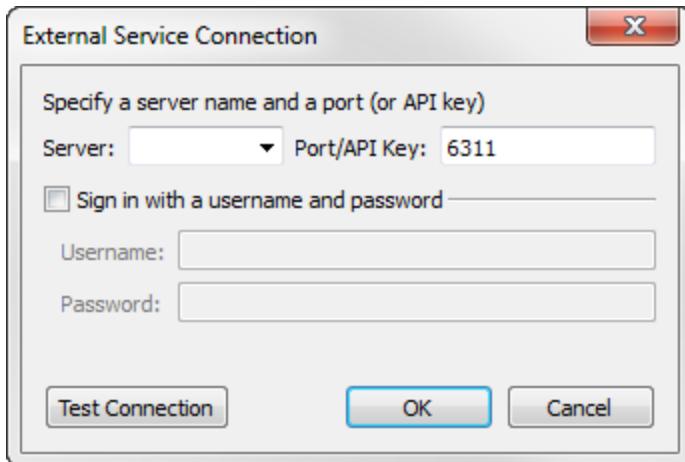
Before you can use any of the SCRIPT functions, you must establish a connection to an Rserve server, which is a server that allows applications to access R functionality. See [Rserve](#) for details. For information on installing, running, and configuring Rserve, as well as on optimizing R scripts and R security, see the Tableau Community post [R Implementation Notes](#). And for more information about using R with Tableau, see the blog post [Tableau 8.1 and R](#).

Tableau has been tested with R versions 3.1 and 3.2, and with Rserver version 0.6-8.

Configure an Rserve Connection

To configure an Rserve connection, follow these steps:

1. On the Help menu in Tableau Desktop choose **Settings and Performance >Manage External Service connection** to open the External Service Connection dialog box:



2. Enter or select a server name using a domain or an IP address. The drop-down list includes **localhost** and the server you most recently connected to.
3. Specify a port. Port 6311 is the default port for Rserve servers.
4. If the server requires credentials, specify a Username and Password.
5. Click **Test Connection**.
6. Click **OK**.

If no connection can be established, an error message is displayed. Click **Show Details** in the message to see any diagnostic information returned by the server.

Share a Workbook That Requires an Rserve Connection

You may need to send a workbook that contains R functionality to other users, who may be using different copies of Tableau Desktop on other computers. Or, users may download a workbook from Tableau Server that contains R functionality. Before users will be able to use the R functionality in workbooks they have received or downloaded to Tableau Desktop, they must configure Rserve connections on their computers.

Publish a Workbook That Requires an Rserve Connection

Before you publish a workbook that relies on an Rserve connection to Tableau Server, you should configure Tableau Server to have its own Rserve connection. You do this by configuring settings with `tabadmin set`. The settings are equivalent to the values you set in the Rserve Connection dialog box. They are:

- vizqlserver.extsvc.host
- vizqlserver.extsvc.port
- vizqlserver.extsvc.username
- vizqlserver.extsvc.password

Note: In versions of Tableau before version 10.1, these settings were named vizqlserver.rserve.host, vizqlserver.rserve.port, vizqlserver.rserve.username, and vizqlserver.rserve.password. Be sure to use these earlier setting names if your Tableau version is older than version 10.1.

The settings vizqlserver.extsvc.username and vizqlserver.extsvc.password should be omitted if Tableau Desktop connects to Rserve without a username and password.

Note: If you configured Tableau Server for an Rserve connection prior to Tableau 10.0, these settings had different names. The configuration vizqlserver.extsvc was introduced in Tableau Server 10.1 as a replacement for vizqlserver.rserve. However, there is reverse compatibility in Tableau Server 10.1. In Tableau Server 10.1, vizqlserver.extsvc settings will be checked for first, but if there are no vizqlserver.extsvc settings then Tableau Server will check for vizqlserver.rserve settings.

For information on how to use the above `tabadmin set` options to configure Tableau Server, see [Change Tableau Server's Configuration from the Command Line](#).

Tableau cannot verify that workbooks that use R will render properly on Tableau Server. There might be scenarios where a required statistical library is available on a user's machine but not on the Rserve instance that Tableau Server is using.

For views that cannot be rendered in Tableau Server because of an R script error, you see a warning error when you publish the workbook:

This worksheet contains R scripts, which cannot be viewed on the target platform until the administrator configures an Rserve connection.

You cannot publish a workbook that contains R scripting to Tableau Online.

Because Tableau Server provides an authentication mechanism, it can be more secure to expose Rserve functionality to users through Tableau Server than in Tableau Desktop.

Troubleshooting R Connections

This topic describes errors you can receive when Tableau is connected to an Rserve server—or attempting to connect to an Rserve server. It also lists issues that you may encounter as you use R with Tableau.

Note: Tableau technical support cannot assist with writing, troubleshooting, or debugging R scripts.

Error Messages below

[Other Issues](#) on page 1625

Error Messages

Error Message	Comments
Unsupported type passed as an argument to SCRIPT function.	Tableau can only export the following Tableau data types to R: <ul style="list-style-type: none">• Number (Decimal)• Number (Whole)• Boolean• String• Date• Date/Time
Unexpected number of results returned by SCRIPT function. Function expected %2 values; %1 values were returned.	The R script result must be either a scalar or vector of length one that is replicated for all rows, or a vector of length equal to the number of rows in the Tableau result table.
The result returned by the SCRIPT function is of an unexpected type.	Occurs when an invalid data type is received. Tableau can only import the following data types from R: <ul style="list-style-type: none">• RDouble• RInteger• RLogical• RCharacter

	This error is also reported if the result is null or if there was a script execution error for which Tableau could not collect an explanation from R.
This worksheet contains R scripts, which are not supported on the target platform.	This error is reported when you attempt to publish a worksheet containing an R script to a server that does not allow R scripts because <code>vizqlserver.script.disabled</code> is set to true.
This worksheet contains R scripts, which cannot be viewed on the target platform until the administrator configures an Rserve connection.	<p>This error is reported when the server is configured to prevent Desktop from publishing worksheets with R scripts because <code>vizqlserver.script.disabled</code> is set to false. (The setting is named contrary to its meaning: true means Desktop can publish worksheets with R scripts, false means Desktop cannot publish worksheets with R scripts.)</p> <p>Setting <code>vizqlserver.script.disabled</code> to false will not prevent Tableau Online users from using <code>tabcmd</code> to publish workbooks with R scripts; however, the resulting views will generate this error when opened in a browser.</p>
An error occurred while communicating with the Rserve service.	Tableau runs all R scripts inside of the "try" R function. This error is displayed along with an R generated error message when the "try" function traps an R evaluation error.
This Rserve connection does not support authentication. Try connecting without specifying a password.	
Authentication failed. Please provide a valid Rserve username and password.	
An unsupported authentication type	Either disable Rserve authentication or change to plaintext password authentication.

is enabled in Rserve.	
No Rserve connection configured. Specify a server name and try again.	See Pass Expressions to R on page 1618 .
The calculation '%1' contains a disabled R script. Configure an Rserve connection to enable custom R scripts.	See Pass Expressions to R on page 1618 .
The workbook you are attempting to publish contains custom R scripts. Custom R scripts are not allowed in public workbooks.	You cannot publish workbooks containing R scripts to Tableau Public.
Tableau Public does not support running R scripts. To take advantage of R integration, upgrade to Tableau Professional.	The "Tableau Public" in this error refers to Tableau Desktop Public.
Tableau Reader cannot load custom R scripts. To take advantage of R integration, upgrade to Tableau Professional.	You cannot view workbooks containing R scripts in Tableau Reader.
Rserve is busy or not responding. Failed to create socket to	Tableau has timed out a read pending on connection to Rserve – the timeout is 250ms. An IPC connection Read of the Rserve protocol header has thrown an exception.

Rserve.	
Unrecognized Rserve signature.	The Rserve header signature must be "Rsrv".
Unrecognized Rserve version.	The Rserve header version must be "0103".
Unrecognized Rserve protocol.	The Rserve header protocol must be "QAP1".
Authentication failure when connecting to R.	Tableau attempted and failed to authenticate with Rserve. Verify that you entered a valid password.
Incorrect number of bytes in parameter/body.	
The length of data which Tableau read does not equal the length promised by the header.	
Unrecognized response type.	The transport protocol type when reading a result was not SEXP as expected.
Excessively long <type>vector.	Unreasonably large number measuring the length of data sent to or from Rserve possibly indicating a corrupt protocol header.
Invalid Rserve command.	Tableau may have improperly implemented the Rserve protocol.
Response from server was Error " << (uint32_t)status << ". See Rsrv.h for details.	Various error conditions are documented in the comments in Rsrv.h.
Excessively long header offset.	This may be due to a garbled header with an unreasonable offset to response data.

Rserve socket failed.	A login, script evaluation, read pending check, result read, or Tableau field to R script argument assignment threw a non-standard exception.
-----------------------	---

Other Issues

SCRIPT Functions Run Even in Logical Statements That Evaluate as False

A function that sends an expression to a running Rserve instance will be executed even when it is within a logical statement that would otherwise prevent it from being executed. This is true for logical functions such as IF, IIF, and CASE. For example:

```
IF 1==0 THEN  
  [[R script code]]  
ELSE  
  "1 does not equal 0"  
END
```

User-defined Encodings Supported by the Statistical File Connector

This topic lists character encodings supported by Tableau Statistical File connector. The list includes single-byte, multi-byte, and Unicode user-defined encodings, as well as single-byte and multi-byte encodings that cannot currently be mapped to the corresponding SAS encodings.

In this topic:

- Single-byte user-defined encodings
 - Multi-byte user-defined encodings
 - Unicode user-defined encodings
 - Single-byte encodings that can't be mapped to SAS encodings
 - Multi-byte encodings that can't be mapped to SAS encodings
-

Single-byte user-defined encodings

ASCII

CSASCII

US-ASCII
US
ISO_646.IRV:1991
ISO646-US
646
ISO-IR-6
IBM367
CP367
ANSI_X3.4-1986
ANSI_X3.4-1968
ISO-8859-1
CSISOLATIN1
LATIN1
L1
ISO_8859-1:1987
ISO8859-1
ISO-IR-100
ISO-8859-1
IBM819
CP819
ISO-8859-15
LATIN-9
ISO_8859-15:1998
ISO_8859-15
ISO8859-15
ISO-IR-203
IBM850
CSPC850MULTILINGUAL
CP850
850
WINDOWS-1252

MS-ANSI
CP1252
ISO-8859-7
CSISOLATINGREEKISO_8859-7:1987
ISO_8859-7
ISO-IR-126
ISO-8859-7
GREEK8
GREEK
ELOT_928
ECMA-118
WINDOWS-1253
MS-GREEK
CP1253
ISO-8859-10
CSISOLATIN6
LATIN6
L6
ISO_8859-10:1992
ISO_8859-10
ISO8859-10
ISO-IR-157
WINDOWS-1257
WINBALTRIM
CP1257
ISO-8859-2
CSISOLATIN2
LATIN2
L2
ISO_8859-2:1987
ISO_8859-2

ISO8859-2
ISO-IR-101
IBM852
CSPCP852
CP852
852
WINDOWS-1250
MS-EE
CP1250
ISO-8859-5
CSISOLATINCYRILLIC
ISO_8859-5:1988
ISO_8859-5
ISO8859-5
ISO-IR-144
CYRILLIC
WINDOWS-1251
MS-CYRL
CP1251
CP866
CSIBM866
IBM866
866
TIS-620
TIS620.2533-1
TIS620.2533-0
TIS620.2529-1
TIS620-0
TIS620
ISO-IR-166
ISO-8859-11

CP874
CSISOLATIN5
LATIN5
L5
ISO_8859-9:1989
ISO_8859-9
ISO8859-9
ISO-8859-9
ISO-IR-148
CSIBM857
IBM857
CP857
857
WINDOWS-1254
MS-TURK
CP1254
CP1129
VPS
WINDOWS-1258
CP1258
ISO-8959-6
CSISOLATINARABIC
ISO_8859-6:1987
ISO_8859-6
ISO8859-6
ISO-IR-127
ECMA-114
ASMO-708
ARABIC
WINDOWS-1256
MS-ARAB

CP1256
ISO-8859-8
CSISOLATINHEBREW
ISO_8859-8:1988
ISO_8859-8
ISO8859-8
ISO-IR-138
HEBREW
IBM864
CSIBM864
CP864
WINDOWS-1255
MS-HEBR
CP1255
IBM862
CSPC862LATINHEBREW0x2E
CP862
862

Multi-byte user-defined encodings

CP936
WINDOWS-936
MS936
GBK
GB2312
CSISO58GB231280
ISO-IR-58
GB_2312-80
CHINESE
ISO-2022-CN
CP950

windows-950
ms-950
ms950
CSBIG5
CN-BIG5
BIGFIVE BIG5
BIG-FIVE
BIG-5
BIG5HKSCS
BIG5-HKSCS
EUC-TW
CSEUCTW
EUCTW
EUC-JP
CSEUCPKDFMTJAPANESE
EXTENDED_UNIX_CODE_PACKED_FORMAT_FOR_JAPANESE
EUCJP
EUC-JP
ISO-2022-JP
CSISO2022JP
ISO-2022-JP
CSSHIFTJIS
SJIS
SHIFT_JIS
SHIFT-JIS
MS_KANJI
CP932
EUC-KR
CSEUCKR
EUCKR
EUC-KR

UHC
CP949
EUC-CN
CSGB2312
GB2312
EUCCN
CN-GB

Unicode user-defined encodings

TF-8
UCS-2
UCS-2BE
UCS-2LE
UCS-4
UCS-4BE
UTF-16
UTF-16BE
UTF-16LE
UTF-32
UTF-32LE
UTF-32BE
UTF-7

Single-byte encodings that can't be mapped to SAS encodings

MACROMAN
CSMACINTOSH
MACINTOSH
MAC
ISO-8859-14
LATIN8
L8

ISO_8859-14:1998

ISO_8859-14

ISO8859-14

ISO-IR-199

ISO-CELTIC

MACGREEK

MACICELAND

ISO-8859-3

CSISOLATIN3

LATIN3

L3

ISO_8859-3:1988

ISO_8859-3

ISO8859-3

ISO-IR-109

ISO-8959-4

CSISOLATIN4

LATIN4

L4

ISO_8859-4:1988

ISO_8859-4

ISO8859-4

ISO-IR-110

ISO-8959-13

LATIN7

L7

ISO_8859-13

ISO8859-13

ISO-IR-179

ISO-8859-13

MACCENTRALEUROPE

MACCROATIAN

IBM855

CSIBM855

CP855

855

KOI8-R

CSKOI8R

MACCYRILLIC

KOI8-U

CSKOI8R

MACUKRAINIAN

ISO-8859-16

LATIN10

L10

ISO_8859-16:2001

ISO_8859-16

ISO8859-16

ISO-IR-226

MACROMANIAN

ARMSCII-8

GEORGIAN-ACADEMY

MACTURKISH

TCVN

VISCII

CSVISCII

VISCII1.1-1

MACARABIC

MACHEBREW

WINDOWS-874

Multi-byte encodings that can't be mapped to SAS encodings

GB18030

HZ

HZ-GB-2312

CSISO2022JP

ISO-2022-JP

JOHAB

JOHAB

CP1361

ISO-2022-KR

CSISO2022KR

ISO-2022-KR

ISO-2022-JP

CSISO2022JP

ISO-2022-JP-1

ISO-2022-JP-2

CSISO2022JP2

ISO-2022-CN

CSISO2022CN

ISO-2022-CN-EXT

Glossary

A

action

An interaction that you can add to your views. There are three types of action: Filter, Highlight, and URL.

ad-hoc calculation

A calculation that you can create and update as you work with a field on a shelf in the view. Also known as type-in calculation or in-line calculation.

aggregation

A result of a mathematical operation applied to a measure. Predefined aggregations include summation and average. You can convert dimensions to measures by aggregating them as a count. For relational data sources, all measures must be either aggregated or disaggregated (unless they appear on the Filters shelf). Tableau aggregates measures, usually as a summation, when you place them on a shelf. For multidimensional (OLAP) data sources, aggregations are defined when the cube is created and cannot be modified in Tableau.

alias

An alternative name that you can assign to a field or to a dimension member.

Analytics pane

A pane on the left side of your workbook that provides quick and easy access to common analytic features in Tableau. From the Analytics pane, you can drag reference lines, box plots, trend lines forecasts, and other items into your view. Toggle between the Data pane and the Analytics pane by clicking one of the tabs at the top of the side bar.

B

bin

A user-defined grouping of measures in the data source.

blending data

The process of combining data from different data source types in a view.

The first data source that you use in the view becomes the primary data source. The remaining data sources become the secondary.

bookmark

A .tbn file in the Bookmarks folder in the Tableau repository that contains a single worksheet. Much like web browser bookmarks, .tbn files are a convenient way to quickly display different analyses.

C

calculated field

A new field that you create by using a formula to modify the existing fields in your data source.

caption

A description of the current view on the active worksheet. For example, "Sum of Sales for each Market". You can automatically generate captions or create your own custom captions. Show and hide the caption by selecting Worksheet > Show Caption.

cell

A basic element of any table that you create in Tableau. You can control cells to enhance your data view, which is useful for text tables and heat maps.

color legend

An area of the view that displays the colors associated with a measure or dimension member. The default legend is modified when you place a dimension or a measure on the Color property.

Color property

A property on the Marks card that enables you to encode data by assigning different colors to the marks in a view. The property accepts measures and dimensions. When you place a dimension on the Color property, Tableau separates the marks according to the members in the dimension, and assigns a unique color to each member. When you place a measure on the Color property, Tableau draws each mark with a different color using a continuous range. In both cases, a legend describes the color encoding.

Columns shelf

A shelf at the top of the workbook that you use to create the columns of a data table. The shelf accepts any number of dimensions and measures. When you place a dimension on the Columns shelf, Tableau creates headers for the members of that dimension. When you place a measure on the

Columns shelf, Tableau creates quantitative axes for that measure. See also Rows shelf.

crosstab

A text table view. Use text tables to display the numbers associated with dimension members.

cube

A data source that is connected to a multidimensional database. Also known as multidimensional data source. For example, data sources that connect to Microsoft Analysis Services or Oracle Essbase are called cubes.

custom geocoding

A process of adding your own location data to extend the built-in geocoding.

D

dashboard

A combination of several views arranged on a single page. Use dashboards to compare and monitor a variety of data simultaneously.

Data Interpreter

A tool that parses your Excel or Google Sheets data source to help prepare your data for analysis.

Data pane

A pane on the left side of the workbook that displays the fields of the data sources to which Tableau is connected. The fields are divided into dimensions and measures. The Data pane also displays custom fields such as calculations, binned fields, and groups. You build views of your data by dragging fields from the Data pane onto the various shelves that are a part of every worksheet.

data source

The source of data outside or inside Tableau.

Data Source page

A page where you can set up your data source. The Data Source page generally consists of four main areas: left pane, join area, preview area, and metadata area.

data view

see: view

Detail property

A property on the Marks card that you can use to separate the marks in a view according to the level of detail (that is, members) of a dimension. The Detail property works only on aggregated data.

dimension

A field of categorical data. Dimensions typically hold discrete data such as hierarchies and members that cannot be aggregated. Examples of dimensions include dates, customer names, and customer segments. See also: measure.

E

encoding

A visual representation of your data. You can encode your data by color, shape, size, and path using the associated worksheet shelves.

extract

A saved subset of a data source that you can use to improve performance and analyze offline. You can create an extract by defining filters and limits that include the data you want in the extract.

F

field

A dimension or a measure in a database. For relational data sources, fields are the columns of a table. For cube (multidimensional) data sources, fields

are the dimensions of a cube. Each dimension or column contains a unique attribute of the data, such as customer name, sales, or product type.

field label

A row or column heading that indicates the data field used to create the view. For example, a view that has rows for East, Central, and West might have a Region field label at the top of the column indicating that each row is a member of the Region field.

Filters shelf

A shelf on the left of the workbook that you can use to exclude data from a view by filtering it using measures and dimensions.

forecast

A calculation that predicts future trends based on current trends and data.

Format pane

A pane that contains formatting settings that control the entire worksheet, as well as individual fields in the view. When open, the Format pane appears on the left side of the workbook.

G

group

A field you can use to combine dimension members into higher level categories. For example, you could group a dimension that contains states into regions. Groups are marked with a paper clip icon in the Data pane.

H

header

A label for member names for each field that you place on Rows or Columns.

hexagonal binning

A technique for clustering data in a two-dimensional plane.

I

in-line calculation

see: ad-hoc calculation

J

join area

An area at the top of the Data Source page where you can drag tables or select queries or cubes to set up your data source. See also: Data Source page.

join condition

A relationship between fields in a join. You can define the relationship in the join area of the data source. See also: joining.

joining

A process of combining data from multiple tables in same-type data sources.

L

level of detail (LOD) expression

A syntax that supports aggregation at dimensionalities other than the view level. With level of detail expressions, you can attach one or more dimensions to any aggregate expression.

LOD

see: level of detail (LOD) expression

M

marks

A part of the view that visually represents one or more rows in a data source. A mark can be, for example, a bar, line, or square. You can control the type, color, and size of marks.

Marks card

A card to the left of the view where you can drag fields to control mark properties such as type, color, size, shape, label, tooltip, and detail.

measure

A field of quantitative data. Measures are fields that are dependent variables. They are typically quantitative fields or calculated fields like sales, temperature, or frequency. You can also create discrete measures in Tableau. See also: dimension.

metadata area

An area that can be accessed from the Data Source page by clicking the metadata button and that displays the fields in your data source as rows so that you can quickly examine the structure of your data source and perform routine management tasks, such as renaming fields or hiding multiple fields at once. When connected to cube data, the metadata displays by default.

See also: Data Source page.

multidimensional data source

see: cube

P

packaged workbook

A single zip file with a .twbx extension that contains a workbook along with any supporting local file data sources and background images. Use this format to package your work for sharing with others who don't have access to the data.

Pages shelf

A shelf to the left of the view that you can use to split a view into a sequence of pages based on the members and values in a discrete or continuous field. Adding a field to the Pages shelf is like adding a field to the Rows shelf, except that a new page is created for each new row.

pane

An area in the table that is created by the intersection of rows and columns. Tables consist of one or more panes. The number of panes in a view depends on the number and type of fields placed on the Rows and Columns shelves.

parameter

A dynamic value that can replace a constant value in calculations, filters, and reference lines.

pass-through function

A function that sends SQL expressions directly to the database to access custom database functions.

Path property

A property on the Marks card that you can use to encode data by connecting marks using a particular drawing order. The property accepts measures and dimensions. Dimensions connect the marks according to the members in the dimension. If the dimension is a date, the drawing order is given by the date order. If the dimension contains words, the line is drawn based on the order of the words in the data source. Measures connect the marks according to the values of the measure. The measure can be aggregated or disaggregated. See also: path.

pill

A field in the view.

preview area

An area at the bottom of the Data Source page where you can review the fields and the first 10,000 rows of the data in the data source. You can also use preview area to make general modifications to your data source, such as hiding or renaming a field, or changing its data type. See also: Data Source page.

primary data source

The first data source that you use in a blended view. See also: blending data, secondary data source.

Q

query

A set of formalized instructions that Tableau uses to communicate with databases. Common query languages include SQL and MDX. Every time you build a view of your data, Tableau translates your actions into queries and retrieves the requested information from the data source. If you are building a dense data view, you can turn queries off until all the fields you want are placed on shelves.

R

relational data source

A data source that is connected to a relational database.

relational database

A database that presents information in tables with rows and columns. Examples of relational databases that Tableau supports are Excel workbooks, Access databases, comma-delimited text files, MySQL database, and Tableau Data Extract files.

Repository

A collection of workbooks, bookmarks, data sources, and logs. By default, the Tableau Repository is located on the drive where Tableau is installed, in the My Documents\My Tableau Repository folder.

Rows shelf

A shelf at the top of the workbook that you can use to create the rows of a data table. The shelf accepts any number of dimensions and measures. When you place a dimension on the Rows shelf, Tableau creates headers for the members of that dimension. When you place a measure on the Rows shelf, Tableau creates quantitative axes for that measure.

S

secondary data source

A second or subsequent data source that you use in a blended view. See also: blending data, primary data source.

set

A custom field that defines a subset of data based on some conditions. A set can be based on a computed condition or on a specific data point in the view. Sets appear at the bottom of the Data pane in the Sets area.

Shape legend

A legend that displays the shapes associated with dimension members. The legend appears on worksheets that have a dimension placed on the Shape shelf.

Shape shelf

A shelf to the left of the view that you can use to encode data by assigning different shapes to the marks in the view. The Shape shelf accepts dimensions only. When you place a dimension on the shelf, Tableau separates the marks according to the members of the dimension, and a legend describes the encoding. You cannot place a measure on the shelf because measures do not contain members.

sheet

A view (also known as worksheet), dashboard, or story. Sheets appear as tabs at the bottom of the workbook.

shelves

Named areas to the left and top of the view. You build views by placing fields onto the shelves. Some shelves are available only when you select certain mark types. For example, the Shape shelf is available only when you select the Shape mark type.

Size shelf

A shelf to the left of the view that allows you can use to encode data by assigning different sizes to the marks in the view. The Size shelf accepts

measures and dimensions. When you place a dimension on the shelf, Tableau separates the marks according to the members in the dimension, and assigns a unique size to each member. When you place a measure on the shelf, Tableau assigns a different size to each mark using a continuous range.

small multiples

A view that contains small multiple charts of the same type.

story

A sheet that contains a sequence of views or dashboards that work together to convey information.

story point

An individual view in a story.

T

table

A visual presentation of a data view. Tables consist of panes, headers, and cells.

table calculation

A computation that uses data from multiple rows in the database and that is applied to the values in the table.

Text shelf

A shelf to the left of the view that you can use to view the numbers associated with a view, and to encode data by assigning text labels to the marks. The shelf accepts measures and dimensions. The most common view using the Text shelf is a text table.

tooltip

Data details that appear when you hover over one or more marks in the view.

type-in calculation

see: ad-hoc calculation

V

view

A representation of your data in a Tableau worksheet or dashboard. You can create data views by placing fields on shelves.

W

workbook

A file with a .twb extension that contains one or more worksheets (and possibly also dashboards and stories).

worksheet

A sheet where you build views of your data by dragging fields onto shelves.

Quick Starts

Quick Starts are one-page articles that let you quickly review how a feature works and that provide links to more detailed information.

This topic lists the Quick Start categories for Tableau Desktop. You can find Quick Starts for other Tableau products here:

- [Tableau Server](#)
- [Tableau Online](#)

Quick Starts about Data Sources and Connections

The first step to analyzing data in Tableau is connecting to a data source.

Quick Starts in this section help you get up and running with specific data sources, and also introduce working with extracts of your data as opposed to working with a live connection, explain ways to keep your data fresh, introduce working with your data in the grid on the Data Source page, and talk about ways to package and publish your data for others to see or work with.

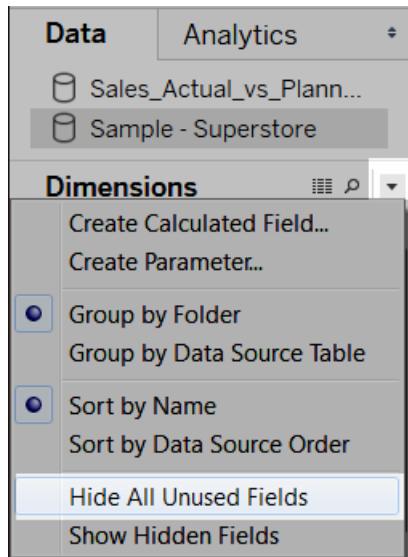
Quick Start: Aggregated Extracts

You can optimize your extracts by only including the data you need. Specifically, you can exclude columns, create filters to limit the number of rows, aggregate data, and roll up dates.

1 Hide unused fields

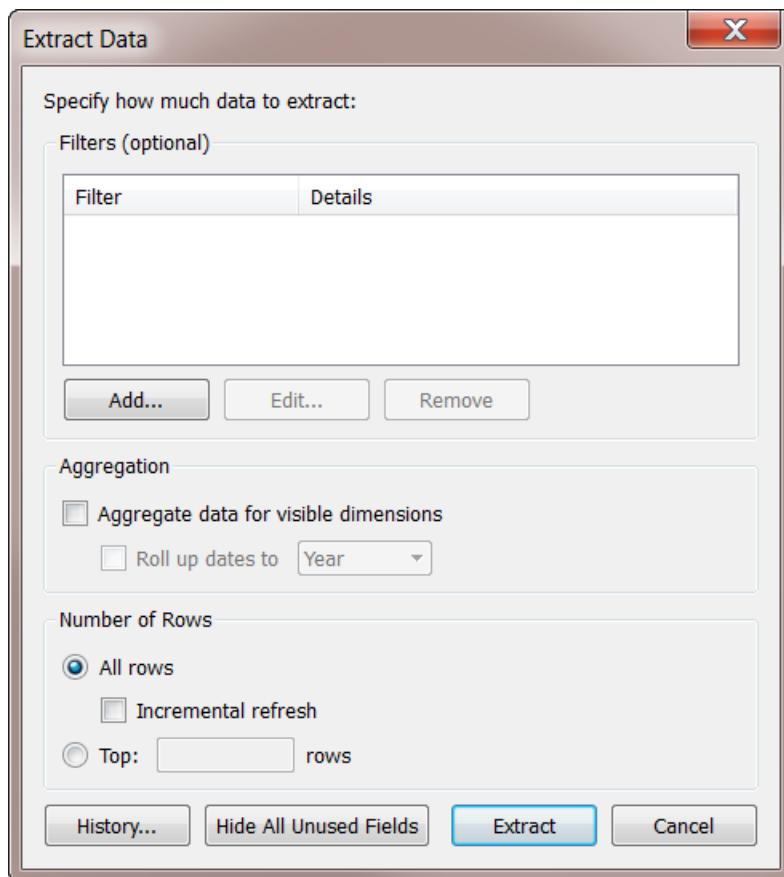
Hidden fields are not included when you create an extract. By hiding unnecessary fields before you create an extract, you make the extract smaller and improve performance.

On the **Data** pane, click the drop-down arrow to the right of Dimensions and select **Hide All Unused Fields** from the context menu.



2 Define filters for the extract

Right-click the data source and select **Extract Data**.



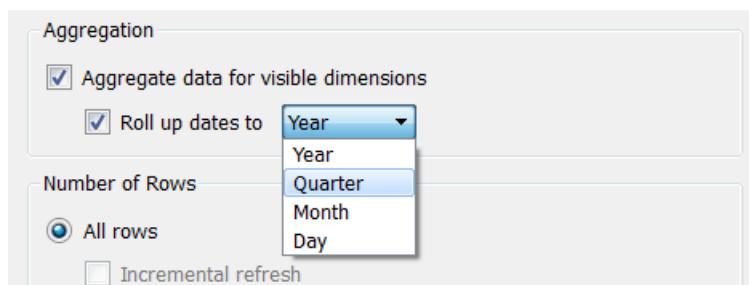
In the dialog box, optionally define one or more filters to limit how much data gets extracted.

Global filters are automatically added as filters on the extract.

3 Aggregate data for visible dimensions

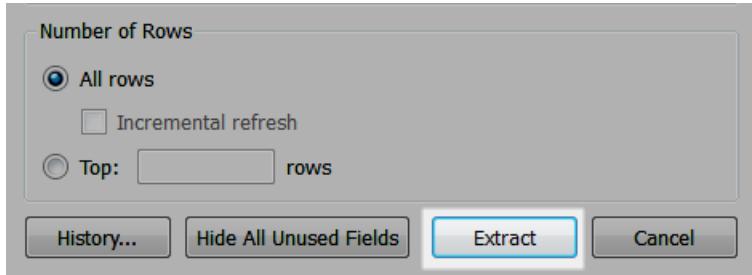
After defining any filters, select **Aggregate data for visible dimensions**. This aggregates the data using the default aggregation for measures.

If the extract contains Date fields, you can also select **Roll up dates to** to adjust date granularity and further minimizing the size of the extract.



4 Click Extract

When ready, click **Extract** and specify a location for the extract (TDE) file. The extract will only contain the visible fields and the data will be aggregated as you specified.



Refresh the extract at any time by selecting **Data > Refresh All Extracts** from the toolbar.

Quick Start: Automatically Update Workbook to Use Published Data Source

Publishing a data source is now as easy as clicking **Publish data source** on the **Server** menu. When you publish a data source, by default your workbook is updated to use the published data source.

Use the published data source (default)

When you publish a data source to Tableau Server or Tableau Online, the publishing process replaces the local version of the data source in your workbook with the published version. This new automated process replaces the manual four-step process you had to follow to update your workbook to use a published version of the data source, rather than the local version.

In addition to streamlining the publishing process, this new functionality also makes it easy for you to keep your workbook data refreshed. Simply schedule the extract for periodic refreshes, and your workbook will be automatically updated with the refreshed data.

Use the local data source

If you need to, you can continue to use the local data source, rather than a published data source. For example, you may want to use a local data source if the purpose of your workbook is to manage the data source.

- If the data source has not yet been published, when you publish the data source to Tableau Server or Tableau Online, clear the **Update workbook to use the published data source** check box.
- If the data source has been published and your workbook has been updated to use the published data source, click the **Undo** button to go back to the local data source. Note that this action does not remove the data source from the server.

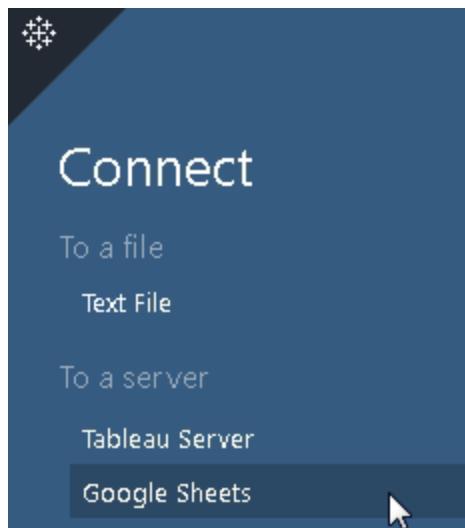
Quick Start: Combine Tables Using Cross-Database Joins

When related data is stored in tables across different databases, you can use a cross-database join to combine the tables.

To create a cross-database, you must create a multi-connection Tableau data source. You do so by adding and then connecting to each of the different databases (including Excel and text files) before you join.

1 Connect to data

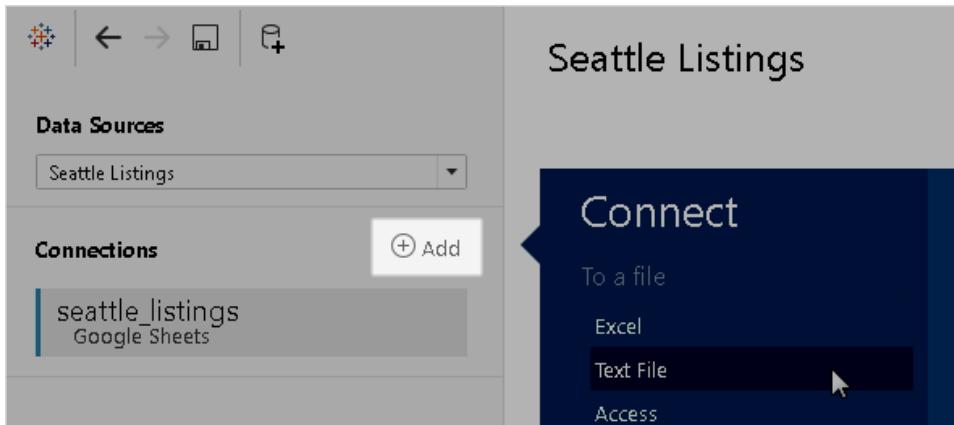
On the start page, under **Connect**, click a connector to connect to your data. This step automatically creates the first connection in the Tableau data source.



2 Add another connection

Add new connections only if the data comes from different databases. For example, add new connections to each database if you need to analyze tables from a text file and tables from Google Sheets.

In the left pane, under **Connections**, click the **Add** button to add a new connection to the Tableau data source.



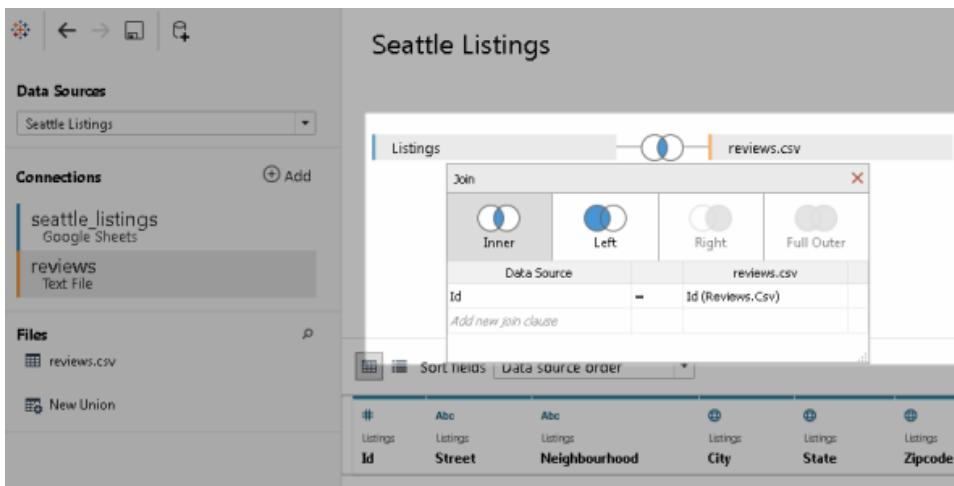
A data source with more than one connection is referred to as a **multi-connection data source**.

Note: If the connector you want is not available from the **Connect** list, cross-database joins are not supported for the combination of sources that you want to join. This includes connections to cube data (e.g., Microsoft Analysis Services), extract-only data (e.g., Salesforce, Google Analytics, and OData), and Tableau Server data sources. Instead of joining tables, consider using data blending. For more information, see [Blend Your Data on page 369](#).

3 Combine your data using a cross-database join

Drag tables from each connection to the canvas in order to create the cross-database join.

After you have joined the tables, you can update the data grid to show the new columns and rows in the multi-connection data source. Tableau colors the tables in the canvas and the columns in the data grid to show you which connection the data comes from.



By default, Tableau creates an inner join, but you can choose a left join.

You can fix broken joins

Before you join tables using a cross-database join, make sure that the data types of the join keys match. If the data types of the join keys don't match, the join breaks, which is indicated by a red exclamation point. To fix a broken join, take one of the following steps:

- For text or Excel-based data, modify the data type of one of the text or Excel fields in the join key using the data type menu in the data grid.
- For most connections, you can use the **Convert to custom SQL** option to change the data type of one of the fields in the join key. The **Convert to custom SQL** option is available only when the data source contains only one connection. In this case, remove the second connection and then select **Data > Convert to custom SQL**.

4 Continue to prepare your multi-connection data source for analysis

After you've created a cross-database join, continue to prepare your multi-connection data source for analysis. You can rename and reset fields, create calculations, clean your Excel data with the Data Interpreter, change the data types of text and Excel-based fields, and so on. Keep the following points in mind:

Calculations and multi-connection data sources: Only a subset of calculations can be used in a multi-connection data source. You can use a specific calculation if it is 1.) supported by all the connections in the multi-connection data source and 2.) supported by Tableau extracts.

Pivot data from within a connection: To pivot data, you must use text columns or Excel columns from the same connection.

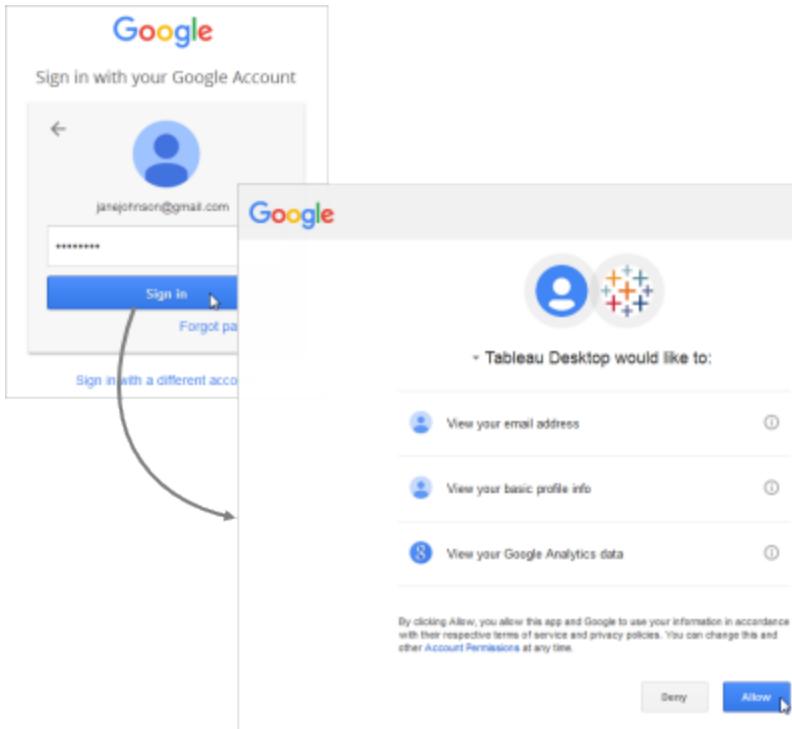
Union data within a connection: To union data, you must use text tables or Excel tables from the same connection.

Quick Start: Connect to Google Analytics

Google Analytics is a service that provides website owners data and detailed statistics about visitors to their website. As a website owner who subscribes to Google Analytics, you can use Tableau Desktop to connect directly to your Google Analytics data, analyze, and obtain rich insights about your website.

1 Sign in to Google

On the start page, under **Connect**, click **Google Analytics**. Then, sign in to Google Analytics with the email and password associated with your Google Analytics account. To complete the initial connection process, click **Accept** to let Tableau Desktop access your Google Analytics account.



2 Select a date range

Select from a list of preset date ranges to limit your data.

Step 2: Select Filters:

Date Range:	Last 30 Days
	Last 30 Days
	Today
Segment:	Yesterday
	Last Week
	Last Month
	Last Year
	This Week to Yesterday
	This Month to Yesterday
	This Year to Yesterday
	Last Week to Yesterday
	Last Month to Yesterday
	Last Year to Yesterday
Sort fields	Fixed Range
	Fixed Start

Google Analytics can provide complete data only up to the previous full day. For example, if you choose **Last 30 days**, data will be retrieved for the last 30-day period ending yesterday.

3 Select dimensions and measures

You can select up to seven dimensions and ten measures. When selecting measures, you can use a predefined measure group or customize your own group.

Step 3: Select up to 7 Dimensions and 10 Measures:

Choose a Measure Group: Custom

Add dimension

Add measure

Enter search text

Acquisition
Advertising
Behavior
Custom Variables
DoubleClick Campaign Manager
Ecommerce
Goal Conversions
Related Products
Social
Time
Users

Some dimensions and measures cannot be used together. For more information, see [Google Analytics Dimensions & Metrics Reference](#) on the Google developer website.

4 Extract data

After completing the steps in the top area of the data source page, Tableau creates an extract of your data.

Data Worksheet Dashboard Story Analysis Map Format Server Window Help

New Data Source Ctrl+D

Paste Data Ctrl+V

Refresh All Extracts...

Edit Relationships...

Replace Data Source...

Upgrade Data Sources...

Tableau Data Server

Google Analytics (Tech Support Test (05823864))

Extract Data... Use Extract Extract Refresh View Data... Rename... Duplicate Close Extract Replace Data Source... Date Properties... Publish to Server... Add to Saved Data Sources... Properties...

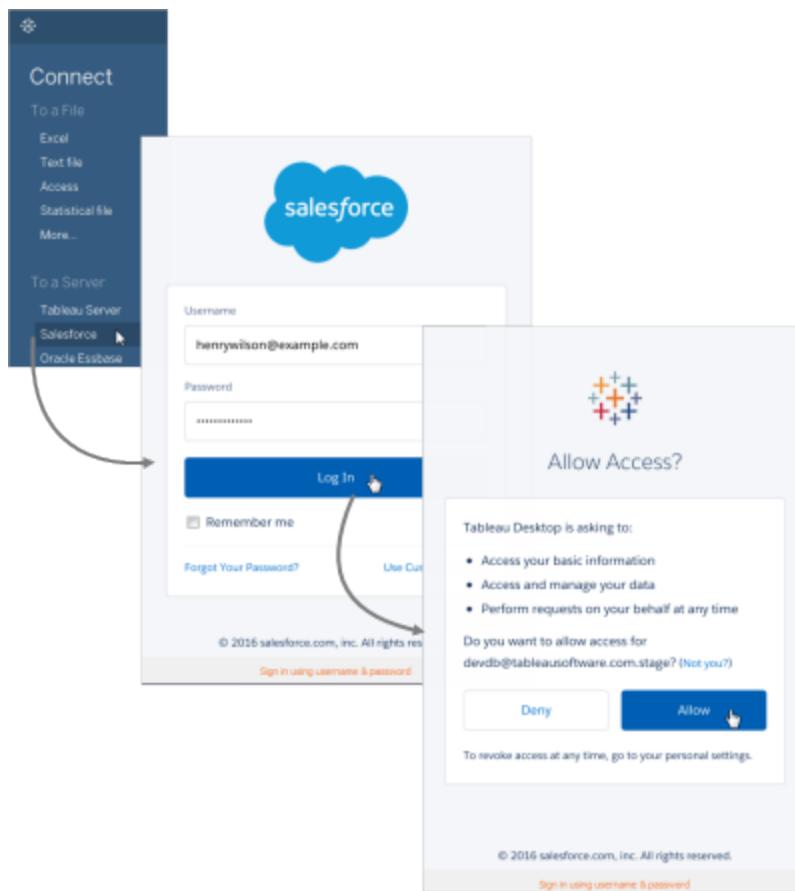
To update your extract you can perform a full or incremental extract refresh.

Quick Start: Connect to Salesforce

Salesforce.com is a web-based customer relationship management (CRM) application. You can use Tableau Desktop to directly connect to Salesforce.com so that you can analyze and answer questions about your accounts, contacts, opportunities, and more. In addition to Salesforce.com, you can use Tableau Desktop to connect to Force.com and Database.com.

1 Log in to Salesforce

On the start page, under **Connect**, click **Salesforce**. Enter the user name and password for your Salesforce account, and then click **Log In**. Then, in the Allow Access dialog box, click **Allow**.



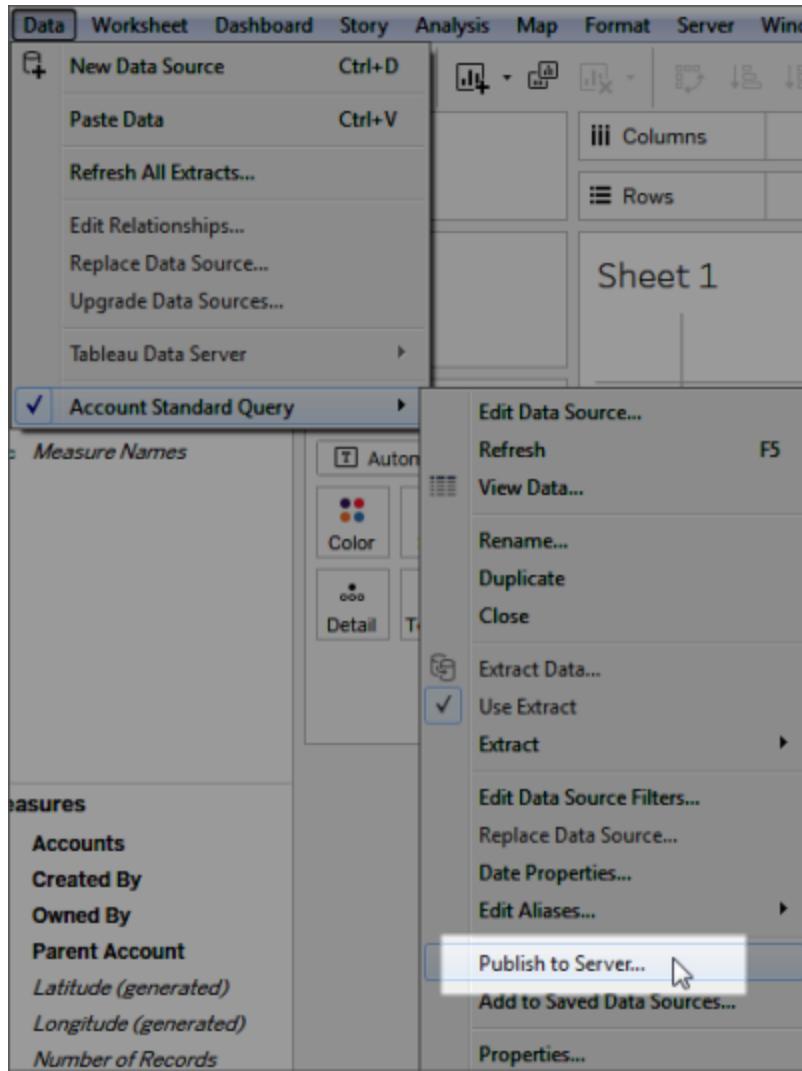
2 Use built-in queries

Connect to your Salesforce data using one of the built-in queries under **Standard Connection**. These queries contain commonly used tables and joins.

Alternatively, you can create your own custom query.

3 Extract data

After you connect to Salesforce and complete the data source page by selecting or creating a query, click the sheet tab to start your analysis. When you click the sheet tab, Tableau creates an extract of your data. **Note:** Because of limitations in Salesforce, the extract process can take a very long time, depending on the query.



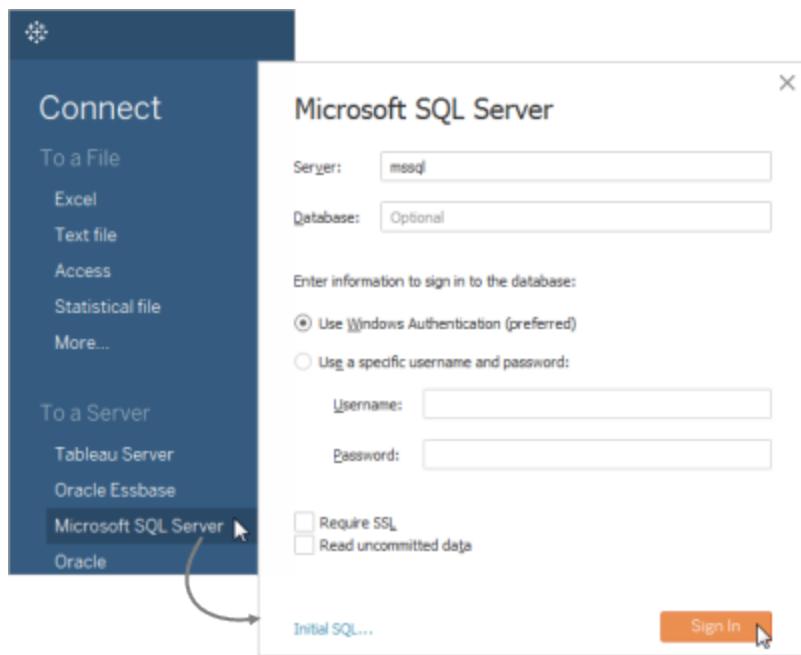
After the initial extract of your Salesforce data is created, we recommend that you publish the extract to Tableau Server and use the extract for any subsequent connections. To update your extract, you can perform a full or incremental extract refresh.

Quick Start: Connect to Your Data

In any analytics project, preparing your data for analysis is just as important as the analysis itself. Tableau makes it easy for you to connect to and bring your data into Tableau, combine, and quickly review the data before you begin analysis.

1 Connect to your data

On the start page, under **Connect**, click a file or database type (in this example, Microsoft SQL Server), and then enter the authentication information to connect.



2 Combine data (joins)

Select the file or database or schema. Then click and drag associated sheets, tables, or stored procedures to join sets. Otherwise, click the join icon to define the join type and clause. Choose from inner, left, right, and full outer. If you don't want to join your data, click the sheet tab to start your analysis.

The data that you drag into the top area of the data source page will be the data source from which you will build your view. Tableau never changes the underlying data.

3 Review the data and change data types

Click **Update Now** to review the first 1,000 rows of the data source (or enter the number you want to see in the **Rows** text box). Every time you make changes to the joins, click **Update Now**. If you want changes in the joins to be automatically reflected in the grid, click **Automatically Update**. You can then make other modifications, such as changing the data type or renaming a column.

The screenshot shows the Power BI Data Source blade. At the top, there are two tables: 'myData' (State, Product Type) and 'Sales' (Product, Sales). Below them are two buttons: 'Update Now' (highlighted with a blue box and a cursor icon) and 'Automatically Update'. A curved arrow points from the 'Automatically Update' button down to the second table. The second table displays four rows of data:

State	Product Type	Product	Sales
CA	Coffee	Amaretto	219.0
NY	Coffee	Columbian	188.0
IL	Tea	Green Tea	250.0
WA	Espresso	Caffe Latte	212.0

4 Connection options and data source filters

At the top of the data source page, select a live or extract connection to the data source. You can also edit your extract and add data source filters. Click the sheet tab to start your analysis.

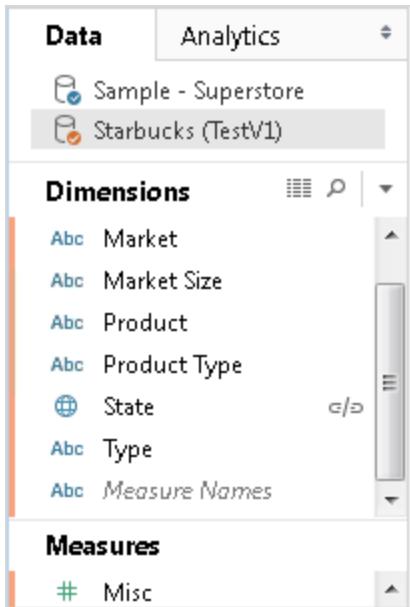
The screenshot shows the DataBlender application window. At the top, there's a menu bar with File, Data, Server, Window, and Help. Below the menu is a toolbar with icons for back, forward, search, and other functions. The main area is titled "My se..." and shows a connection status with "Live" selected. There are tabs for "Connections" (with "mssql" selected) and "Extract". A "Filters" section with a "0" count and an "Add" button is also present. On the left, a sidebar titled "Database" shows a dropdown for "TestV1" and a list of tables: Election, FischerIris, Loan, myData, mySe...Data, NumericBins, and New Custom SQL. Below that is a section for "Stored Procedures" with "LargeResults" and "nonRQL" listed. A "Go to Worksheet" button is located here. The central part of the screen displays a table titled "myData" with columns: Market, State, Product Type, and Sales. The data rows are: West, CA, Coffee, Amaretto; East, NY, Coffee, Columbian; Central, IL, Tea, GreenTea; West, WA, Espresso, Caffe Latte. At the bottom, there's a "Data Source" dropdown set to "Sheet 1" and a set of sheet navigation icons.

Quick Start: Data Blending

If you need to analyze data from different databases, but the databases are not supported in a multi-connection data source, or the tables you need to analyze are at different levels of detail, use data blending. Data blending allows you to combine data from multiple data sources onto a single sheet. The **Data** pane includes a list of all the data sources you're connected to. Relationships between the data sources are automatically created--or you can define custom relationships.

1 Connect to Multiple Data Sources

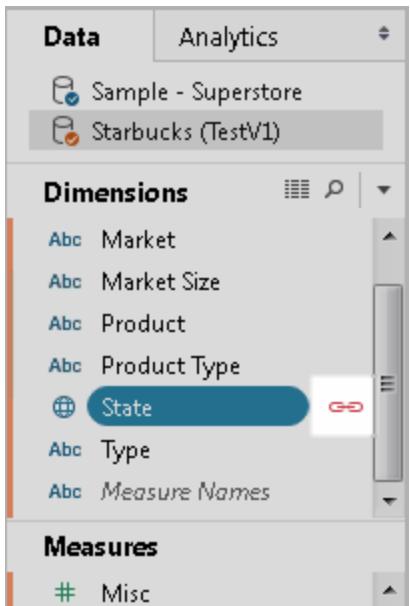
Select **Data > Connect to Data** and connect to your data. Then connect to another data set. The data sources are shown in a drop-down list at the top of the **Data** pane.



After you add a field to the view, the **Data** pane shows the secondary data source with a colored bar down the left side. If you don't see such a bar, it means that Tableau has not yet determined which data source is the primary data source. As you start adding fields to the view the bar will appear.

2 Use Fields From Both Data Sources

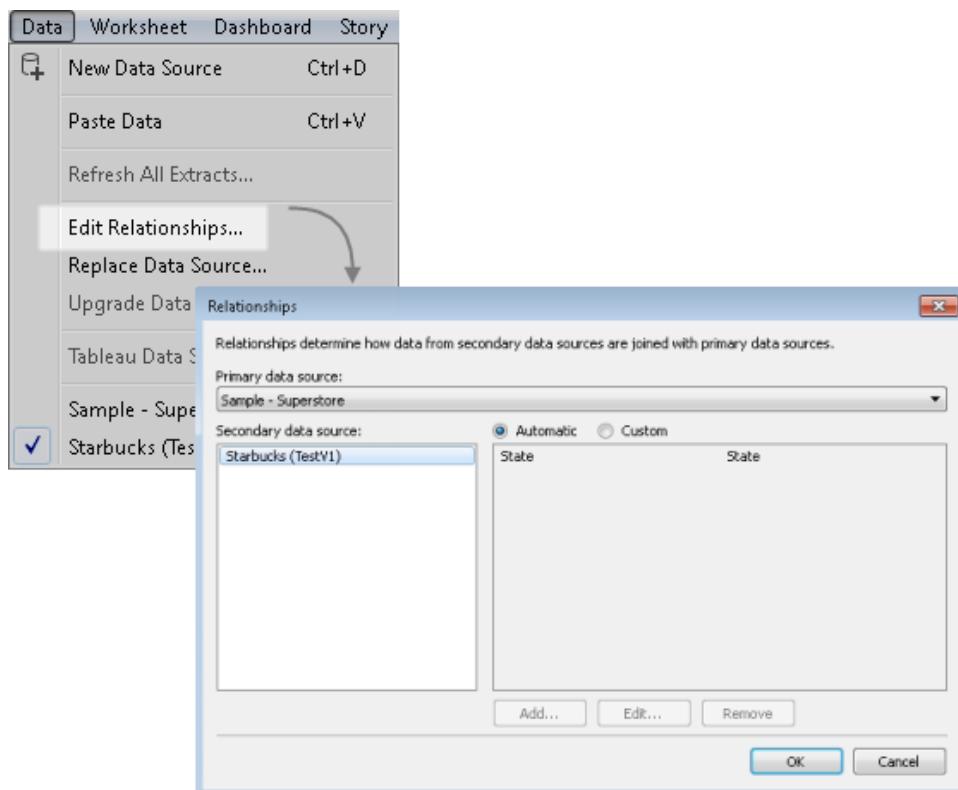
When you drag a field from a secondary data source into the view, Tableau automatically creates relationships between the data sources (if it can). The related fields are marked with a Link icon in the **Data** pane.



In this view, the bars are created using the Sales values from the primary connection (Superstore Sales) while the reference lines show the Sales Plan values from the secondary connection (Sales Plan).

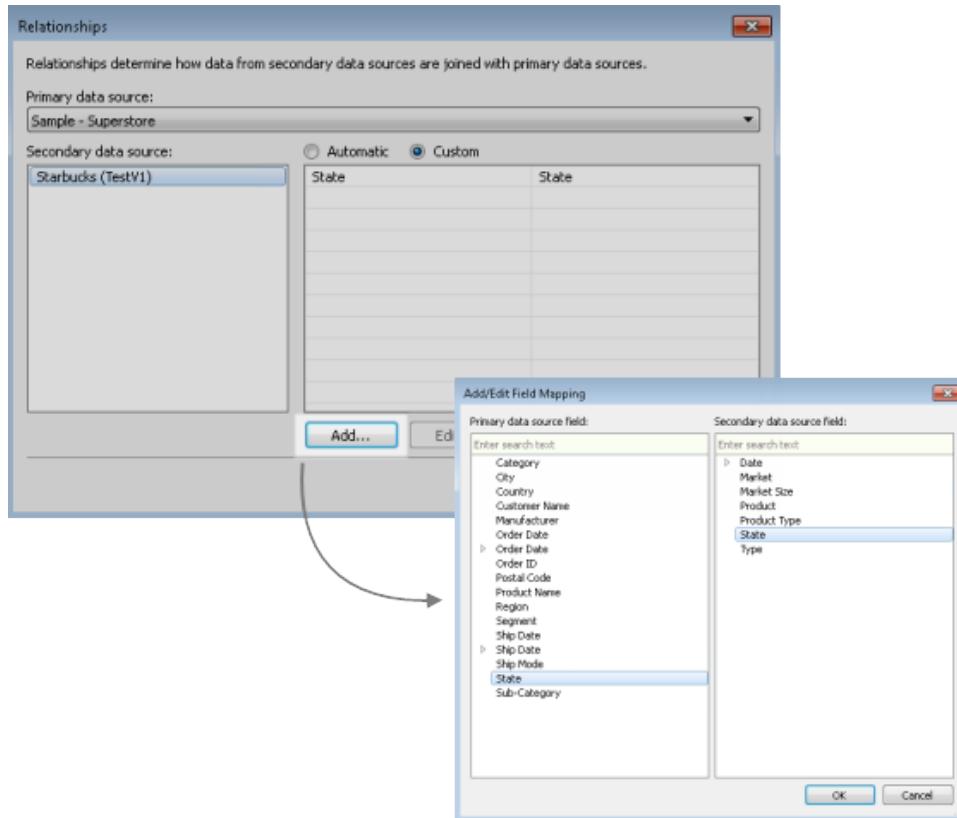
3 Create Custom Relationships

Relationships are automatically created based on field names. But you can define custom relationships by choosing **Data > Edit Relationships**.



4 Edit and Define Relationships

In the Relationships dialog box, select the primary data source in the drop-down at the top of the dialog box. Then select the secondary data source. Automatic relationships are shown, but you can select **Custom** and then click **Add** to link pairs of fields from the respective data sources.



Quick Start: Data Grid Enhancements

The following enhancements made to the grid on the **Data Source** page help you better see the data in your data source and prepare it for analysis.

A Sort columns and rows

Sort columns: Sort columns in the grid and metadata grid by selecting a sort option from the **Sort fields** drop-down list. Sort the columns by table or data source order.

Sort rows: Sort rows by clicking the sort button. Click the sort button once to sort rows in ascending order, click the sort button again to sort rows in descending order, and then click the sort button a final time to clear the sort.

A Modified sort state can occur when some tasks cause new columns to be added to the grid.

B Change or reset field names

Rename column: Double-click the name of the column to rename the field.

Reset name: If you've renamed a field, click the column drop-down menu, and then select **Reset Name** to revert back to the original name of the field. You can also select multiple columns and perform the same action.

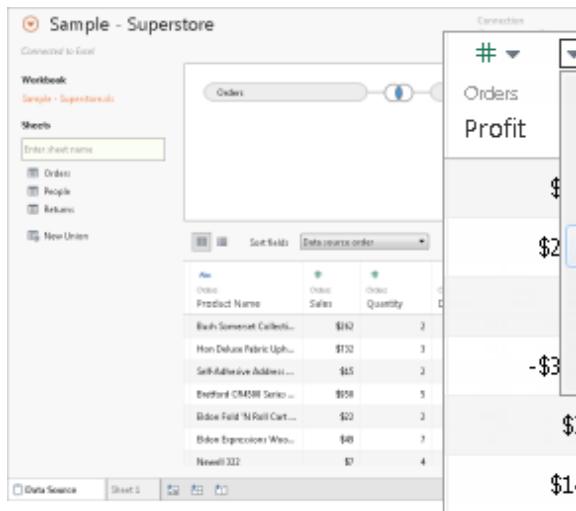
Original name indicates the name specified in the underlying data. You can use the **Revert command to reset field names that have changed as a result of naming improvements Tableau has automatically made.**

to the data source. For more information, see [Field Type Detection and Naming Improvements](#).

C Create new columns and see extract data

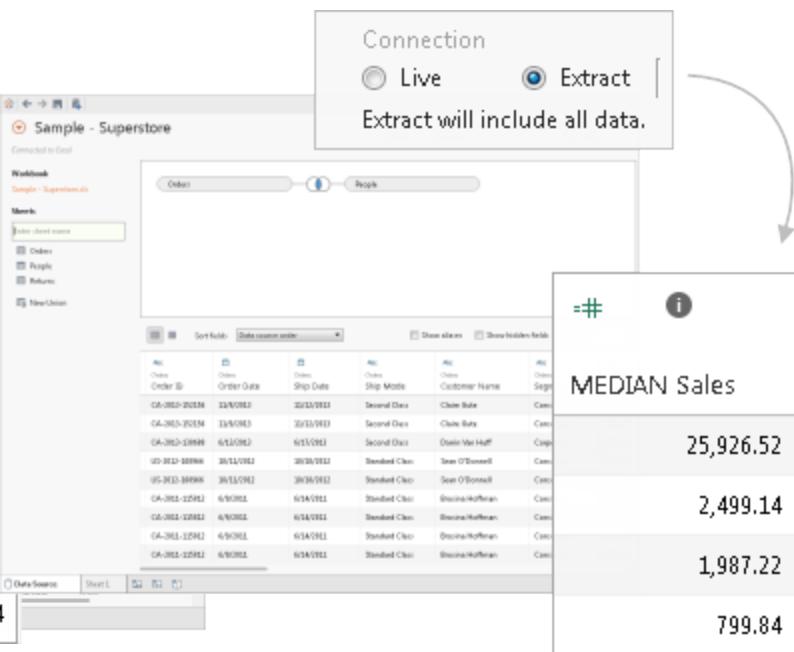
Create new calculations: Create new calculations, groups, or bins based on existing fields in the data source. Click the column drop-down arrow and select one of the respective options.

If you create an aggregate calculation, the values displayed in the grid are not meaningful until the calculation is used in context of the view's level of detail.



See extract data: For the Web

Data Connector, file- and relational-based data sources in Extract mode, you can see extract data in the grid, including extract only calculations like MEDIAN.



When in Extract mode, the row order of the values in the data source might be different than the row order of values in Live mode.

D Copy values

Copy values: Copy values in the grid by selecting the values and then pressing **Ctrl+C** (Command-C on a Mac). Alternatively, to copy values in the metadata grid, select the values, right-click, and then select **Copy**.

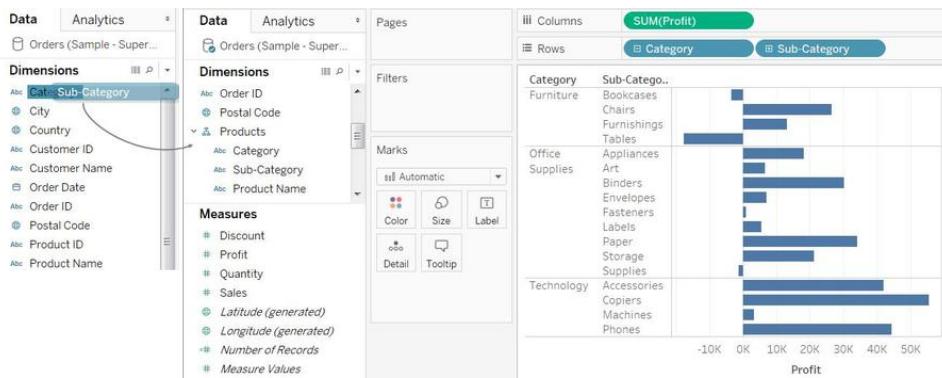
Quick Start: Hierarchies

When you connect to a data source, Tableau automatically separates date fields into hierarchies so you can easily break down the view by year, quarter, month, etc. You can also create your own custom hierarchies. For example, if you have a set of fields named Region,

State, and County, you can create a hierarchy from these fields so that you can quickly drill down between levels.

Drag and drop fields in the Data pane

Create hierarchies simply by dragging fields on top of other fields in the **Data** pane. After that you can reorder the levels of the hierarchy by dragging and dropping within the **Data** pane.



When you drag a member of the hierarchy to the view, you can use the plus and minus (+/-) buttons on the field on the shelf to drill through the levels.

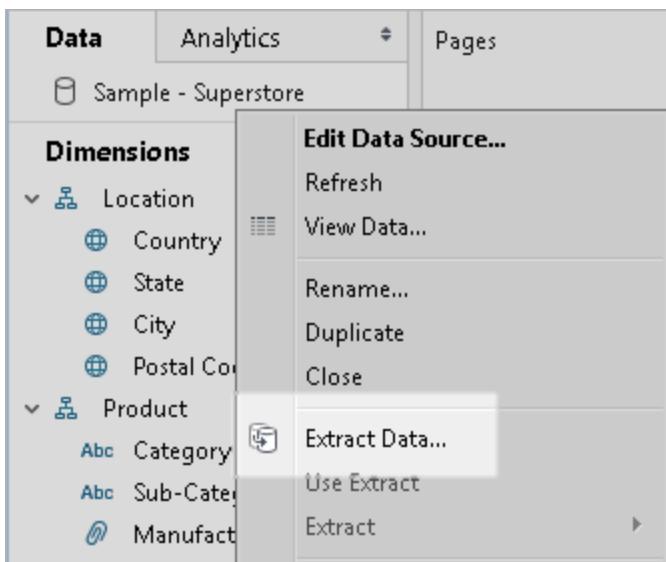
Note: You can also create hierarchies by selecting multiple fields in the **Data** pane, right-clicking and selecting **Create Hierarchy**.

Quick Start: Incremental Extract Updates

When you import all or some of your data into Tableau's data engine, you create a data extract. After you create the initial extract, you can set up an incremental refresh so that importing new data doesn't require you to rebuild the entire extract. An incremental refresh can be defined by the values in a specified column. For example, if you've created an extract that has date values, you can define the incremental refresh to only add new rows if there are additions in the date column.

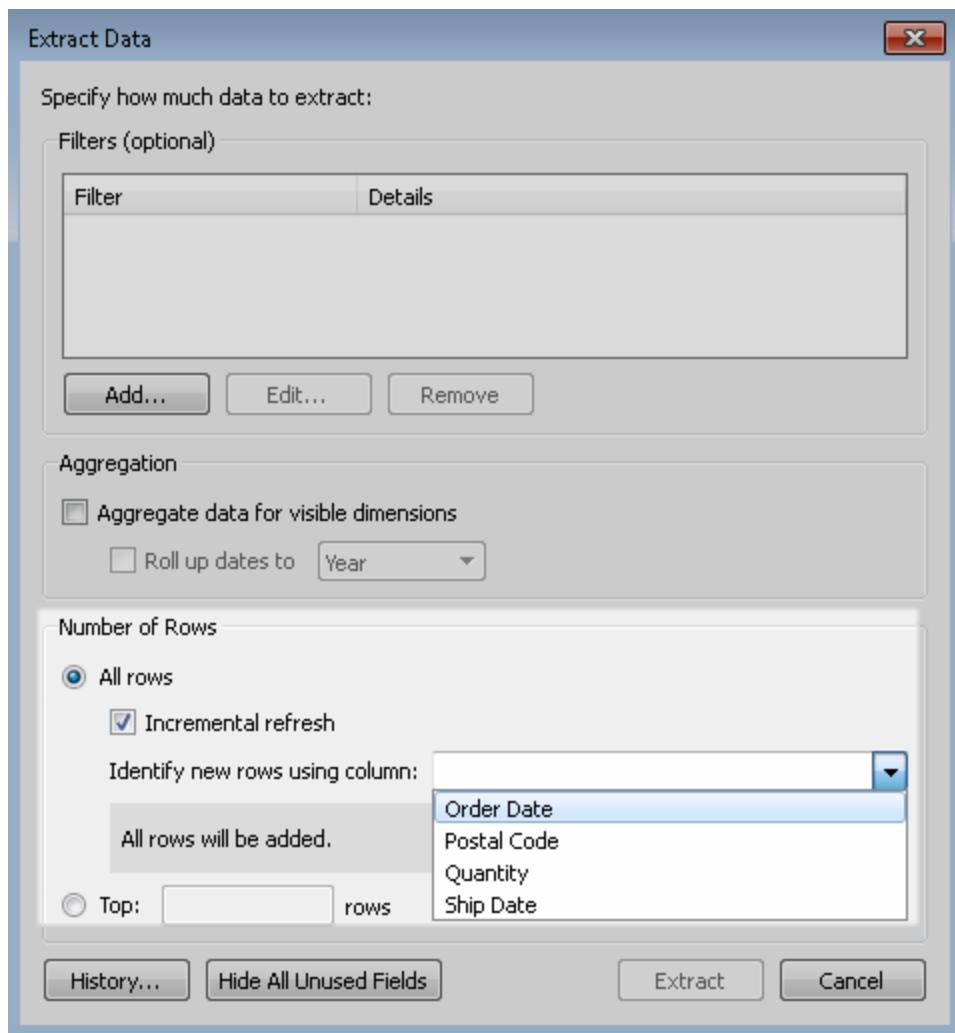
1 Create an Extract

Right-click the data source (or control-click on a Mac) and choose **Extract Data** to create the initial extract. You must do a full extract before you can set up an incremental refresh.



2 Configure the Incremental Refresh

After the data is imported, open the extract dialog box again and choose **Incremental refresh**. Select the column you want to use to identify new records from the drop-down. When finished, click **Extract**.



3 View Extract History

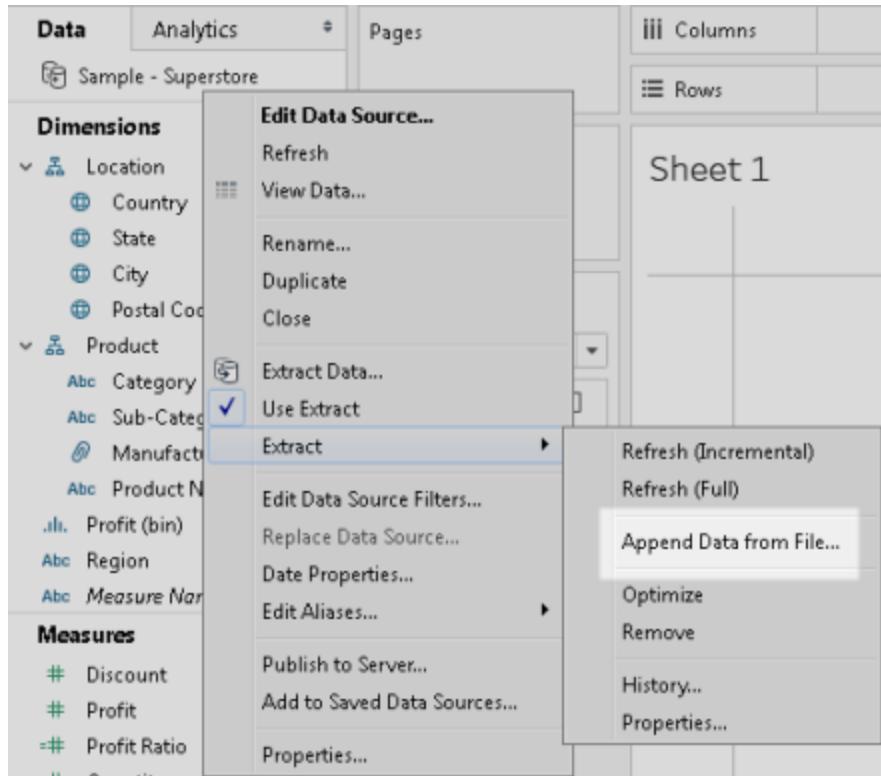
You can see a list of the updates that have been made since the initial extract by right-clicking the extract data source and choosing **Extract > History**.

The screenshot shows the Tableau Data pane with the 'Data' tab selected. A context menu is open over the 'Extract' data source, which is highlighted with a blue selection bar. The menu items are:

- Edit Data Source...
- Refresh
- View Data...
- Rename...
- Duplicate
- Close
- Extract Data... (highlighted with a blue selection bar)
- Use Extract (highlighted with a blue selection bar)
- Extract
 - Refresh (Incremental)
 - Refresh (Full)
- Edit Data Source Filters...
- Replace Data Source...
- Date Properties...
- Edit Aliases...
- Publish to Server...
- Add to Saved Data Sources...
- Properties...

4 Add Data from a File

If you regularly receive updates in a CSV or Excel file, you can add data to the extract directly from the local file by right-clicking the extract data source and choosing **Extract > Append Data from File**.



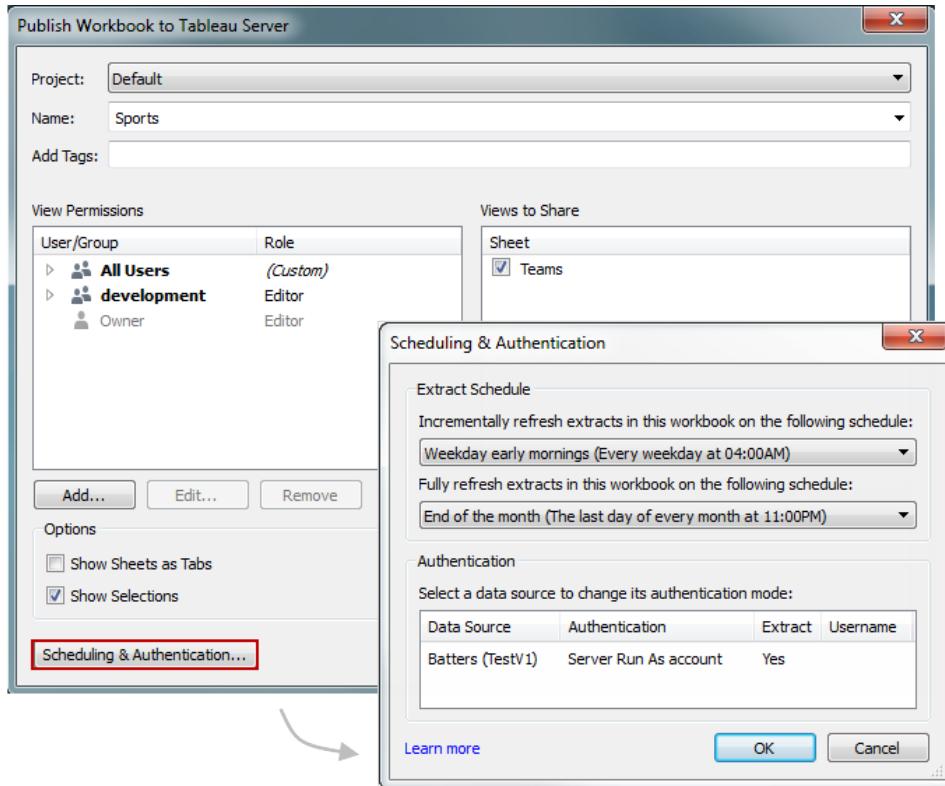
The local file must have the same columns as the original data source.

Quick Start: Manage Incremental Extracts

When you publish a workbook that has an incremental extract, you can associate it with up to two refresh tasks that Tableau Server will handle for you: An incremental refresh of the extract and a full refresh. After you publish the workbook, you or a Tableau Server administrator can modify any tasks that are associated with the workbook. You can also delete tasks or add more.

1 Publish and Assign a Schedule

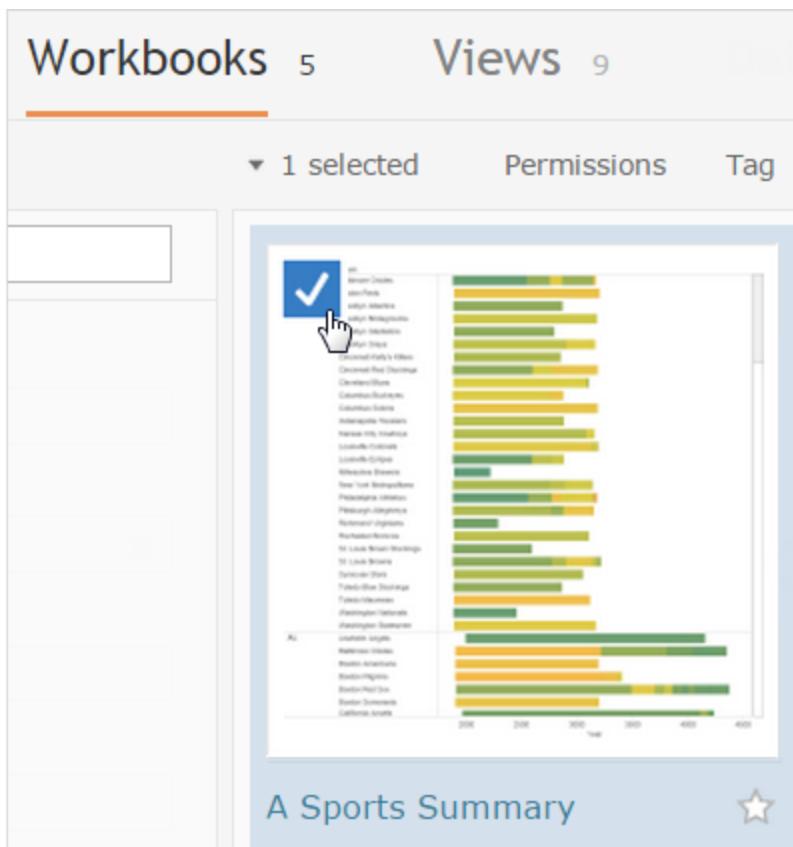
In Tableau Desktop, after you create a workbook that uses an extract, go to **Server > Publish Workbook**, and click **Scheduling & Authentication**. Next, choose schedules for your refreshes and click OK.



After you publish in Tableau Desktop and choose your refresh schedules, Tableau Server handles the refresh tasks for you.

2 Select the Workbook

To modify a workbook's scheduled task, sign in to Tableau Server and on the **Workbooks** page, select the workbook:



3 Access the Refresh Schedule

Click **Refresh Schedule**.

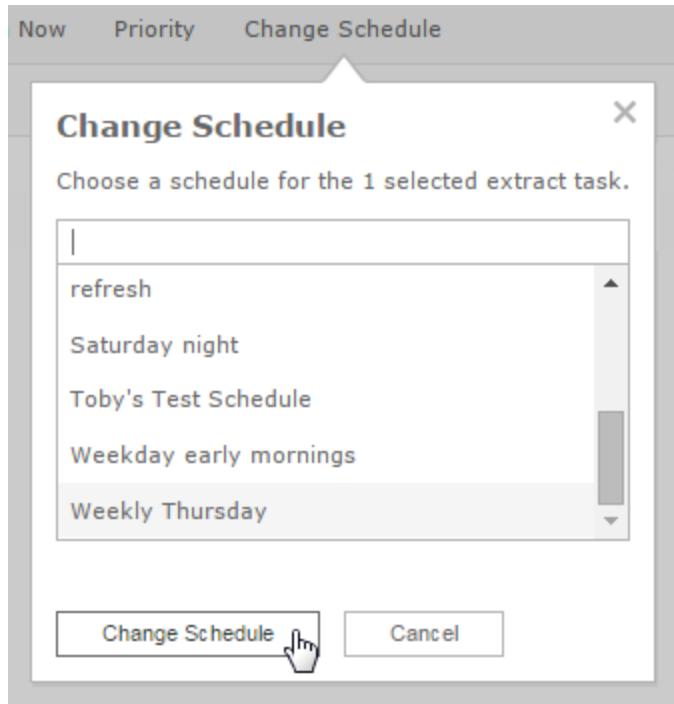


Select the check box for the refresh task you want to modify:

Refresh Type	Schedule
<input type="checkbox"/> Full refresh	End of the month - Once a month
<input checked="" type="checkbox"/> Incremental refresh	Saturday night - Weekly

4 Edit, Delete, or Add More Tasks

Select the action you want to take—for example, **Change Schedule**—and make your selection. You can also delete the task, change its priority, or add more refresh tasks.



Quick Start: Manage Metadata

Use the metadata area to quickly examine and change the structure of the Tableau data source. You can use the metadata area to review the columns in your data and their data types, do routine management tasks such as renaming a column, hiding a column, changing the data type of a column, and changing the geographical role of the column.

Note: Metadata management tasks vary when working with a multi-dimensional data source.

1 Go to the metadata area

After you have set up your data source, click the metadata area button. If you are connected to a cube (multidimensional) data source, the metadata area is shown by default. For other data source types, such as Salesforce or Tableau data extracts (TDE), the default view will vary.

The screenshot shows the Tableau Data Source Editor interface. On the left, there's a sidebar with 'Connections' (mssql, Microsoft SQL Server) and 'Database' (TestV1). The main area is titled 'REI' and shows the 'REI' table selected. Below it is a grid of metadata:

Field Name	Table	Remote Field Name
Date	REI	Date
Abc Category1	REI	Category1
Abc Category2	REI	Category2
Abc Category3	REI	Category3

2 Review metadata

The columns of the data source are represented as rows in the metadata area. Each row shows the data type of the column, the column name in the Tableau data source, the name of the table in the underlying data from which the column comes, and the name of the column in the underlying data.

The screenshot shows a comparison between raw data and Tableau metadata. At the top is a preview of raw data with four columns: Date, Category1, Category2, and Category3. Below it is the Tableau metadata grid:

Field Name	Table	Remote Field Name
Date	REI	Date
Abc Category1	REI	Category1
Abc Category2	REI	Category2
Abc Category3	REI	Category3
Abc Category4	REI	Category4

A curved arrow points from the raw data preview down to the metadata grid.

3 Manage metadata

Depending on the type of data you're connected to, you can perform routine management tasks like hiding multiple columns at once or quickly renaming columns. For example, to hide multiple columns at once, press the CTRL or Command key, select the columns you want to hide, click the drop-down arrow next to a column name, and then select **Hide**. Changes that you make in the metadata area of the data source do not modify your underlying data.

Field Name	Table	Remote Field Name
Date	REI	Date
Abc Category1	REI	Category1
Abc Category2		Rename
Abc Category3		Copy Values
Abc Category4		Hide
Abc Category5		Aliases...
Abc Trans Description		Create Calculated Field...
# Total Units		Create Group...
# Total Sale Revenue		Split
		Custom Split...
		Describe...

Quick Start: Packaged Workbooks

Often, workbooks reference files that are stored on your local computer. These files might be Excel or Access files you used as data sources, Tableau extracts, or background image files. You can save copies of these local files along with your workbook in a single file called a **packaged workbook**. Packaged workbooks are easy to share with others.

1 Save As a Packaged Workbook

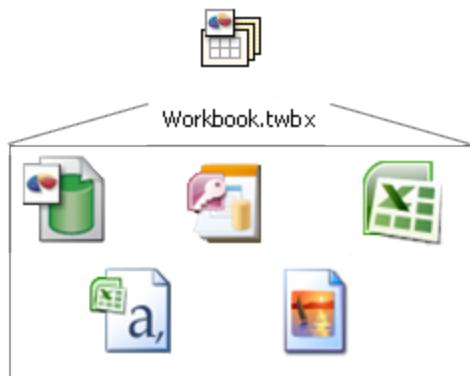
On the top menu in your workbook, click **File > Save As** and select **Tableau Packaged Workbook** as the type.



The file extension for packaged workbooks is **.twbx**.

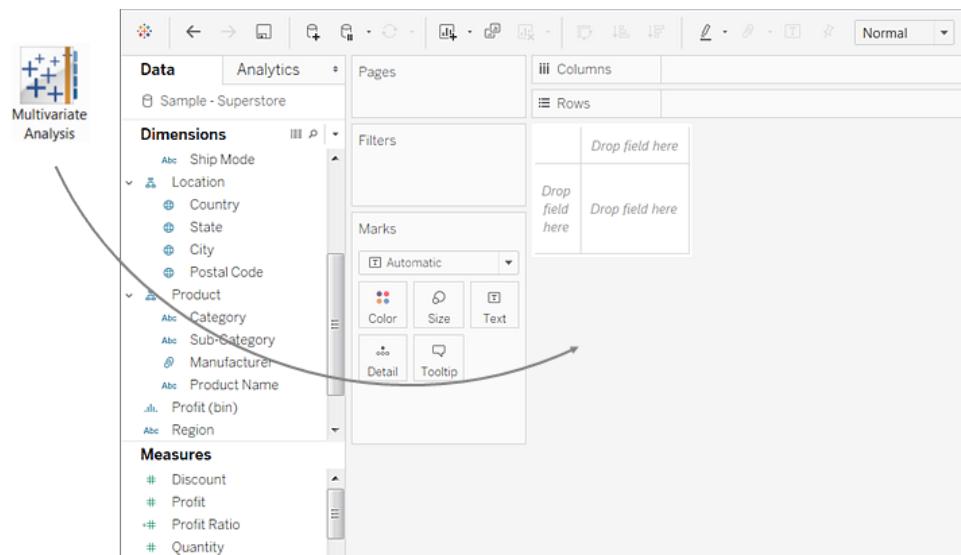
2 Share Your Work with Others

A packaged workbook is a single file that contains a copy of all of the referenced local files, so you can easily share it using email, a shared file system, or by publishing to Tableau Server.



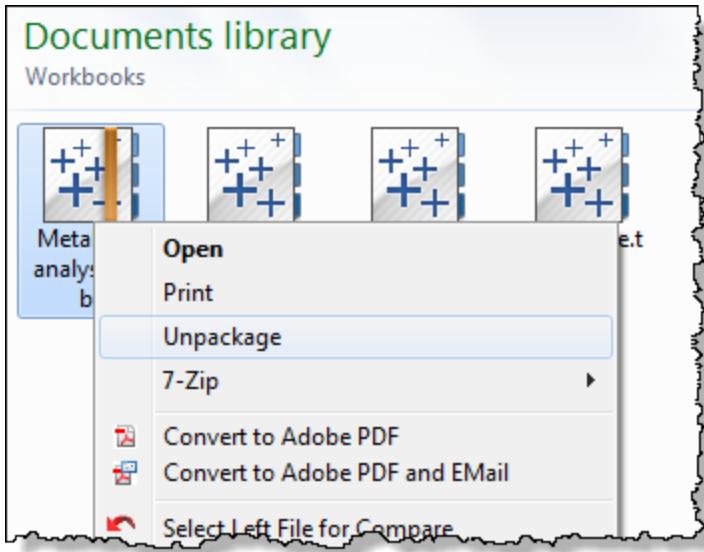
3 Opening Packaged Workbooks

To open a packaged workbook in Tableau Desktop, double-click the file on your computer, or drag and drop the file onto the running application.



4 Unpacking Workbooks

You can access the packaged files at anytime by unpackaging the workbook. Right-click the packaged workbook file on your PC (or control-click on a Mac) and select **Unpackage**.



Unpacking produces a workbook file (twb) along with a folder containing all of the referenced files.

Quick Start: Pivot Data from Columns to Rows

Sometimes, analyzing data that is stored in a crosstab format can be difficult in Tableau. When working with this type of data, you can pivot your data from crosstab format into columnar format.

For example, suppose you have the number of devices sold by quarter for three vendors in three separate fields. You can pivot your data so that the vendor is in one field and the number of devices sold is in another field.

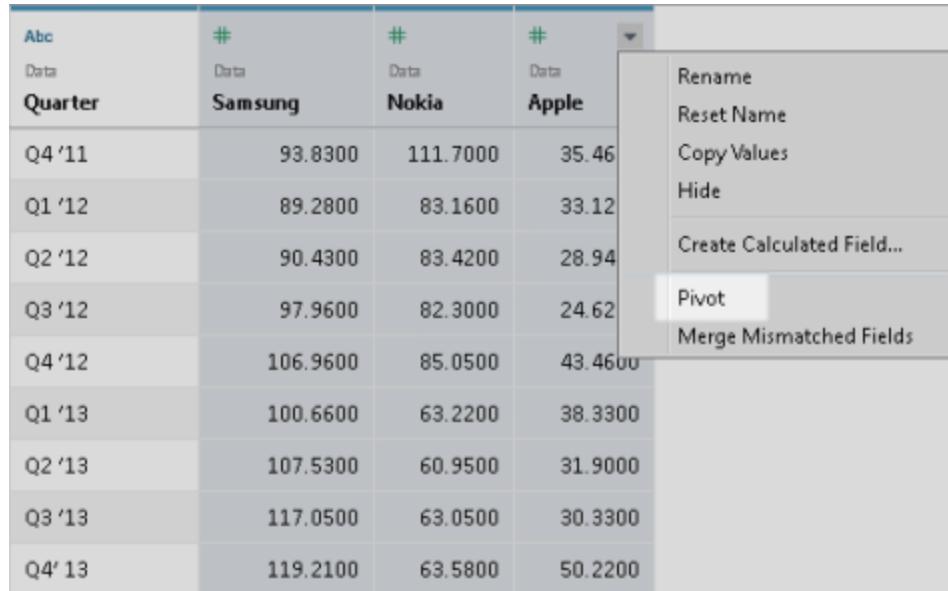
Quarter	Samsung	Nokia	Apple
Q1 '12	89.2800	83.1600	33.1200
Q2 '12	90.4300	83.4200	28.9400
Q3 '12	97.9600	82.3000	24.6200
Q4 '12	106.9600	85.0500	43.4600
Q1 '13	100.6600	63.2200	38.3300
Q2 '13	107.5300	60.9500	31.9000
Q3 '13	117.0500	63.0500	30.3300
Q4 '13	119.2100	63.5800	50.2200

Quarter	Pivot Field Names	Pivot Field Values
Q4 '12	Apple	43.460
Q1 '13	Apple	38.330
Q2 '13	Apple	31.900
Q3 '13	Apple	30.330
Q4 '13	Apple	50.220
Q1 '10	Nokia	110.110
Q2 '10	Nokia	111.470
Q3 '10	Nokia	117.460
Q4 '10	Nokia	122.280

1 Pivot the data

After you have set up the data source, in the grid, select two or more columns. Click the drop-down arrow next to the column name, and then select **Pivot**. New columns called "Pivot field

"names" and "Pivot field values" are created and added to the data source. The new columns replace the original columns that you selected to create the pivot.



A screenshot of a Microsoft Excel PivotTable context menu. The menu is open over a table with four columns: Quarter, Samsung, Nokia, and Apple. The 'Quarter' column header has a dropdown arrow. The menu options are: Rename, Reset Name, Copy Values, Hide, Create Calculated Field..., Pivot (which is highlighted with a white background), and Merge Mismatched Fields.

Quarter	Samsung	Nokia	Apple
Q4 '11	93.8300	111.7000	35.46
Q1 '12	89.2800	83.1600	33.12
Q2 '12	90.4300	83.4200	28.94
Q3 '12	97.9600	82.3000	24.62
Q4 '12	106.9600	85.0500	43.46
Q1 '13	100.6600	63.2200	38.3300
Q2 '13	107.5300	60.9500	31.9000
Q3 '13	117.0500	63.0500	30.3300
Q4 '13	119.2100	63.5800	50.2200

Pivot is available only for non-legacy Microsoft Excel and text file data sources.

2 Add to the pivot

To add more data to the pivot, select another column, click the drop-down arrow next to the column name, and then select **Add to Pivot**. Make sure that the pivot columns and values look as expected before you begin your analysis.

Abc	#	Abc	#
Quarter	Data	LG	Pivot Field Values
Q1 '10	27.19	Rename	8.270
Q2 '10	29.37	Copy Values	8.740
Q3 '10	27.48	Hide	13.480
Q4 '10	30.12	Create Calculated Field...	16.010
Q1 '11	24.00	Create Group...	16.880
Q2 '11	24.4200	Create Bins...	19.630
Q3 '11	21.0100	Add data to pivot	17.300
Q4 '11	16.9400	Describe...	35.460
Q1 '12	14.7200	Apple	33.120

To remove a pivot, click the drop-down arrow next to the name of a pivot column, and then select **Remove Pivot**.

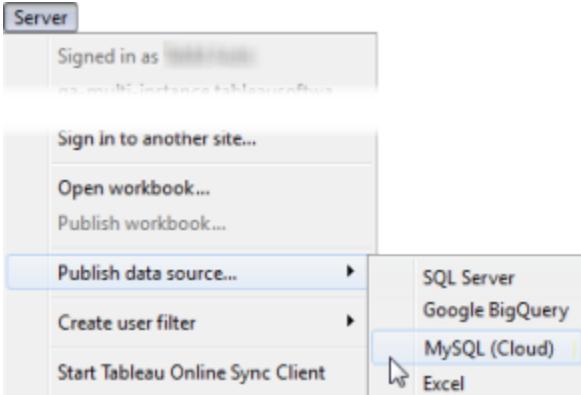
Quick Start: Publish a Data Source to Tableau Online

You can publish data sources so that other Tableau Desktop users can connect to your data, and to have updates you make to the published data source flow to all workbooks that rely on it. This topic assumes you are familiar with setting up your data model and signing in to Tableau Online. For more information, see the related links at the bottom.

For help publishing data to Tableau Server, see [Quick Start: Publish a Data Source to Tableau Server](#).

1 Sign in and initiate publishing

1. If you're not already signed in to the site you want to publish to, select **Server > Sign In**, and sign in the way you normally do Tableau Online.
2. Select **Server > Publish Data Source**. If your workbook is connected to multiple data sources, you can select the data source you want to publish.

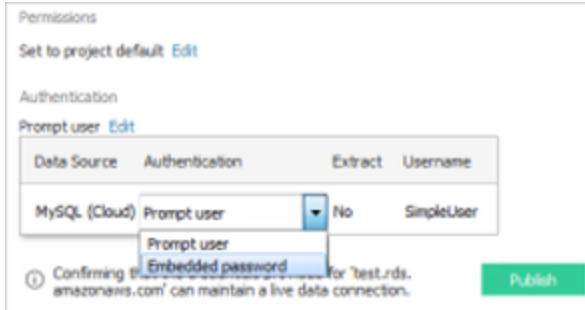


3. In the **Publish data source to Tableau Server** dialog box, do the following:
 - Select the project you want to publish to and enter the data source name.
 - Add a description and tags that will help other users find your data source

2 Select the authentication type

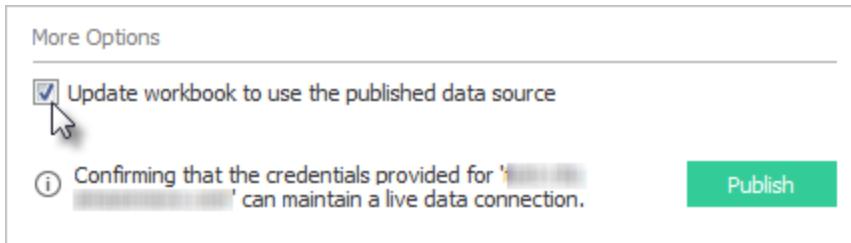
The options available for accessing the data source depend on the type of data you publish. For descriptions of all of the authentication types, see [Set Credentials for Accessing Your Published Data](#) in the Tableau Desktop Help.

Information appears at the bottom of the dialog box to let you know whether you need to take further action, such as adding Tableau Online to your data provider's authorized list.



3 Use the published data source

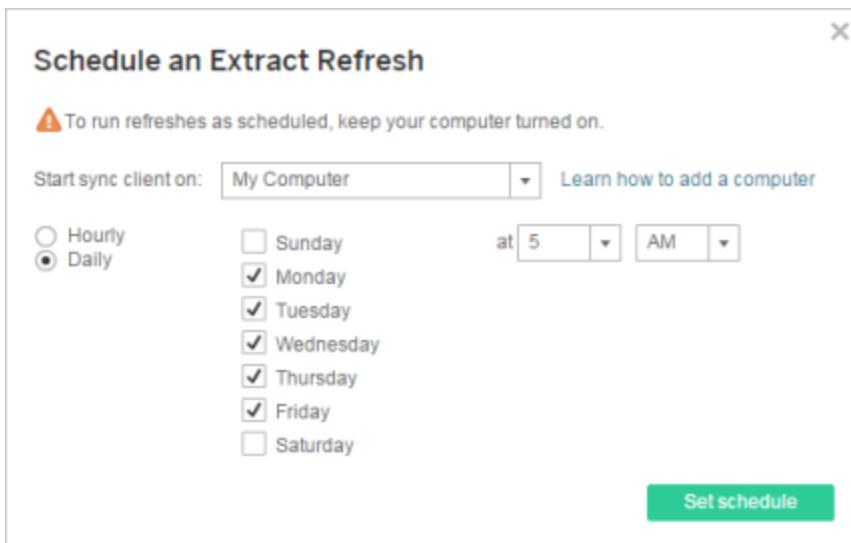
By default, when you publish, Tableau updates the workbook you're publishing from to use the published data source, and then closes the local data source. You can clear this check box if you want the workbook to continue to use the local data source.



For more information, see [Quick Start: Automatically Update Workbook to Use Published Data Source](#).

4 Set up a refresh schedule

As you publish an extract of a data source that Tableau Online cannot reach directly, such as one that you maintain inside your organization's firewall, you are prompted to start the Tableau Online sync client and set up a scheduled refresh for that data source. Select the computer on which you want to run the sync client (**My Computer** is the computer you are publishing from).



See also

Q-

- [Publish Data Sources and Workbooks](#)
- [Publish a Data Source](#)
- [Allow Direct Connections to Data Hosted on a Cloud Platform \(Tableau Online\)](#)
- [Keep Data Fresh \(Tableau Online\)](#)

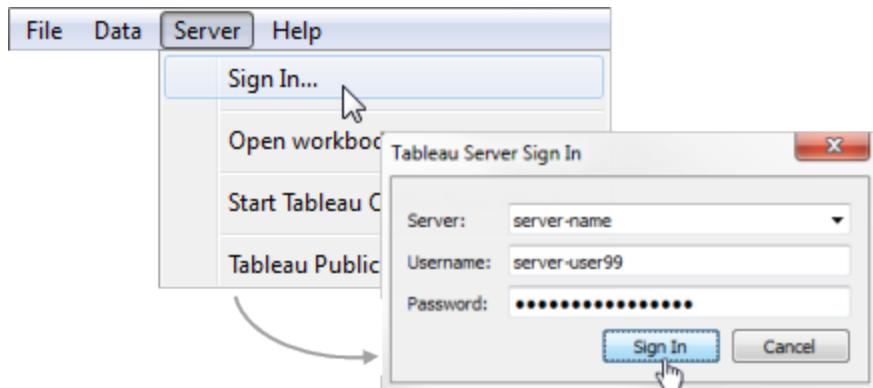
Quick Start: Publish a Data Source to Tableau Server

You can publish data sources so that other Tableau Desktop users can connect to your data, and to have updates you make to the published data source flow to all workbooks that rely on it.

If you want to publish to Tableau Online, see [Quick Start: Publish a Data Source to Tableau Online](#) on page 1680.

1 Sign in to the server

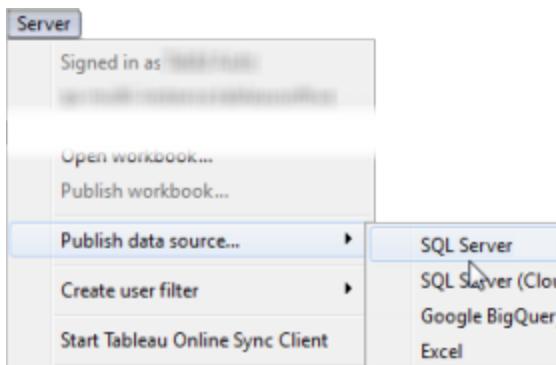
If you're not already signed in to the server you want to publish to, select **Server > Sign In**, and enter your server name and user name and password.



2 Start the publishing process

1. Select **Server > Publish Data Source**.

If your workbook is connected to multiple data sources, select the data source you want to publish from the submenu.

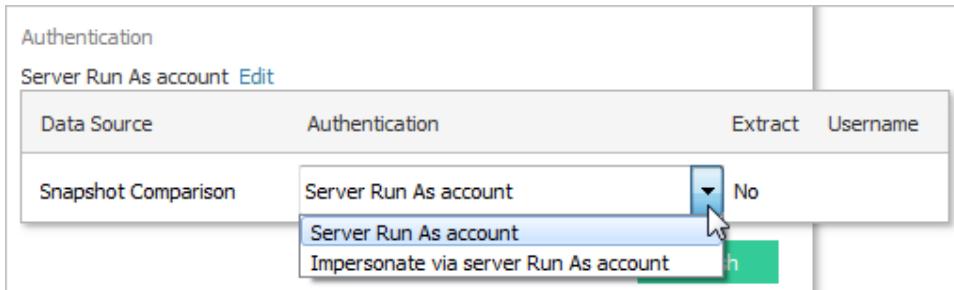


2. In the **Publish data source to Tableau Server** dialog box, do the following:

- Select the project you want to publish to and enter the data source name.
- Add a description and tags that will help other users find your data source

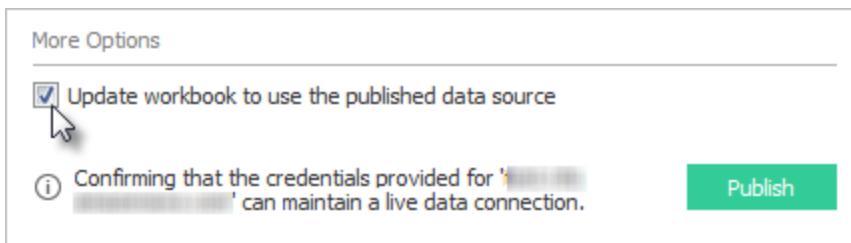
3 Select the authentication type

The options available for accessing the published data source depend on the type of data you publish. The following image shows options for a SQL Server data source



4 Use the published data source

By default, when you publish, Tableau updates the workbook you're publishing from to use the published data source and closes the local data source. You can clear this check box if you want the workbook to continue to use the local data source.



For more information, see [Quick Start: Automatically Update Workbook to Use Published Data Source](#) on page 1650.

See also

- [Publish Data Sources and Workbooks](#)
- [Publish a Data Source](#)

Q-

Quick Start: SAP HANA Single Sign-On

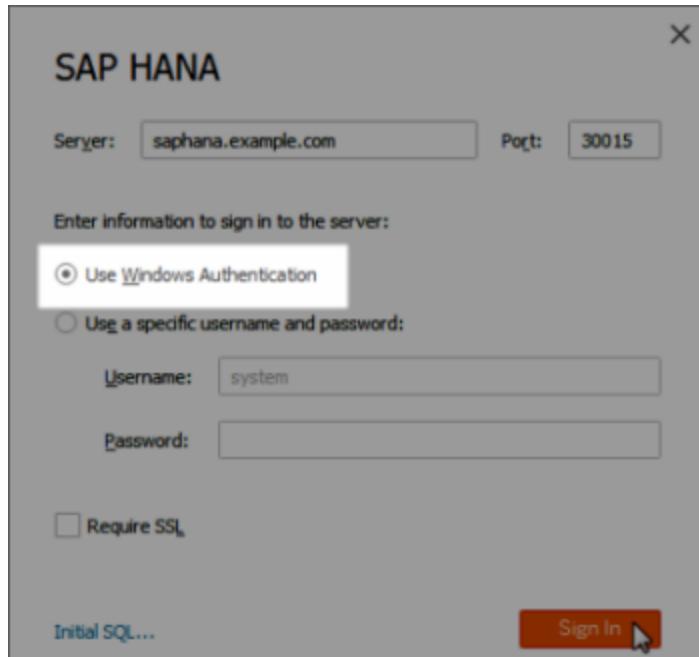
When SAP HANA is configured to support single sign-on (SSO), after you sign in to the SAP HANA server, you can access data, and publish data sources and workbooks to Tableau Server, without having to re-enter your user name and password. And, you can publish a data source or workbook so that other users with SSO can access the published data sources and workbooks without having to enter their user names and passwords.

Important: Your environment must be correctly configured to support SSO for SAP HANA:

- Tableau Desktop requires SAP HANA driver version 1.00.85 and later.
- Tableau Server must be configured to support SSO for SAP HANA. For information, see [Configure SAP HANA for Single Sign-On](#) in the Tableau Server Help.

1 Connect to SAP HANA using SSO

On the start page in Tableau Desktop, under **Connect**, select **SAP HANA**, specify a server name, select **Use Windows Authentication**, and then click **Sign In**.



2 Create a data source or a workbook

After you connect to SAP HANA data, you can set up a data source and publish it to Tableau Server for others to use. You can also create a view and publish the workbook to Tableau Server.

A screenshot of the Tableau Desktop interface. On the left, the "Connections" pane shows a connection to "saphana.example.com (SAP HANA)". The "Schema" dropdown is set to "TESTV1". The "Table" pane shows a list of tables: "Calcs (TESTV1.Calcs)", "GFLYNN (... GFLYNN)", "Collection...lections", "DataBins...DataBins", "DateTime...ateTime)", "DEFECT45", and "Electon (... Election)". A "New Custom" button is visible. The main workspace shows a data source named "Dates" with a "Connection" type of "Live". It includes a "DateBins" section with a switch between "Date" and "DateTime", and a "Filters" section with a count of 0. Below this is a table view with columns: "Key", "Deadline", "Date", "Deadline", "Value", "ID", "DateTime", "DateTime", "DateTime", "DateTime", "DateTime", "DateTime". Buttons for "Update Now" and "Automatically Update" are present. At the bottom, there are buttons for "Data Source", "Sheet1" (which is highlighted in orange), and other worksheet tabs.

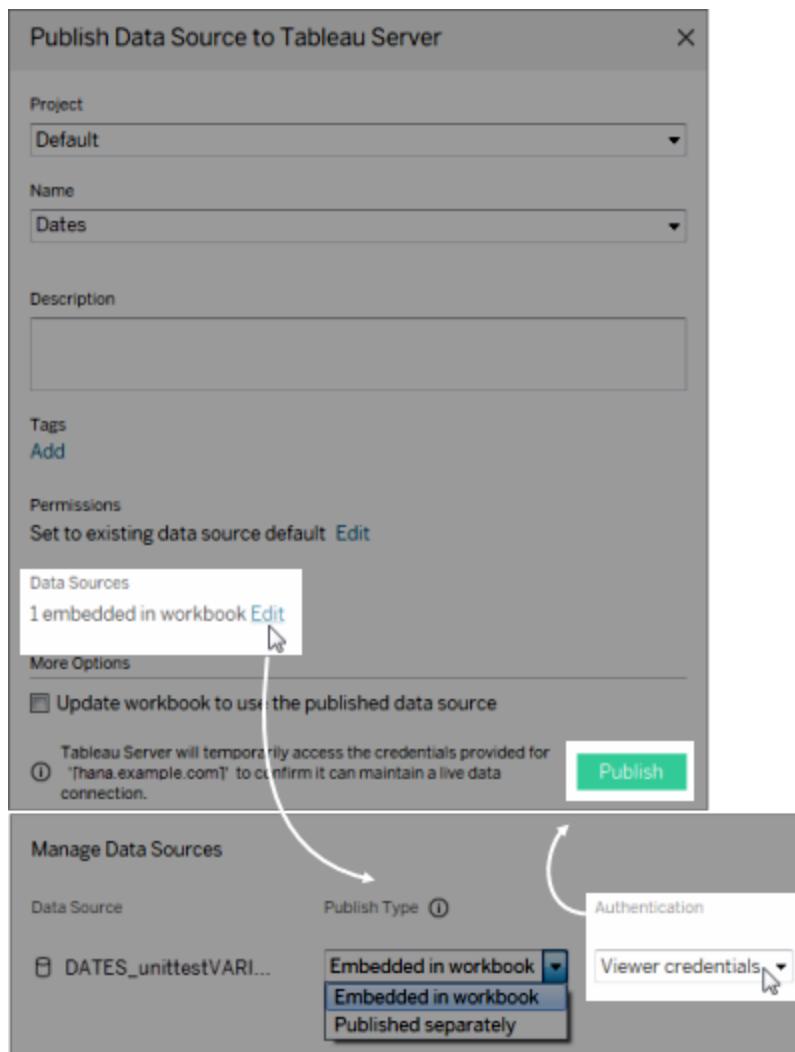
3Publish to Tableau Server with SSO enabled

In an SSO environment, users don't have to enter their credentials to access a data source or a workbook if you publish it to Tableau Server with the correct authentication mode.

[Data Source](#)

To enable SSO when you publish a data source to Tableau Server, follow these steps:

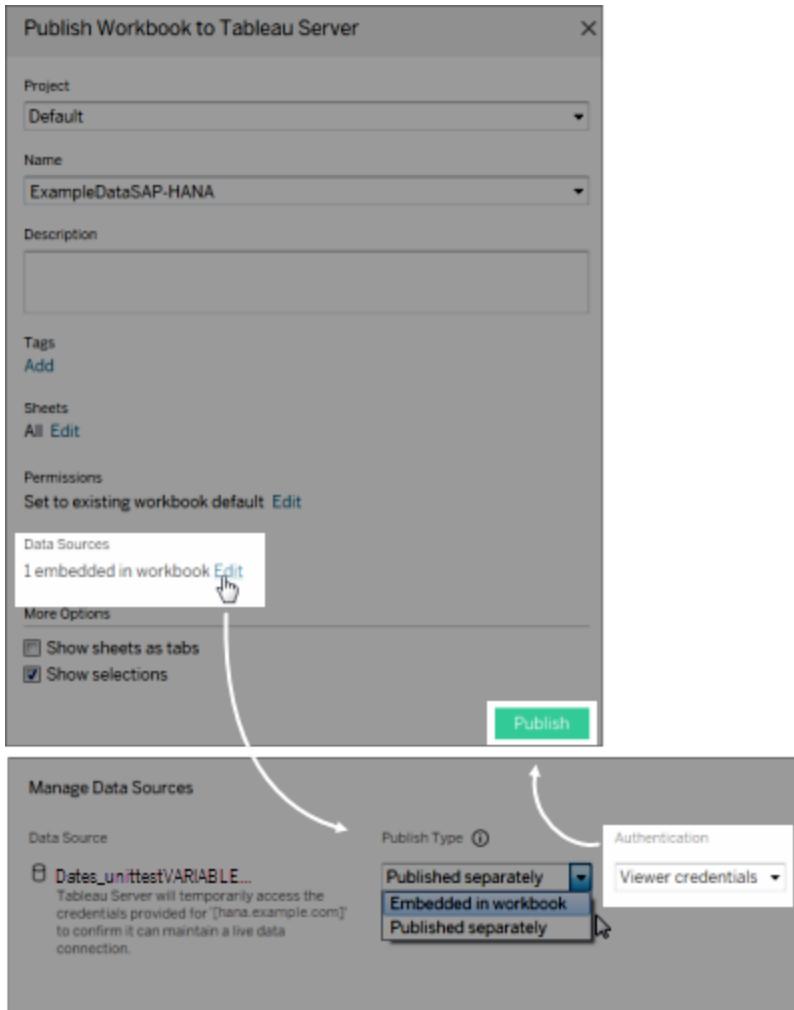
1. Select **Server> Publish Data Source**
2. Under Data Sources, click **Edit**.
3. In the Manage Data Sources dialog box, do the following:
 1. Select the **Publish Type: Embedded in workbook or Published separately**.
 2. Select **Viewer credentials for Authentication**.
4. Click **Publish**.



Workbook

To enable SSO when you publish a workbook to Tableau Server, follow these steps:

1. Select **Server > Publish Workbook**.
2. Under Data Sources, click **Edit**.
3. In the Manage Data Sources dialog box, do the following:
 1. Select the **Publish Type: Embedded in workbook or Published separately**.
 2. Select **Viewer credentials** for **Authentication**.
4. Click **Publish**.



Quick Start: Split a Field into Multiple Fields

Sometimes, it is easier to analyze a string fields if its values are separated into multiple fields. Use the Split and Custom Split commands to split string values from one field into multiple fields.

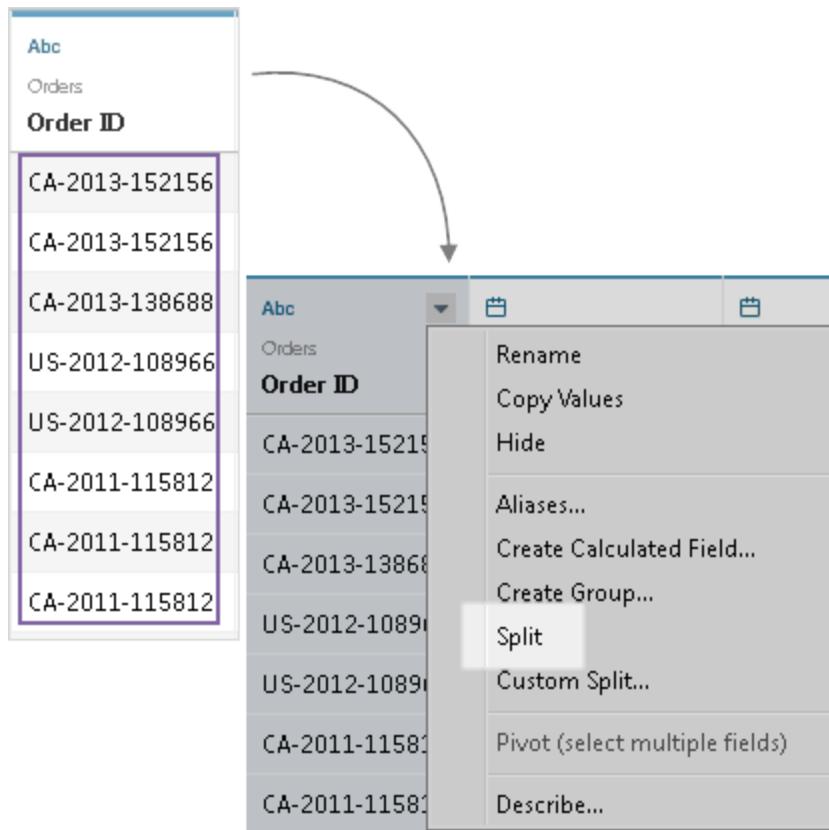
1 Split the field

After you have set up the data source, you can have Tableau automatically split the field based on a common separator or do a custom split of a field by specifying the common separator and the number of fields.

Note: The split and custom split commands are available for the following data sources types: Tableau data extracts, Microsoft Excel, Text, Salesforce, OData, Microsoft Azure Market Place, Google Analytics, HP Vertica, Oracle, MySQL, PostgreSQL, Teradata, Amazon Redshift, Aster Data, Google Big Query, Cloudera Hadoop Hive, Hortonworks Hive, and Microsoft SQL Server.

Split fields automatically

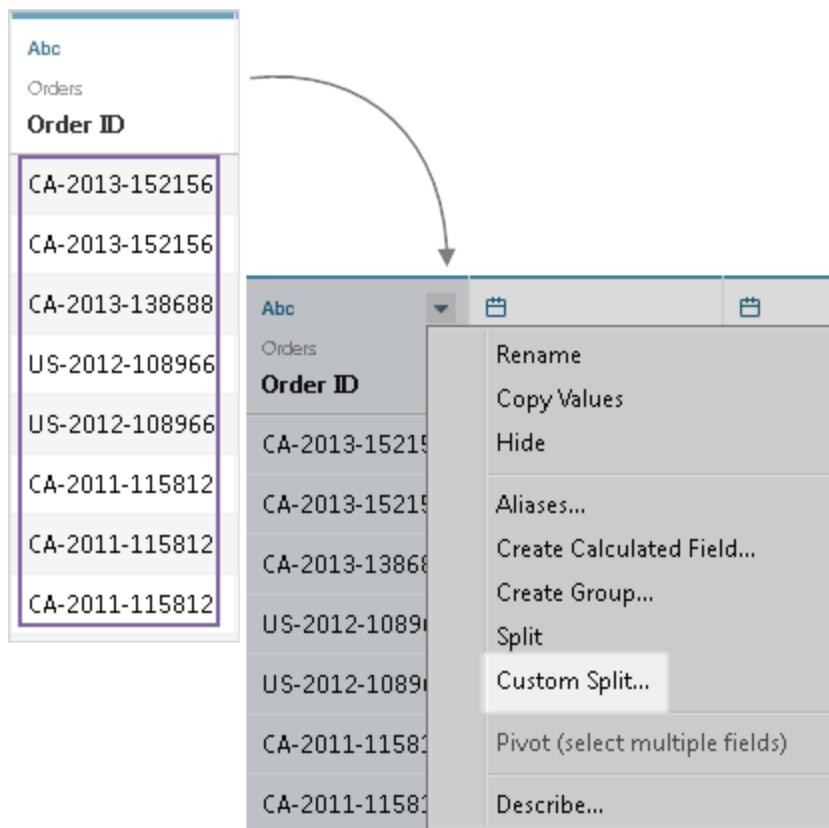
In the grid, click the drop-down next to the column name. Select **Split**.



If Tableau cannot identify a common separator among the string values you are prompted to create a custom split.

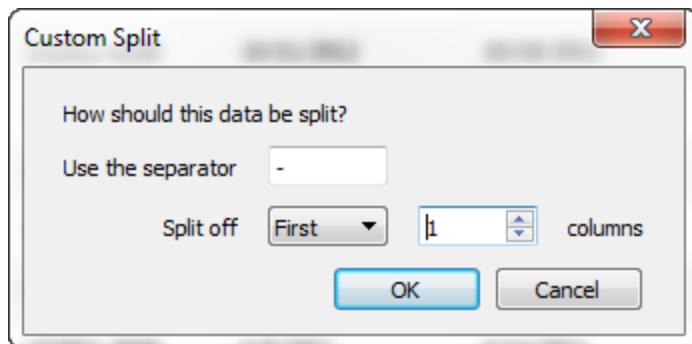
Use a custom split

In the grid, click the drop-down arrow next to the column name of the column you want to split. Select **Custom Split**.



In the **Custom Split** dialog box, specify the following:

1. In the **Use the separator** box, enter the separator by which to separate the values in the field. The separator can be a character or a combination of characters or phrases.
2. Under **Split off**, select whether to split the string values for every instance (**All**) of the separator, the first (**First**) *n* instances of the separator, or the last *n* instances (**Last**) of the separator.
3. Enter the number of fields you want to generate from the split.



2 Review the results

New fields are created and added to the data source as calculated fields. If the new fields do not look as expected, you can undo the changes. After you verify that the split values look as expected, begin your analysis by clicking the sheet tab.

Note: You can also make modifications to the new split fields by editing the calculated fields from the Data pane in the sheet tab.

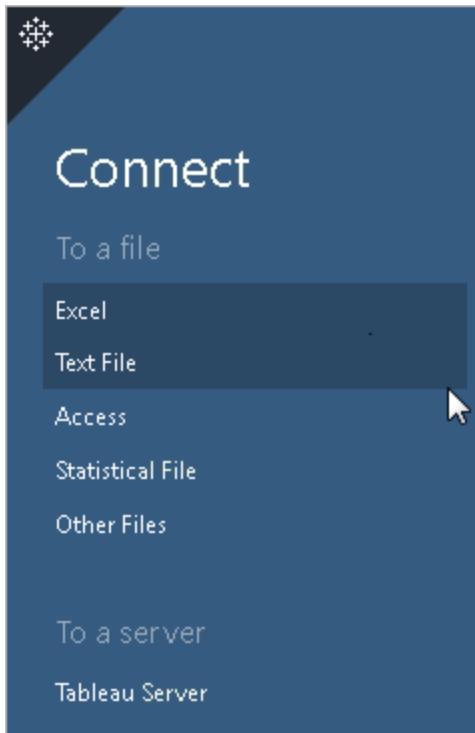
Abc Orders Order ID	=Abc Calculation Order ID - Split 1	=# Calculation Order ID - Split 2	=# Calculation Order ID - Split 3
CA-2013-152156	CA	2013	152156
CA-2013-152156	CA	2013	152156
CA-2013-138688	CA	2013	138688
US-2012-108966	US	2012	108966
US-2012-108966	US	2012	108966
CA-2011-115812	CA	2011	115812
CA-2011-115812	CA	2011	115812
CA-2011-115812	CA	2011	115812

Quick Start: Union Tables—Now Automatically

When you create joins, you combine two or more tables of data by adding fields (columns). With unions, you can combine two or more tables of data by appending values (rows) from one table to another. You can union Microsoft Excel or text file-based tables that share the same fields. That is, to union two or more disparate tables, the related fields in each table must have matching field names and data types. For example, before you can union two single-field tables that contain customer names, both field names must be called "Customer Name" and both field data types must be *string*.

1 Connect to your Excel or text file data

On the start page, under **Connect**, click **Excel or Text File** to connect to your Excel or text file-based data.



2 Create a union

Double-click **New Union** to set up the union.

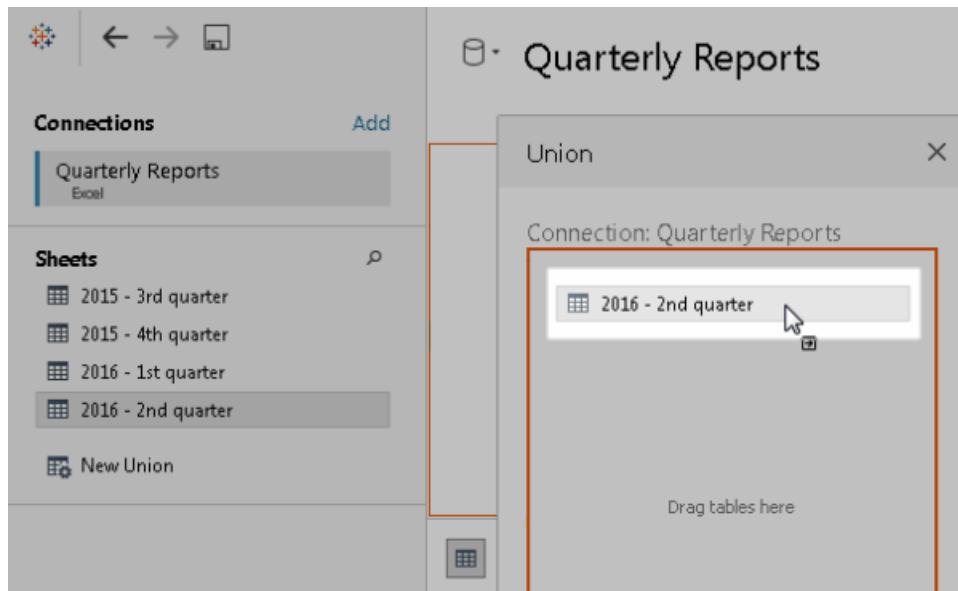
A screenshot of the Tableau Data Source pane. At the top, there are navigation icons: a gear, back, forward, a square, and a refresh symbol. Below that is a dropdown menu labeled "Data Sources" with "Quarterly Reports" selected. Under "Connections", there is a list with "Quarterly Reports" (Excel) selected. In the "Sheets" section, there is a text input field "Enter sheet name" and a list of four sheets: "2015 - 3rd quarter", "2015 - 4th quarter", "2016 - 1st quarter", and "2016 - 2nd quarter". At the bottom, there is a button labeled "New Union".

3 Union tables

You can union tables using one of two methods:

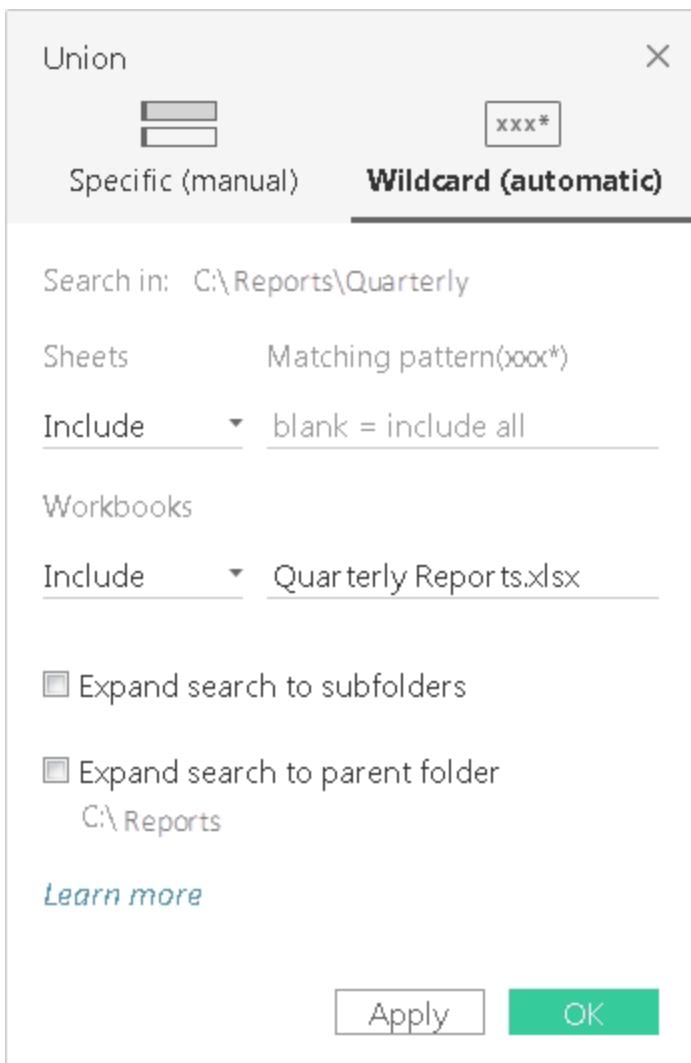
Specific (manual) —for distinct files or sheets

Select individual tables from the left pane and drag it into the Union dialog box. To add multiple tables to a union at the same time, in the left pane, press **Shift** or **Ctrl**, select the tables you want to union, and then drag them below the first table. Click **Apply** or **OK** to complete the union. Use the Specific option when you have distinct files and sheets you want to union from.



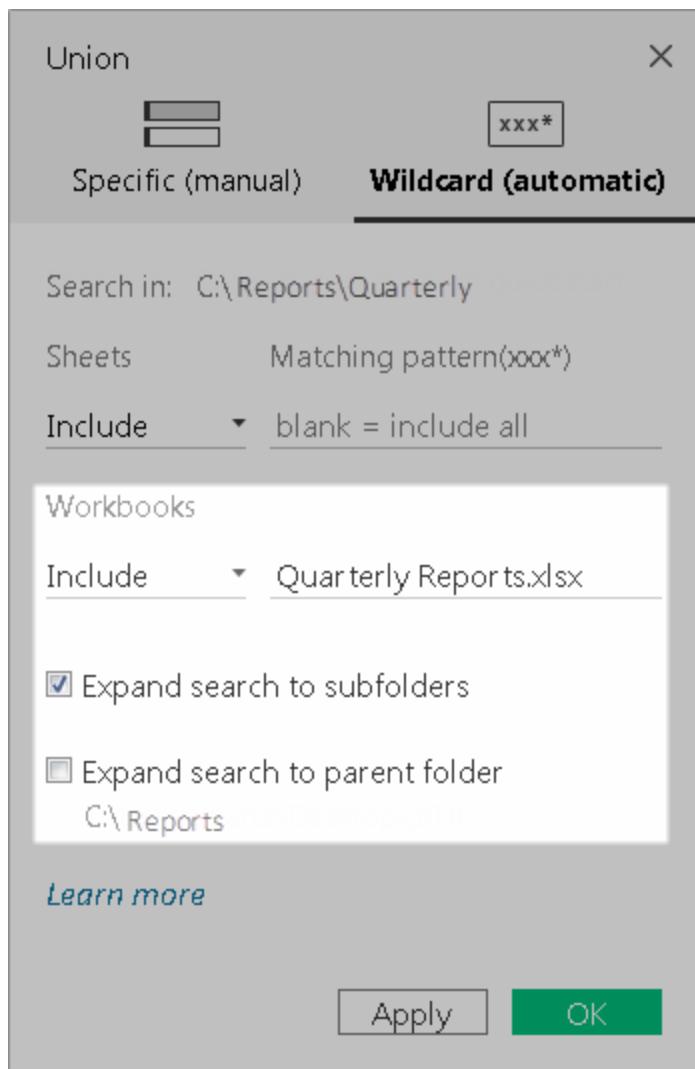
Wildcard (automatic) —across files in different directories or across workbooks

Set up search criteria to include tables in your union. You can use the wild character * to match a sequence or pattern of characters in the text file or Excel workbook names, and/or the Excel worksheet name. Click **Apply** or **OK** to complete the union. Use the Wildcard option to union text and Excel files across folders and Excel sheets across workbooks that are scoped to the connection.



Find more tables

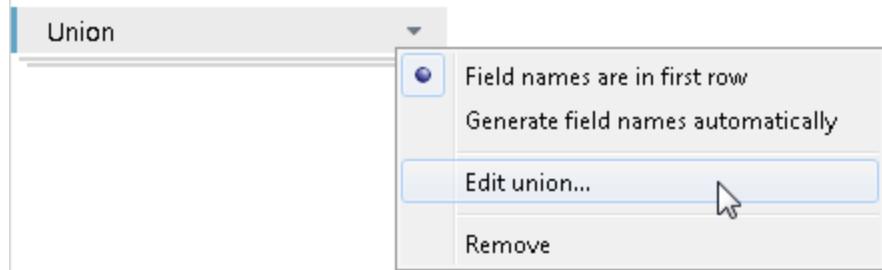
The tables that are initially available to union are scoped to the connection you've selected. If you want to union more tables that are located outside of the folder (for text and Excel files) or in a different workbook (for Excel worksheets), you can select a combination of the **Expand search to subfolders** and **Expand search to parent folder** check boxes to broaden the search criteria. For more information, see [Union Your Data on page 392](#).



4 Edit the union

Click the union drop-down arrow and then select **Edit union** to rename the union, add or remove individual tables in the union, or change the search criteria to include or exclude more tables in the union.

Union (Quarterly Reports)



After you have unioned the tables, additional fields are added automatically generated to help identify where in the original data the values in the union are derived coming from. Review these values and the other values in the grid.

Continue to prepare your data source for analysis by joining other tables to the union table and making general modifications to the fields in the grid, such as pivot and split. When you are finished, click the sheet tab to start your analysis.

Project	Union	Abc	Abc	Abc	Abc
Start time & Start	End time	ID#	Member type	Sheet	Table Name
9/30/2015 11:58:00 PM	10/1/2015 12:05:00 AM	W00630	Registered	2015 - 3rd quarter	2015 - 3rd quarter
9/30/2015 11:53:00 PM	10/1/2015 12:11:00 AM	W01456	Registered	2015 - 3rd quarter	2015 - 3rd quarter
12/1/2015 11:58:00 PM	12/2/2015 12:05:00 AM	W01301	Subscriber	2015 - 4th quarter	2015 - 4th quarter
12/2/2015 11:56:00 PM	12/3/2015 12:02:00 AM	W01398	Subscriber	2015 - 4th quarter	2015 - 4th quarter
12/2/2015 11:56:00 PM	12/3/2015 12:03:00 AM	W00098	Subscriber	2015 - 4th quarter	2015 - 4th quarter
3/31/2016 11:58:00 PM	4/1/2016 12:00:00 AM	W00749	Registered	2016 - 1st quarter	2016 - 1st quarter
3/31/2016 11:52:00 PM	4/1/2016 12:08:00 AM	W01048	Casual	2016 - 1st quarter	2016 - 1st quarter
6/30/2016 11:59:00 PM	7/1/2016 12:03:00 AM	W20490	Subscriber	2016 - 2nd quarter	2016 - 2nd quarter
6/30/2016 11:58:00 PM	7/1/2016 12:30:00 AM	W00304	Subscriber	2016 - 2nd quarter	2016 - 2nd quarter

You can troubleshoot unions

If you notice some fields with null values, it could be because field names in the union table do not match. In this case, you can merge the non-matching fields by selecting two or more columns in the grid, clicking the column drop-down arrow, and then selecting **Merge mismatched fields**. When you use this option, a new field replaces the fields with the null values in to the Tableau data source. **Note:** You can also create a calculation to merge fields or combine fields by modifying the underlying Excel or text file data.

The diagram illustrates the process of merging non-matching fields in a unioned data source. On the left, two separate tables are shown under the 'Union' project. The first table has a column 'Start time' and the second has a column 'Start'. Both columns contain time values. An arrow points from these two tables to the right, indicating the merge process. On the right, the resulting single table is shown under the 'Project' project. It contains a single column named 'Start time & Start' which now contains all the time values from both original columns, effectively merging them into one.

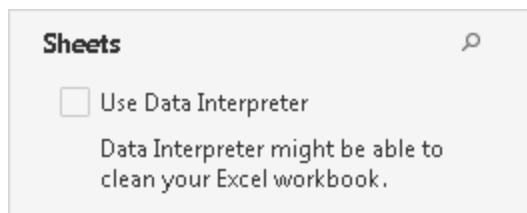
Union	Union	Project
Start time	Start	Start time & Start
9/30/2015 11:58:00 PM	null	9/30/2015 11:58:00 PM
9/30/2015 11:53:00 PM	null	9/30/2015 11:53:00 PM
12/1/2015 11:58:00 PM	null	12/1/2015 11:58:00 PM
12/2/2015 11:56:00 PM	null	12/2/2015 11:56:00 PM
12/2/2015 11:56:00 PM	null	12/2/2015 11:56:00 PM
null	3/31/2016 11:58:00 PM	3/31/2016 11:58:00 PM
null	3/31/2016 11:52:00 PM	3/31/2016 11:52:00 PM

Quick Start: Use Data Interpreter to Prepare the Data Source for Analysis

Sometimes, the format of the data in your Microsoft Excel or Google Sheets data makes it difficult to analyze in Tableau. For example, your data might include sub-tables, hierarchical headers, extraneous headers and footers, or blank rows and columns. The Data Interpreter draws out sub-tables and removes some of that extraneous information to help prepare the data source for analysis.

1 Turn on the Data Interpreter

After you have connected to your data and set up the data source, if Tableau detects sub-tables, unique formatting, or some extraneous information, the Data Interpreter option becomes available. Select the **Use Data Interpreter** check box.



2 Review results

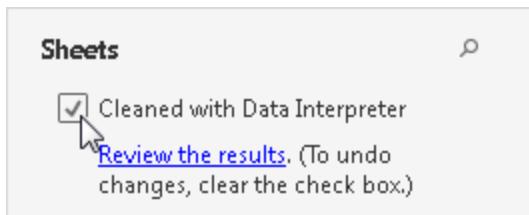
The Data Interpreter draws out sub-tables and removes the extraneous information from your Tableau data source and automatically updates the grid with its interpretation. You can also click the **Review results** link to see a copy of the data source. The data source copy contains annotation definitions and annotations that describe how your data was interpreted.

The screenshot shows the 'Sheets' tab of the Tableau Data Interpreter settings, with the 'Cleaned with Data Interpreter' checkbox checked. Below the checkbox is a link: 'Review the results. (To undo changes, clear the check box.)'. A curved arrow points from this link to a preview of the data grid below. The data grid has four columns: 'Year', 'Number o', 'Number o', and 'Number o'. The rows show data for years 94-95 through 00-01, with values decreasing over time.

3	Year	Number o	Number o
4	94-95	47,236	137,795
5	95-96	43,099	132,675
6	96-97	36,125	118,946
7	97-98	26,677	114,730
8	98-99	21,134	113,370
9	99-00	19,452	107,949
10	00-01	18,775	108,884

3 Begin your analysis

If you like the interpretations made by the Data Interpreter, you can begin your analysis by clicking the sheet tab. Alternatively, you can clear the **Use Data Interpreter** check box to use the original data source.



Quick Start: Use Initial SQL Parameters

You can pass parameters to your data source in an initial SQL statement. There are several reasons why this is useful:

- You can configure impersonation using the **TableauServerUser** or **TableauServerUserFull** parameters.
- If your data source supports it, you can set up row-level security (for example, for Oracle VPD or SAP Sybase ASE) to make sure that users see only the data that they are authorized to see.
- You can provide more details in logging, for example, the Tableau version or the workbook name.

Note: If you use the **TableauServerUser**, **TableauServerUserFull**, or **WorkbookName** parameter in an initial SQL statement, you will create a dedicated connection that can't be shared. This will also restrict cache sharing, which can enhance security, but may also slow performance.

Supported parameters

Parameter	Description	Example of returned value
TableauServerUser	The user name of the current server user. Use when setting up impersonation on the server. Returns an empty string if the user is not signed in to Tableau Server.	asmith
TableauServerUserFull	The user name and domain of the current server user. Use when setting up imper-	domain.lan\asmith

	sonation on the server. Returns an empty string if the user is not signed in to Tableau Server.	
TableauApp	The name of the Tableau application.	Tableau Desktop Professional Tableau Server
TableauVersion	The version of the Tableau application.	9.3
WorkbookName	The name of the Tableau workbook. Use only in workbooks with an embedded data source.	Financial-Analysis

Examples

The following examples show different ways you can use parameters.

- This example sets the security context on Microsoft SQL Server:

```
EXECUTE AS USER = [TableauServerUser] WITH NO REVERT;
```

- This example can be used to help set up row-level security for Oracle VPD:

```
begin
  DBMS_SESSION.SET_IDENTIFIER([TableauServerUser]);
end;
```

- This example shows how, on a DataStax data source, you can use parameters to add detail to logging or to set up a session variable to keep track of the data:

```
SET TABLEAUVERSION [TableauVersion];
```

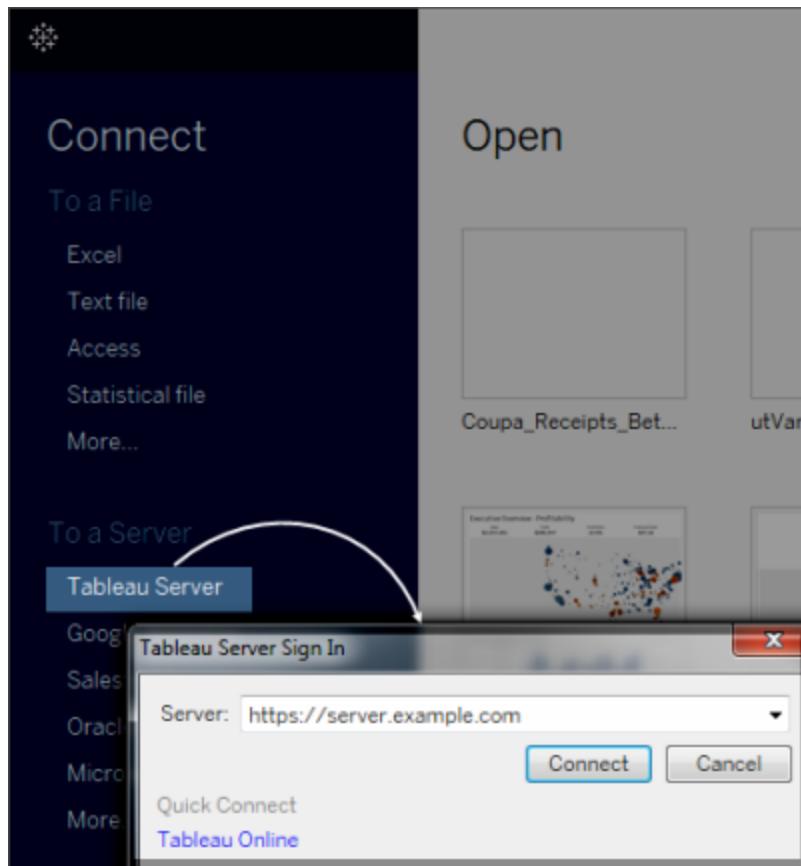
Note: Oracle PL/SQL blocks require a trailing semicolon to terminate the block. Consult Oracle documentation for the proper syntax.

Quick Start: Use Tableau Server Data Sources

You may be able to use Tableau Server data sources that were created by an administrator or another workbook author. Because the data extract and any required database connection information or database drivers are stored within the data source on the server, using a Tableau Server data source is a fast way to quickly connect to data and start authoring. Tableau Server data sources can also contain a layer of workbook customizations such as calculations. Although you cannot alter customized fields that come with the data source, you can create additional customizations. You can also create a local copy of the data source, update or enhance it, and then republish it.

1 Connect to Tableau Server

On the start page in Tableau Desktop, under Connect, click Tableau Server, and then specify a server name and provide any necessary sign-in information.



You may need to provide a user name and a password, depending on how Tableau Server is authenticating users.

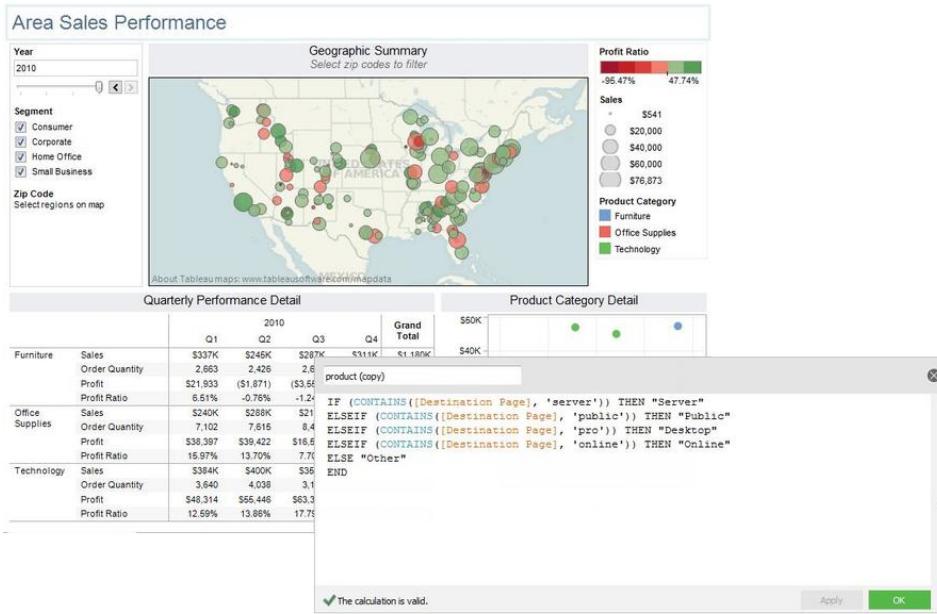
2 Select your Data Source

After you log in to Tableau Server, select a data source. It will load in the **Data** pane with a Tableau Server icon.

The screenshot shows the Tableau Server interface. At the top, there are navigation icons: a gear, back, forward, and refresh. Below that, a header bar displays "Connected to Tableau Server as Henry Wilson" and the URL "https://server.example.com". A search bar is present. The main content area is titled "Data Source" and lists several sources. One source, "Accidents", is highlighted with a blue selection bar and has a circular arrow icon next to it. A callout arrow points from the "Accidents" link in the list to the "Accidents" entry in the "Dimensions" list below. The "Data" tab is selected, and the "Analytics" tab is visible. The "Dimensions" list contains three items: "Accidents", "CL_TWAY_ROUTE", and "HARM_EV".

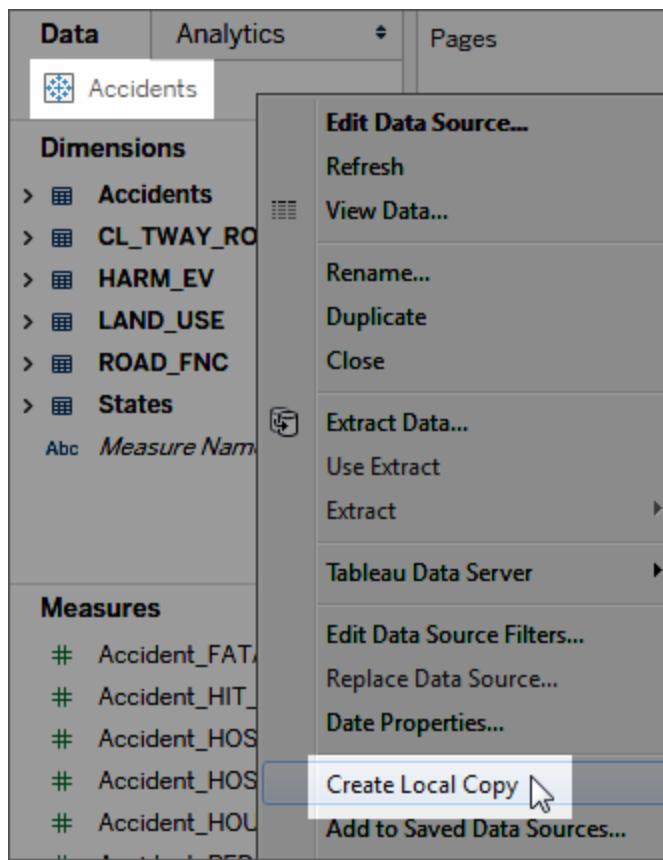
3 Create and Customize the View

Create one or more views in Tableau Desktop. You won't be able to modify customizations that came with the data source, such as calculations, groups, and sets, but you can add new customizations of your own.



4 Publish a Modified Version of the Data Source

To create a data source that adds your customizations to the original, right-click the data source (or control-click on a Mac) and select **Create Local Copy**. A local, writable copy of the data source is created.

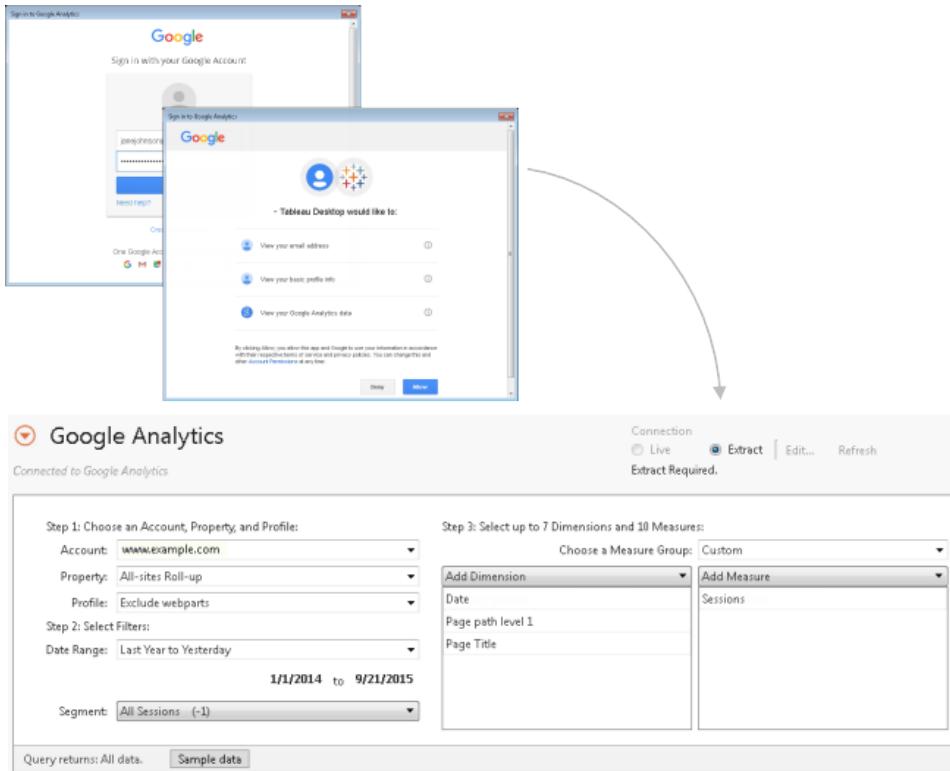


Quick Start: Query All Data from Google Analytics

When working with Google Analytics (GA) data, GA can restrict the amount of data that it returns from a query and provide sampled data instead. Performing analysis on sampled data can skew results and cause inaccurate inferences about the entire data set. In version 9.2, Tableau has made enhancements to GA queries to avoid returning sampled data by taking a single GA query, creating multiple queries from it, and then combining the results from the queries to bypass the query restrictions that cause sampling to occur.

1 Connect to GA data

From Tableau Desktop, on the start page and under **Connect**, click **Google Analytics**. Sign in to Google, select a date range, and then select the necessary dimensions and measures to create your query.



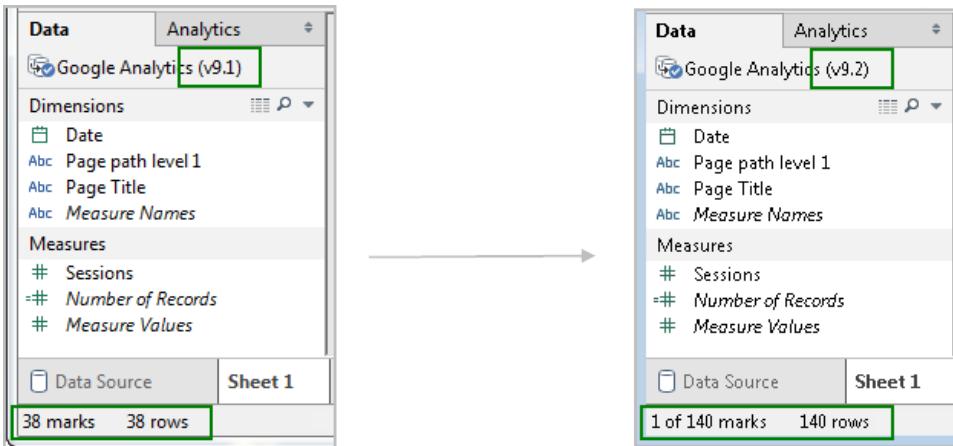
2 Compare queries

Tableau detects that your GA query might return sampled data, and, by default, returns all data instead.

Query returns: All data. Sample data

If the GA query stays within the boundaries of the query restrictions, GA doesn't return sampled data and you do not see the above message.

The image below compares the same Google Analytics data set in version 9.1 and 9.2. The query in version 9.1 returns sampled data while the query in version 9.2 returns all data.



3 Troubleshoot issues with returning all data

If your query continues to return sampled data, consider the following:

- **Missing date dimension** – You must use the date dimension in your query to return all data.
- **Too much data** – Your query might contain too much data. Reduce the date range.
Note: The minimum date range is one day.
- **Non-aggregatable dimensions and measures** – Some dimensions and measures cannot be separated into multiple queries. If you suspect a problematic dimension or measure in your query, hover over the **All data** button to see the tooltip that shows which dimensions or measures to remove from your query. For more information about these types of dimensions and measures, see [Sampled Data from Google Analytics](#).
- **Legacy workbooks** – Workbooks created in Tableau Desktop 9.1 and earlier cannot return all data. Open the legacy workbook in Tableau Desktop 9.2, and save the workbook.

4 Return sampled data instead

In some cases when workbook performance is critical or there are specific dimensions and measures you want to use in your query that are not supported by Tableau's default query process, use sampled data instead. To return sampled data, click the **Sample data** button.



Quick Starts about Visual Analysis

There are many features and techniques you can use as you explore your data in Tableau. Some of these are easiest to get through from the Analytics pane, and others become available after you've gotten a little deeper into your data.

The following Quick Starts highlight specific chart types, introduce specific tools, and explain ways to group and filter data.

Quick Start: Advanced Selection Tools

Tableau has a set of advanced selection tools that display in the view toolbar in the upper left corner of the view. Use the advanced selection tools to select multiple marks in the view.

1 Show the View Toolbar

By default, the view toolbar displays when you hover over a map view. To display the view toolbar in another view, right-click anywhere in the view and select **Show View Toolbar**.

The **Show View Toolbar** settings also apply to the view in Tableau Server and Tableau Online.

2 Open the Selection Tool Menu

Hover over the arrow on the view toolbar.



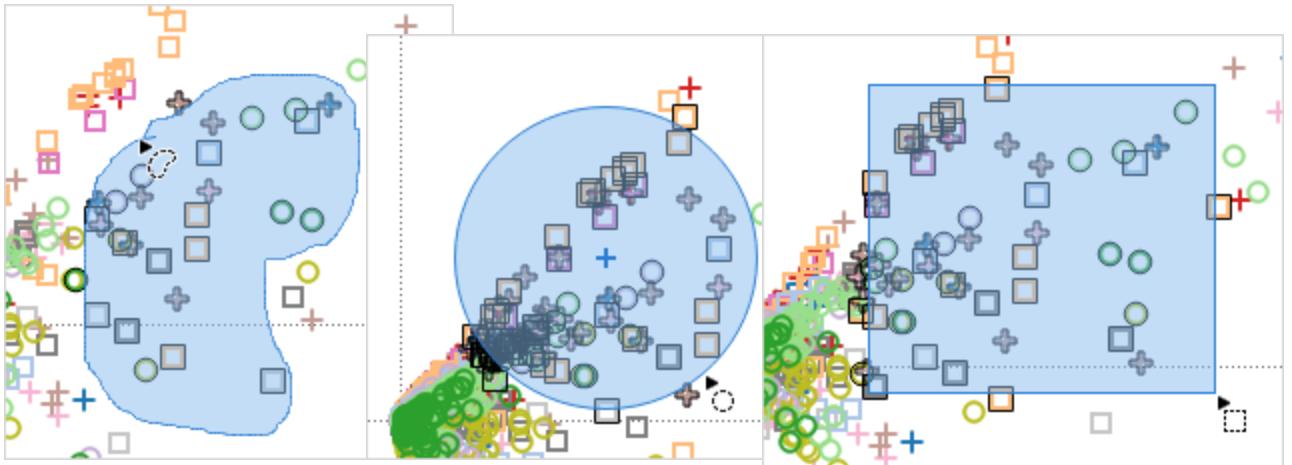
3 Pick a Selection Tool

Pick one of the following selection tools.

Lasso

Radial

Rectangular



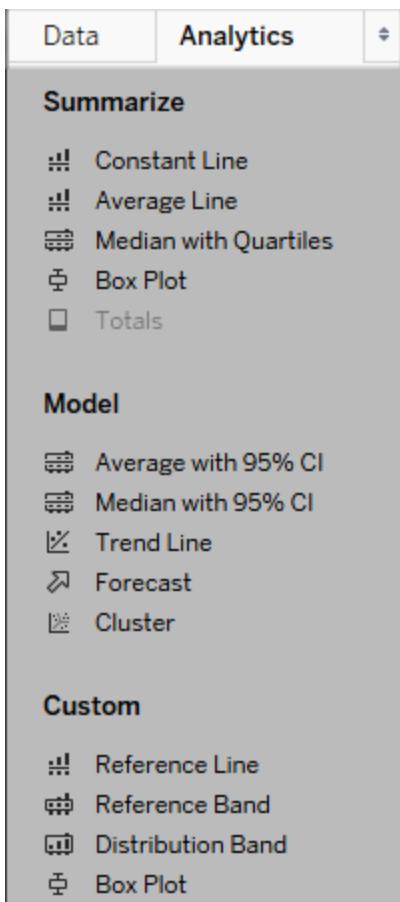
Click and drag across the view to select marks.

Quick Start: Analytics Pane

The Analytics pane provides quick and easy access to common analytic objects in Tableau. You can drag reference lines, forecasts, trend lines, and other objects into your view from the Analytics pane, which appears on the left side of the workspace.

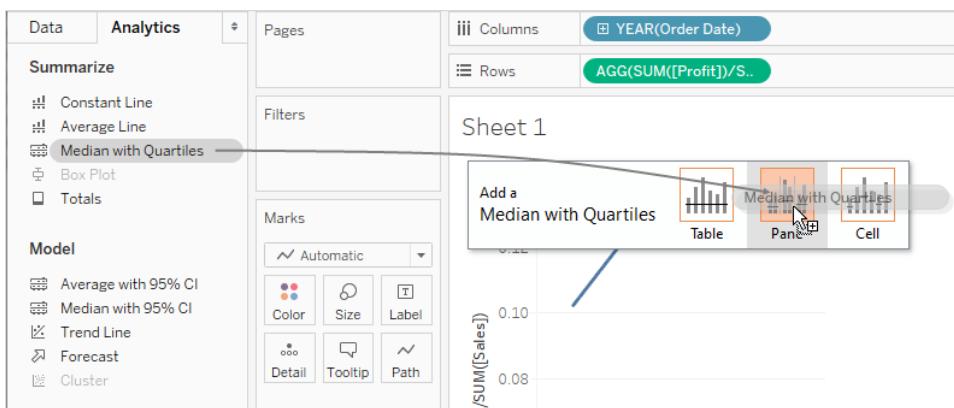
1 Analytics at Your Fingertips

Toggle between the Data pane and the Analytics pane by clicking one of the tabs at the top of the Side Bar:



2 Add an Analytics Object

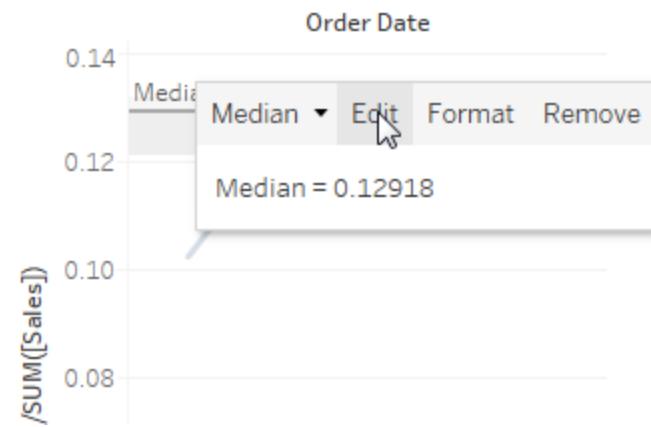
To add an object from the Analytics pane, drag it into the view. When you drag an object from the Analytics pane, Tableau shows the possible destinations for that object in a drop target area in the upper left section of the view—drop the object within one of the red boxes.



The terms Table, Pane and Cell define the scope for the object.

3 Edit an Analytics Object

To edit an object you have added from the Analytics pane, click the object and choose **Edit**.



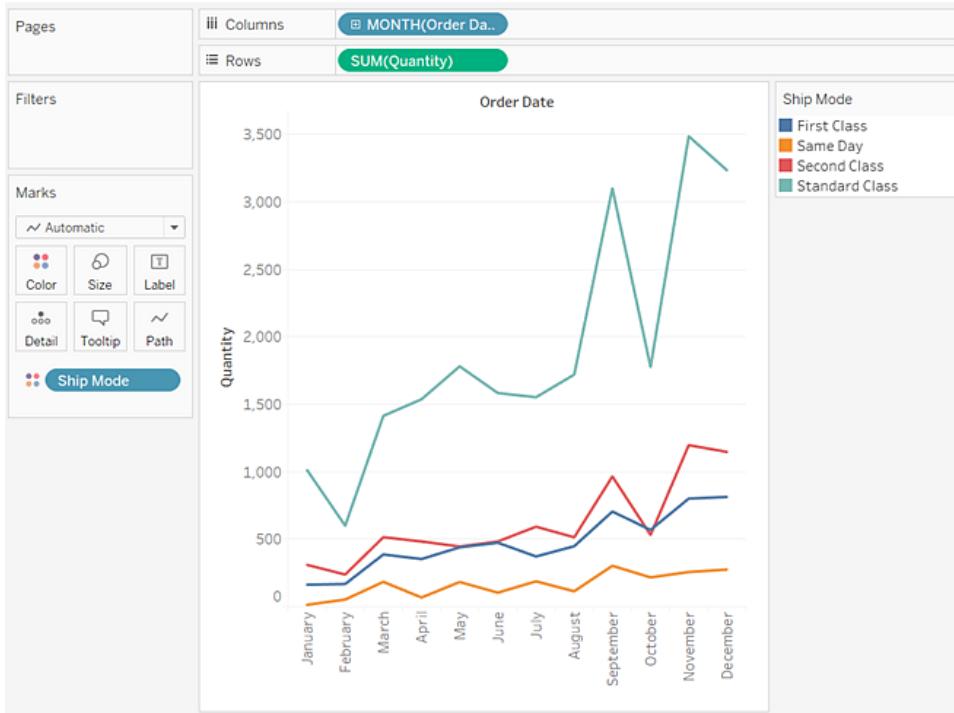
To edit a distribution band, you must click on the outer edge--clicking in the middle of the band has no effect.

Quick Start: Area Charts

An area chart is a line chart where the area between the line and the axis are shaded with a color. These charts are typically used to represent accumulated totals over time and are the conventional way to display stacked lines. Follow the steps below to create an area chart.

1 Build a Stacked line chart

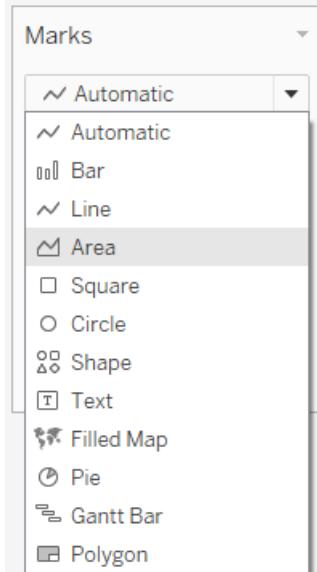
Drag a dimension to the Columns shelf and a Measure to the Rows shelf. Then drag another dimension to **Color** on the Marks card.



This example shows the sum of total quantity by ship mode.

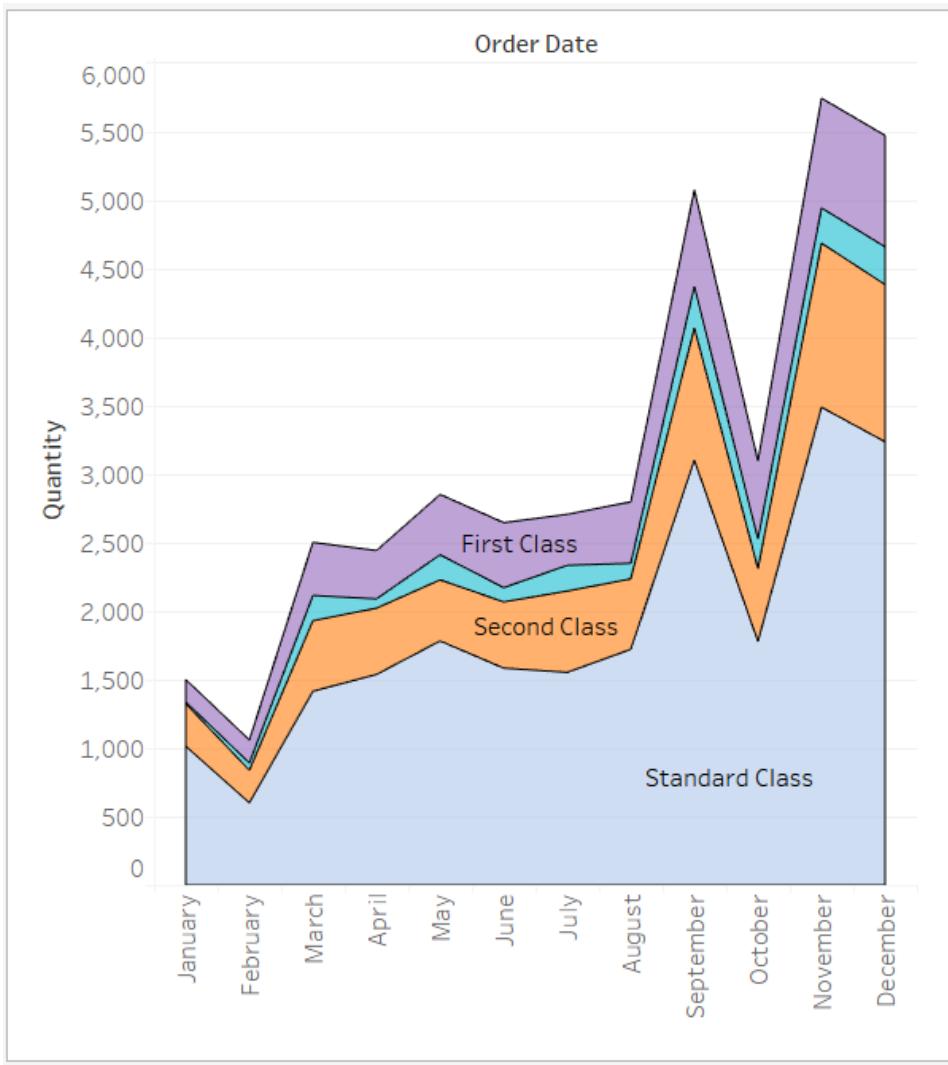
2 Select the Area mark type

The Automatic mark type will show this type of view as lines. On the **Marks** card, select **Area** to show the data as an area chart.



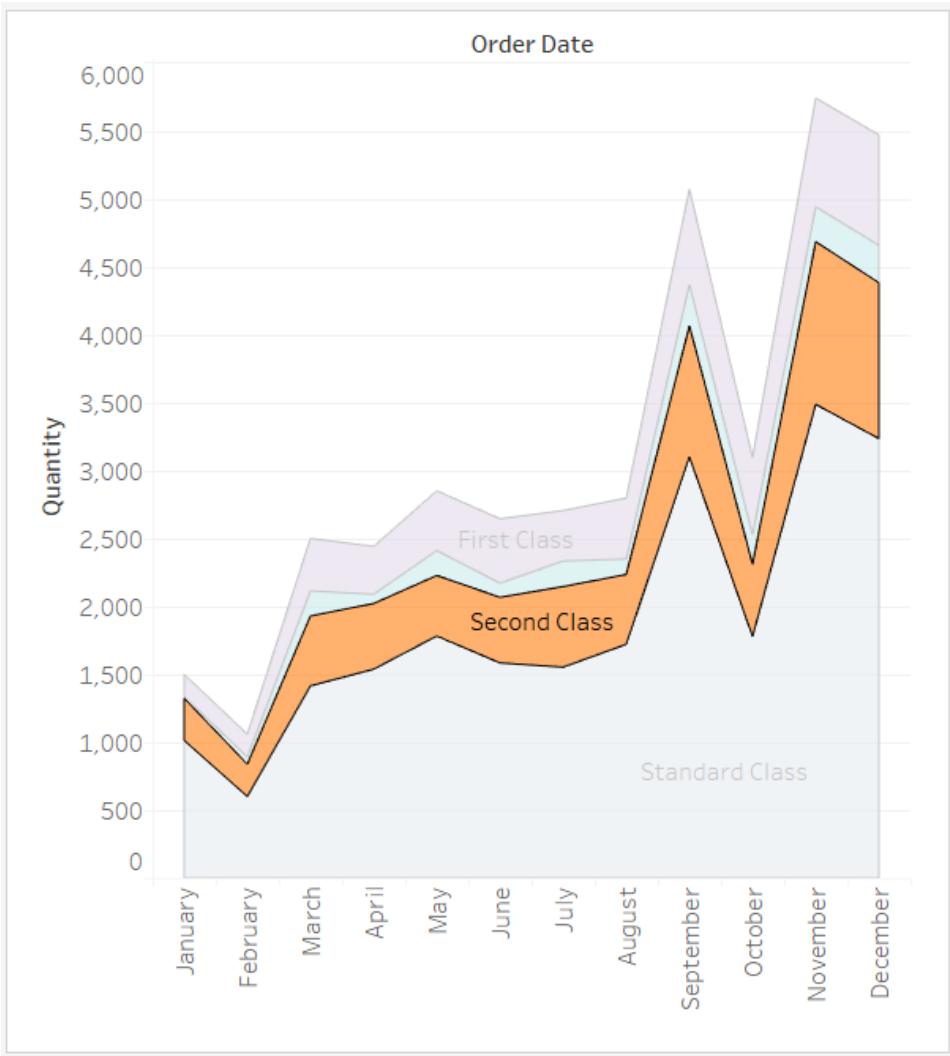
3 Adding formatting

You can add formatting to an area chart. For example, you can edit the color legend and turn on mark labels and borders.



4 Highlight the areas

You can also use highlight actions with area charts. For example, selecting a color in the legend or turning on the highlighter will highlight the entire area instead of just the line.

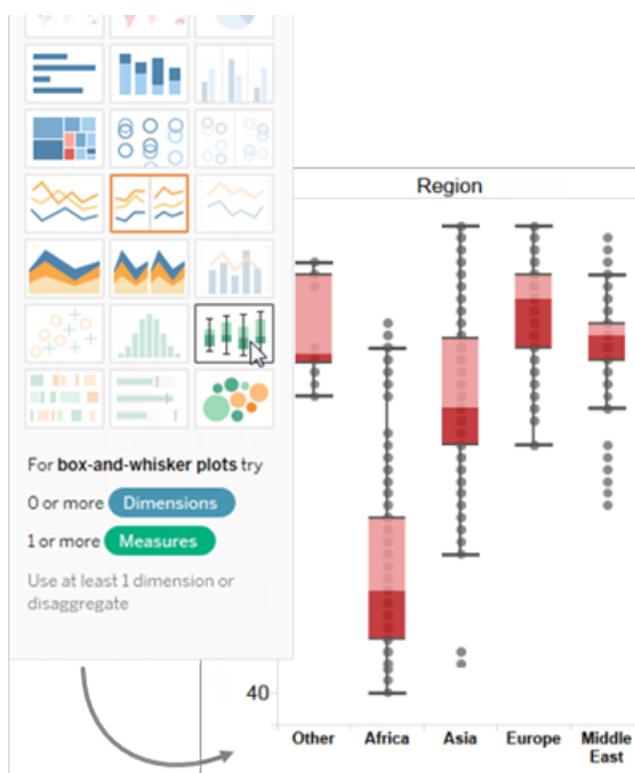


Quick Start: Box Plots

A box plot, also known as a box-and-whiskers plot, is a graphical display type well known to statisticians. Box plots provide quick insight into the distribution of data.

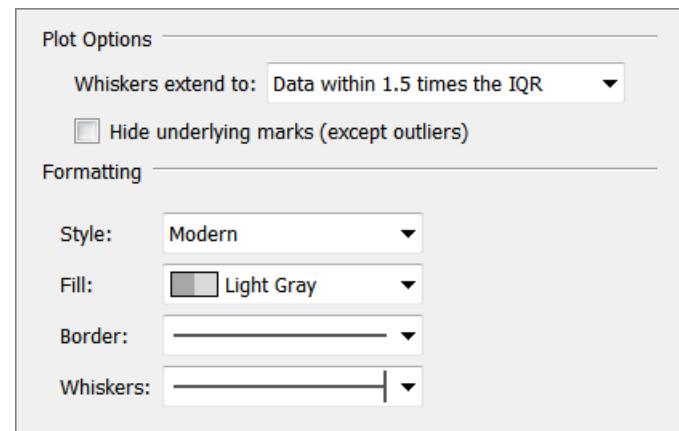
1 Use Show Me

Use Show Me to create a box plot.



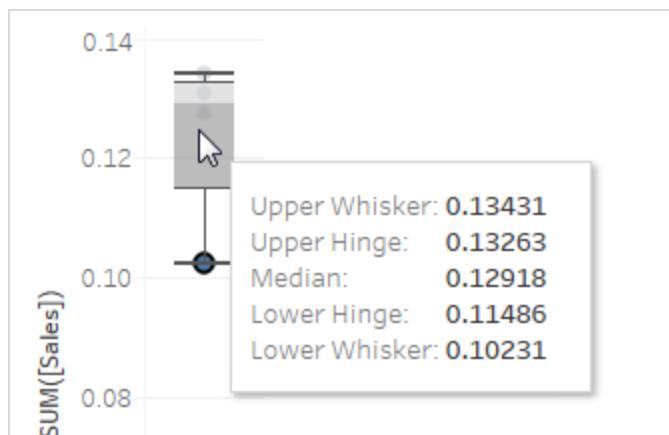
2 Set Box Plot Options

To change the appearance of a box plot, select the box plot in the view, and then click **Edit**. Set the whisker distance, specify whether to show individual marks, and select the style of the box plot.



3 View Box Plot Statistics

To see statistics for a box plot, hover the cursor over it.

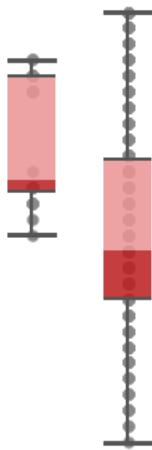


Box Plot Styles

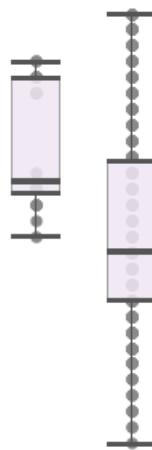
Tableau offers a range of different styles to choose from when you format a box plot.

Q-

Modern



Glass



Classic



Classic with Dual Fill

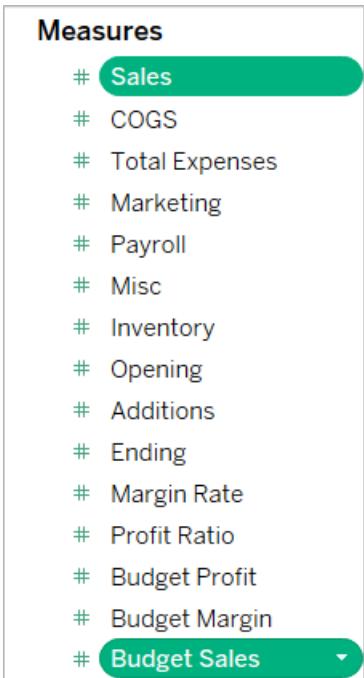


Quick Start: Bullet Graphs

A bullet graph is a variation of a bar graph developed to replace dashboard gauges and meters. A bullet graph is useful for comparing the performance of a primary measure to one or more other measures. Below is a single bullet graph showing how actual sales compared to estimated sales. Follow the steps below to learn how to create a bullet graph.

1 Select two measures to compare

Hold down the Ctrl key (⌘ on the Mac) on your keyboard and select two measures in the Data pane.



2 Click Show Me on the toolbar

In **Show Me**, click the **Bullet Graph** view type.



For **bullet graphs** try

0 or more **Dimensions**

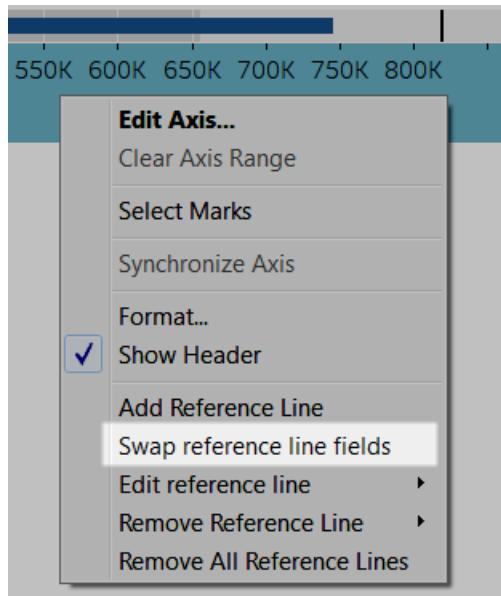
2 **Measures**

Right-click the continuous axis
to swap reference lines

Two types of reference lines are added: a line to indicate a target or goal, and a distribution to show ranges of performance.

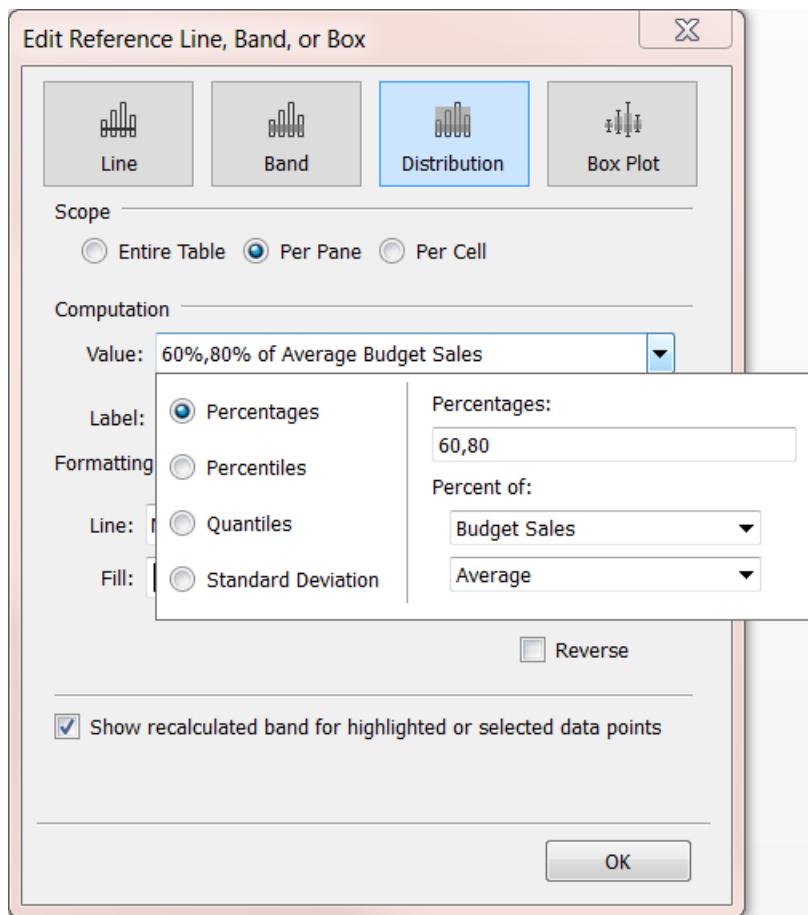
3 Swap Reference Line Fields

Sometimes you might want to swap the reference lines fields. For example, the actual sales is shown as a reference distribution instead of a bar. To swap the two measures, right-click (control-click on the Mac) the axis and select **Swap Reference Line Fields**.



4 Refine by editing the reference lines

Right-click (control-click on the Mac) the axis in the view and select **Edit Reference Line**, and then select one of the reference lines to modify.



Add a measure to **Detail** to make it available as the target measure.

Quick Start: Find Clusters in Data

Cluster analysis partitions marks in the view into clusters, where the marks within each cluster are more similar to one another than they are to marks in other clusters.

1 Add clusters to a view

Drag **Cluster** from the **Analytics** pane into the view, and drop it on in the target area in the view:

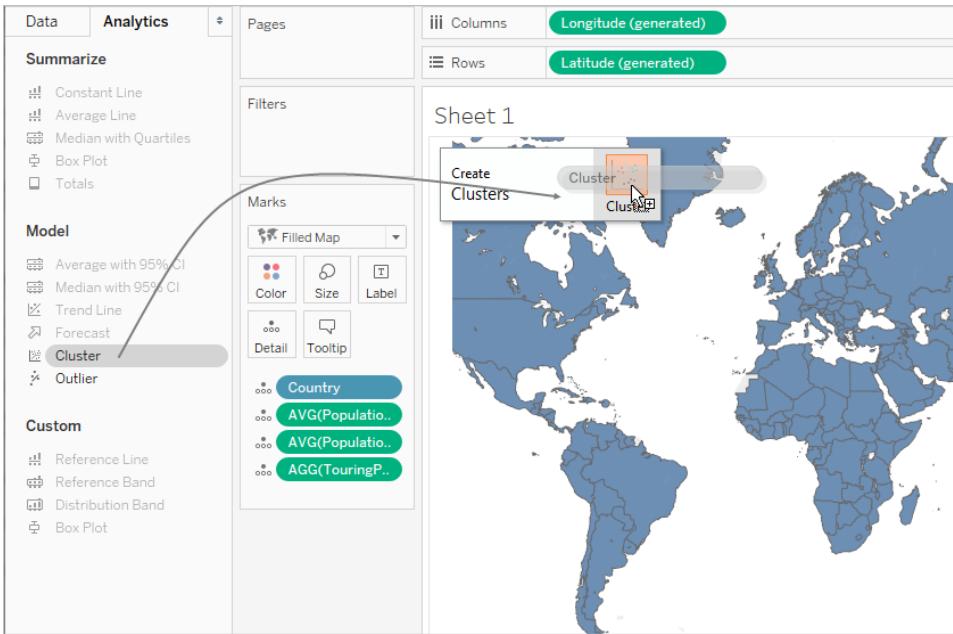
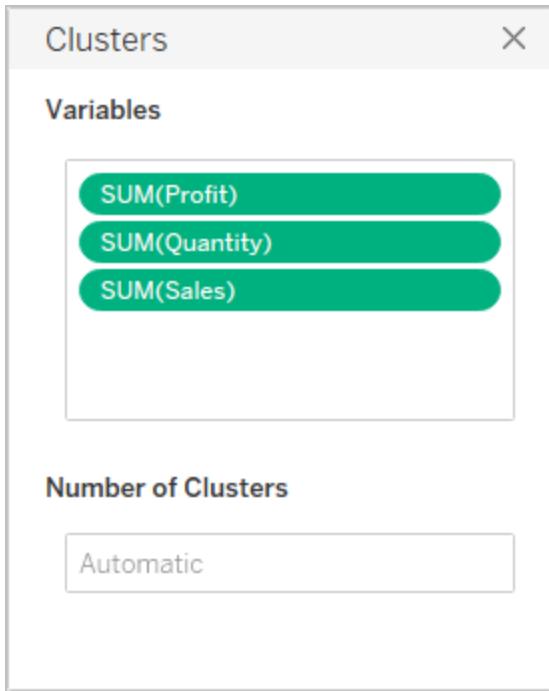


Tableau adds **Clusters** on **Color**, and colors the marks in your view by cluster.

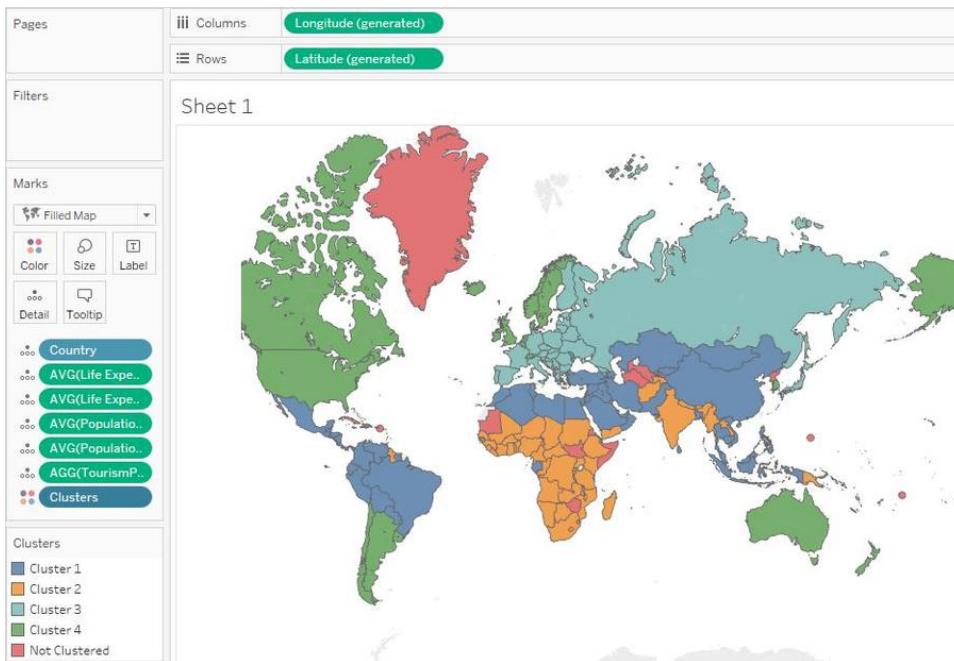
2 Configure clustering results

Let Tableau determine the optimal number of clusters, or specify exactly how many clusters you want, in the Clusters dialog box. You can also add or remove variables—the fields that determine the clusters. The default variables are the measures in the view when you create the clusters.



3 Assess the result

Does the view give you the insight you were aiming for? If not, you can continue to fine-tune the clusters by changing the number of clusters or adding or removing variables. To re-open the Clusters dialog box, click the **Clusters** field on the Marks card and choose **Edit clusters**.



4 See what the statistics say

It's not enough to know which marks are in what cluster. You also need to know what factors influenced the results. To get another perspective on clustering results, click the **Clusters** field on the **Marks** card and choose **Describe clusters**.

The table at the bottom of the **Models** tab in the Describe Clusters dialog box shows the average value for each variable in each cluster:

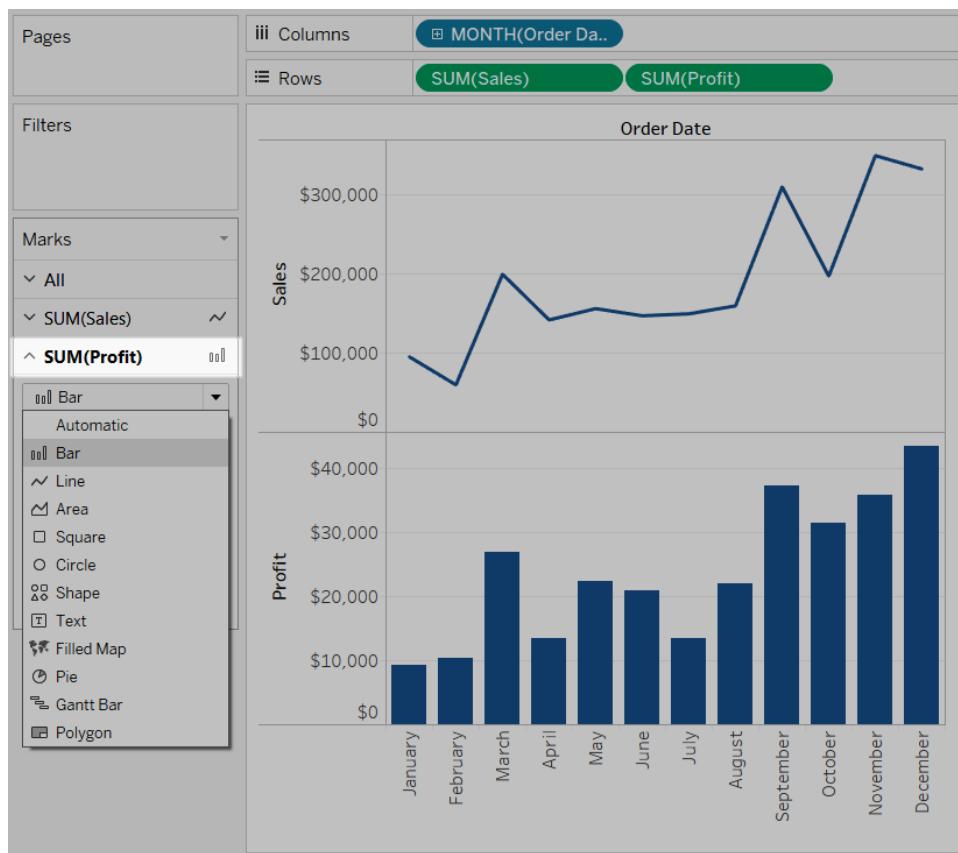
Clusters	Number of Items	Avg. Life Expectancy Female	Avg. Life Expectancy Male	Avg. Population 65+	Avg. Population Urban	TourismPerCapita
Cluster 1	69	74.216	69.003	0.054943	0.5325	197.12
Cluster 2	52	56.75	54.429	0.031889	0.33127	20.775
Cluster 3	29	79.164	71.706	0.15493	0.65532	320.92
Cluster 4	30	81.287	76.355	0.11606	0.87123	1360.4
Not Clustered	28					

Quick Start: Combination Charts

Combination charts are views that use multiple mark types in the same sheet. For example, you may show sum of profit as bars with a line across the bars showing sum of sales. You can also use combination charts to show multiple levels of detail in the same view. For example, you can have a line chart with individual lines showing average sales over time for each customer segment, then you can have another line that shows the combined average across all customer segments.

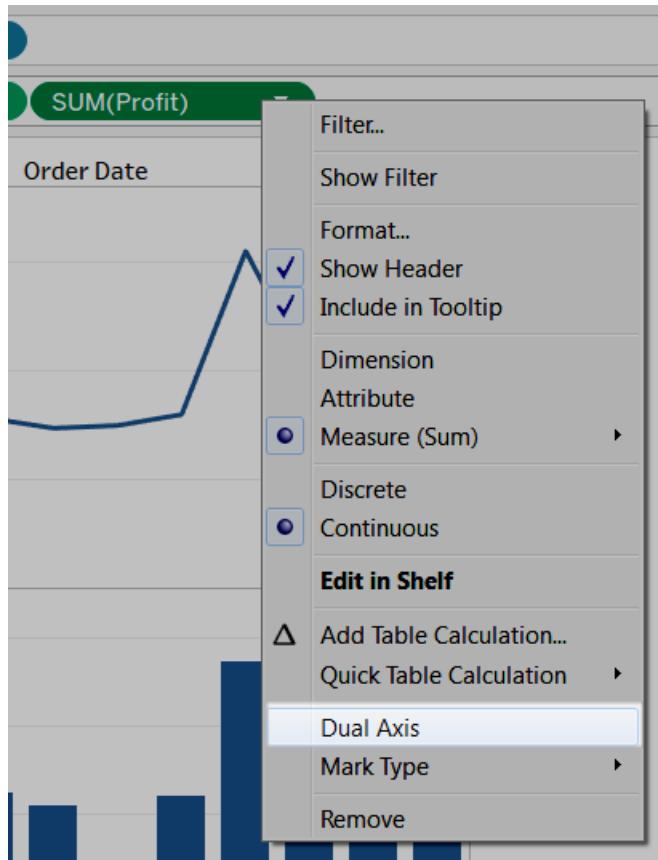
1 Customize marks

Click one of the measure names on the **Marks** card, and then select a mark type that is different than the other measure on the **Marks** card.



2 Layer multiple panes

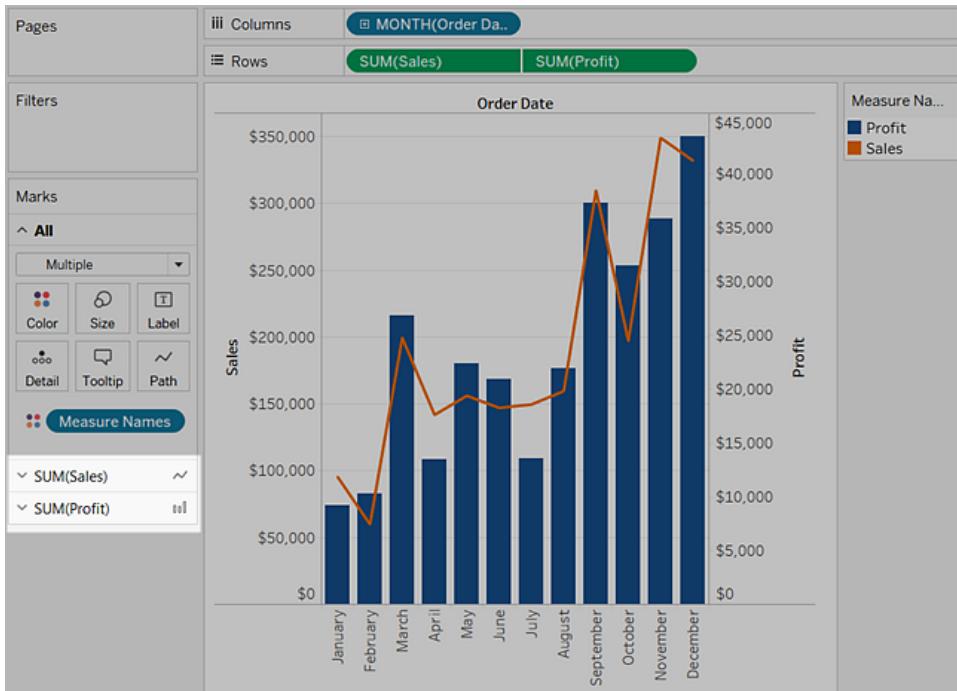
Right-click a measure on the **Rows** or **Columns** shelf and select **Dual Axis**. The two mark types are now layered on top of each other.



Some marks can be hidden behind others. To move the marks forward or backward, right-click (control-click on a Mac) an axis in the view and select **Move Marks to Back** or **Move Marks to Front**.

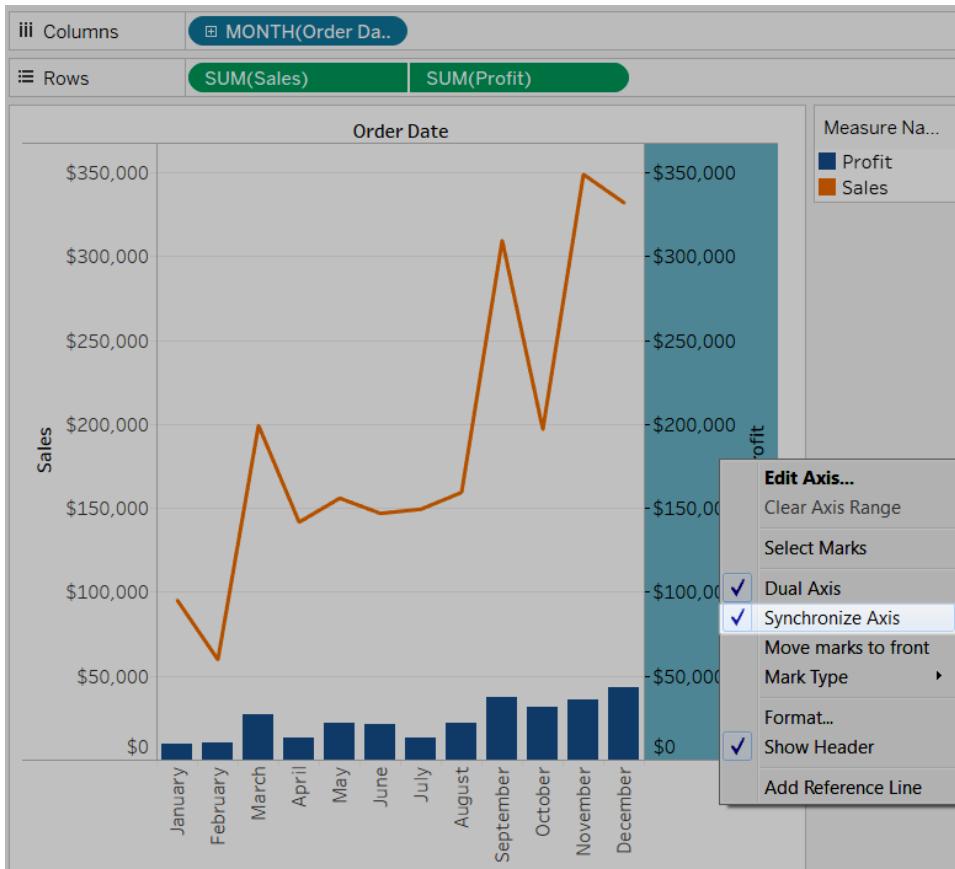
3 Switch between panes

To switch between the layered panes, on the **Marks** card, select a measure name. For each pane you can change the mark type, colors, shapes, sizes, and other options.



4 Synchronize the axes

Right-click the secondary axis in the view and select **Synchronize Axis** to keep the two axes on the same scale.



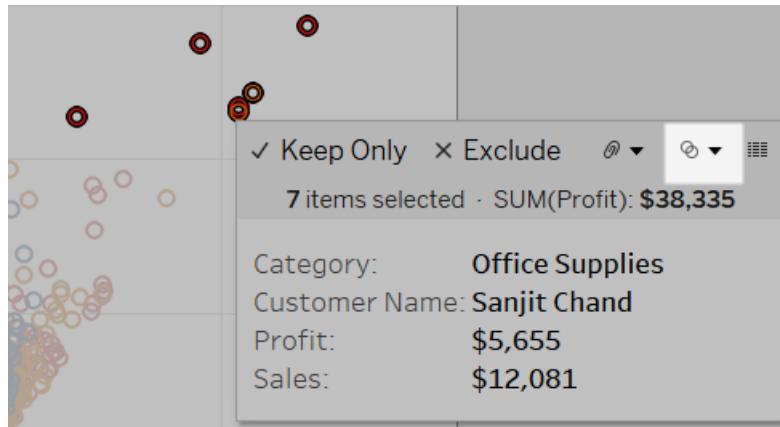
Quick Start: Combine and Compare Sets

Sets are subsets of your data that meet certain conditions. A set can be based on a computed condition, such as a set that contains only customers with sales over a certain threshold.

Computed sets update as your data changes. Alternatively, a set can be based on specific data points in your view. Follow the steps below to create a new set, add and remove data from sets, and combine sets.

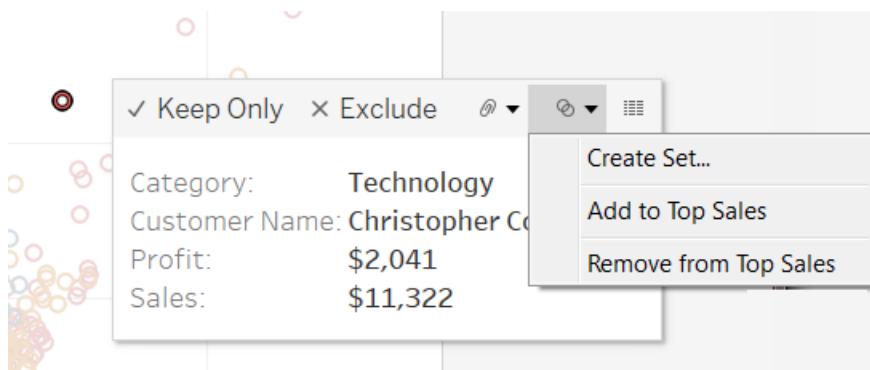
1 Create a Set

There are many ways to create a set. To create a set with specific data points, select the marks in the view and then select **Create Set** on the tooltip. Alternatively, right-click (control-click on Mac) a dimension in the **Data** pane and select **Create Set** to create a computed set. Complete the **Create Set** dialog box.



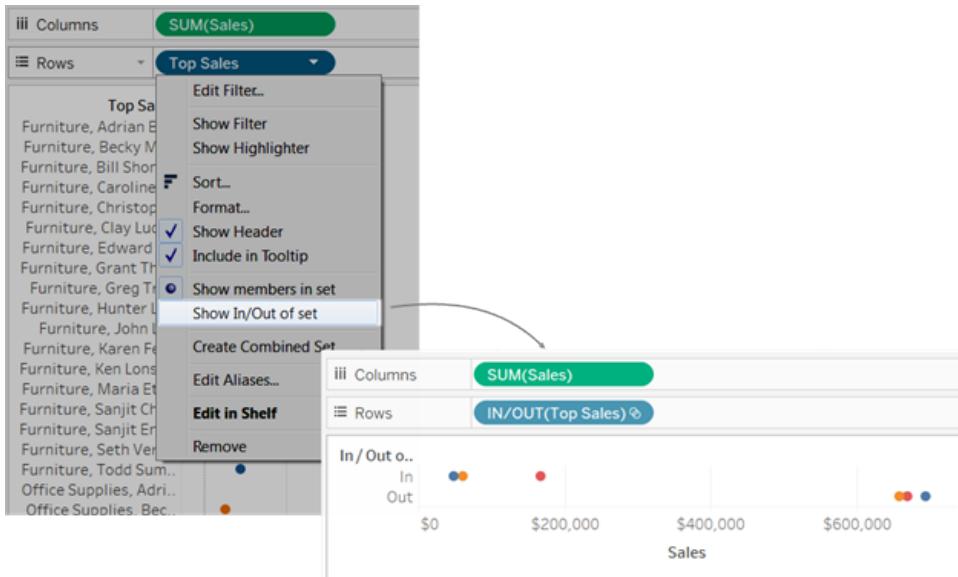
2 Add and remove from Sets

If you created a set using specific data points, you can add more data to or subtract data from the set. After you select marks in the view, click the **Sets** drop-down menu on the tooltip and click **Add to** or **Remove from** to add or remove data from a particular set.



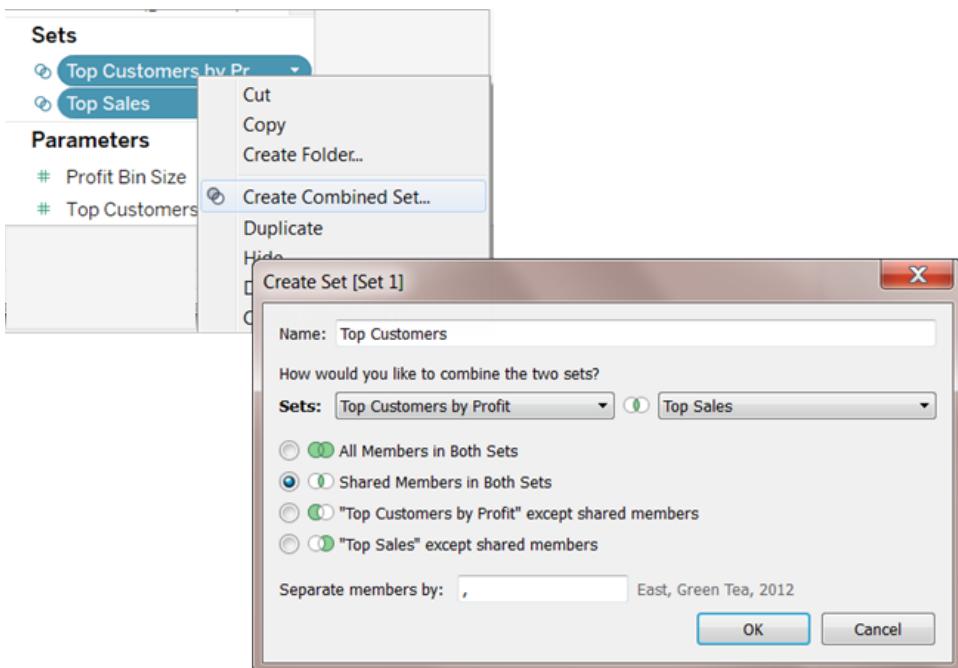
3 Compare In/Out values

Instead of looking at the individual members of a set, you can instead compare the members that are in the set to the ones that are not. To show in/out values in a set, right-click (control-click on Mac) a set in the view and select **Show In/Out of Set**.



4 Create combination sets

To combine multiple sets into a new combination set, right-click (control-click on Mac) two sets in the **Data** pane and select **Create Combined Set**. In the **Create Set** dialog box, specify how you want to combine the two sets.



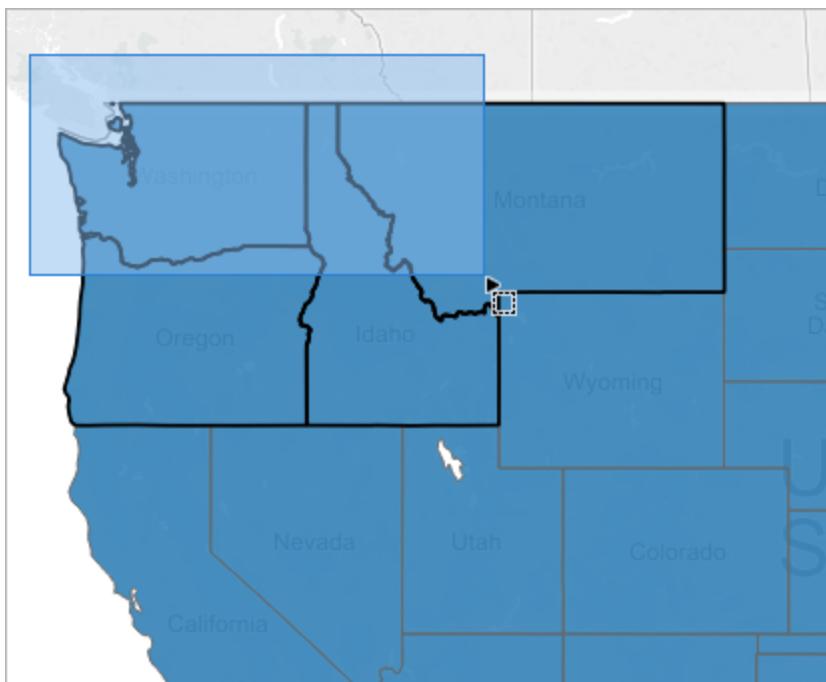
Create Custom Territories on a Map

You can create custom territories on a map by grouping locations together. For example, if you have a map of total sales for each state in the United States, you can combine several or all of those states into a group to create a sales territory.

When you create territories, Tableau adds a separate geographic group field in the **Data** pane. Your existing geographic fields are not changed.

1 Select locations to group into a territory

On a map view, select several locations that you want to group into a custom territory.



2 Create the first group (your first territory)

On the tooltip that appears, click the **Group Members** icon, and then select the level of detail you want to group the locations by.



A new group field is added in the **Data** pane with a group icon next to it. The new field is also added to **Color** on the **Marks** card.

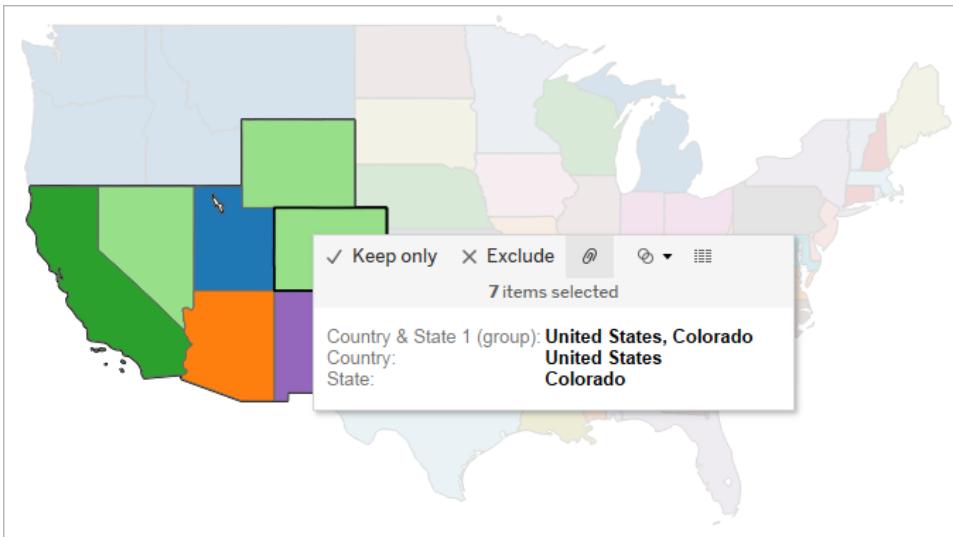
Group	Field
Location	Country & State (group)
Color	Country
Color	State
Group	Country & State 1.

To edit the locations in your territories at any time, right-click the group field in the **Data** pane and select **Edit group**.

3 Continue grouping data to add additional territories

Repeat steps 1 and 2 to create additional territories. Each territory is added as a group within the group field you created in step 2. You can create as many territories as you want.

Note: You might have to right-click the group field on the Marks card and select **Include 'Other'** to continue grouping locations.



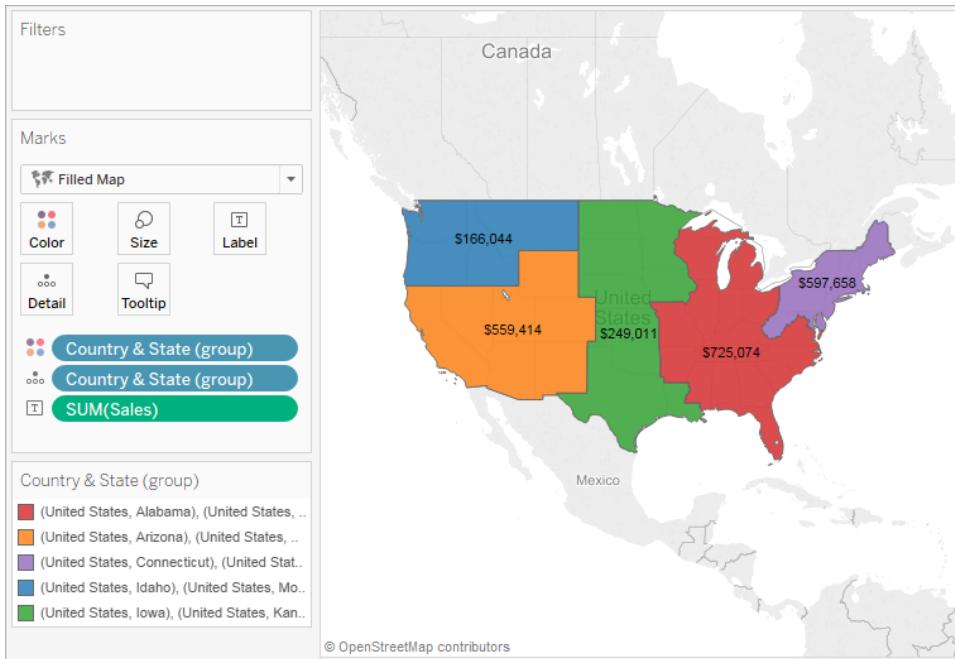
To edit the colors of your territories, click **Color** on the **Marks** card and select **Edit Colors**.

4 Add the group field to the view

From the **Data** pane, drag the newly created group field to **Detail** on the **Marks** card. Next, to make sure that the territory appears correctly, you must also remove the related geographic field(s) from the view.

For example, if you grouped states at the State, Country level, you must remove the State and Country fields from the view. The group field replaces these fields on **Detail** on the **Marks** card.

In the image below, notice that the State and Country fields are no longer on **Detail** on the **Marks** card because they have been replaced by the Country & State (group) field.



For this example, there are five groups within the Country & State (group) field, so five territories appear on the map. Aggregations, such as SUM(Sales), are computed at the level of the territory, rather than separately for each location within the territory. For example, in the image above, the orange territory includes the sum of sales for all states in that territory combined: 559,414 USD.

Create custom territories in the Group dialog box

Q-

You can also create custom territories by creating groups in the Data pane.

1. In the **Data** pane, right-click a geographic field (such as City or State) and select **Create > Group**.
2. In the Create Group dialog box, type a name for your group and then select the locations you want in your first territory and click **Group**. Each group you create represents a territory.
3. Repeat step 2 until you've created all of your territories, and then click **OK**.

The new group field is added to the **Data** pane.

4. From the **Data** pane, drag the newly created group field to **Detail** on the **Marks** card.

You can also drag the field to **Color** or **Label** on the **Marks** card to help differentiate between each territory.

See Also:

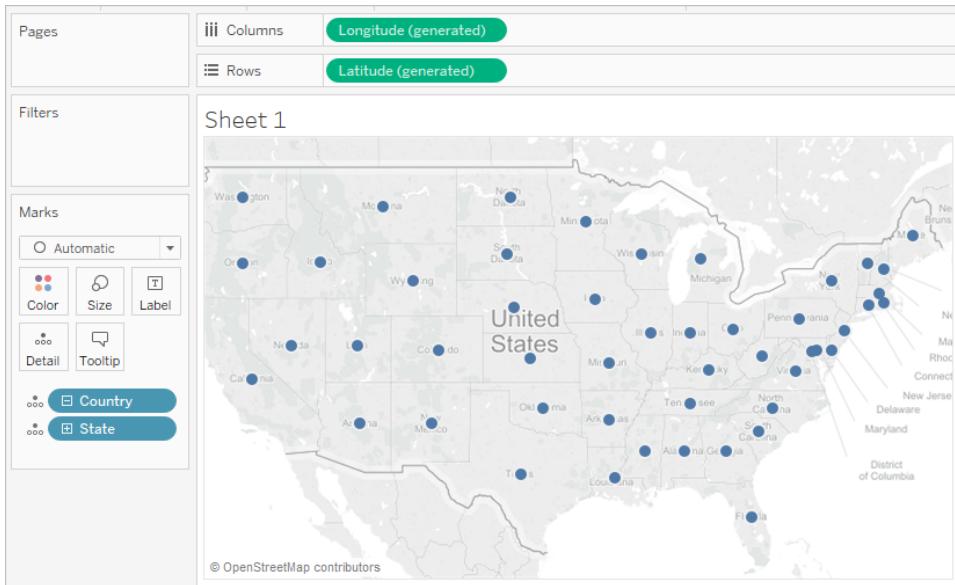
- [Create Territories on a Map](#) on page 774
- [Build a Map View](#) on page 744

Quick Start: Filled Maps

When mapping data in Tableau, a mark is shown for each location. The mark can be a shape such as a circle, square, pie, etc. Alternatively, you can fill the polygon for the geographic region with a color based on a data value. For example, you can fill states with a color based on how profitable they are.

1 Build a Map View

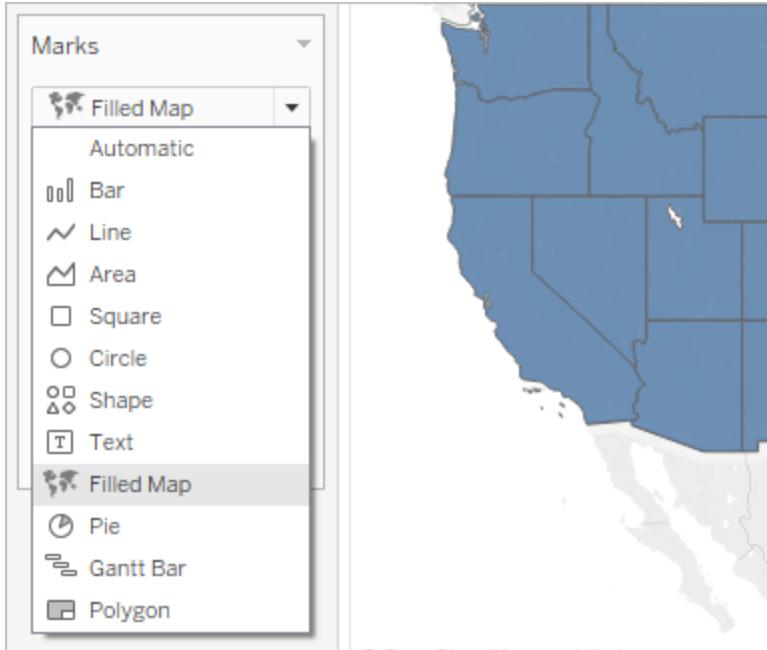
Double-click a geographic field such as State, Area Code, Zip Code, etc.



The generated **Latitude** and **Longitude** fields are moved to the **Rows** and **Columns** shelves and the geographic field is moved to **Detail** on the **Marks** card.

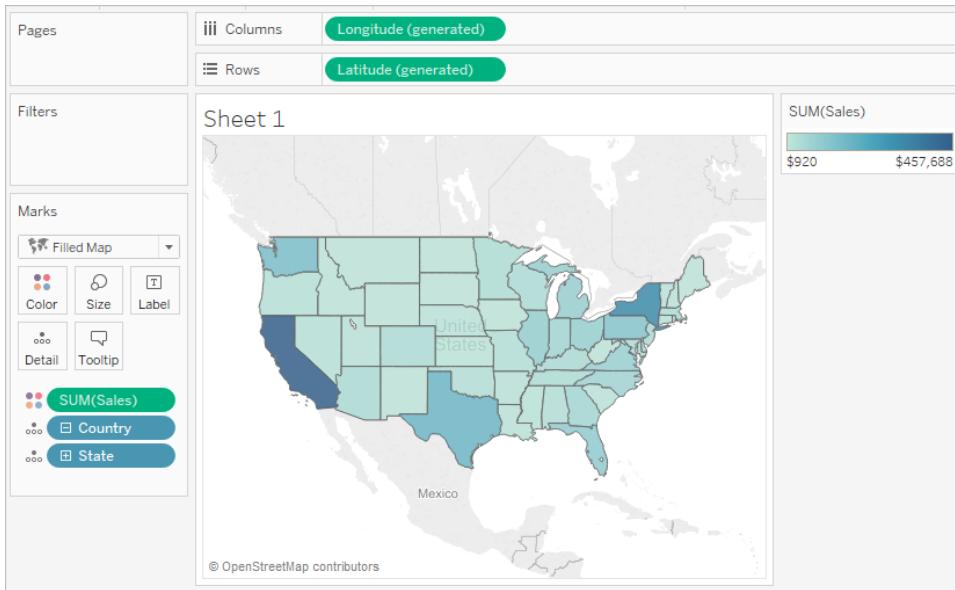
2 Select the Filled Map Mark Type

The Automatic mark type will show this type of view as circles over a map. On the Marks card, click the Mark Type drop-down menu and select **Filled Map** to color the geographic areas.



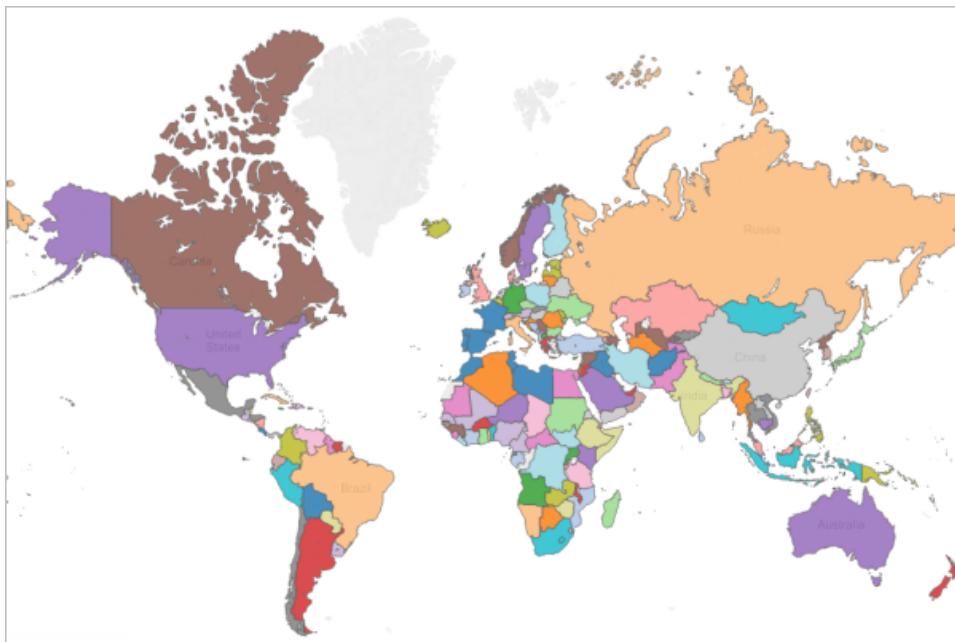
3 Drag a Field to the Color Shelf

Define how the locations are colored by dragging another field to **Color** on the **Marks** card.



4 Worldwide Regions

Filled maps are available at the country and state/province level worldwide.



Quick Start: Filter Data Across Multiple Data Sources

You can filter data across multiple primary data sources in Tableau.

For example, suppose you have a worksheet that shows population size by each county in California, and that uses data source A. Suppose you also have a worksheet that shows the

average education level by county in California that uses data source B. You can filter both of those worksheets by creating a county filter from data source A and applying it to data source B.

You can do this because the two worksheets use data sources that have data in common (county names).

Before you start: See the [Cross data-source filtering FAQs](#) forum post in the Tableau Community.

1 Create relationships between your data sources

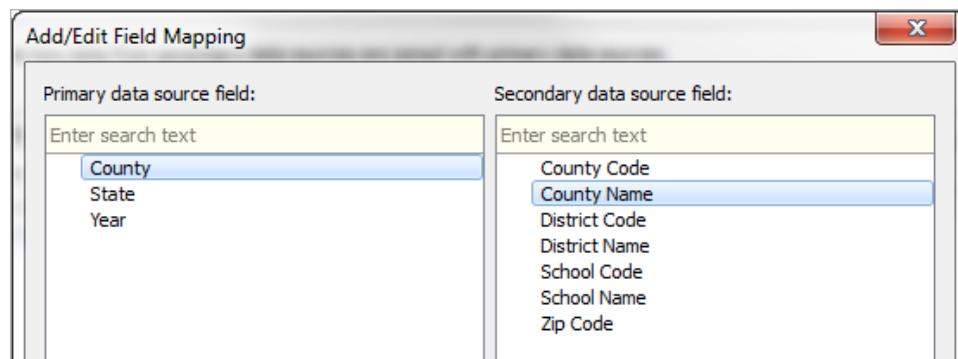
To filter data across multiple primary data sources, you must first create a relationship between one or multiple fields. Sometimes Tableau does this automatically because your data sources have fields in common, as in the above example. This is called an automatic relationship.

If Tableau doesn't automatically create relationships between your data sources based on a field they have in common, you can create the relationship between the two data sources manually.

To create a relationship between your two data sources:

1. Select **Data > Edit Relationships**.
2. In the Edit Relationships dialog box, select a data source to be the primary and a data source to be the secondary, click **Custom**, and then click **Add**.
3. In the Add/Edit Field Mapping dialog box, do the following, and then click **OK**:
 - Under **Primary data source field**, select a field.
 - Under **Secondary data source field**, select a field with similar data to the field you selected for the primary data source.

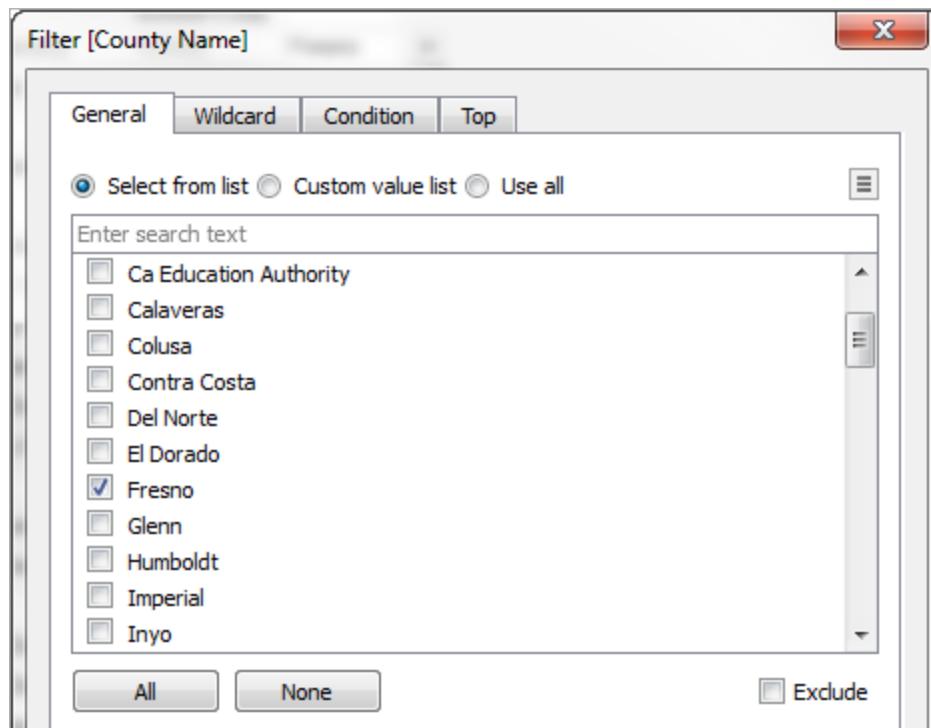
For example, in the image below, the field, **County**, in the primary data source is related to the field, **County Name**, in the secondary data source.



2 Add a field to the Filters shelf

From the **Data** pane, drag a dimension field to the **Filters** shelf. In the Filter dialog box that opens automatically, select the members you want to include or exclude from the view, and then click **OK**.

Note: If the Filter dialog box does not open automatically, right-click the field on the Filters shelf and select **Edit filter**.



3 Apply filter to views with related data sources

To apply the filter to all the worksheets in your workbook that use related data sources, right-click the field on the Filters shelf and select **Apply to worksheets > All using related data sources**.

Related data sources in this case are data sources in your workbook that have a relationship with the field on the Filters shelf. If you recall, a relationship is created by defining fields that your data sources have in common.

After you apply a filter to all related data sources, an icon appears next to the field on the **Filters** shelf to indicate that it is a cross data source filter.



To apply the filter to select worksheets in your workbook, select **Apply to worksheets > Selected worksheets**. In the Apply Filter to Worksheets dialog box, you can select to apply the filter to sheets that use the same data source or that use related data sources.

After you apply a filter to selected worksheets with related data sources, a different icon appears next to the field on the **Filters** shelf to indicate that it is a cross data source filter.

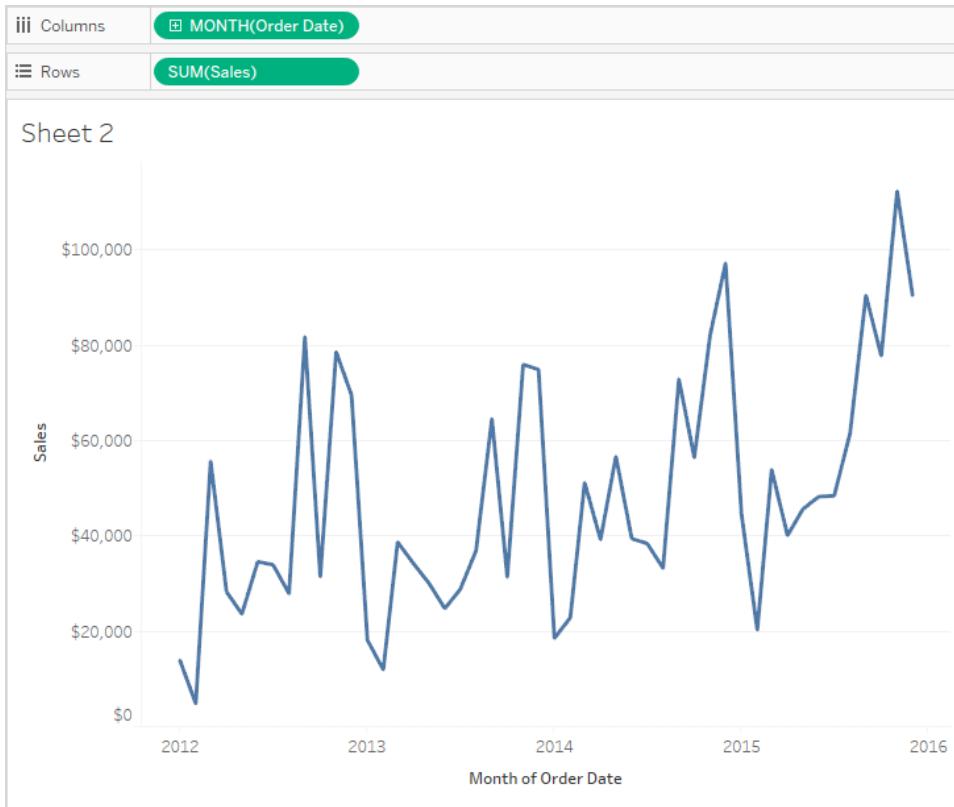


Quick Start: Forecasting

Forecasting is a calculation that predicts future trends based on current trends and data. Follow the steps below to add a forecast to your view.

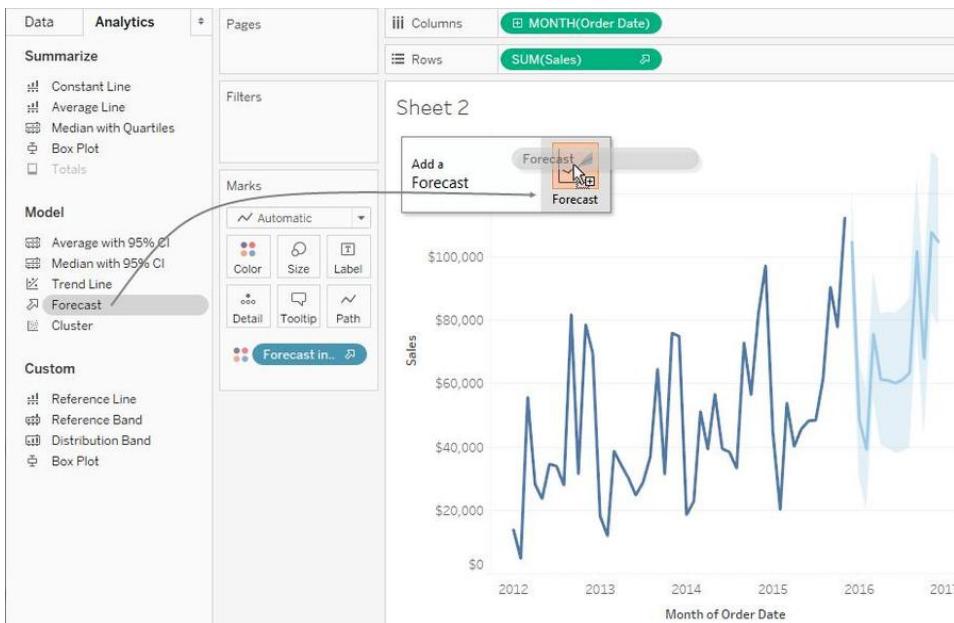
1 Build the View

Build a view that shows a measure over time. The date or time field can be discrete or continuous. For example, the view below shows sales by Month and Year.



2 Add a Forecast to the View

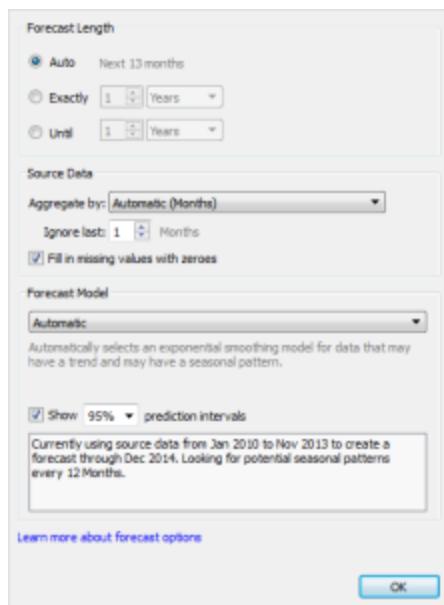
To add a forecast to the view, drag **Forecast** from the **Analytics** pane and drop it on **Forecast** in the view. The time axis is extended and estimated values are shown.



You can also add a forecast to the view by selecting **Analysis > Forecast > Show Forecast**.

3 Change the Forecast Options

You can modify the forecast options by selecting **Analysis > Forecast > Forecast Options**. In the **Forecast Options** dialog box, you can specify how far into the future to estimate, the date part that the forecast is based on, and the forecasting model to use.



4 View Forecast Details

To view the forecast model and the parameters used to create the forecast, select **Analysis > Forecast > Describe Forecast**.

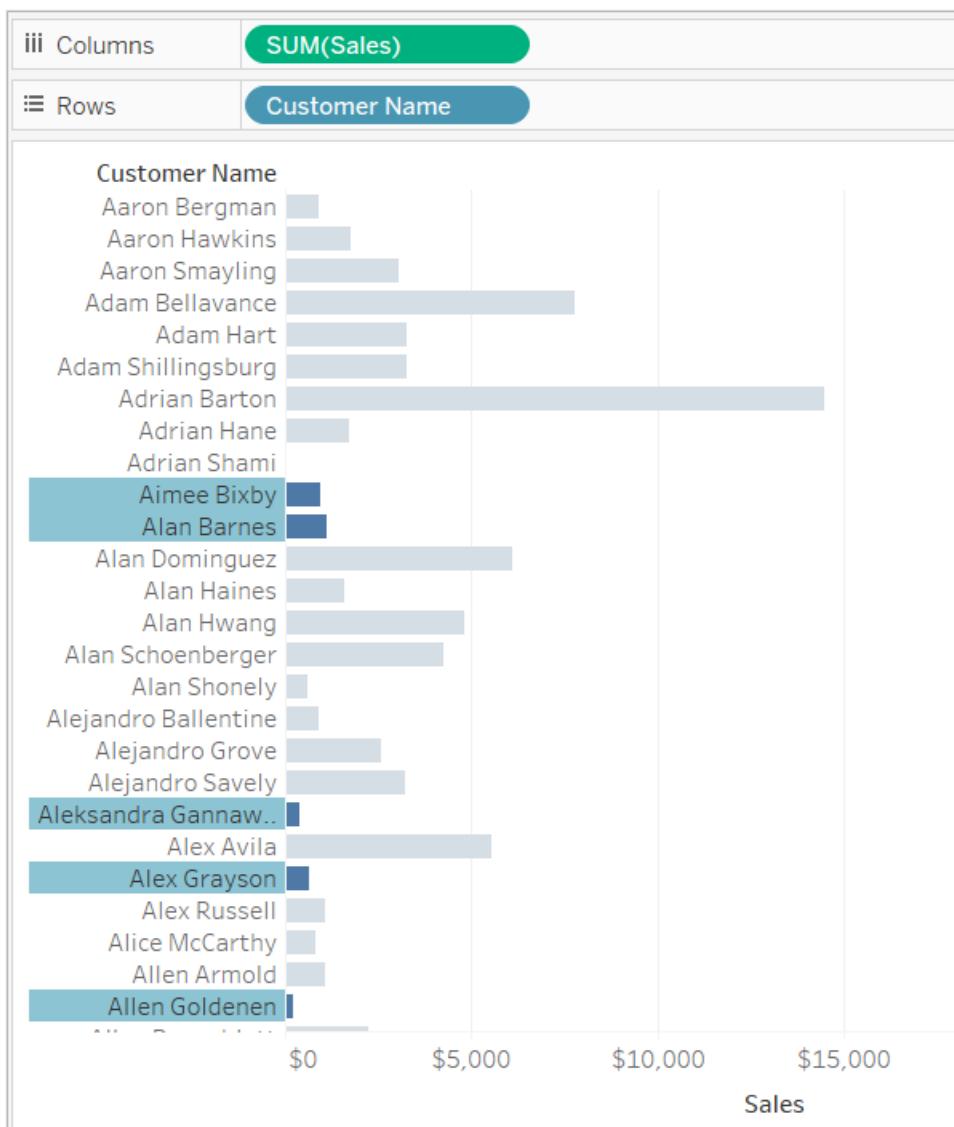


Quick Start: Groups

Sometimes you will want to group several dimension members into categories that don't already exist in your data. For example, when looking at a list of customers, you may want to group them by outside information, such as the type of customer they are: regular, employee, etc.

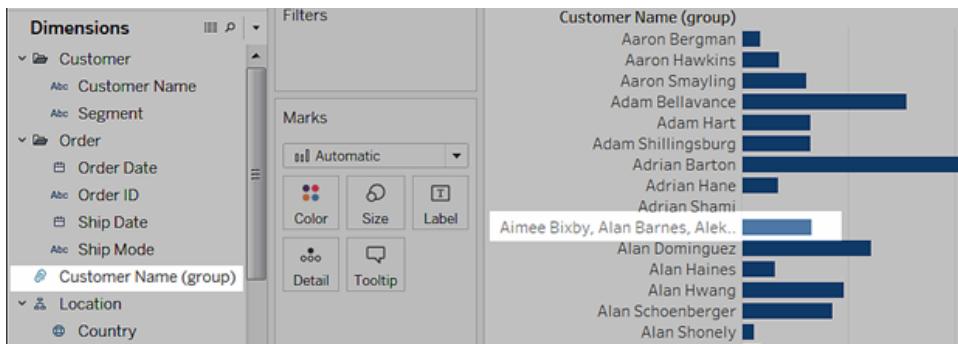
1 Select the members to group

Select one or more members that you want to group.



2 Click Group on the toolbar

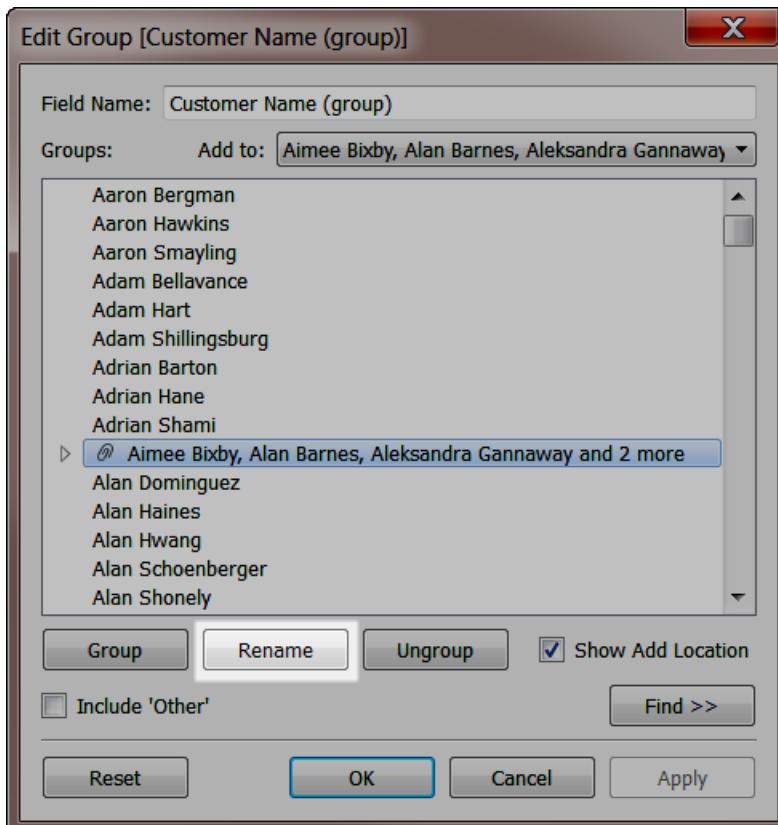
Click the **Group Members** icon  on the toolbar.



The members are grouped into a single member in the view and the new group field is shown in the Data pane.

3 Edit the name of the group

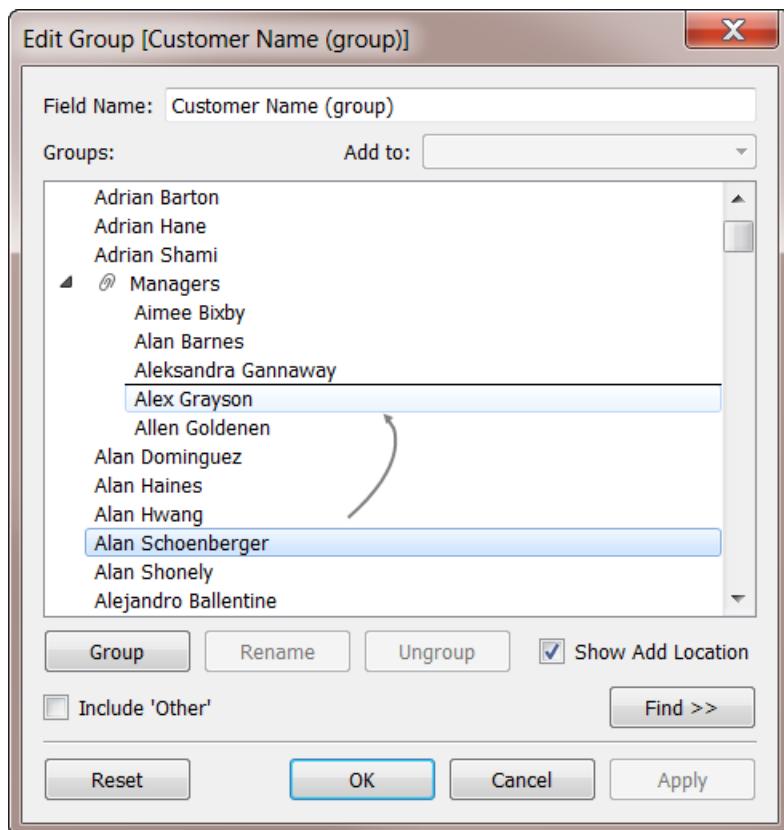
In the Data pane, right-click (control-click on Mac) the grouped field and select **Edit Group**. Select the group and then click, **Rename**.



You can also rename the name of the field itself using the text box at the top of the dialog box.

4 Add and remove members

In the **Edit Group** dialog box, add and remove members by dragging and dropping the selected members in the list to and from the group.



To quickly add members to a group, right-click (control-click on Mac) a member in the **Edit Group** dialog box and select **Add to**, then in the **Add To Group** dialog box, select the group you want to add them to.

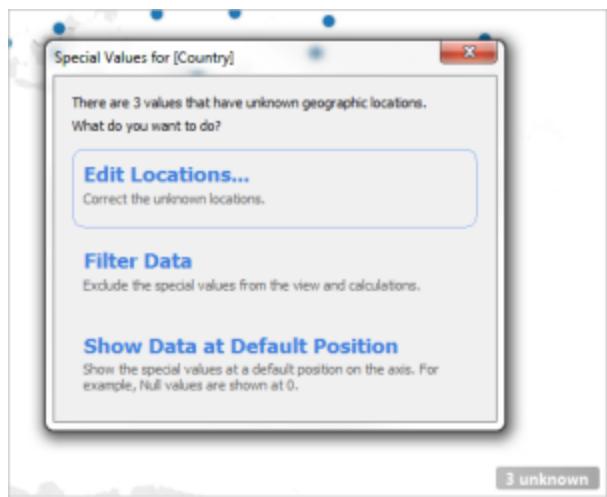
Finding members

If you are working with a dimension that has a lot of members, you can search and select members that match a specific criteria using the **Find** button in the **Edit Group** dialog box.

Q-

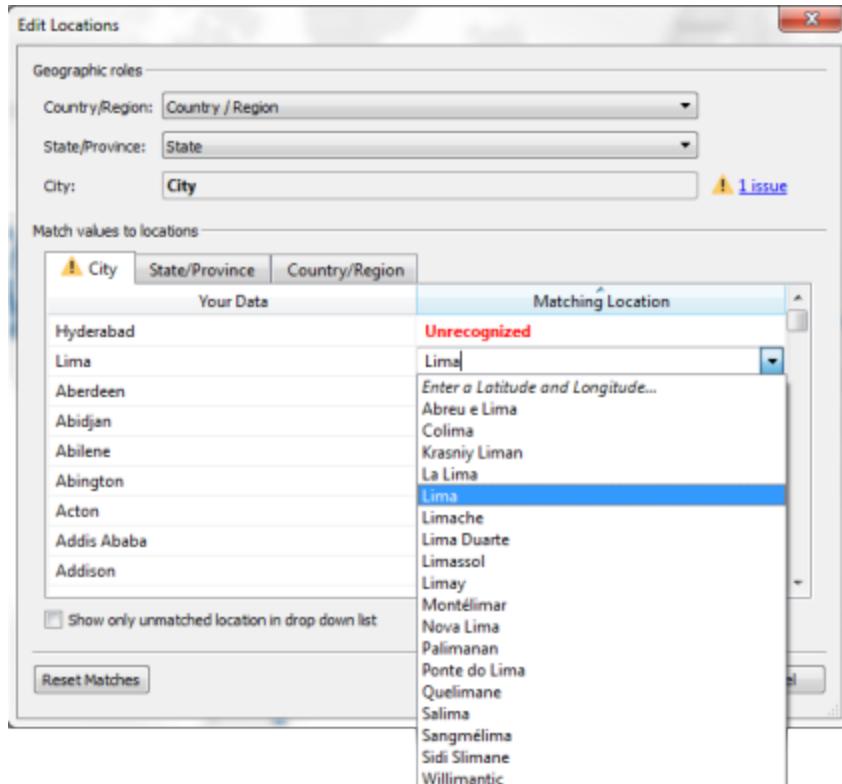
Quick Start: Null Values

Null values along an axis are indicated in the lower right corner of the view. Click the indicator to open a dialog box that will help you decide how to handle these values. The options available in the dialog box depend on the type of data in the view. The options for geographic data, numbers, and dates are described below.



Unknown Geographic Locations

If you are mapping data using the built-in geographic roles in Tableau, you may have some unrecognized locations that cannot be mapped. To edit these locations, click **Edit Locations** in the **Special Values** dialog box. In the **Edit Locations** dialog box, match your data to known locations. For example, your data may include “Mass” instead of “Massachusetts” as a state name.

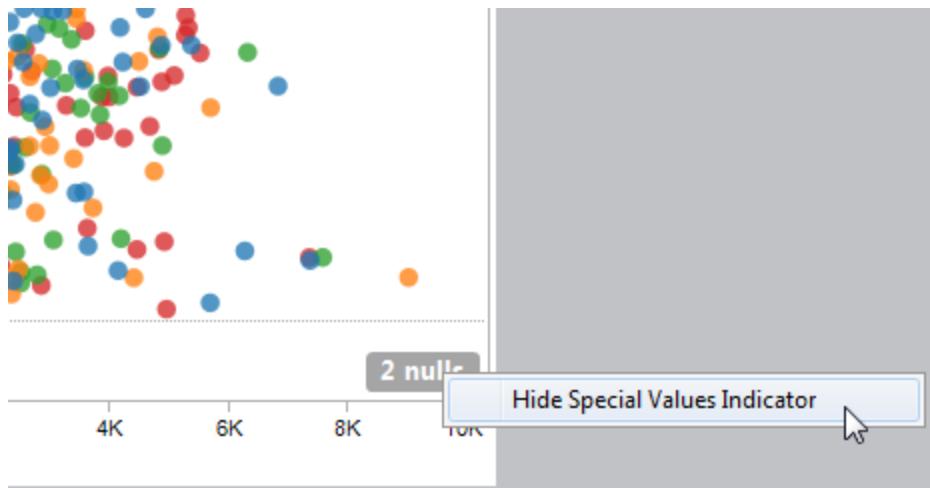


Null Numbers and Dates

If a numeric or date axis contains null values, you can choose to filter the data, which removes null values from the view and all calculations, or show them at a default position. The default positions are described in the following table.

Type of Data	Default Position
Numbers	0
Dates	12/31/1899
Negative Value on a Log Axis	1
Unknown Geographic Locations	(0, 0)

By default, the values are shown with the indicator in the lower right corner of the view and are included in calculations. Right-click (control-click on Mac) the indicator to hide it.



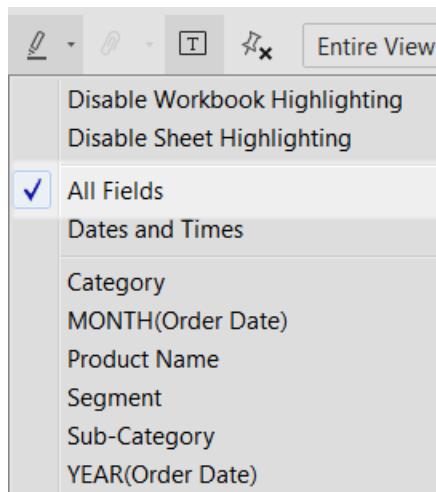
Quick Start: Highlight Actions

You can highlight data in your views to make information stand out using a variety of methods. For example, you can use Highlight Actions to highlight related data in a variety of views based on your selection in the current view. Highlight Actions are particularly useful on a dashboard, where the highlight provides rich visual interaction between the views on the sheet.

For information about the different methods that you can use to highlight data, see [Highlight Actions on page 1120](#)

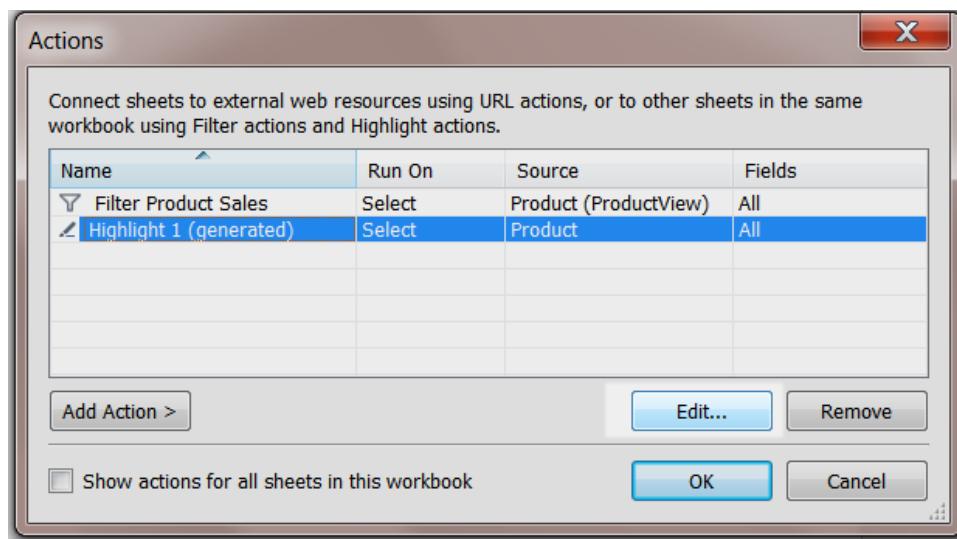
1 Turn on Highlighting

To turn on highlighting in a dashboard, click the **Highlight** icon on the toolbar and select an option. Select **All Fields** to highlight all fields in the view, select **Dates and Times** to highlight dates and times in the view, or select a specific field in the view to highlight.



2 Refine the Highlight Action

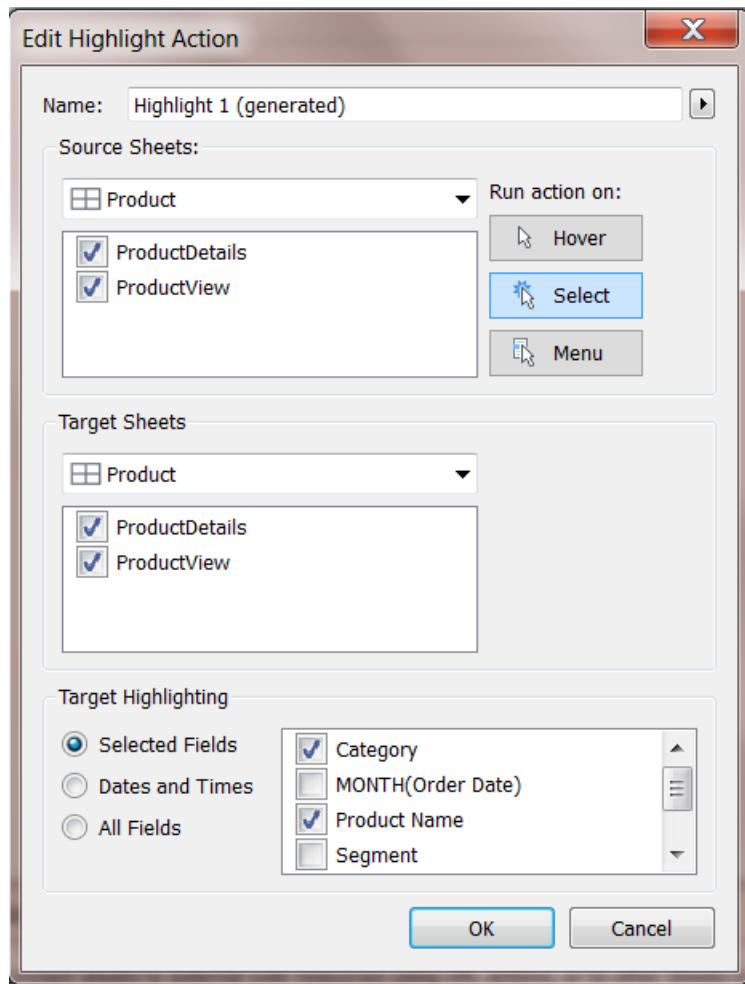
Select **Dashboard > Action** to open the **Actions** dialog box. Select an action and then click **Edit**.



Any action you create appears in the **Actions** dialog box. You can add Filter, Highlight, or URL actions.

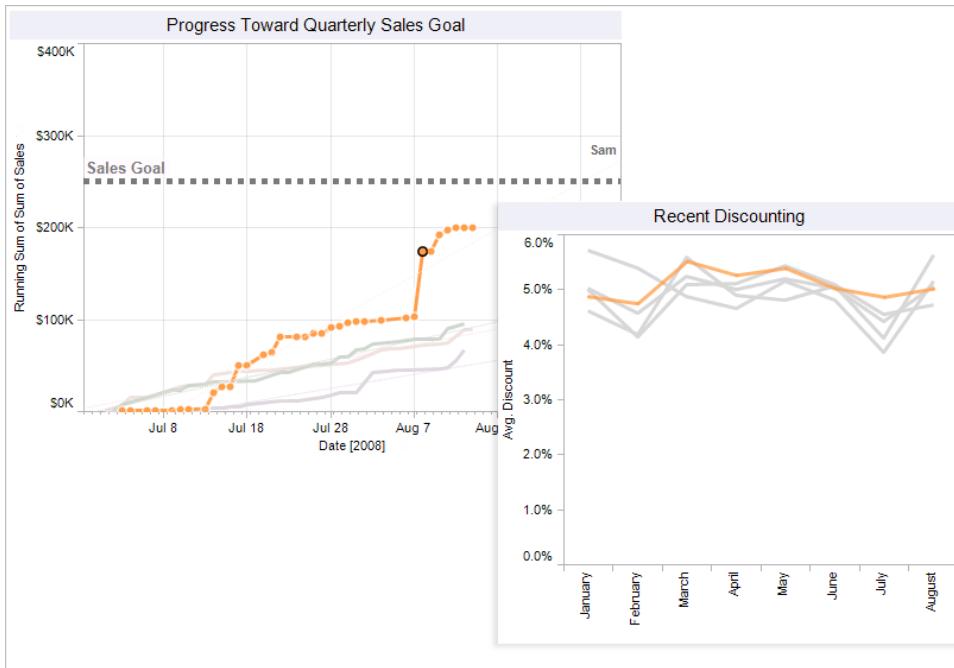
3 Specify the Action Settings

In the **Edit Highlight Action** dialog box, select the source sheets, target sheets, and the fields to highlight. You can also select whether to run the action when you hover, select, or right-click (control-click on Mac) the source sheet. When finished, click **OK**.



4 See It in Action

In your dashboard, hover, select, or right-click (control-click on Mac) the source view to see the related data highlighted in the target views.



A selection in the source view highlights the related information in all of the target views.

Quick Start: Measure Distance in Maps with the Radial Tool

While you explore data in Tableau maps, you might have questions about how the data relates to its surroundings.

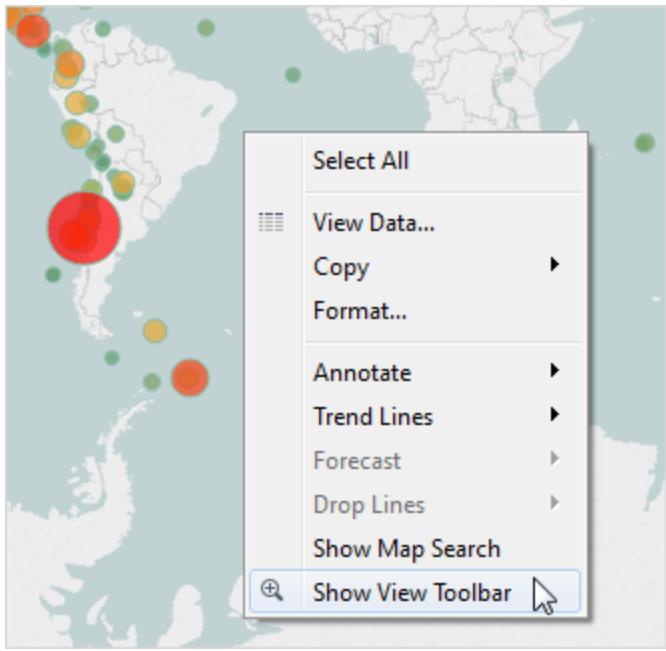
For example, you might wonder how many earthquakes have occurred within 25 miles of a city, or how many public transit stops are within a kilometer of a popular tourist destination.

You can use the Radial tool to measure approximate distances between your data and the locations or landmarks in your map view.

1 Show the View Toolbar

The Radial tool button is located on the view toolbar, which appears in the top-left corner of the view when you hover over a map. If the view toolbar doesn't appear in the view, it is hidden, and you can choose to show it.

To show the view toolbar, right-click (control-click on Mac) anywhere in the view and select **Show View Toolbar**.

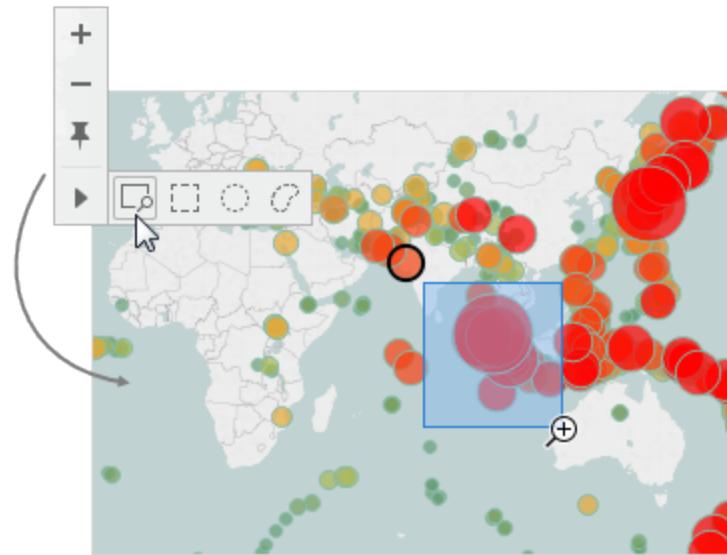


You can also use keyboard shortcuts to navigate the view and select marks. For more information, see [Keyboard Shortcuts](#) on page 1611.

2 Zoom In to an Area or Location

Before you can measure distance in your map view using the Radial tool, you need to first zoom in to a particular area or location in the map.

To zoom in, click the **Zoom In** button on the view toolbar. To zoom quickly to an area of the view, double-click the area on the map.

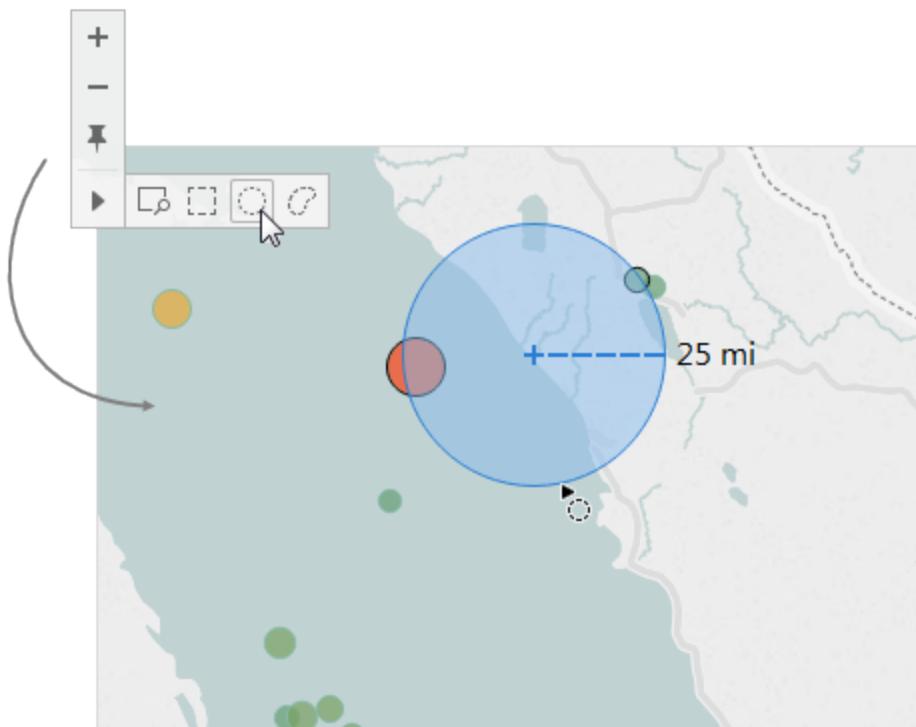


You might need to zoom in several times before you can measure distance with the Radial tool. For more information, see the [Measurement accuracy](#) on page 798 section of the [Measure Distances between Data Points and locations in a Map](#) topic.

3 Select the Radial Tool

To select the Radial tool, hover over the arrow on the view toolbar and click the Radial tool button. Next, click a location, landmark, or area on the map that you want to measure from, and then drag across the view.

As you drag, the Radial tool selects the marks that are within the radius of the circle. In the example below, the radius is 25 miles. This means that the two marks selected are within approximately 25 miles of the chosen location.



Units of Measurement

By default, your workbook locale determines which units the Radial tool uses to measure distance. If your workbook locale is set to a country that uses the Imperial system, such as the United States or the United Kingdom, the Radial tool measures distance in feet and miles. If your workbook locale is set to a country that uses the Metric system, such as Brazil, the Radial tool measures distance in meters and kilometers.

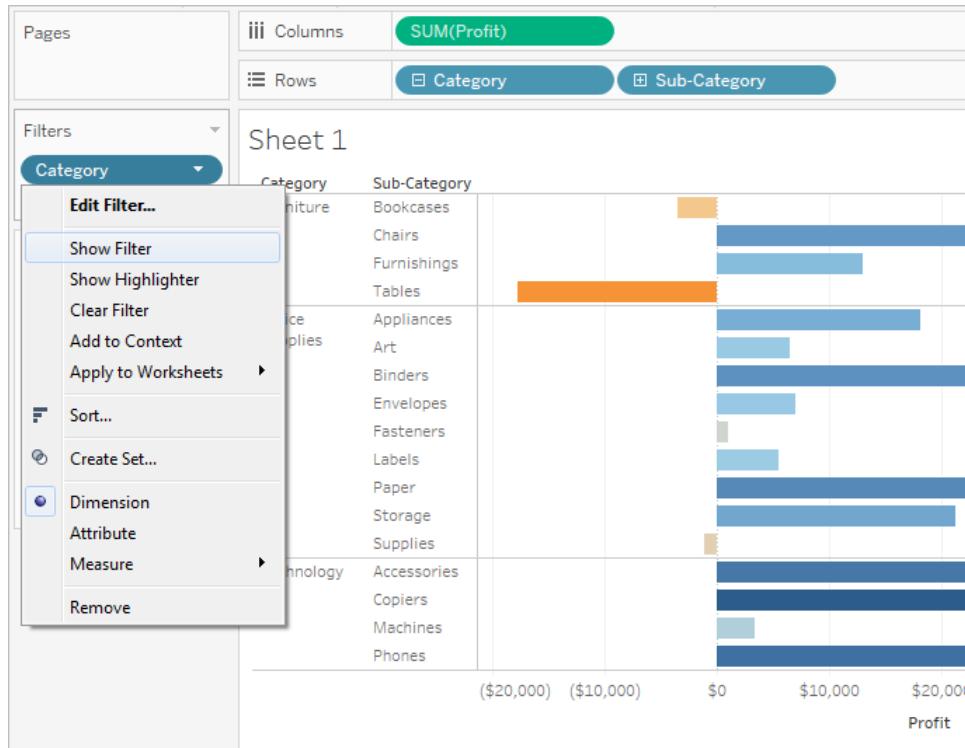
To change the units the Radial tool uses to measure distance, select **Map > Map Options**. Next, under Units, click the drop-down menu and select an option. For more information, see [Change the units of measurement](#) on page 798.

Quick Start: Filter Modes

When you show a filter in the view, you can choose a filter mode that best fits your data. This quick start shows you the filter modes for categorical (dimension) filters. Quantitative (measure) filters have different filter modes.

1 Show a filter in the view

In the **Data** pane, or on the **Filters** shelf, click the field drop-down menu and select **Show Filter**. A filter card appears to the right of your visualization.



You can show filters for any field, whether the field is in the view or not.

2 Select a filter mode

In the view, click the filter card drop-down menu and select one of the following seven filter modes:

Single Value (List)

Select one value at a time in a list.

Single Value (Dropdown)

Select a single value in a drop-down list.

Single Value (Slider)

Drag a horizontal slider to select a single value.

Category
<input type="radio"/> (All)
<input checked="" type="radio"/> Furniture
<input type="radio"/> Office Supplies
<input type="radio"/> Technology

Category
Furniture

Category
Furniture

↔

Multiple Values (List)

Select one or more values in a list.

Multiple Values (Drop-down)

Select one or more values in a drop-down list

Multiple Values (Custom List)

Search and select one or more values.

Category
<input type="checkbox"/> (All)
<input checked="" type="checkbox"/> Furniture
<input checked="" type="checkbox"/> Office Supplies
<input type="checkbox"/> Technology

Category
(Multiple values)
<input type="checkbox"/> (All)
<input checked="" type="checkbox"/> Furniture
<input checked="" type="checkbox"/> Office Supplies
<input type="checkbox"/> Technology

Category
Furniture
Office Supplies

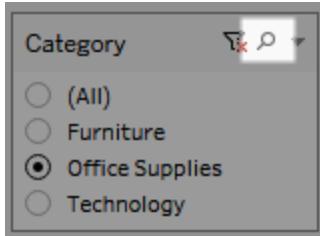
Wildcard Match

Select values containing the specified characters.

Category
furniture

3 Search filter values

Some filter modes support search so you can quickly find and select values. To search a filter shown in the view, click the **Search** icon in the title area of the filter in the view, and then type the value you want to find. The matching results display below the search box where you can select or clear them.



Use the **(All)** option to quickly select and clear all values in the list.

Quick Start: Rank and Percentile

Rank

Use rank when you are interested in seeing the order of values in a list. For example, you could apply a rank table calculation to a table to see which product category ranked highest in sales over a four-year period.

Right-click a field (or control-click on a Mac) and choose **Add Table Calculation**. Then choose **Rank** as the **Calculation Type** and **Table (Down)** as the value for **Running along**.

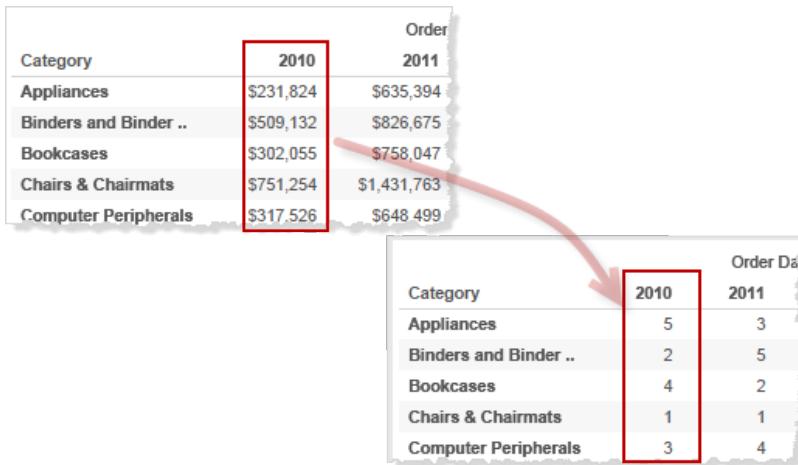


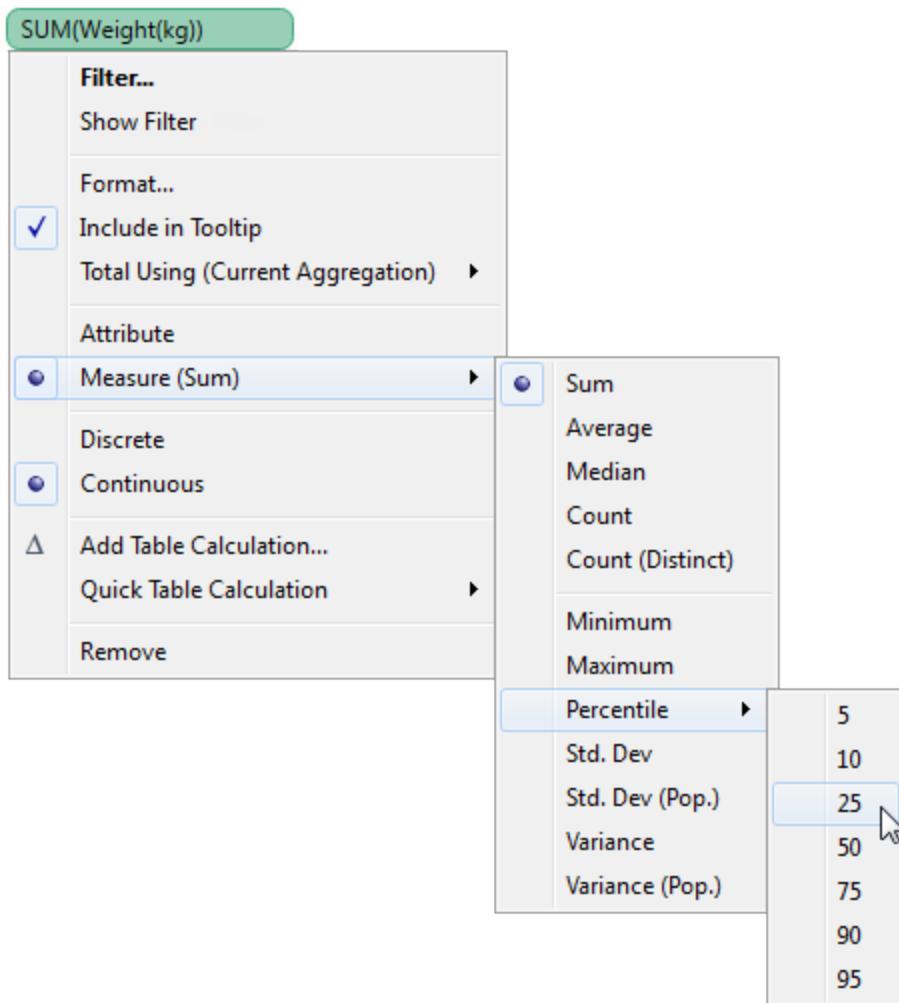
Tableau supports a range of ranking options, so you can specify how to rank data that contains duplicate values. For example, if you had a set of five values, and three of them were the same (10, 15, 15, 15, 24), you could configure the table calculation to rank them in any of the following ways:

- 1, 2, 2, 2, 5 (Competition ranking)
- 1, 4, 4, 4, 5 (Modified competition ranking)
- 1, 2, 2, 2, 3 (Dense ranking)
- 1, 2, 3, 4, 5 (Unique ranking)

Ranking functions that support these options are also available for use in calculated fields (for example, DENSE_RANK, MODIFIED_RANK).

Percentile

Percentile is available as an aggregation and as a table calculation. To use one of the built-in percentile aggregations, right-click a measure in the view and select **Measure > Percentile** and then choose one of the numeric options.



The Percentile aggregation requires an extract.

If none of the numerical values listed is what you want, you can right-click a measure in the view and choose **Add Table Calculation** to configure a custom percentile.

You can compute multiple percentiles on the same measure to show different percentiles simultaneously. The view below shows Weight at three different percentages, to display a range of percentiles over time.

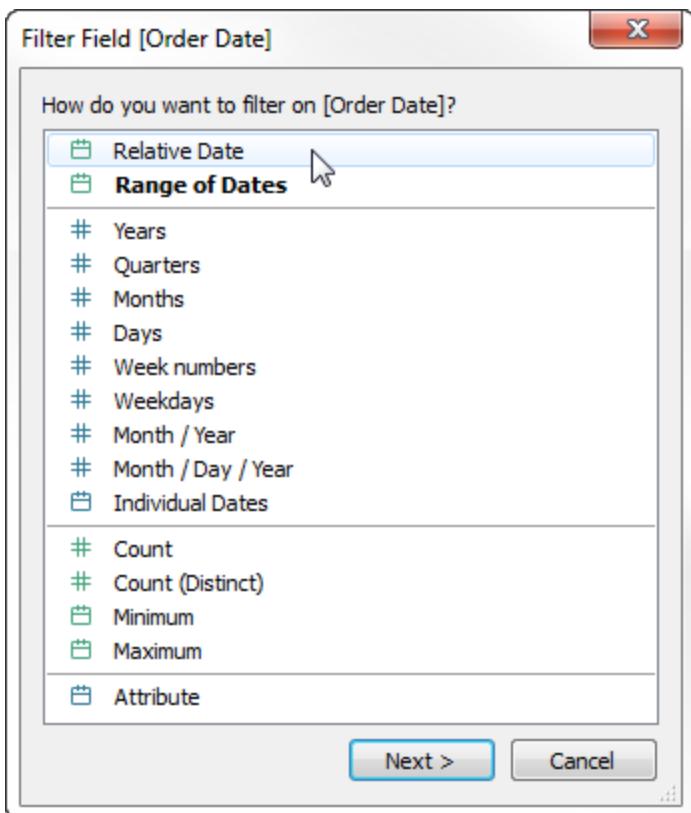


Quick Start: Relative Date Filters

Create filters to show a date period that is relative to when you open the view. For example, create dynamic filters to only show the current week, the year to date, or the past 10 days. Relative date filters make it easier to create lasting views that you can publish and share.

1 Drag a Date Field to the Filter Shelf

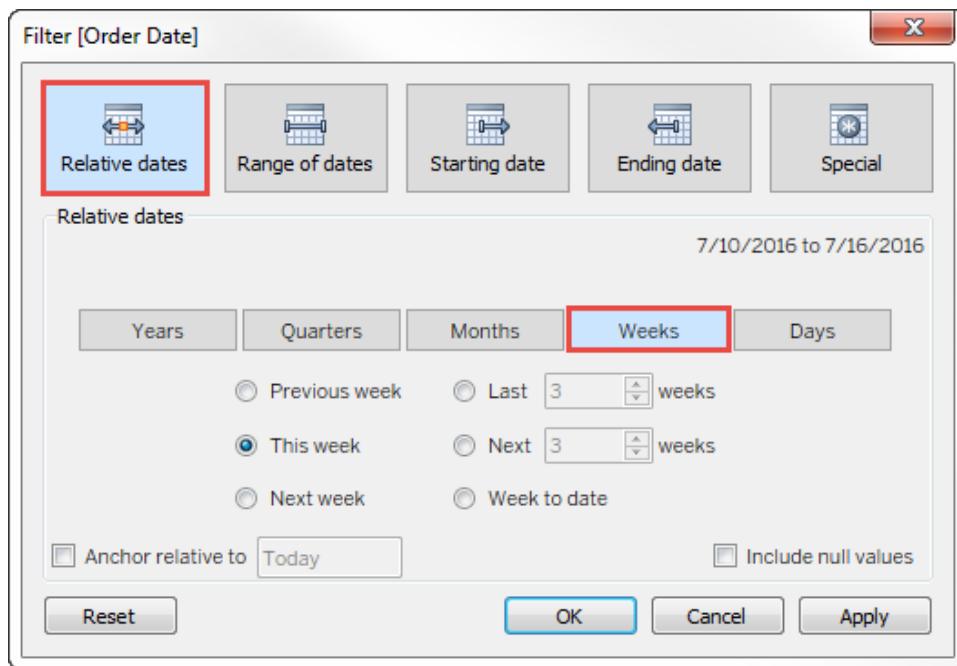
Right-click (control-click on a Mac) and drag a date field from the **Data** window to the Filters shelf, and then click **Relative Date** in the Filter Field dialog box.



Then click **Next**.

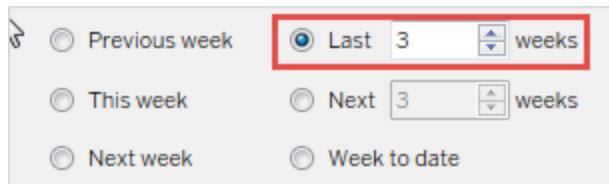
2 Select a Time Unit

In the Filter dialog box, click **Relative dates** and then select the unit of time you want to base your filter on. For example, to filter to only show the three most recent weeks, select **Weeks**.

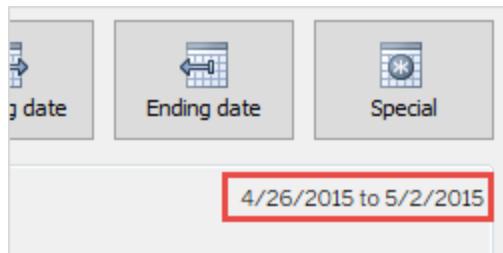


3 Define the Date Period

Use the options in the lower part of the Filter dialog box to specify which weeks to include in the view. To show the last three weeks click **Last** and then select the number **3**.



Date periods include the current unit of time. For example, if you select the last three weeks Tableau will include the current week and the two previous weeks. The range of time that you have selected is displayed in the upper right of the Filter dialog box.

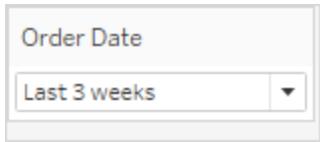


4 Watch the View Update

The filter will now update to always show the three weeks preceding the current date.



Relative date filters, once created, can be displayed in the view as filter cards.

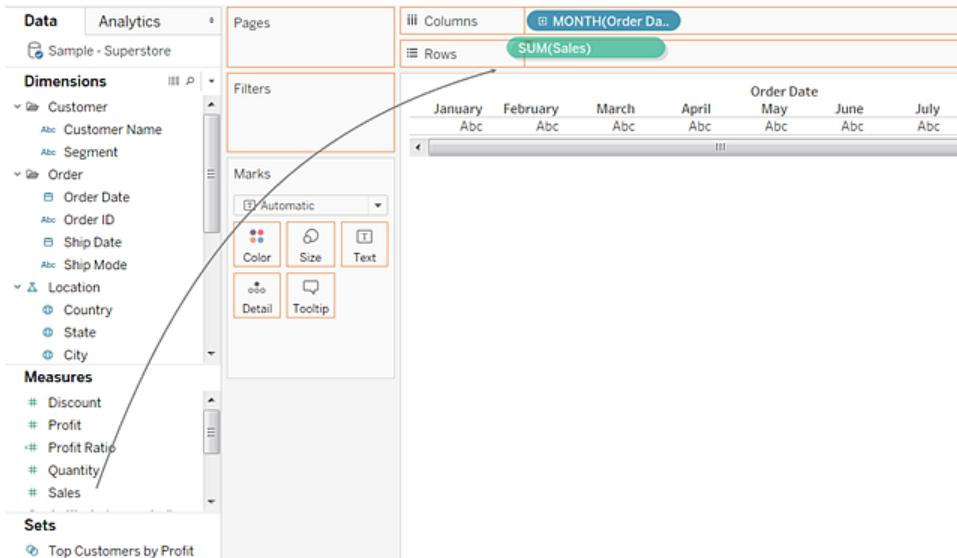


Quick Start: Secondary Axes

In addition to blending multiple measures on the same axis, you can add a secondary axis, or dual axis to a view to better compare measures of different scales. Follow the steps below to add and remove a secondary axis.

1 Drag a measure to the view

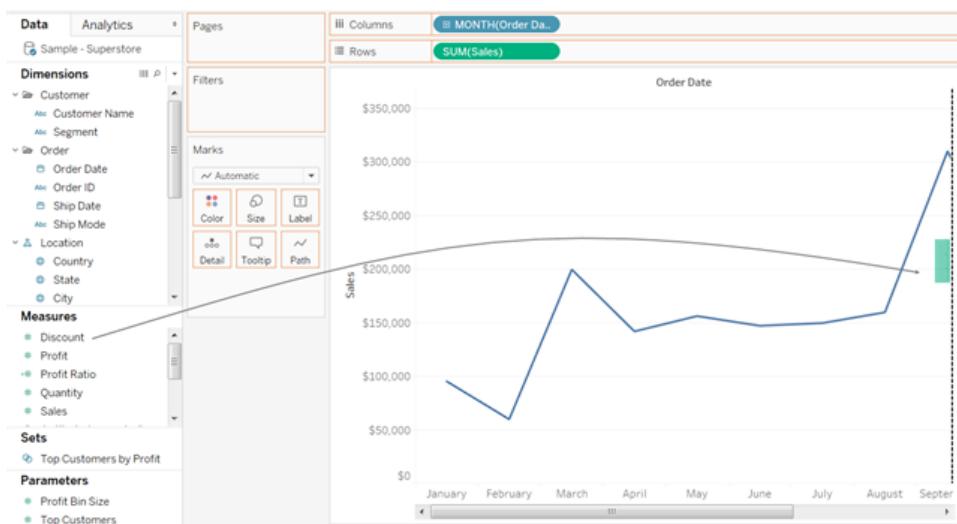
Drag a continuous field from the Data pane to the view.



When you add a measure or continuous field to the view, an axis is created.

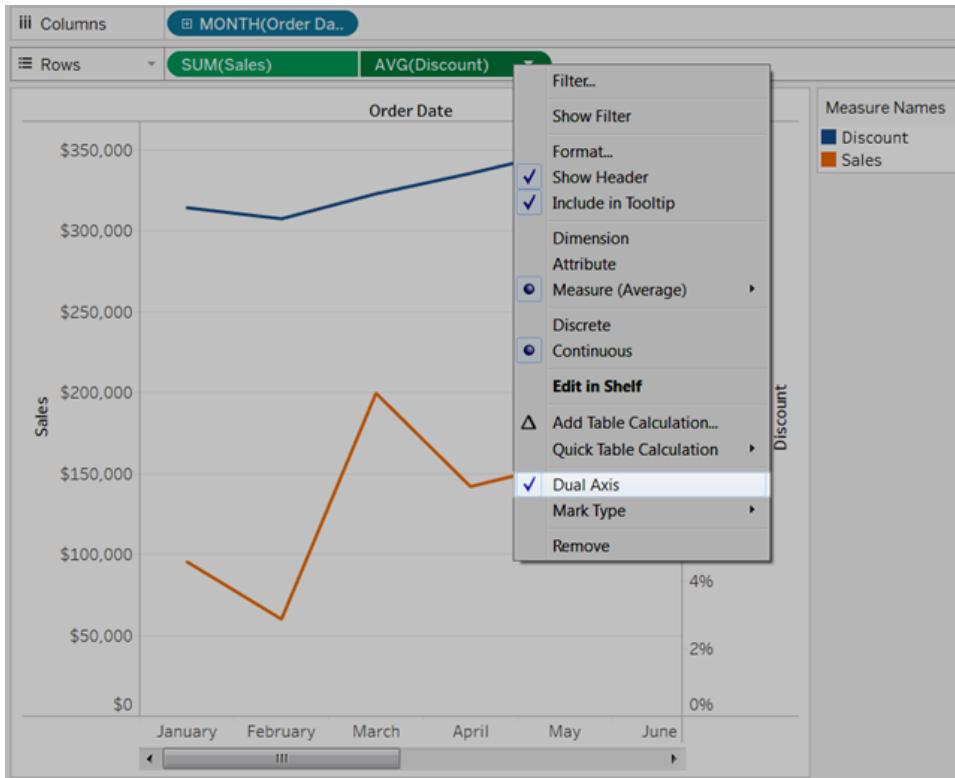
2 Add a second measure to the view

Drag a second measure to the view, but this time drag it to the far right side of the view. A dashed line and a ruler icon displays when you hover over the right side of the view.



3 Remove the secondary axis

To remove a secondary axis at any time, right-click (control-click on Mac) the secondary field on the **Rows** or **Columns** shelf and clear the check mark for **Dual Axis**.



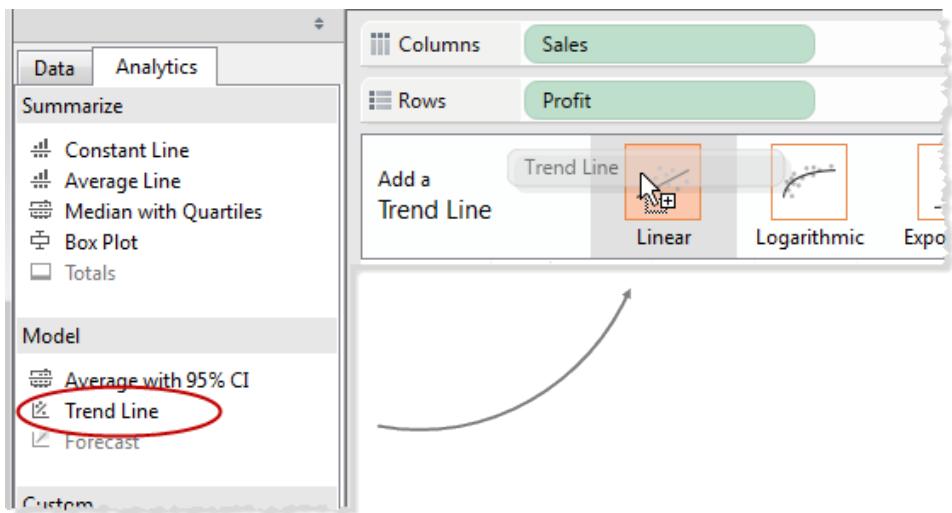
To edit each axis individually, right-click (control-click on Mac) one of the axes and select **Edit Axis**. In the **Edit Axis** dialog box, you can specify the scale, tick marks, title, and more.

Quick Start: Trend Lines

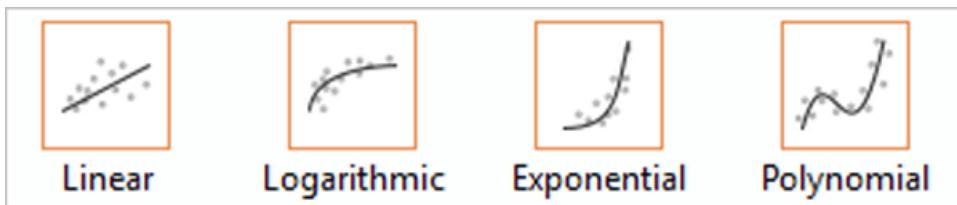
Trend lines, also known as best fit lines, are computed lines that predict data trends. Follow the instructions below to learn how to add trend lines to your view.

1 Add a Trend Line

From the **Analytics** pane, drag the **Trend Line** model to the view and drop it on a model type.



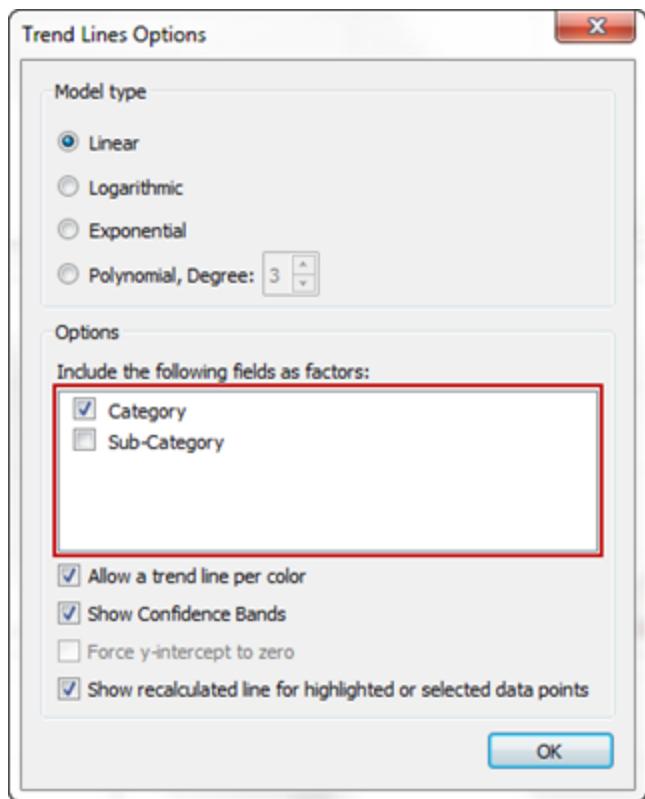
You can select one of the following model types to add to the view.



To remove a trend line, drag the trend line off the view, or select it in the view and click **Remove**.

2 Exclude Fields from the Trend Line

To edit a trend line, click it and select **Edit**. In the **Trend Lines Options** dialog box, if the trend is considering multiple factors, you can clear any dimensions that you don't want to include as factors in the trend line model.



3 Specify Additional Options

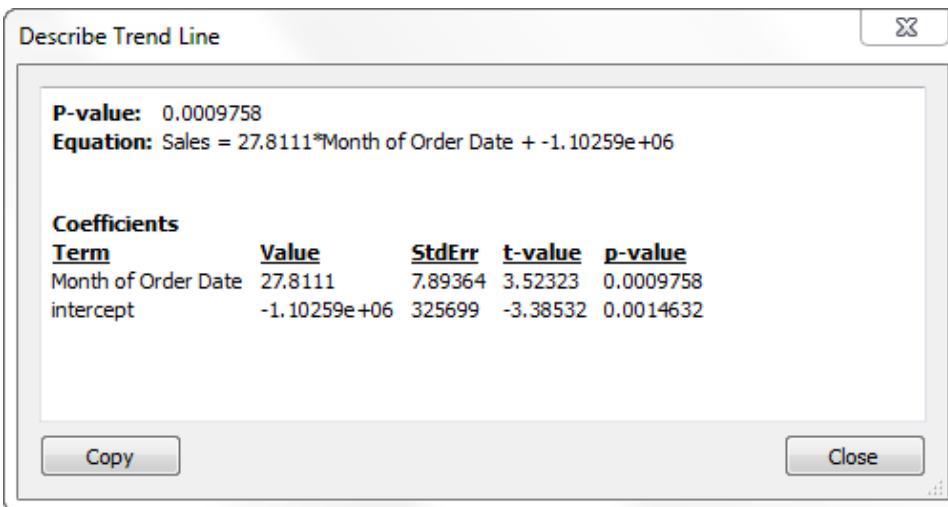
Another option in the **Edit Trend Lines** dialog box is **Allow a trend line per color**. When you have color encoding in your view, you can use this option to add a single trend line that models all of the data and ignores the encoding.



By default, trend lines are shown with upper and lower 95% confidence lines. In a view with multiple trend lines, these confidence lines can make the view "noisy" and difficult to read. If this is the case, clear the **Show Confidence Bands** option.

4 Describe the Trend Line

After you add trend lines, you can use the Describe feature to display statistics on the trend line. For example, you can see the formula as well as r-squared and p values. To view a description of a single trend line, right-click the trend line in the view and select **Describe Trend Line**.



To view a description of the entire trend line model, select **Analysis > Describe Trend Model**.

Quick Start: Turn Off Pan and Zoom in Maps

Sometimes you want to control how your audience interacts with your map view.

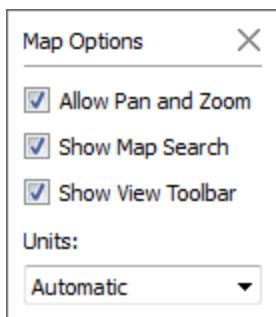
For example, if your audience is using a mobile device to explore your workbook, you might want to turn off pan in your map view so they can move from sheet to sheet with ease.

Similarly, if your map view is zoomed in to a particular city, you might want to ensure that people won't zoom out.

In cases like these, you can turn off pan and zoom in a map view, as well as in a background image.

1 Clear the Allow Pan and Zoom Option

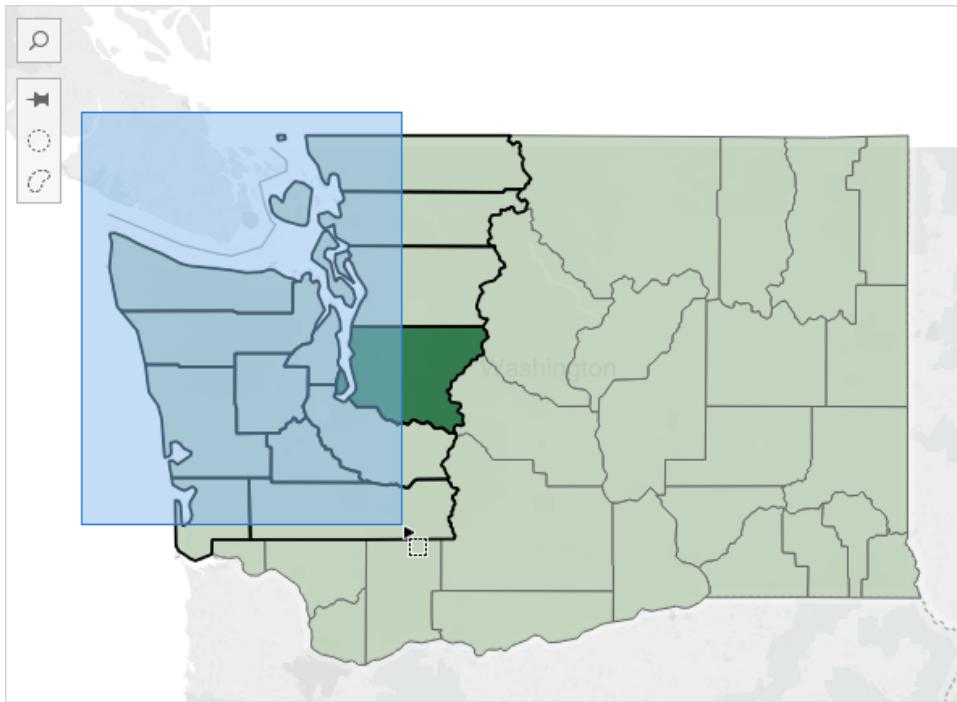
To turn off pan and zoom in your map view or background image, select **Map > Map Options**, and then clear **Allow Pan and Zoom**.



In a dashboard, select the map view first, and then select **Map > Map Options > Allow Pan and Zoom** to clear the check mark.

2 Review the changes to the view toolbar

When you turn off pan and zoom in a map view or background image, the pan tool and all zoom controls are removed from the view toolbar. Keyboard shortcuts for zooming in and out of the view, or panning, no longer work. For more information about the view toolbar and default tools, see [View Toolbar on page 565](#) and [Select Marks on page 566](#).



You can still use the view toolbar to select marks in the view, lock the map in place, or automatically zoom the map to all of your data. You can also click the map search icon to search for locations in the view.

Quick Start: Use Mapbox Maps

You can use Mapbox maps to create map views in Tableau.

For example, if you have a Mapbox map that fits your data or corporate style, you can connect to it in Tableau Desktop and use it as a background map.

You can add a Mapbox GL map or a classic Mapbox map. For this example, learn how to add a classic Mapbox map. For information on how to add a Mapbox GL map, see [Use Mapbox Maps on page 788](#).

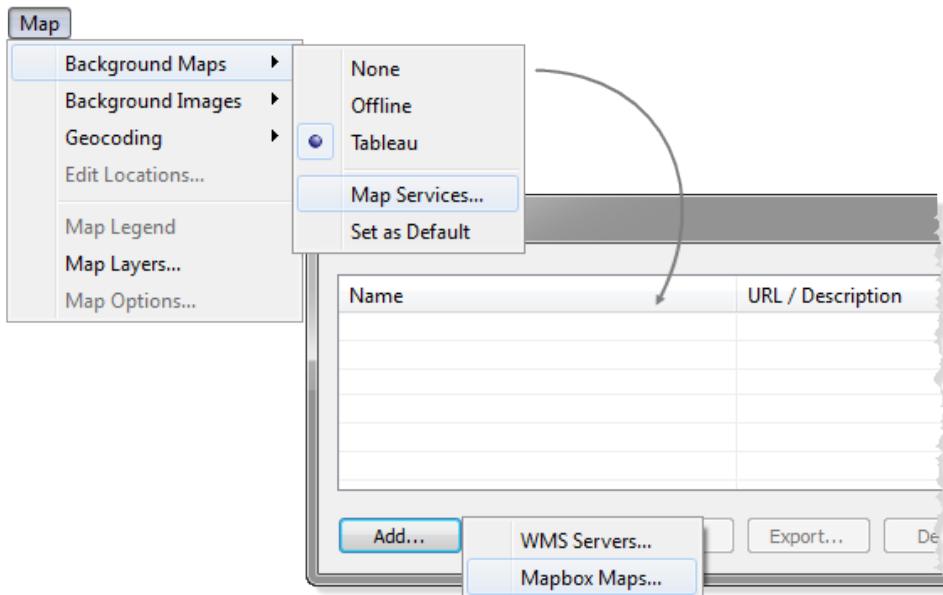
When you use a Mapbox map in one of your views, you can publish that view to Tableau Server, Tableau Online, or Tableau Public so your audience can view your data and your Mapbox map without needing to have a Mapbox account.

Connect to your Mapbox map

1 Open the Add Mapbox Map dialog box

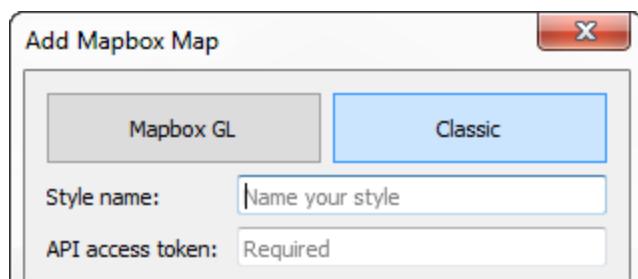
Click **Map > Background Maps > Maps Services**. This opens the Map Services dialog box.

In the Map Services dialog box, click **Add > Mapbox Maps**. This opens the Add Mapbox Map dialog box.



2 Enter a style name and API access token

In the Add Mapbox Map dialog box, click **Classic**. Give the map a style name, and then enter the API access token for the classic Mapbox map you want to add.



3 Add custom layers or choose a preset style

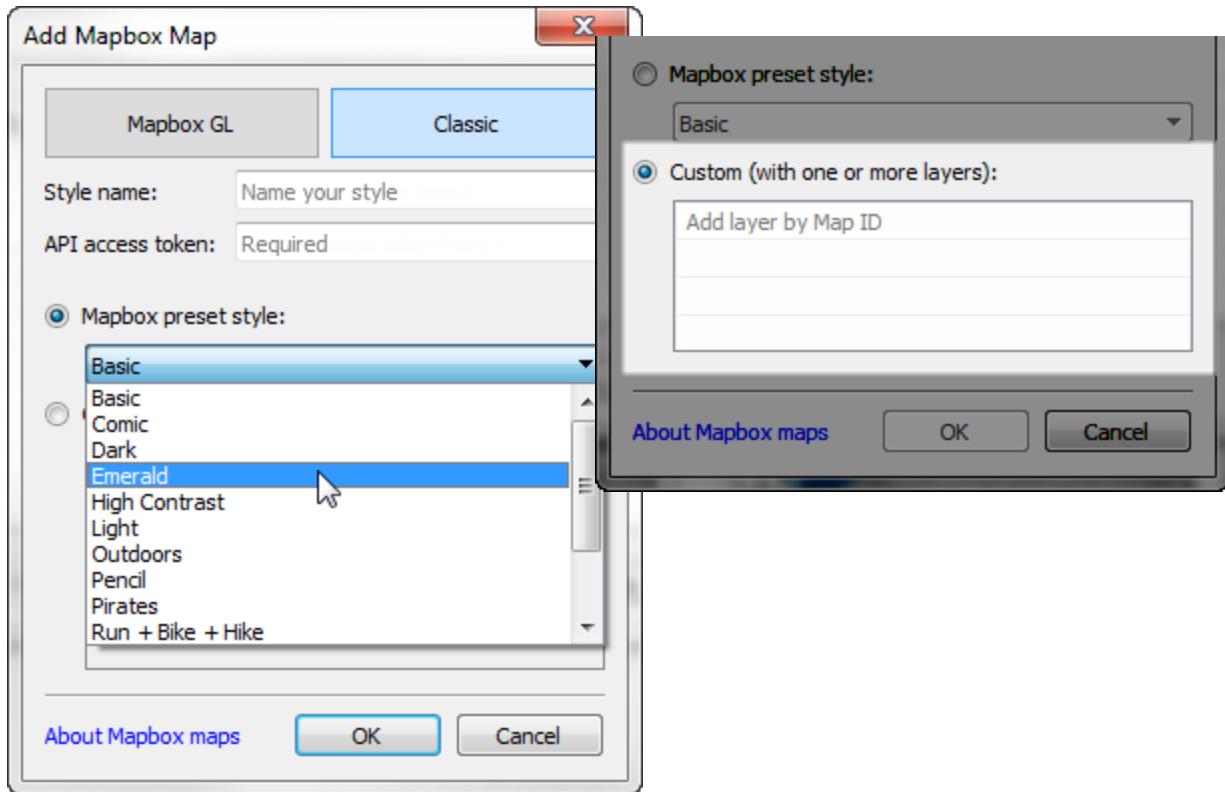
You can choose to add one or more map layers, or you can choose to use a Mapbox preset style.

To use a Mapbox preset style:

In the Add Mapbox Map dialog box, click **Mapbox preset style**, and then select a preset style from the drop-down menu.

To add one or more map layers:

In the Add Mapbox Map dialog box, click **Custom**, and then enter one or more map IDs. For more information about map IDs, see the [Mapbox API help](#) on the Mapbox website.



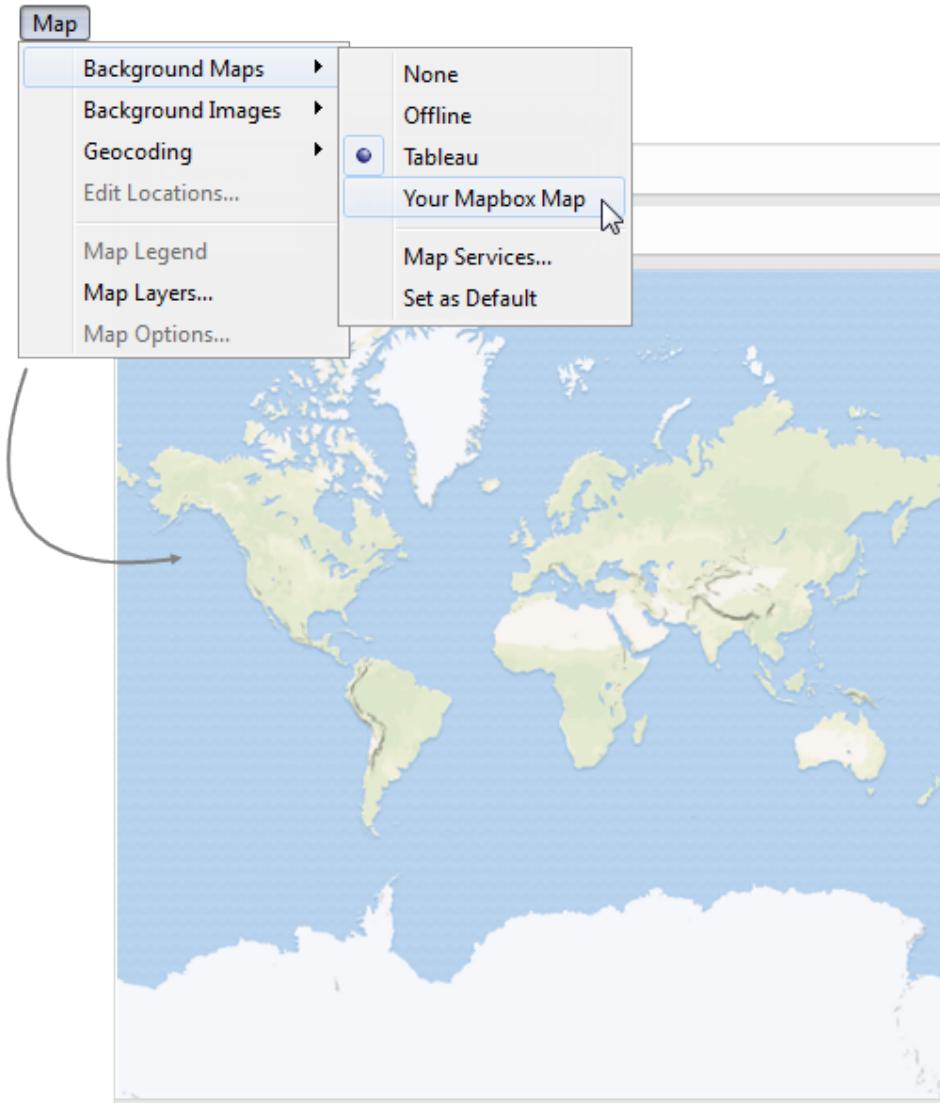
When finished, click **Okay** to exit the Add Mapbox Map dialog box, and then click **Close** to return to the view.

You can add as many Mapbox maps as you want to a workbook. Each Mapbox map you add appears as a background map in the **Background Maps** menu.

Use your Mapbox map

1 Build the map view

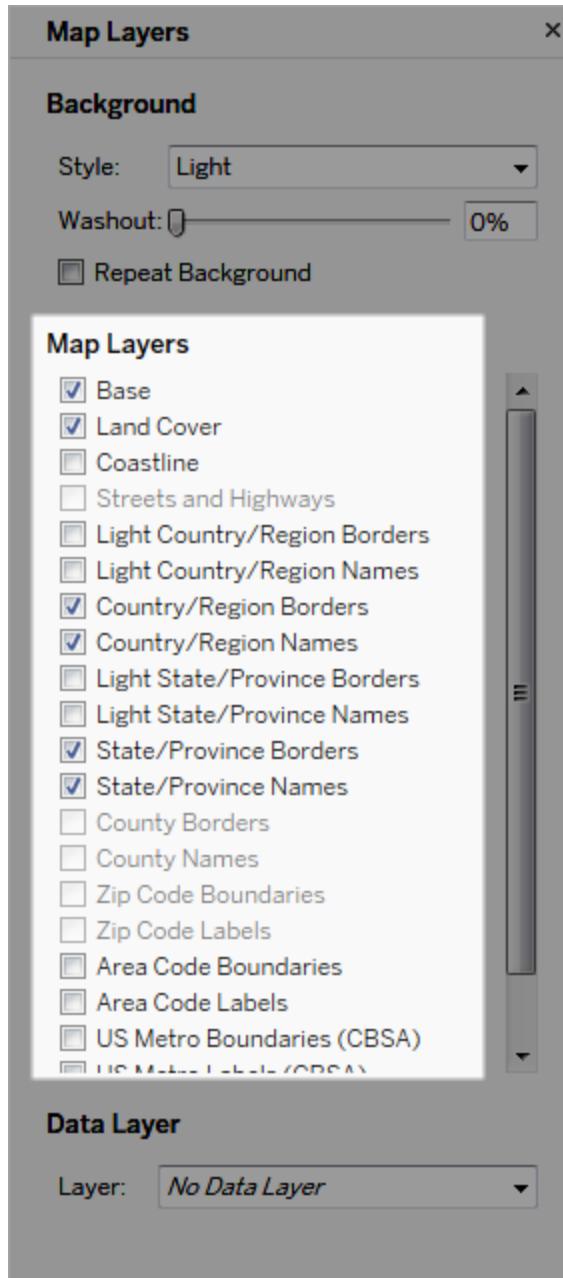
Select **Map > Background Maps**, select the Mapbox map you added from the list, and then build a map view. For more information about how to build map views, see [Build a Map View on page 744](#).



2 Select which map layers appear in the view

Once you are connected to your Mapbox map, you might want to add or subtract map layers from the view.

To toggle between custom Mapbox layers, click **Map > Map Layers**, and then under the **Map Layers** section, select the layers you want to appear in the view.



By default, all map layers appear in the view when you first add a Mapbox map to your workbook.

Save your Mapbox map as a Tableau Map Source

Q-

After you add a Mapbox map to your workbook, it will be saved with the workbook and available to anyone with whom you share the workbook. You can also save a Mapbox map as a Tableau Map Source (.tms) that you can share with others so they can quickly connect to it and use it in their own workbooks.

1. Select **Map > Background Maps > Map Services**.
2. In the Map Services dialog box, select the Mapbox map you want to save, and then click **Export**.
3. In the Export Connection dialog box, enter a name for the file, choose a location for the file, and then click **Save**.

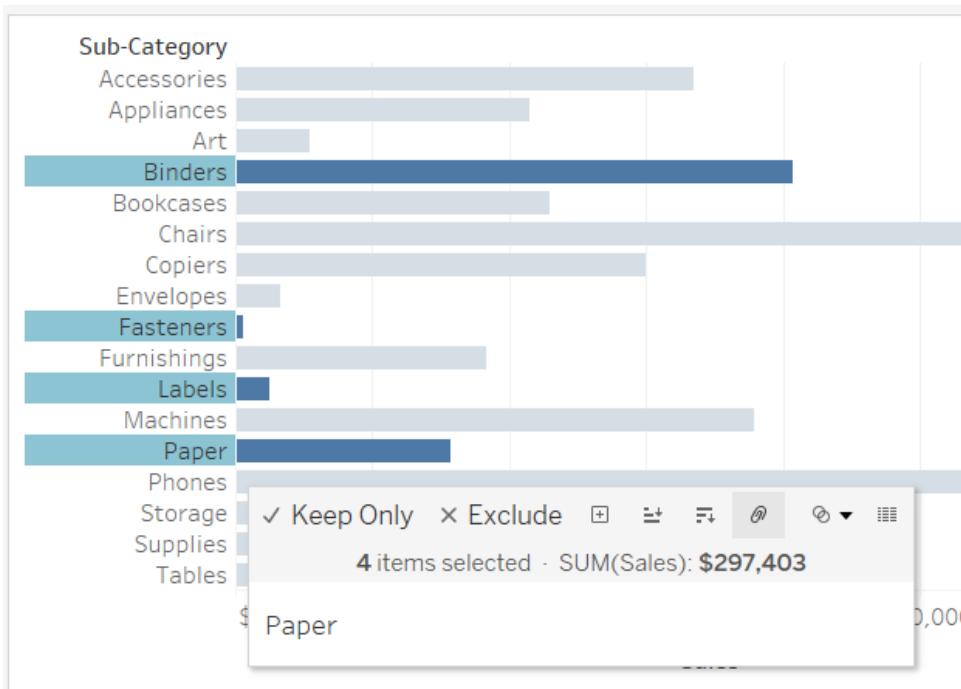
For more information, see [Save a Map Source](#) on page 799.

Quick Start: Visually Grouping Data

Groups combine dimension members into categories and enable you to analyze data as a group instead of as individual marks. For example, if you are looking at retail data, you might have a list of all your products. You can use groups to combine several of your small products into a group to analyze them as one mark, like combining binders, fasteners, labels and paper to create a group called organizational supplies. There are many ways to create groups in Tableau. Here are three options.

1 Select headers

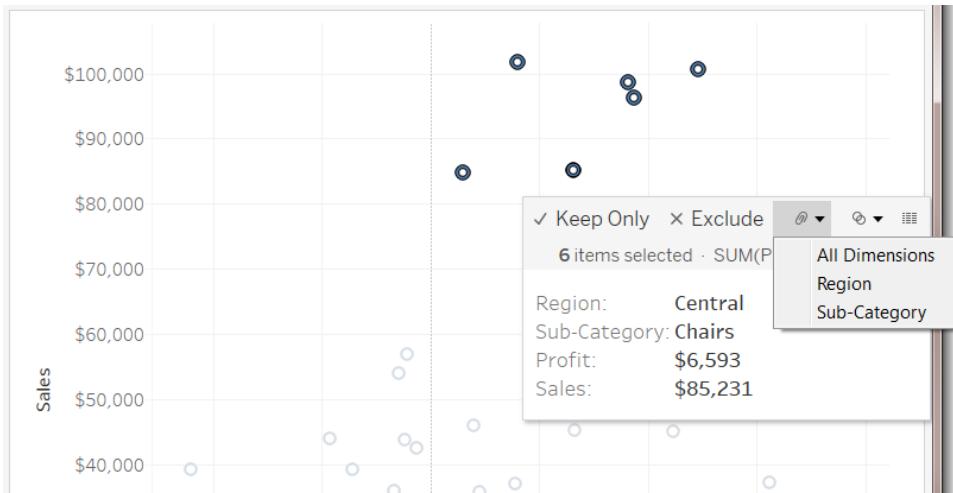
One way to create a group is by selecting multiple headers in a table or values in a legend and then clicking the Group option (which looks like a paperclip) in the tooltip. This method is useful for fixing simple errors in your data or answering “what if.?” questions.



The group's name is concatenated based on the members in the group. To rename the group, right-click it in the view (control-click on a Mac), and select **Edit Alias**. The group is also added to the Data pane as a separate field..

2 Select marks in the view - all dimensions

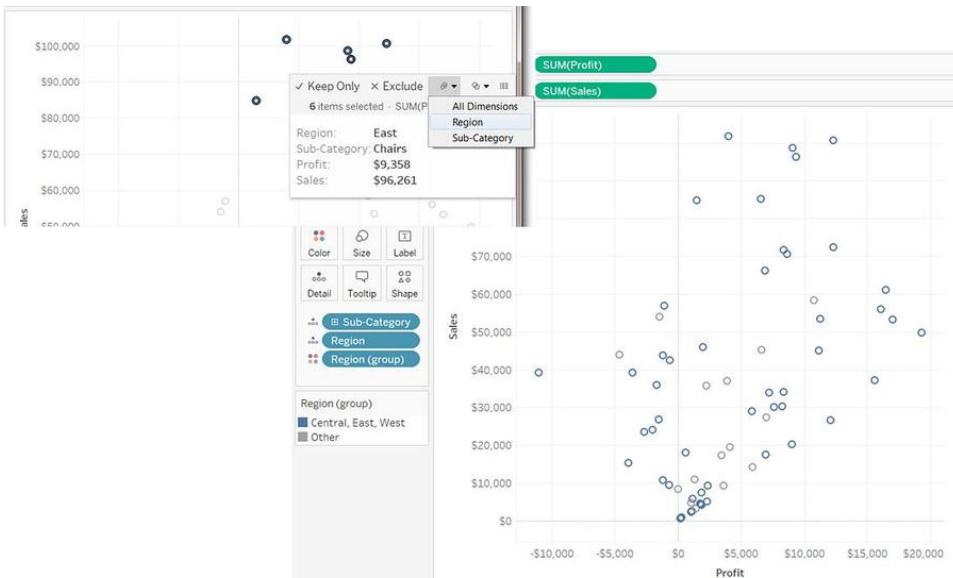
If you want to visually call out a set of marks in your view instead of combining them into a new mark you can create groups by selecting marks directly in the view and then select Group on the tooltip. A new group field is created automatically and placed on the color shelf. The group combines and aggregates the selected marks into one category and assigns all other values to an Other category. If the selected marks represents multiple dimensions (for example, Sub-Category and Region), you can either group on all dimensions or on a specific dimension.



Grouping on all dimensions creates a category that includes each relevant combination of Sub-Category and Region. These combinations are grouped into a single category and everything else is put in the Other category.

3 Select marks in the view - specified dimensions

Grouping selected marks on a specific dimension and then selecting a specific dimension from the Group menu on the tooltip creates a group based on that dimension. In this example, the specified dimension is Region and the selected marks contain product values including the Central, East and West regions. All qualifying marks are grouped into a category and all non-qualifying marks are assigned to the Other category.



The marks assigned to the group don't all need to be included in the selection you used when you created the group. This is because the group is based on values for the dimension, not the marks you selected.

Quick Starts about Calculations

After you master the basics of building views in Tableau, you can modify the fields that Tableau extracts from your data source to dig deeper with your analysis. You can create new fields that don't exist in your data source to explore new dimensions and find facts.

Use the calculation editor for customizing and creating fields, or, to stay in the flow and change course quickly, create ad hoc calculations that you can easily save and reuse.

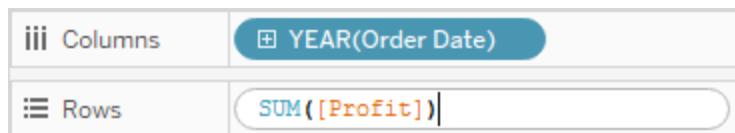
The following Quick Start topics are designed to get you started using calculations.

Quick Start: Ad-Hoc Calculations

Ad-hoc calculations are calculations that you can create and update as you work with a field on a shelf in the view. Ad-hoc calculations, also known as type-in or in-line calculations, can be useful for testing a hunch, trying a what-if scenario, or debugging a complex calculation.

1 Double-Click to Start Editing an Existing Field

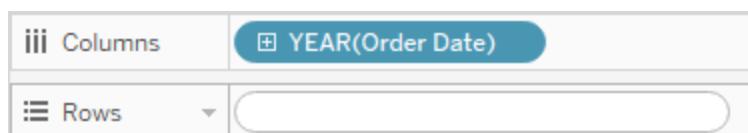
Double-click on an existing field in the view to start editing.



Ad-hoc calculations are supported for fields on the **Rows**, **Columns**, **Marks**, and **Measure Values** shelves; they are not supported for fields on the **Filter** or **Pages** shelves.

2 Double-Click to Create a New Calculation

Alternatively, you can double-click on an empty shelf or on an empty part of a shelf to create a new ad-hoc calculation.



3 Type or Drag Content to the Calculation

Type to update the expression, or drag new fields into the expression from the **Data** pane or elsewhere in the view.

4 Commit the Calculation to Update the View

When you're satisfied with the expression, press Enter or Tab, or click outside the expression to commit the expression and update the view.

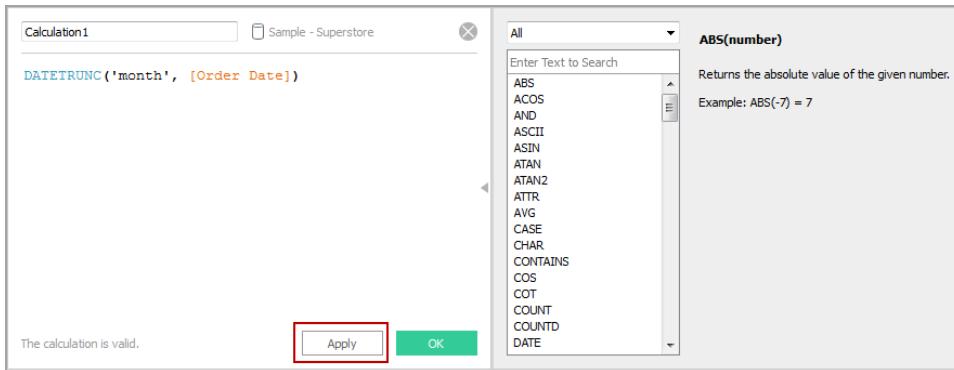
Ad-hoc calculations are not named, but are saved when you close the workbook. If you want to save an ad-hoc calculation for use in other workbook sheets, copy it to the **Data** pane. Tableau will prompt you to name it.

Quick Start: New and Improved Calculation Editor

The calculation editor in Tableau Desktop has been redesigned to provide interactive editing, intelligent formula-completion, and drag-and-drop support. The editor is now also available when you're editing a view in Tableau Server or Tableau Online.

1 Interactive Editing

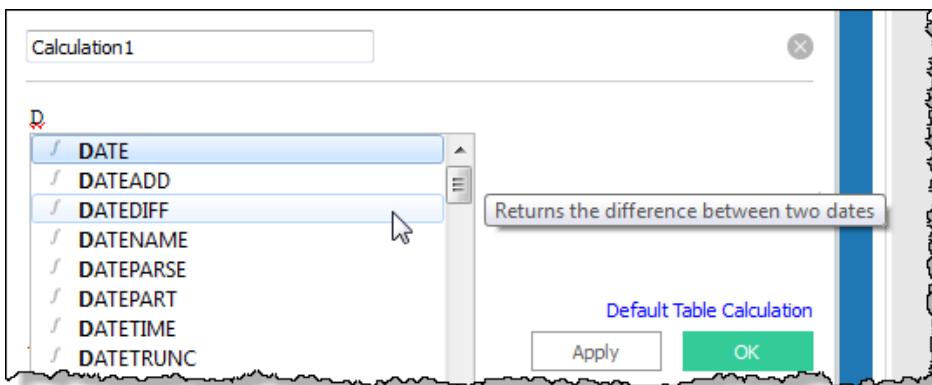
To open the calculation editor, click the drop down to the right of Dimensions on the **Data** pane and choose **Create Calculated Field**. The calculation editor is now non-modal, which means you can work in your view and work with a calculation at the same time. If you edit a field that's in the view and then click **Apply**, the view changes immediately and the editor remains open.



2 Auto-Completion for Formulas

As you work on a formula in the calculation editor, Tableau displays a list of options for completing the formula.

As you scroll the list, using mouse or keyboard, Tableau shows a short description when the current item is a function:

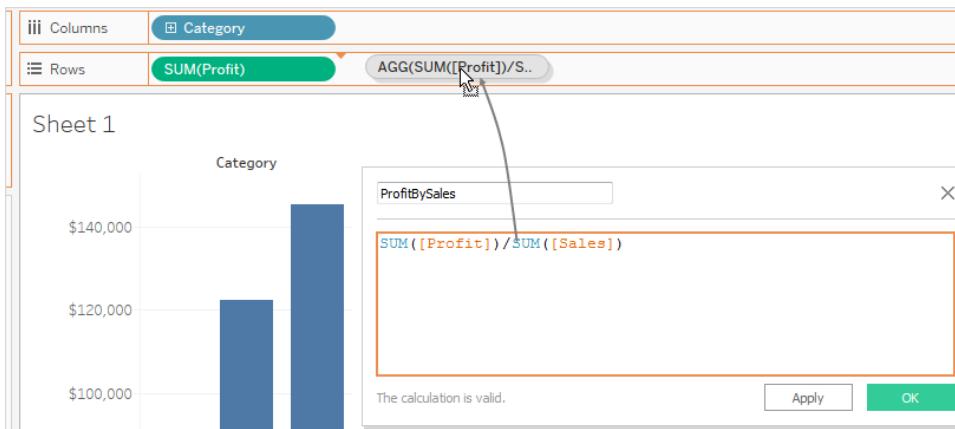


Click a keyword in the list or press Enter to select it. If the keyword is a function, Tableau displays syntax information so you know how to proceed:



3 Drag and Drop Between Editor and View

You can drag fields from the **Data** pane or from any of the shelves in the Tableau workspace into the calculation editor, or drag part or all of a calculation from the editor to the **Data** pane or to a shelf.



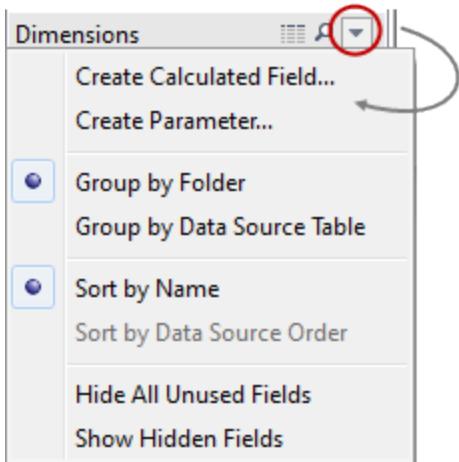
Quick Start: Creating Calculated Fields - Cubes

Tableau supports calculated fields on multidimensional data sources. There are two kinds of calculated fields: calculated measures and calculated dimensions. When you define a new calculated field, you create a new field in your data source based on existing measures in the data source as well as standard functions and operators. The steps below explain how to create calculated measures using Tableau formulas.

Tableau support multidimensional data sources only in Windows.

1 Open the Calculated Field Dialog Box

Click **Analysis > Create Calculated Field**.



2 Define a Formula

In the **Calculated Field** dialog box, give the new field a name and define a formula.



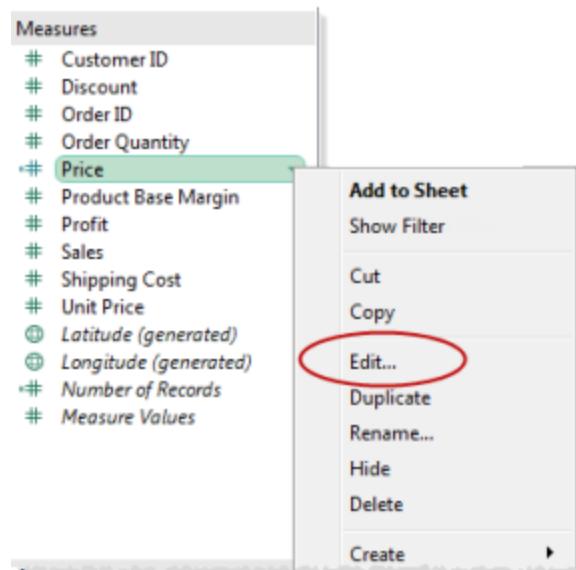
3 Use the New Field

Check that the formula is valid, and then click **OK** to save the new field. It will save to **Measures** on the **Data** pane.

Measures
Customer ID
Discount
Order ID
Order Quantity
=# Price
Product Base Margin
Profit
Sales
Shipping Cost
Unit Price
(# Latitude (generated))
(# Longitude (generated))
=# Number of Records
Measure Values

4 Edit the Formula

If you need to change the formula for a calculated field, right-click (control-click on the Mac) the field in the data pane and select **Edit**. Keep in mind that the changes you make will affect any worksheets that use the new field.



Quick Start: Creating Calculated Members with MDX

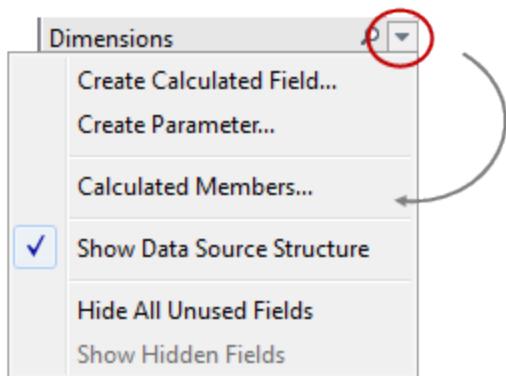
Rather than use Tableau formulas to create new fields, you can also create calculated members using Multidimensional Expressions (MDX). With MDX you can create more complex calculations and reference both existing measures and dimensions. Calculated members can

either be new measures or new dimension members depending on the hierarchy that you choose. Follow the steps below to learn how to create calculated members using MDX.

Tableau support multidimensional data sources only in Windows.

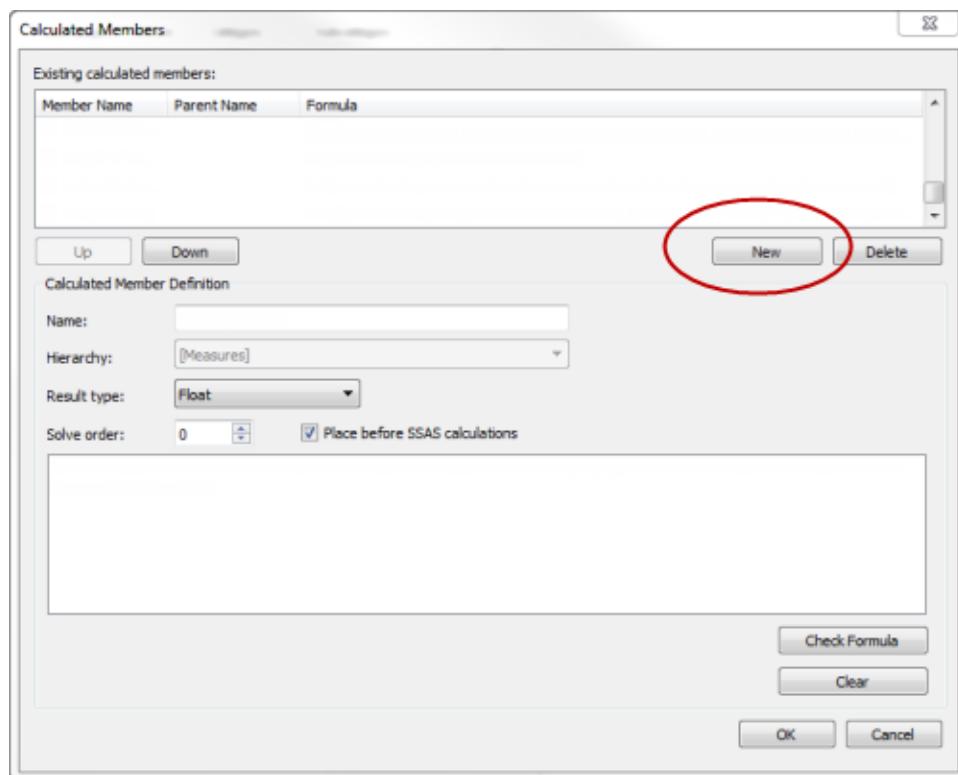
1 Open the Calculated Members Dialog Box

Click the Dimensions drop-down menu and select **Calculated Members**.



2 Create a New Calculated Member

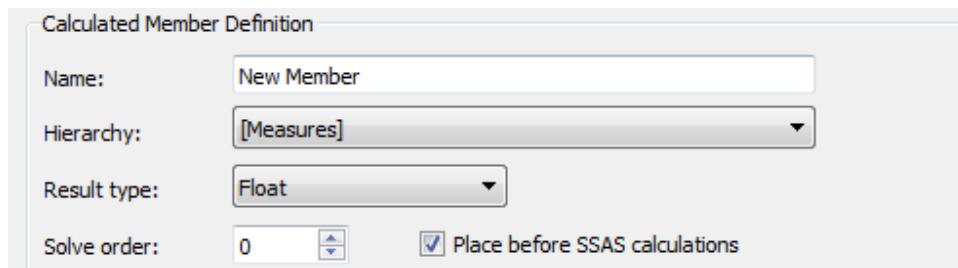
In the Calculated Member dialog box, click **New**.



3 Specify a Calculated Member Definition.

Name the new member and select its location in the hierarchy. For calculated dimension members, specify a parent member. For calculated measure members, specify a result type.

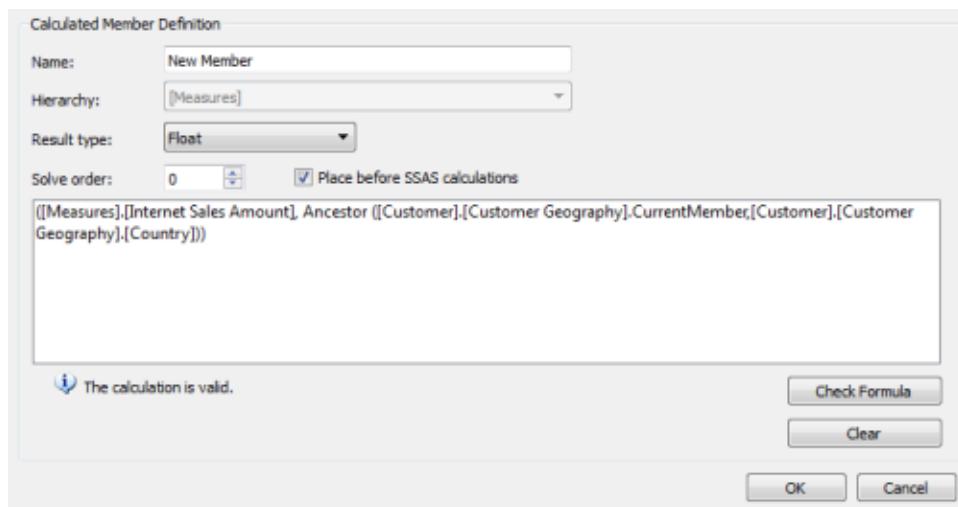
Finally select a solve order for the member.



Note: Sometimes a single cell in your data source can be defined by two different formulas. The solve order defines the precedence given to each formula. Formulas with a lower solve order are solved first. The default solve order is zero.

4 Define a Formula

Type a formula that defines the new member and click **Check Formula** to validate. When finished, click **OK**. The new member displays in the Data pane under the parent member and hierarchy you specified.



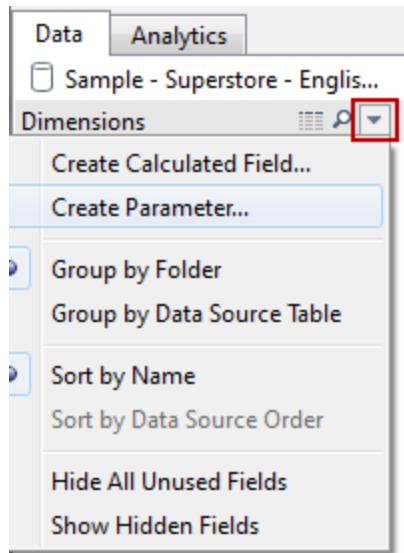
Quick Start: Parameters

Parameters are dynamic values that can replace constant values in calculations. For example, you can create a calculated field that returns true if Sales is greater than \$500,000 and

otherwise return false. Then you can replace the constant value of “500,000” in the formula with a parameter that users can change dynamically using a parameter control.

1 Create a Parameter

Click the drop down to the right of Dimensions on the **Data** pane and choose **Create Parameter**



2 Define the Parameter

Specify the following properties to define the parameter:

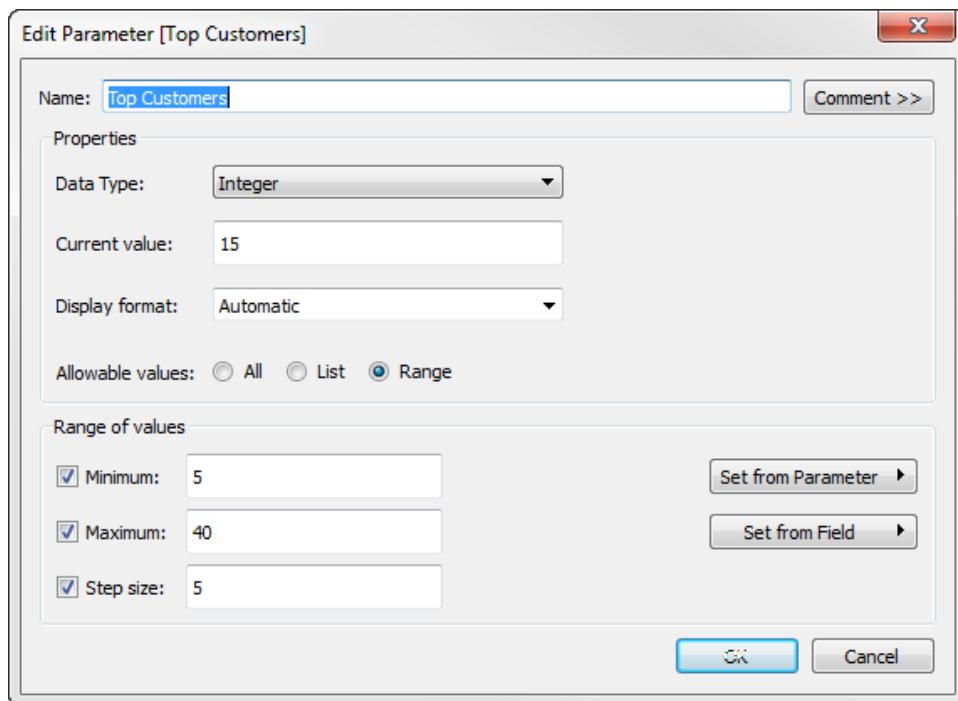
Name - A descriptive name for the parameter.

Data type - Parameters can be integers, floating point numbers, strings, Boolean values, dates, or date/times.

Current Value - An initial value for the parameter.

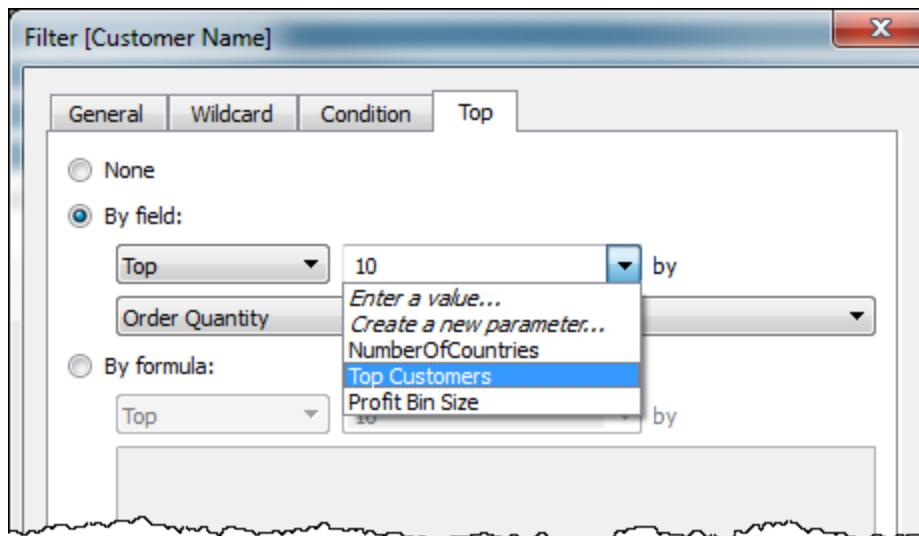
Display format - How to format the values.

Allowable values - The type of control to be used for selecting values. Parameters can be defined a text field (All), a List, or a Range of defined values.



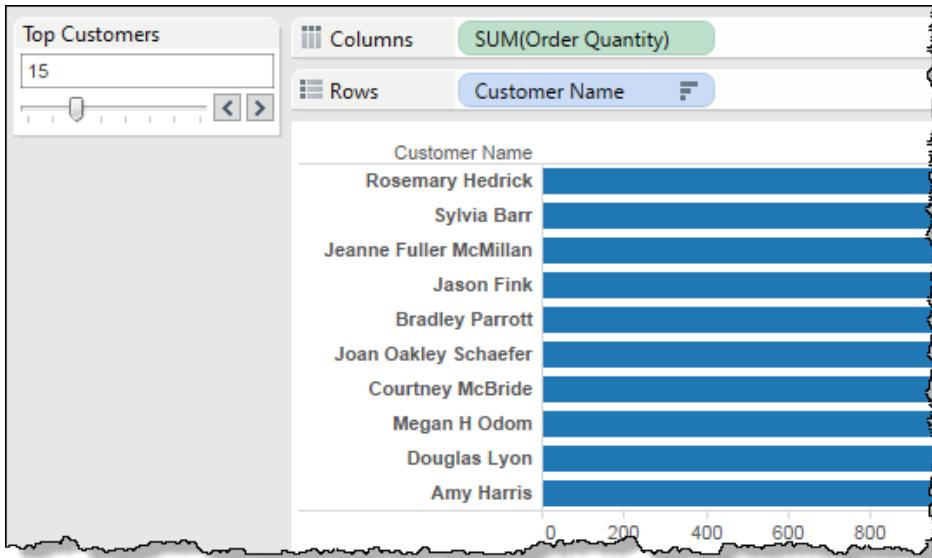
3 Use the Parameter in a Calculation

Replace the constant value in a calculated field or filter with the parameter. You can double-click the parameter in the list to add it to the formula.



4 Show the Parameter Control

Parameters are global across the entire workbook and are shown at the bottom of the **Data** pane. Right-click a parameter and select **Show Parameter Control** to display a card where users can modify the parameter value.



You can move a parameter control to different locations in the view.

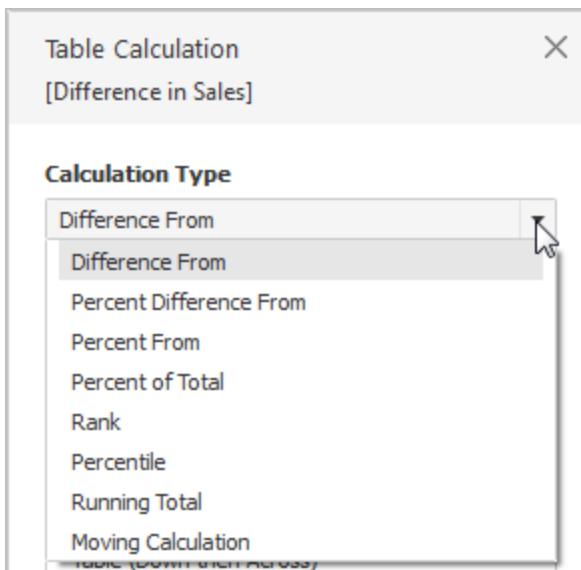
Q-

Quick Start: Table Calculations

Table Calculations are computations that are applied to the values in the entire table and are often dependent on the table structure itself. For example, in a sales environment, you can use table calculations to compute the running total of sales across a specified date range or to compute each product's contribution to the total sales in a quarter.

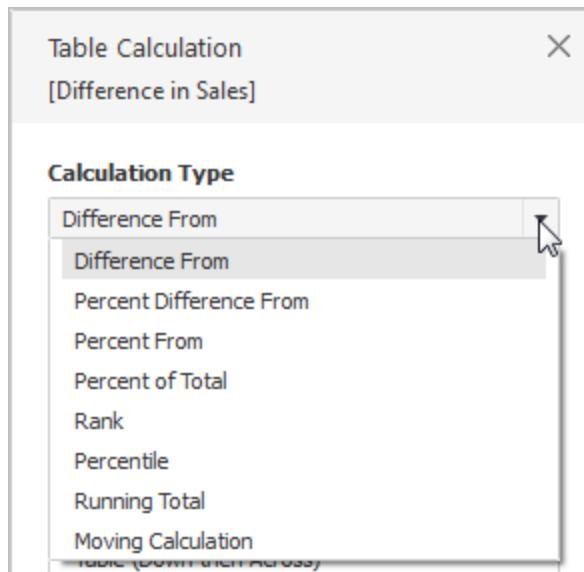
1 Open the Table Calculation Dialog Box

Right-click a measure in the view (control-click on a Mac) and select **Add Table Calculation**.



2 Choose a Calculation Type

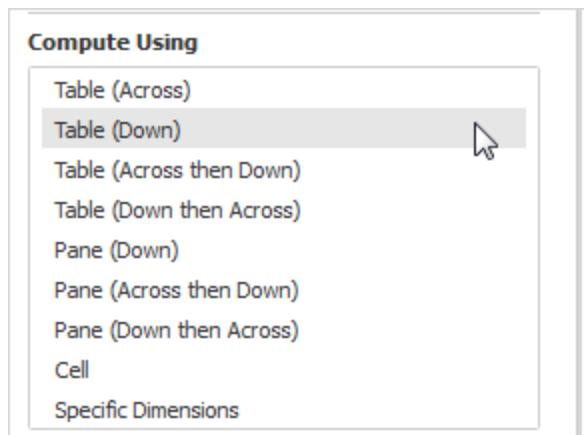
In the Table Calculation dialog box, choose the type of calculation you want to apply.



- **Difference From** - show absolute change.
- **Percent Difference From** - show rate of change.
- **Percent From** - show as % of other specified value.
- **Percent of Total** - show values as % of the total.
- **Rank** - rank values numerically
- **Percentile** - compute percentile values
- **Running Total** - show a cumulative total.
- **Moving Calculation** - smooth short fluctuations to identify long term trends.

3 Choose one of the Compute Using options

In the bottom half of the Table Calculation dialog box, define the calculation using the drop-down lists. The options vary for different types of calculations.



4 Check your view and decide

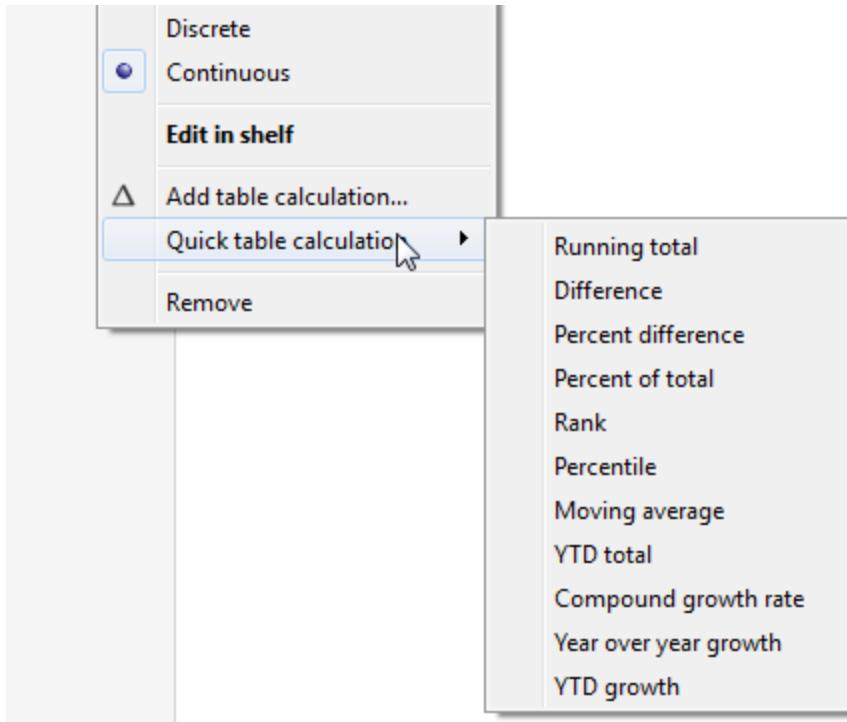
As you experiment with different calculation types and **Compute Using** options, highlighting in the view shows the effect of your choices. Continue to experiment and try different options until the result is the one you want.

The screenshot shows a 'Compute Using' dropdown menu on the left side of the screen. The menu items are: Table (Across), Table (Down) (which is selected and highlighted in yellow), Table (Across then Down), Table (Down then Across), Pane (Down), Pane (Across then Down), Pane (Down then Across), Cell, and Specific Dimensions. A black arrow points from the 'Table (Down)' option in the menu to the '2011' column header in a table on the right.

Quarter of O..	Month of Or..	Order Date			
		2011	2012	2013	2014
Q1	January				
	February	-\$9,138	\$5,963	\$4,325	\$24,420
	March	\$50,880	\$26,256	\$28,319	\$33,625
Q2	April	-\$27,396	-\$4,272	-\$11,938	-\$13,797
	May	-\$4,647	-\$4,064	\$17,442	\$5,539
	June	\$10,947	-\$5,334	-\$17,261	\$2,609
Q3	July	-\$649	\$3,968	-\$990	\$169
	August	-\$6,037	\$8,133	-\$5,175	\$13,088
	September	\$53,868	\$27,698	\$39,643	\$28,973
Q4	October	-\$50,324	-\$33,191	-\$16,445	-\$12,695
	November	\$47,175	\$44,568	\$25,729	\$34,533
	December	-\$9,083	-\$1,053	\$15,045	-\$21,852

The "Quick" Alternative

Tableau offers common pre-configured table calculations that you can apply quickly without having to go through the steps for defining the calculation. If the result meets your needs, you're done. Right-click a measure in the view, choose **Quick table calculation**, and pick a calculation type. If not, open the calculation in the Table Calculations dialog box.



Quick Start: Table Calculations Enhancements

Table calculations got a lot easier in Tableau 10. Here's how:

1 Instant updates as you configure a calculation

Click a measure in the view and choose **Add Table Calculation** to open the Table Calculation dialog box. While you work in the redesigned dialog box, you can see how your **Calculation Type** and **Compute Using** selections affect the view. Continue to experiment and try different options until you see the result that you want.

The screenshot shows a data table with columns for Quarter, Month, and years 2011 through 2014. A 'Table Calculation' dialog box is overlaid on the right, with a red arrow pointing to the 'Table Across' option under 'Compute Using'.

Quarter of O..	Month of Or..	2011	2012	2013	2014
Q1	January	2.88%	3.86%	3.05%	6.09%
	February	0.99%	2.60%	3.76%	2.76%
	March	11.50%	8.18%	8.41%	7.35%
Q2	April	5.84%	7.27%	6.45%	5.47%
	May	4.88%	6.40%	6.32%	6.22%
	June	7.14%	5.23%	5.23%	5.23%
Q3	July	7.01%	6.11%	6.11%	6.11%
	August	5.76%	7.84%	7.84%	7.84%
	September	16.89%	13.73%	13.73%	13.73%
Q4	October	6.50%	6.67%	6.67%	6.67%
	November	16.24%	16.15%	16.15%	16.15%
	December	14.36%	15.92%	15.92%	15.92%

Table Calculation

[Total Sales]

Calculation Type

Percent of Total

At the level

Compute total across all pages

Compute Using

Table (Across) **Table (Down)**

Table

Pane (Down)

Pane

2 Highlighting

What is the difference between **Table (Across)** and **Pane (Across)**? Or between **Pane (Across then Down)** and **Pane (Down then Across)**? To make it easy to see what these options mean, Tableau now highlights part of your view while you work in the Table Calculations dialog box to show exactly how the current **Compute Using** option defines the scope of your calculation.

Order Date						
Quarter of O..	Month of Or..	2011	2012	2013	2014	
Q1	January	18.73%	26.40%	20.03%	37.60%	
	February	6.46%	17.74%	24.70%	17.06%	
	March	74.81%	55.87%	55.28%	45.34%	
Q2	April	32.70%	38.37%	28.99%	29.93%	
	May	27.33%	33.81%	41.88%	34.06%	
	June	39.98%	27.82%	29.13%	36.01%	
Q3	July	23.63%	22.08%	26.58%	24.16%	
	August	19.43%	28.33%	23.00%	30.69%	
	September	56.93%	49.59%	50.42%	45.15%	
Q4	October	17.51%	17.23%	23.94%	27.72%	
	November	43.77%	41.68%	34.84%	40.03%	
	December	38.72%	41.10%	41.22%	32.24%	

3 With good visual feedback, you can get the result you want

Table calculations are powerful, but often you don't know exactly what you're aiming for until you can see it in front of you. Now table calculations can be part of your creative "flow" in Tableau.

The screenshot shows the Table Calculation dialog box with the formula [% of Total Sales]. The Calculation Type dropdown is set to 'Percent of Total'. The Compute Using dropdown is set to 'Table (Down)'. To the right, a preview of a table titled 'Order Date' is shown, which contains four columns: Quarter of O., Month of Or., 2011, 2012, 2013, and 2014. The data is grouped by quarter and month, with values like 2.88% for January 2011 and 14.36% for December 2014.

Quick Start: Computing Totals

Tableau supports grand totals and subtotals for both relational and multidimensional data sources.

Grand Totals

Grand totals can be computed for rows or for columns.

Row Grand Totals

For Rows: Choose **Analysis > Totals > Show Row Grand Totals**.

Continent							
Container	Department	Africa	Asia	Europe	North America	South America	Grand Total
Jumbo Box	Furniture	\$46,239	\$157,241	\$75,163	(\$1,358)	\$4,731	\$278,833
	Technology	\$12,783	\$37,662	\$18,627	\$46,609	\$24,851	\$140,531
Jumbo Drum	Furniture	\$85,769	\$250,496	\$61,124	\$119,529	\$51,796	\$577,977
	Office Supplies	\$7,894	\$43,400	\$17,333	\$27,538	\$38,669	\$134,232
	Technology	\$170,958	\$316,764	\$40,826	\$198,078	\$82,310	\$808,936
Large Box	Furniture	\$8,434	\$147,055	\$19,328	\$53,560	\$41,627	\$270,004
	Office Supplies	(\$3,964)	\$1,278	\$5,763	(\$56,751)	(\$15,325)	(\$32,961)
	Technology	\$16,079	\$97,707	\$101,492	\$179,650	\$122,807	\$538,766
Medium Box	Furniture	\$20,189	\$19,064	\$7,644	\$44,942	\$3,920	\$96,376
	Office Supplies	\$318	\$23,778	\$5,045	\$8,098	\$2,079	\$39,319
	Technology	\$4,505	\$119,792	\$8,181	\$49,947	\$53,027	\$236,167
Small Box	Furniture	\$4,843	\$17,992	\$7,750	\$28,253	\$14,546	\$73,540
	Office Supplies	\$143,447	\$697,153	\$88,570	\$531,999	\$311,863	\$1,788,075
	Technology	\$46,321	\$551,146	\$147,000	\$390,937	\$202,191	\$1,345,595
Small Pack	Furniture	\$7,753	\$31,674	\$5,325	\$29,157	\$6,499	\$80,367
	Office Supplies	\$6,011	\$15,672	\$1,543	\$3,837	\$1,022	\$28,096
	Technology	\$9,994	\$84,756	\$16,800	\$46,690	\$38,398	\$197,930
Wrap Bag	Furniture	\$530	\$8,394	\$5,652	(\$3,137)	\$1,876	\$14,037
	Office Supplies	\$7,770	\$27,015	\$4,869	\$8,586	\$6,040	\$54,257
	Technology	\$1,292	\$25,071	\$4,716	\$12,391	\$3,936	\$47,406

Column Grand Totals

For Columns: Choose **Analysis > Totals > Show Column Grand Totals**.

		Continent				
Container	Department	Africa	Asia	Europe	North America	South America
Jumbo Box	Furniture	\$46,239	\$157,241	\$75,163	(\$1,358)	\$4,731
	Technology	\$12,783	\$37,662	\$18,627	\$46,609	\$24,851
Jumbo Drum	Furniture	\$85,769	\$250,496	\$61,124	\$119,529	\$51,796
	Office Supplies	\$7,894	\$43,400	\$17,333	\$27,538	\$38,669
	Technology	\$170,958	\$316,764	\$40,826	\$198,078	\$82,310
Large Box	Furniture	\$8,434	\$147,055	\$19,328	\$53,560	\$41,627
	Office Supplies	(\$3,964)	\$1,278	\$5,763	(\$56,751)	(\$15,325)
	Technology	\$16,079	\$97,707	\$101,492	\$179,650	\$122,807
Medium Box	Furniture	\$20,189	\$19,064	\$7,644	\$44,942	\$3,920
	Office Supplies	\$318	\$23,778	\$5,045	\$8,098	\$2,079
	Technology	\$4,505	\$119,792	\$8,181	\$49,947	\$53,027
Small Box	Furniture	\$4,843	\$17,992	\$7,750	\$28,253	\$14,546
	Office Supplies	\$143,447	\$697,153	\$88,570	\$531,999	\$311,863
	Technology	\$46,321	\$551,146	\$147,000	\$390,937	\$202,191
Small Pack	Furniture	\$7,753	\$31,674	\$5,325	\$29,157	\$6,499
	Office Supplies	\$6,011	\$15,672	\$1,543	\$3,837	\$1,022
	Technology	\$9,994	\$84,756	\$16,800	\$46,690	\$38,398
Wrap Bag	Furniture	\$530	\$8,394	\$5,652	(\$3,137)	\$1,876
	Office Supplies	\$7,770	\$27,015	\$4,869	\$8,586	\$6,040
	Technology	\$1,292	\$25,071	\$4,716	\$12,391	\$3,936
Grand Total		\$597,165	\$2,673,108	\$642,751	\$1,718,555	\$996,863

Subtotals

Subtotals can be turned on for a selected dimension or for all qualifying dimensions.

Subtotals for a Selected Dimension

When Grand Totals are on for a dimension in the view, right-click the dimension (or control-click on a Mac) and then choose **Subtotals** to turn on subtotals for the selected dimension.

Columns **Continent**

Rows **Container**

Container	Department	Africa	Europe	North America	South America
Jumbo Box	Furniture	\$46,212	\$163	(\$1,358)	\$4,731
	Technology	\$12,720	\$627	\$46,609	\$24,851
	Office Supplies	\$7,833	\$124	\$119,529	\$51,796
Jumbo Drum	Technology	\$170,936	\$333	\$27,538	\$38,669
	Furniture	\$8,426	\$326	\$198,078	\$82,310
	Office Supplies	(\$3,916)	\$328	\$53,560	\$41,627
Large Box	Technology	\$16,045	\$763	(\$56,751)	(\$15,325)
	Furniture	\$20,144	\$492	\$179,650	\$122,807
	Office Supplies	\$3,344	\$644	\$44,942	\$3,920
Medium Box	Technology	\$4,545	\$181	\$8,098	\$2,079
	Furniture	\$4,875	\$750	\$49,947	\$53,027
	Office Supplies	\$143,475	\$570	\$531,999	\$311,863
Small Box	Technology	\$46,321	\$551,146	\$147,000	\$390,937
	Furniture	\$7,753	\$31,674	\$5,325	\$29,157
	Office Supplies	\$6,011	\$15,672	\$1,543	\$3,837
Small Pack	Technology	\$9,994	\$84,756	\$16,800	\$46,690
	Furniture	\$530	\$8,394	\$5,652	(\$3,137)
	Office Supplies	\$7,770	\$27,015	\$4,869	\$8,586
Wrap Bag	Technology	\$1,292	\$25,071	\$4,716	\$12,391
	Furniture	\$530	\$8,394	\$5,652	(\$3,137)
	Office Supplies	\$7,770	\$27,015	\$4,869	\$8,586
Grand Total	\$597,165	\$2,673,108	\$642,751	\$1,718,555	\$996,863

Subtotals for All Relevant Dimensions

To turn on subtotals for all relevant dimensions, choose **Analysis > Totals > Add All Subtotals**.

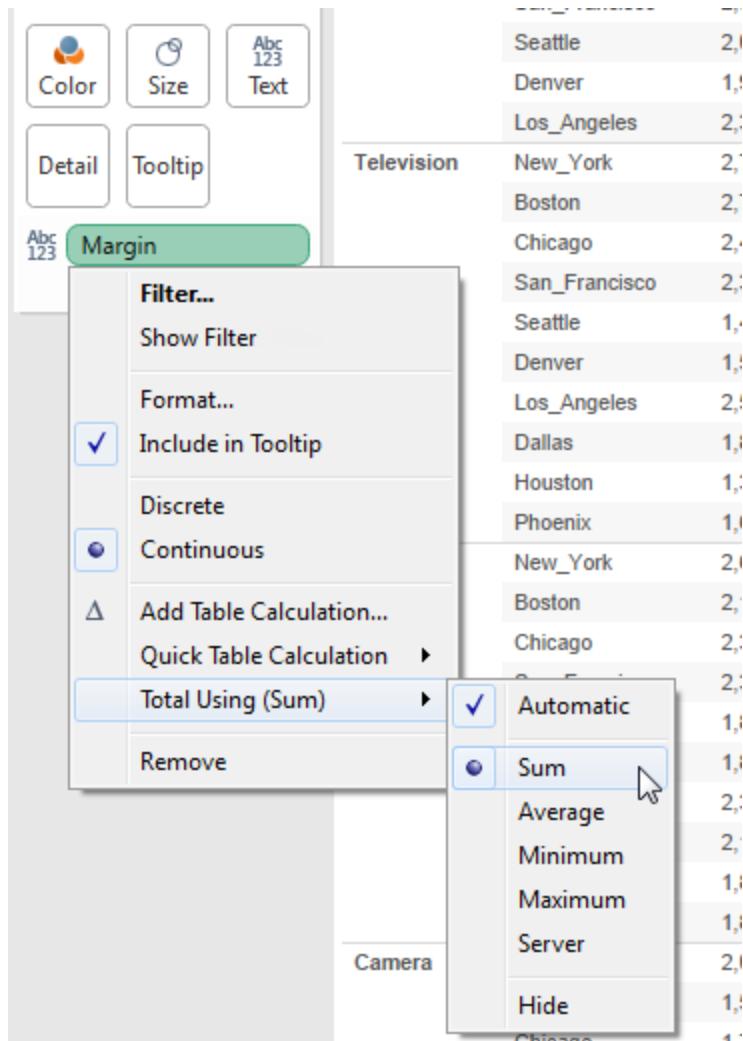
		Continent				
Container	Department	Africa	Asia	Europe	North America	South America
Jumbo Box	Furniture	\$46,239	\$157,241	\$75,163	(\$1,358)	\$4,731
	Technology	\$12,783	\$37,662	\$18,627	\$46,609	\$24,851
	Total	\$59,022	\$194,903	\$93,789	\$45,251	\$29,582
Jumbo Drum	Furniture	\$85,769	\$250,496	\$61,124	\$119,529	\$51,796
	Office Supplies	\$7,894	\$43,400	\$17,333	\$27,538	\$38,669
	Technology	\$170,958	\$316,764	\$40,826	\$198,078	\$82,310
	Total	\$264,621	\$610,659	\$119,283	\$345,145	\$172,775
Large Box	Furniture	\$8,434	\$147,055	\$19,328	\$53,560	\$41,627
	Office Supplies	(\$3,964)	\$1,278	\$5,763	(\$56,751)	(\$15,325)
	Technology	\$16,079	\$97,707	\$101,492	\$179,650	\$122,807
	Total	\$20,550	\$246,039	\$126,583	\$176,458	\$149,109
Medium Box	Furniture	\$20,189	\$19,064	\$7,644	\$44,942	\$3,920
	Office Supplies	\$318	\$23,778	\$5,045	\$8,098	\$2,079
	Technology	\$4,505	\$119,792	\$8,181	\$49,947	\$53,027
	Total	\$25,011	\$162,634	\$20,870	\$102,988	\$59,026
Small Box	Furniture	\$4,843	\$17,992	\$7,750	\$28,253	\$14,546
	Office Supplies	\$143,447	\$697,153	\$88,570	\$531,999	\$311,863
	Technology	\$46,321	\$551,146	\$147,000	\$390,937	\$202,191
	Total	\$194,612	\$1,266,291	\$243,320	\$951,189	\$528,600
Small Pack	Furniture	\$7,753	\$31,674	\$5,325	\$29,157	\$6,499
	Office Supplies	\$6,011	\$15,672	\$1,543	\$3,837	\$1,022
	Technology	\$9,994	\$84,756	\$16,800	\$46,690	\$38,398
	Total	\$23,758	\$132,102	\$23,669	\$79,684	\$45,919
Wrap Bag	Furniture	\$530	\$8,394	\$5,652	(\$3,137)	\$1,876
	Office Supplies	\$7,770	\$27,015	\$4,869	\$8,586	\$6,040
	Technology	\$1,292	\$25,071	\$4,716	\$12,391	\$3,936
	Total	\$9,592	\$60,480	\$15,237	\$17,840	\$11,852
Grand Total		\$597,165	\$2,673,108	\$642,751	\$1,718,555	\$996,863

Totals on Multidimensional Data Sources

When you turn on grand totals or subtotals using the above instructions, the computation is performed on the server. If you are using a multidimensional data source, it is sometimes impossible for the server to compute totals accurately. In this case you can specify that the computation be performed locally using the data that you see in the table.

Q=

To do this, right-click the measure being totaled and select **Total Using**. Then select an aggregation from the sub-menu.



Quick Starts about Dashboards and Stories

After you create your data visualization, there are many ways to use it. For example, you might want to compare it with other visualizations of the same data to see how different conclusions jump out from different ways of looking at the data.

You might want to share your findings. You can present your visualization to your team, share it with the public, or even arrange visualizations into a sequence to support a narrative. Dashboards and stories are excellent tools for analyzing and presenting data.

- A **dashboard** is a collection of several worksheets and supporting information shown in a single place so that you can compare and monitor a variety of data simultaneously.
- A **story** is a sheet that contains a sequence of worksheets or dashboards that work together to convey information.

The following Quick Starts give you a glimpse into the power of presentation. To learn even more, see the [Dashboards](#) on page 1149 and [Stories](#) on page 1202 sections of the documentation.

Quick Start: Design Device Layouts for Dashboards

You can create a dashboard that adapts its layout and contents to any device that displays the dashboard. After you publish the dashboard to Tableau Server, people who view it experience a dashboard tailored for viewing on their phone, tablet, or desktop. As the author, you only have to create a single dashboard and deliver a single URL.

1 Create a Dashboard

Click the **New dashboard** tab at the bottom of your workbook then click and drag worksheets from the **Dashboard** pane to the dashboard on the right.

Dashboa... Layout

Layouts

Default dashboard

Show Preview

Size

Desktop browser (1000 x 800) ▾

Sheets

- Hourly Bike Acitivity
- Station Map
- What days and how ...
- KPI: Total Trips per Day
- KPI: Subscribers per ...
- KPI: Public Customer...
- KPI: Male Riders per D...
- KPI: Female Riders Per...
- KPI: Trip Duration
- Sheet 16
- Gender Usage
- Sheet 7
- Heat Map - End Time ...
- Heat Map - Start time...
- Age of Bike Users

Click the image to replay it.

2 Preview Device Layouts

Previewing is the first step to adding a new layout. Click **Show Preview**, then click a **Device type** arrow to advance through the available layouts.

Dashboard Layout Device Preview Device type ▲ Phone ▾

Chicago Bike Sharing

Divvy is a new bike sharing system featuring thousand Hyde Park, available 24/7, 365 days a year.

Date range: 7/9/2013 12:00:00 AM — 8/29/2013 11:59:59 PM

219,501 TRIPS 25.88 AVG RIDE (MIN) 76,805 MEMBERS

Station map

Layout - Phone

Default **Custom**

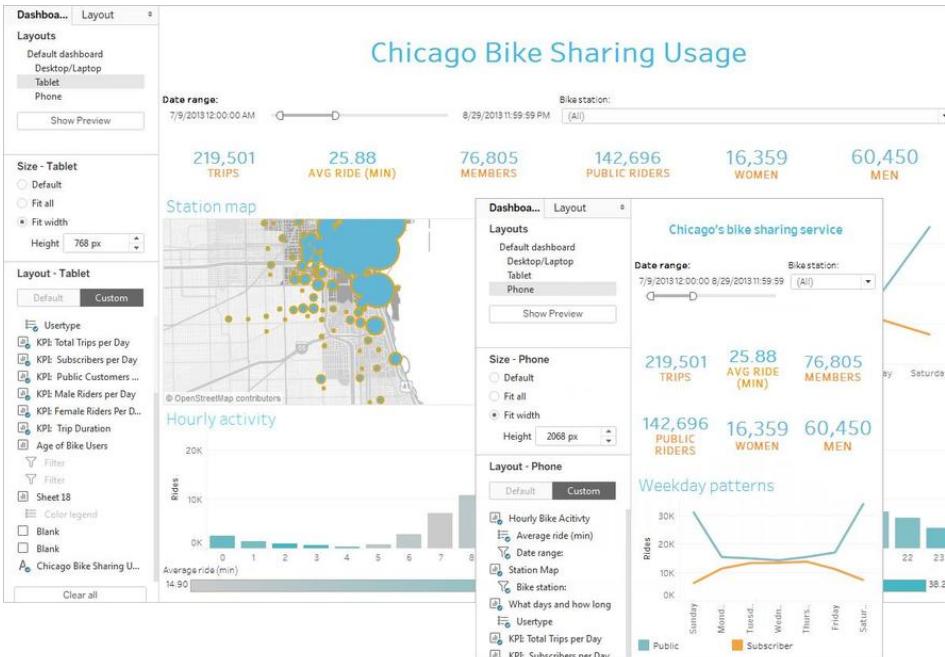
The content and layout of the dashboard on all devices will match the Default dashboard.

Click "Custom" to change how the dashboard will look for the devices.

The frame shows the display size for the device you select.

3 Add Device Layouts

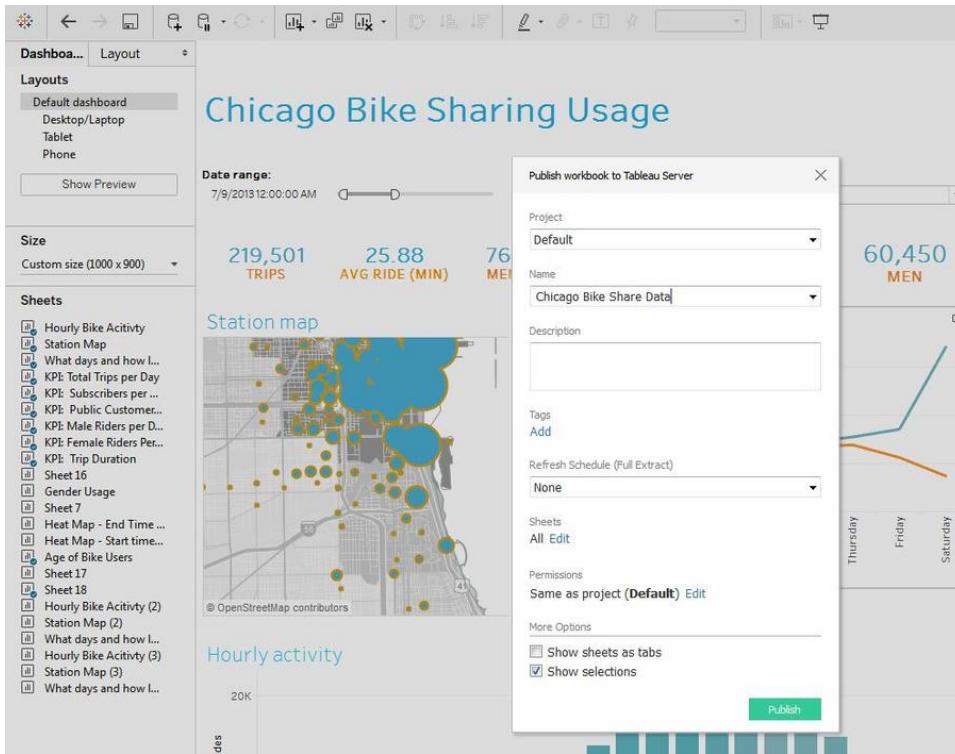
To create custom layouts for phones, tablets, or laptops, click **Add Layout** and start moving sheets in or out of the worksheet. **Fit all** is an easy way to adjust the dashboard to fit a specific screen size.



Before you can add an object or sheet to your layout, it must exist in the default dashboard.

4 Publish the Dashboard

When you've finished creating layouts, publish your dashboard to Tableau Server by selecting **Server > Publish workbook**.



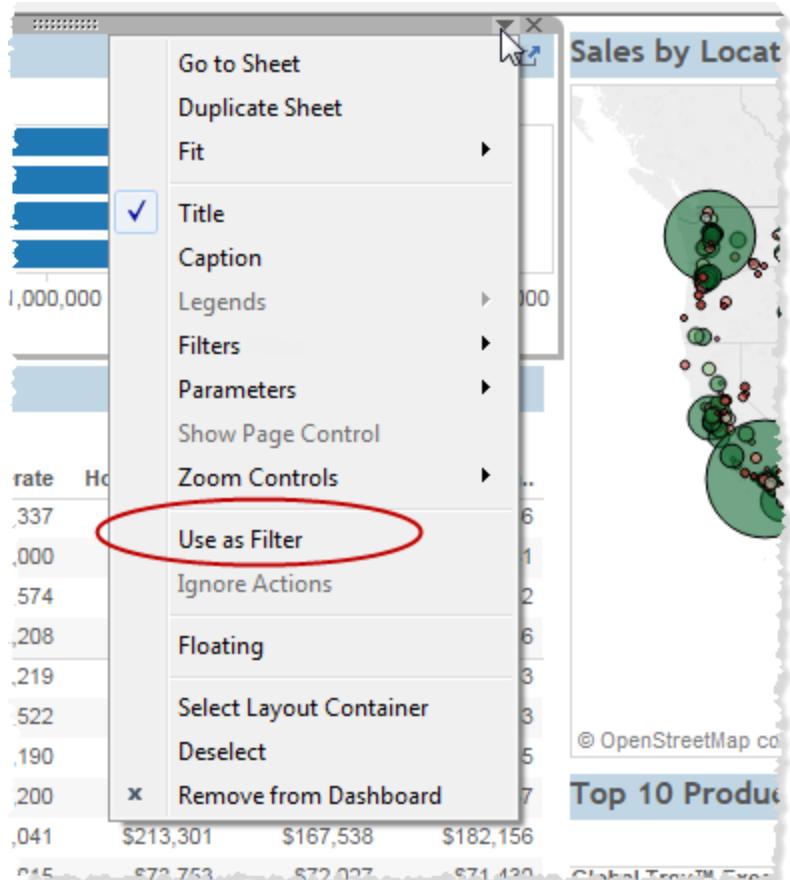
People viewing your dashboard can use a single URL. The layout they see is appropriate for their device screen size.

Quick Start: Filter Actions

Use the Filter Actions filter to show related information between a source sheet and one or more target sheets. This type of action works well when you are building guided analytical paths through a workbook or in dashboards that filter from a master sheet to show more details.

1 Select a View to Use as a Filter

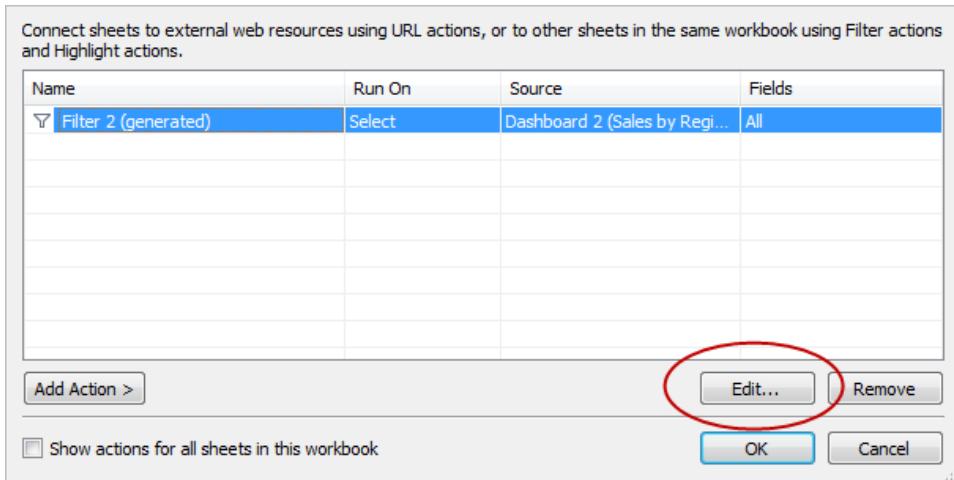
On a dashboard, click the drop-down menu of a sheet and select **Use as Filter**.



A filter action is created so when you select a mark in the view the rest of the dashboard updates to show only related data.

2 Edit the Action

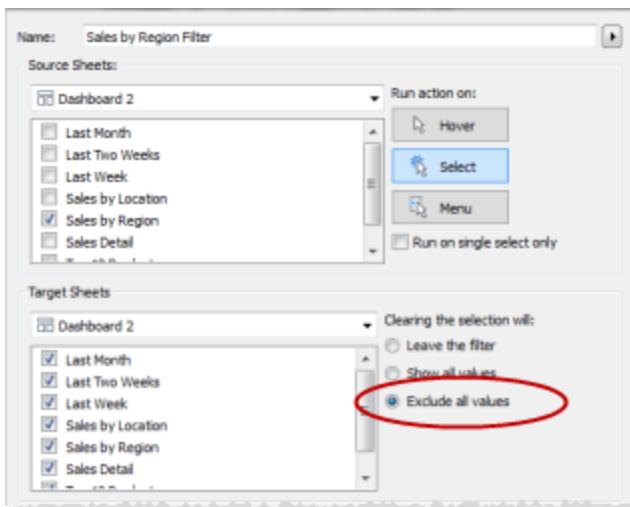
Select **Dashboard > Actions** to open the **Actions** dialog box. Select an action and then click **Edit**.



Any action you create displays in the **Actions** dialog box. You can add Filter, Highlight, or URL actions.

3 Specify the Action Settings

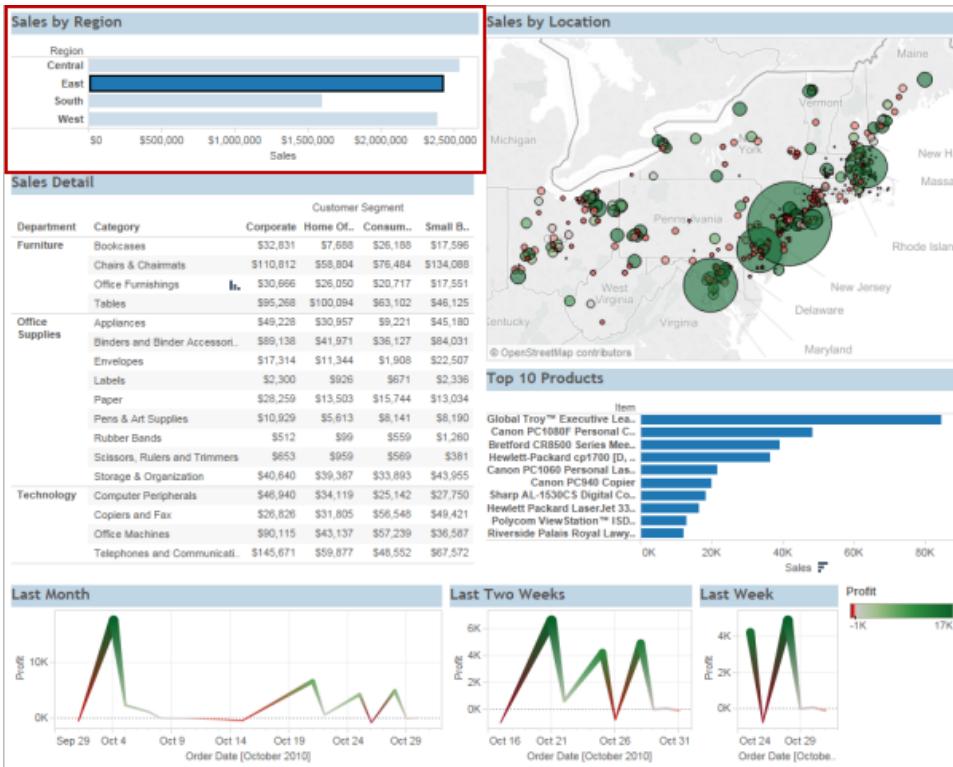
In the **Edit Filter Action** dialog box, select the source sheets, target sheets, and the fields to filter. You can also select whether to run the action when you hover, select, or right-click the source sheet. When finished, click **OK**.



To remove the filter when you deselect the data in the source sheet, select **Exclude All Values** in the **Clearing the Selection Will** section.

4 See it in Action

Hover, select, or right-click the source sheet and watch the other sheets update to show related data.



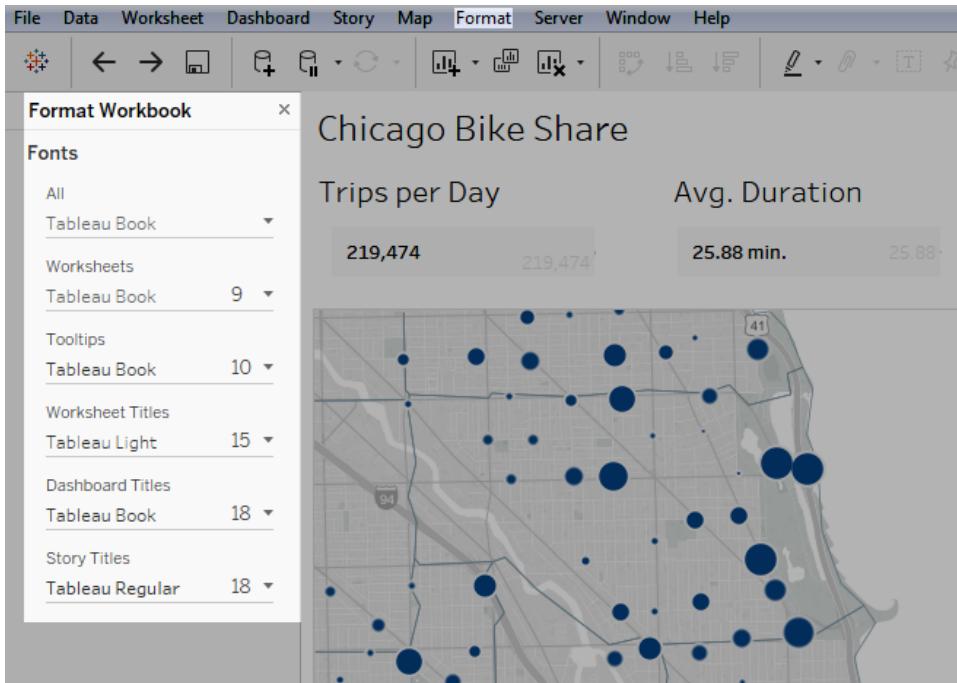
In this example, when you select the **East** region in the top sheet, the rest of the sheets filters to show sales information for this region.

Quick Start: Workbook Formatting

You can change the look of views and dashboards across your entire workbook at once by using formatting at the workbook level, instead of applying formatting individually at the view, dashboard, or story level. For example, you might want to change a workbook's font so that all the views, dashboards, and stories in the workbook adhere to your company's brand.

1 Open the Format Workbook pane

In Tableau Desktop, open an existing workbook or create a new one by clicking the **New Workbook** icon. On the **Format** menu, click **Workbook** to open the **Format Workbook** pane.



2 Change the Font Workbook-Wide

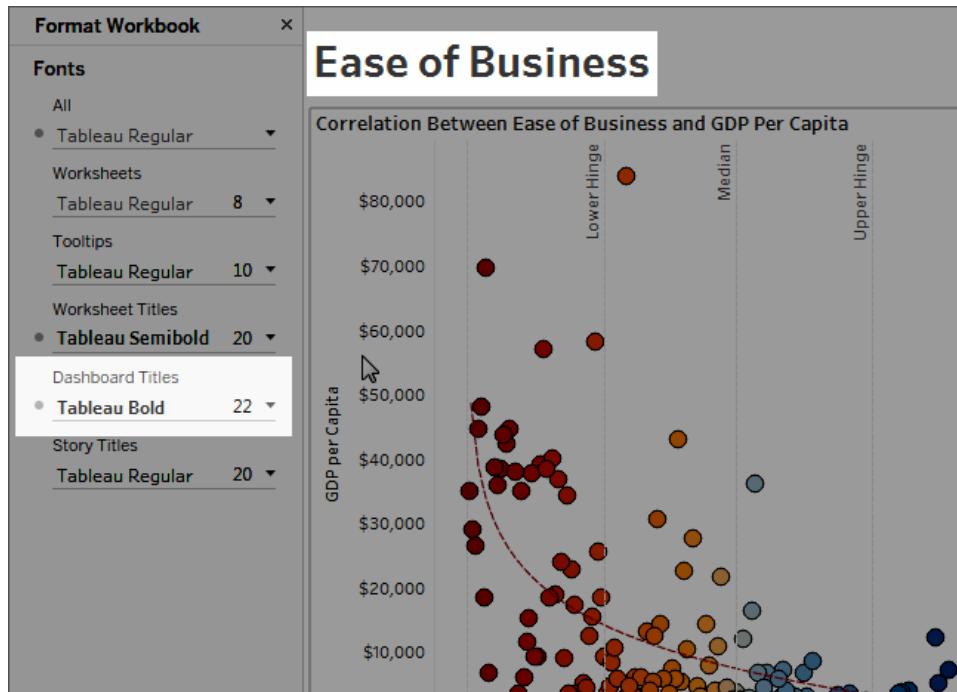
Under **Fonts**, click the **All** drop-down arrow to select a new font. Your font choice is applied to all text that appears in your workbook's views.



The Tableau typeface was designed to be paired with data visualizations and is optimized to be highly legible at small sizes.

3 Fine-Tune Title Fonts and Sizes

After you've changed the font used throughout your workbook, you can adjust just your workbook's titles. For example, you can fine-tune the fonts and sizes used by worksheet titles vs. dashboard titles.



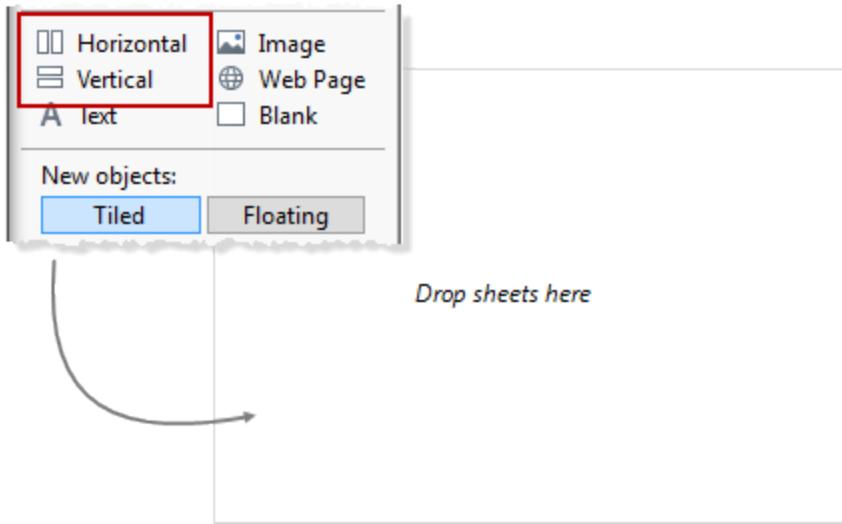
The gray dot indicates that the font has been changed from the default. Clicking the dot returns the setting to its default.

Quick Start: Layout Containers

Layout containers help you organize sheets and other objects on a dashboard. These containers create an area in the dashboard where objects automatically adjust their size and position based on the other objects in the container. For example, a dashboard with a filter action that changes the size of a view will automatically adjust the other views when the filter is applied.

1 Add a Layout Container

Create a new dashboard and drag a horizontal or vertical layout container to the view.



2 Add Sheets

Drag sheets into the layout container. A gray rectangle indicates where the sheet will be placed in the layout container.

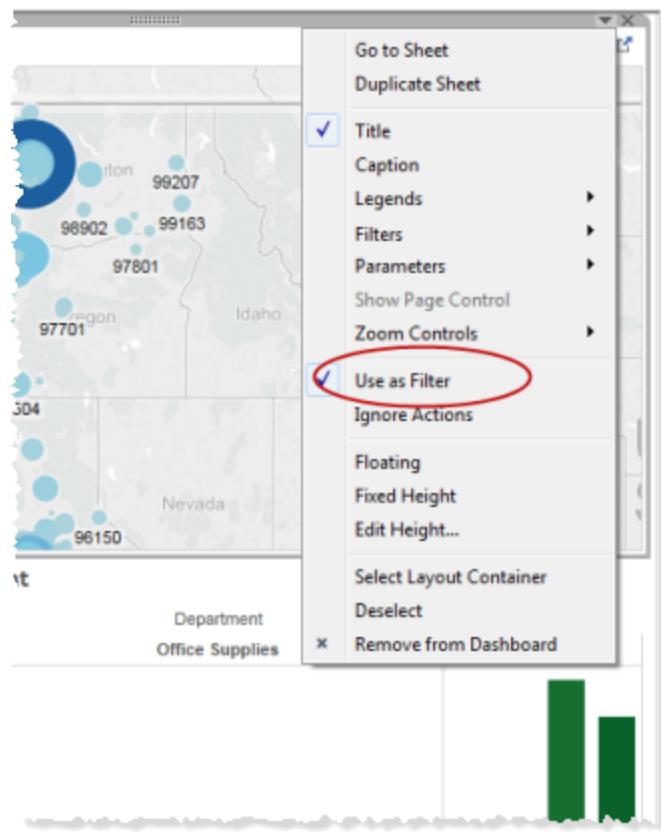


The corresponding quick filters and legends are automatically added with each sheet.

3 Create and Edit a Filter Action

To have the views automatically resize depending on what is selected in the view, create a filter action. To create a filter action in the dashboard, click the drop-down menu of a sheet and

select **Use as Filter**. Next, go to **Dashboard > Actions** and edit the generated filter action you just created. In the **Edit Filter Action** dialog box, select **Exclude all values**.



4 Watch Objects Move and Resize

Click a mark in the view with the filter applied to it to update the other view. In the example, when a mark is selected in the map view, the bar chart updates to display the profit and sales for that mark; when no marks are selected in the map view, the map automatically expands to fill the dashboard.

With mark selected

No selection



Remove a Layout Container

Q-

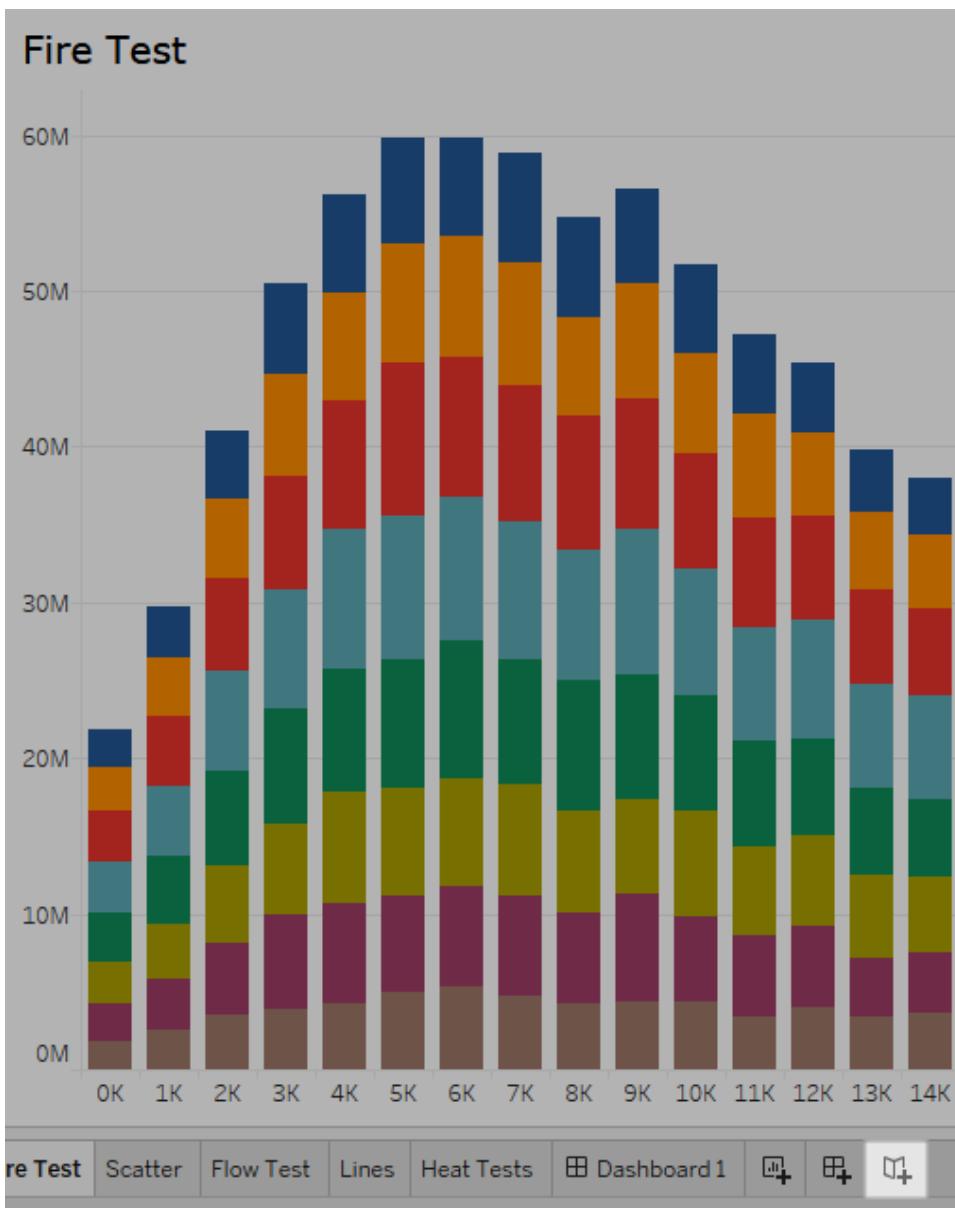
To remove a layout container, click the drop-down menu of a sheet and select **Select Layout Container**. The layout container is highlighted blue. Click the layout container drop-down menu and select **Remove from Dashboard**. The container and all of its contents are removed.

Quick Start: Stories

A story is a sheet that contains a sequence of worksheets and dashboards that work together to convey information. Use stories to make your case more compelling by showing how facts are connected, and how decisions relate to outcomes. You can then publish your story to the web, or present it to an audience.

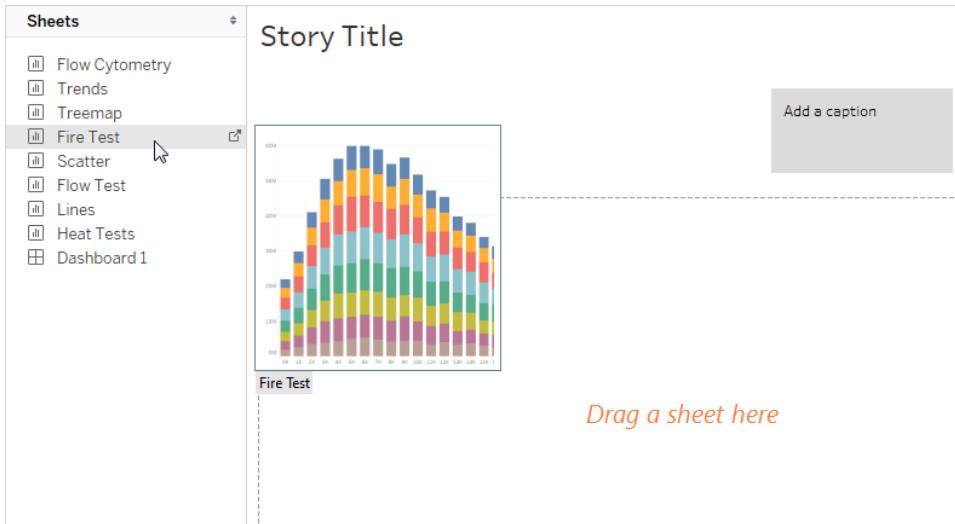
1 Create a New Story

To create a story, click the (new story) tab.



2 Add the First Story Point

To create the first point in your story, drag a worksheet or dashboard to the story and drop it in the center of the view. Add a caption, and then customize your story point.



Hover over a worksheet or dashboard to see a preview before adding it to your story.

3 Add More Story Points to Build Your Story

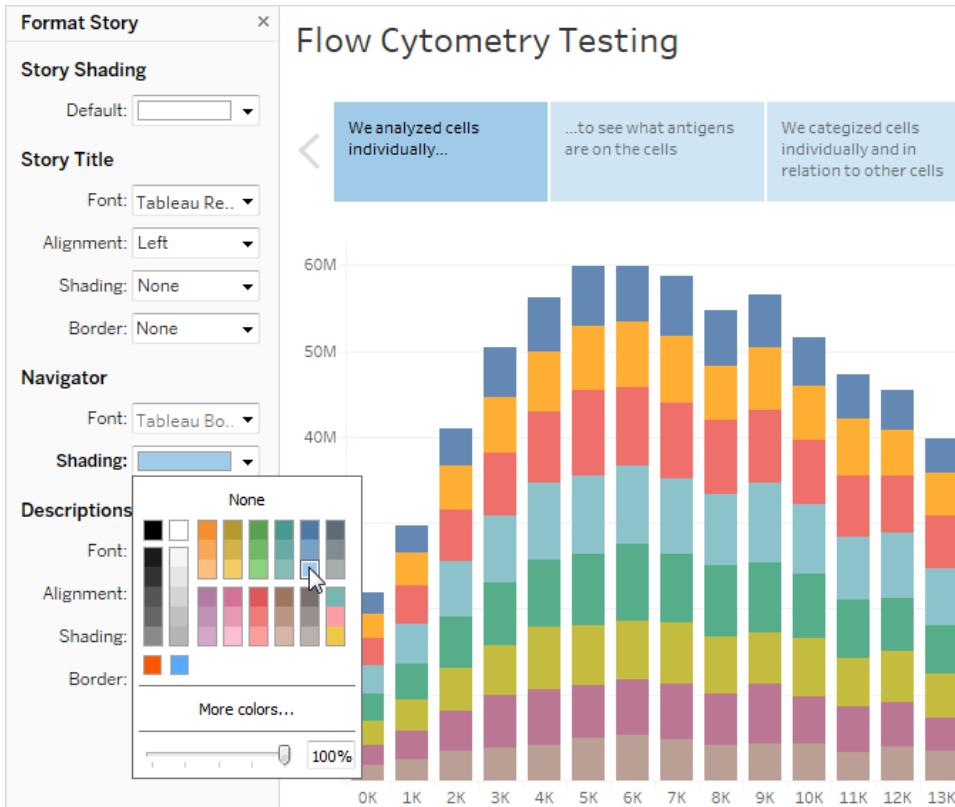
To add a new story point based on a different sheet, click **New Blank Point**. To use the same sheet in the next story point, click **Duplicate**.



Use the navigator to move through the story.

4 Format Your Story

To format your story, click **Format > Story**. In the **Format Story** pane, you can change the background colors for the story and the navigator bar, and change the appearance of the story title, the captions in the navigator, and the descriptions.



Publish Your Stories to the Web

Q-

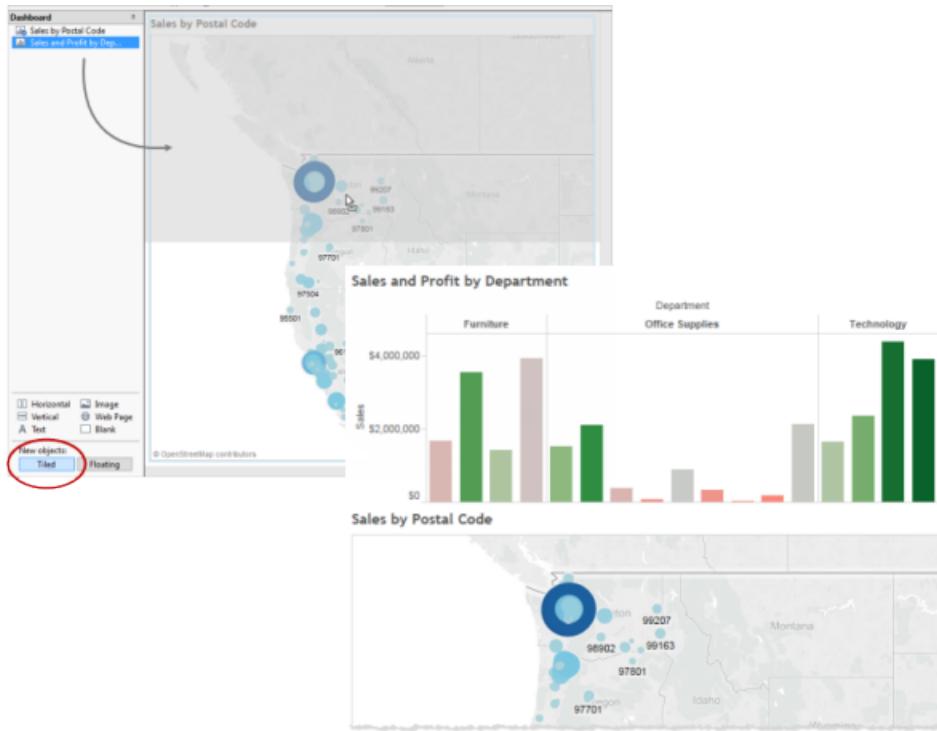
You can save your workbook that contains the story to Tableau Public or publish it to Tableau Server and Tableau Online.

Quick Start: Tiled and Floating Dashboard Layouts

A dashboard is a collection of worksheets and objects on a single sheet so you can compare and monitor a variety of data simultaneously. Dashboard objects can be tiled or floating. Tiled objects are arranged in a grid while floating objects can be layered on top of other objects.

1 Add a Sheet as Tiled

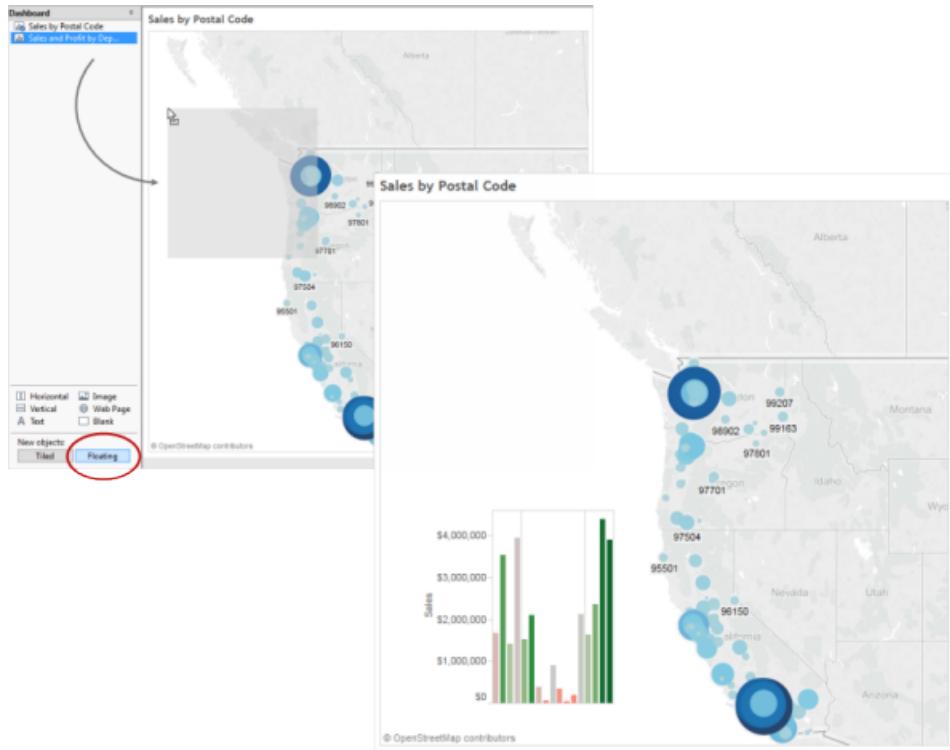
To add a sheet to the tiled layout, click **Tiled** and then drag a sheet to the view.



Adding a tiled sheet places it in a layout container and arranges it into a grid.

2 Add a Sheet as Floating

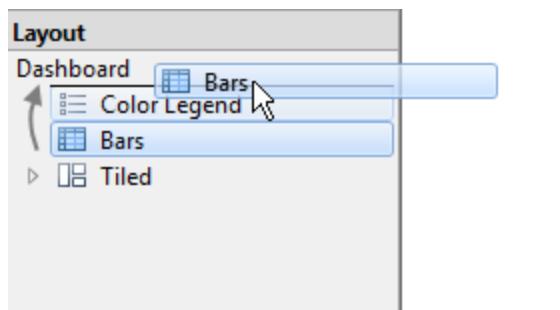
To add a sheet to the floating layout, click Floating and then drag a sheet to the view.



Adding a floating sheet places it on top of the other objects in the dashboard.

3 Reorder Floating Objects

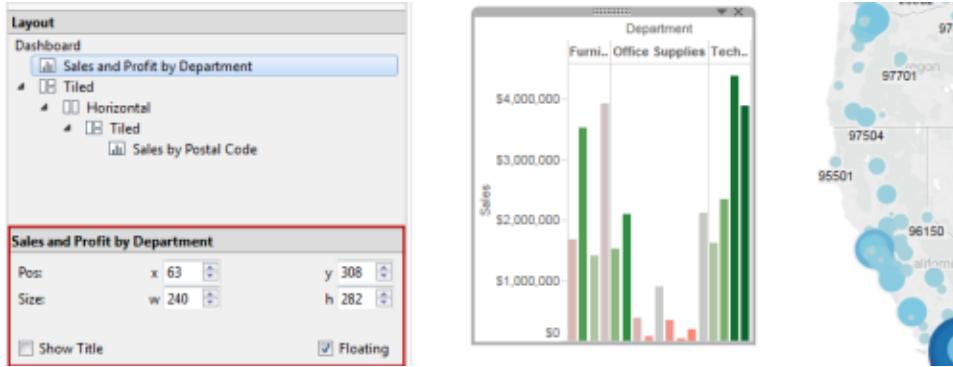
You can change the order of floating objects by rearranging the sheets in the **Layout** area of the **Dashboard** pane. Objects at the top of the list display in the front, while objects at the bottom of the list display in the back.



Right-click (control-click on the Mac) an object in the layout area to show/hide legends and filters, go to sheet, and more.

4 Resize Floating Objects

When you select an object in the dashboard, the size and position properties are shown at the bottom of the **Dashboard** pane. You can make small adjustments to the size and position as well as toggle between whether it is floating or tiled.



Holding the shift key while dragging an object toggles whether it is floating or tiled.

Quick Starts about Using Tableau Server

Tableau Server is an online solution for sharing, distributing, and collaborating on content created in Tableau.

You can create views in Tableau Desktop and then publish them to Tableau Server.

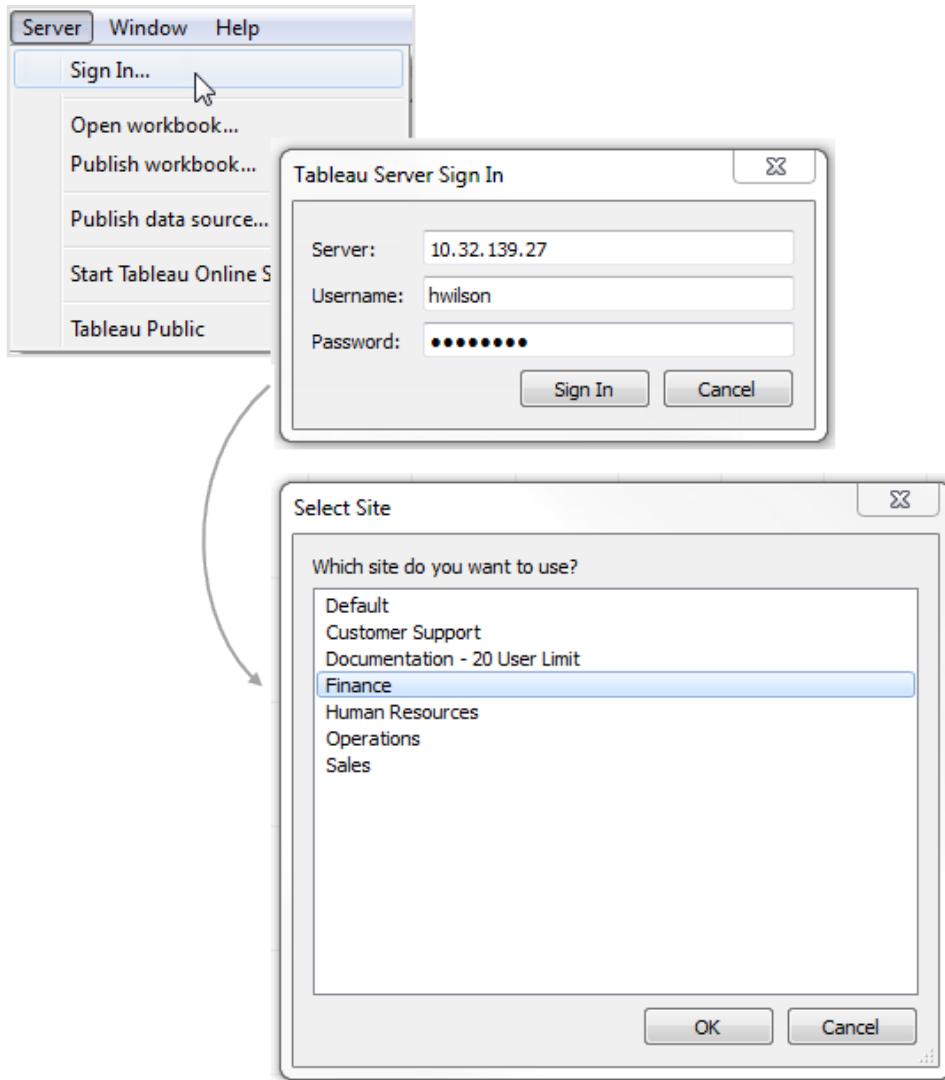
Quick Start: Stay Connected with Automatic Sign-In

When you connect to Tableau Server or Tableau Online, Tableau Desktop keeps you signed in from session to session, if you don't sign out. Next time you launch Tableau Desktop, you are automatically signed in to your most recent server connection. As you sign in to different servers and sites, you can easily switch between your available server and site connections.

Note: Automatic sign-in is available for servers configured for Windows Authentication only, and must be enabled by a server administrator in the server's **Settings** page, under **Connected clients**. Automatic sign-in is not available for servers that use Kerberos or SAML authentication.

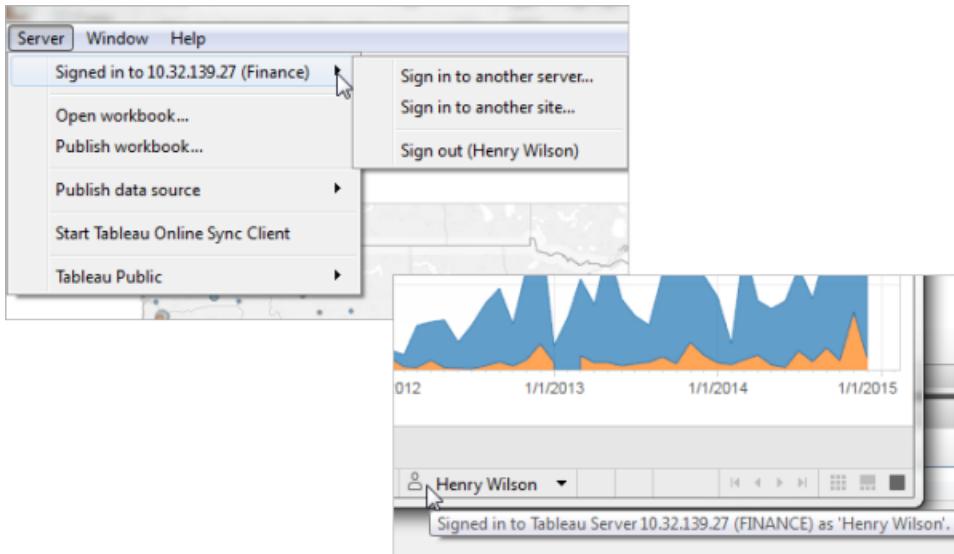
1 Sign in

Launch Tableau Desktop and then sign in to Tableau Server or Tableau Online. If you have access to multiple sites on the server, select the site you want to use.



2 Launch Tableau Desktop again

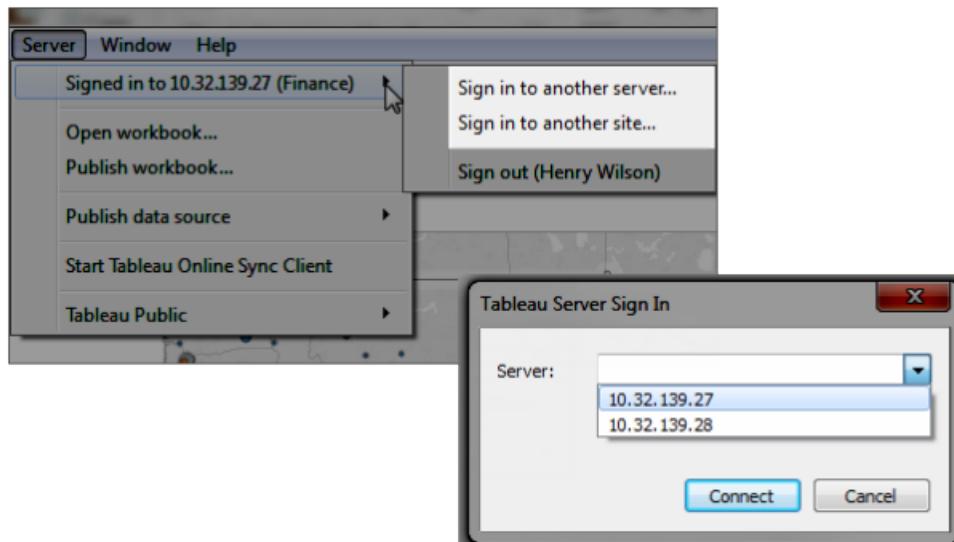
Next time you launch an instance of Tableau Desktop, you are signed in to your most recent server connection – no need to enter your credentials. You can see the server and site you are signed into, as well as who you are signed in as in two places. The **Server** menu shows the server and site that you are signed in to. **Sign out** shows you are signed in as. If you hover



Hover over the user name in the status bar to see the currently signed in user, the server, and the site.

3 Switch servers, switch sites

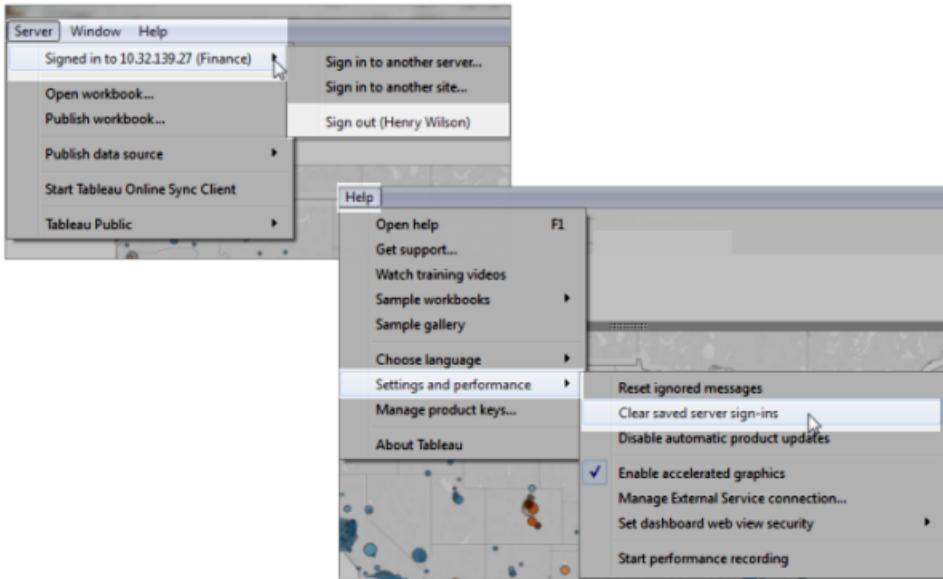
To switch servers, on the **Server** menu, click **Sign in to another server**, and then enter the server name and your sign-in credentials, or select a different server name from the drop-down menu. To switch sites, on the **Server** menu, click **Sign in to another site**, and then select the site.



4 Sign out or clear all saved connections

Click **Server > Sign out** to clear your sign-in credentials for the current connection. To clear all sign-in credentials, click **Help > Settings and performance > Clear saved server sign-**

ins.



More about automatic sign-in

- When you sign in to a server, your server authentication token is stored in secure storage **U-** on your computer. This token is used to authenticate you every time you connect to the server.
- If you are signed in to a server or site when you close Tableau Desktop, your sign-in credentials are saved. Also, when you switch sites or servers, Tableau Desktop saves your sign-in credentials.
- When you click **Sign Out** on the **Server** menu, Tableau Desktop deletes your sign-in credentials for that connection. To delete credentials for every server and site you have previously signed into, click **Help > Settings and performance > Clear saved server sign-ins**.
- If you have access to a single-site server, or you have access to only one site on a server you are signing in to, **Sign in to another site** will not be available.

Upgrade Tableau Desktop

This section includes information on upgrading from a previous or a beta version, and on how to turn the product update feature off or on.

Upgrade from a Previous or Beta Version

When you upgrade to Tableau 10.1 from previous versions, your repository is upgraded. Any bookmarks, workbooks, and data sources that you had in your old repository will still be

accessible by the application. In addition, the new sample data sources and workbooks will replace the old samples unless you have modified them and saved them as your own.

Note: When you download a new version of Tableau Desktop, the product update installers are downloaded to the Downloads/TableauAutoUpdate folder. If the Downloads folder doesn't exist, the installers are downloaded to the TEMP/TableauAutoUpdate folder.

If you have participated in the Tableau Software Beta program, you also have a beta repository. While this folder will still exist after you install Tableau 10.1, the application will no longer access it. To make your beta workbooks accessible in Tableau 10.1, copy the workbooks from the beta repository to your new 10.1 repository.

Turn Product Updates Off or On

To ensure that you always have the most up-to-date features, security resolutions, and corrected issues, Tableau Desktop includes a product update feature. When you start Tableau Desktop, product updates prompts you to download an updated maintenance version of Tableau Desktop, if one exists. The update downloads immediately and then installs when you exit Tableau. You can also choose to postpone or skip the update.

Note the following:

- The product update installers are downloaded to the Downloads/TableauAutoUpdate folder. If the Downloads folder doesn't exist, the installers are downloaded to the TEMP/TableauAutoUpdate folder.
- Updates are not downloaded and installed on your computer if your Product Maintenance has expired. For more information, see the [Product Maintenance FAQ](#).
- You may not be prompted for product updates. There are a number of reasons why this may be so. For details, see the **Troubleshoot maintenance updates** section in the [Troubleshoot Tableau Desktop Installation](#) article in the Tableau Desktop Deployment Guide.

Turn off product updates

Product updates is on by default. You can turn off product updates from the Help menu, or on Windows, by running the installer.

Use the Help menu

Select **Help > Settings and Performance > Disable Product Updates for Me**.

Run the installer (Windows only)

To turn off product updates on Windows, run the Tableau Desktop installer, and then click **Customize**. In the Custom setup dialog box,

1. Clear the **Check for Tableau product updates** check box.
2. Click **Install**.

To turn product updates on again, run the installer, click **Customize**, and select the check box.

Administrators control product updates

As an administrator, you can turn product updates off or on for your users. You can also determine the Tableau Desktop version that your users will update to. Rather than having users update to the version of their choice (or choose not to update), you can make sure that your users update to the version you choose. For more information, see the [Control Product Updates](#) article in the Tableau Desktop Deployment Guide.

Accessing the Help

Tableau offers multiple ways to access the help topics. First, you can always view the online help by selecting **Help > Open Help** in the application. You must have an internet connection to view these pages. However, you can download a copy of the help files to your hard drive so that you can open them even when you are not connected to the internet. Finally, you can choose to download a printable PDF version of the help. This section explains each of the three ways to view the help.

Viewing the Online Help

The online help is the best way to access the most current documentation. You can open the help by selecting **Help > Open Help** in the application or by pressing **F1** on your keyboard. You must have an internet connection to access this help.

Download the Offline Help

Although it is more efficient to view the help online, there may be times when you do not have access to the internet. In these cases, you may want a copy of the help stored on your computer. Download the offline help from the [Product Help](#) page on the Tableau website.

You must have an internet connection to initially download the help, but any subsequent times you want to access the help you are not required to be online.

If you choose to download the help and use the offline version, please remember to update your local files periodically in order to ensure you are viewing the most recent information. Anytime you want to update your offline help files, repeat the download and overwrite the existing files.

If you download the offline help, Tableau will always open the local files when you select **Help > Open Help** in the application. You can view the online help by visiting the [Product Help](#) page on the Tableau website

Download a PDF

You must have Adobe Reader to view this file. Adobe Reader is a free software that can be found on Adobe's Web site: www.adobe.com. You can download the PDF from the [Product Help](#) page on the Tableau website.

Optimize Workbook Performance

When we talk about *performance*, we mean the speed with which you can work in Tableau. That might mean speed of data analysis, for example, if you work in Tableau Desktop on a large corporate database that takes forever in real-time when you access it remotely. Or it might mean simply the speed of loading views or dashboards that you access on your desktop or from Tableau Server.

If you are working with small data volumes, many of the recommendations in this section are optional for you. Your workbook performance is probably as speedy as you expect.

But if you work with hundreds of millions of records, designing an efficient workbook is critically important to the speed at which you can work. We recommend that you review the topics and tips in this section before you start building your first view.

Tableau provides a number of features designed to help you optimize performance. These features are discussed throughout the help within the descriptions of tasks and workflows where you would most likely encounter them. They are also collected here in case you want make performance your focus.

Making performance improvements almost always means making trade-offs. For example: Do you sacrifice speed to make sure you have the most current data? Is it important to include *all* the data for analysis, or will a speedier subset do?

The bigger and more complicated the data, the longer it takes to interpret and render, but there are things you can do to accelerate the process. Think of fine-tuning your workbook performance as working on a puzzle with multiple pieces. This section provides you with the pieces of information, and you pick and choose which pieces solve your particular puzzle.

This section starts by discussing the big picture and then drills down to specific functionality. We start with databases, data, and extracts, and then proceed through things you can do that affect your data source, your workbooks, your calculations, and visualizations. Next, we talk about how to record and analyze workbook performance. Finally we look at how to reduce upload times to a Tableau server.

Know Your Data at the Database Level

Understanding the impact that certain factors in the design of your database have on Tableau performance might give you information you can use to work with your database team to optimize data at the database level.

Enable support for referential integrity

Databases that support referential integrity support the Tableau Assume Referential Integrity feature, which improves the performance of inner joins. Joins cost time and resources to process on the database server. When you join multiple tables in a data source, Tableau uses functionality that is generally invisible to the user, called *join culling*. Join culling queries only the relevant tables instead of all tables defined in your join. Join culling only occurs where referential integrity is defined between tables. For more information, see [Assuming Referential Integrity on page 367](#).

Make sure database permissions support creating temp tables

Does your database grant users permission to create and drop temporary tables, also known as temp tables, and does the environment have sufficient spool space for the queries being run? Tableau creates temp tables to help improve performance and add functionality. The temp tables temporarily hold information specific to a connection. If the creation of temp tables in the database is restricted, the overall performance of workbooks and views isn't as fast as it could be.

Create indexes for tables

Index the tables in your relational database. To successfully index your data set, identify the fields that you frequently filter on and add them to the index. If you have a field that you use as a context filter often, consider setting it as your primary index. If you are working with Access tables that have more than 200,000 rows of data, consider setting indexes on the tables. You can learn how to do this by searching for “index” in the Access online help. You can officially store 2 GB of data (approximately 1-2 million rows) in an Access database, but it performs poorly well below this limit.

Many database management system (DBMS) environments have management tools that will look at a query and recommend indexes that would help.

Break up your data

You can often improve performance if you partition a large database table into multiple smaller tables. For example, you can create a cluster of Access tables that addresses specific subsets of your data.

Use a database server

If you have a lot of data, you might consider storing it in a database server, such as Oracle, MySQL, or Microsoft SQL Server. The Professional Edition of Tableau can connect to these larger database servers.

Test Your Data and Use Extracts

You probably have no control over the hardware, processing power, and disk space that your database runs on. Or how many people access the database at any given time, how many services are running, or how robust the network is. But you can test the *raw performance* of the data before you begin to build a workbook, so that you understand the environment you're working in. And there are things you can do to affect the performance of the workbooks you create, starting with how you choose to connect to your data: the drivers you use, and whether your connection type is *live* or *extract*.

Test as close to the data as possible

A good way to test the raw performance of the data is to install Tableau Desktop on the computer where the data resides and to run some queries. This eliminates external factors such as network bandwidth and latency from performance results, and help you to understand the raw performance of the query in the data source.

Additionally, you can use the *localhost* name for the data instead of the DNS name to help determine if environmental factors such as slow name resolution or proxy servers are slowing performance.

Connect with native database drivers

When you connect to data with native database drivers, you will often experience significantly faster performance than when you connect to the same data with ODBC drivers. When you use native database drivers, make sure that you're using the latest version. Database vendors are always working to improve their drivers. For a list of drivers, see [Drivers and Activation](#) on the Tableau website.

Work with extracts instead of live data

Depending on your data, you can choose between a live or extract connection on the data source page. A *live* connection is a direct connection to your data. A Tableau data *extract* is a compressed snapshot of data stored locally and loaded into memory as required to render a Tableau visualization. Extracts are designed to use all parts of your computer's memory optimally.

There are several reasons to use an extract, but the main performance-related reason is if your query execution is slow. The extract data format is designed to provide a fast response to analytic queries. In this case, you can think of the extract as a query acceleration cache.

For more information about Tableau extracts, see [Extract Your Data](#) on page 440. Another reference is a three-part blog post about extracts that starts with the first post, [Understanding Tableau Data Extracts](#).

Reduce the amount of data

When you create an extract, use filters to exclude data that you don't need. Also, ask yourself if you need all of the records in a data source, or if you can limit the extract to a representative sample. For more information, see [Extract Your Data on page 440](#).

Hide unused fields

Hidden fields are not included when you create an extract. Use the **Hide All Unused Fields** option to hide unnecessary fields before you create an extract. This makes the extract smaller, which improves performance. For more information, see [Hide or Unhide Fields on page 313](#).

Optimize extracts

Since an extract is a columnar store, the wider the data set, the slower the query time. You can optimize an extract by selecting a data source on the **Data** menu and then selecting **Extract > Optimize**. Optimizing an extract creates secondary structure in the extract that speeds up future queries. For more information, see [Optimize Extracts on page 448](#).

Use extracts for file-based data

In general it's best practice to import file-based data—text files such as CSV, Microsoft Excel spreadsheets, and Microsoft Access files, as well as statistical files like SPSS, SAS, and R—into Tableau. This makes queries perform much faster and also results in a much smaller file to store the data values.

If your data is a large text or Excel file, using an extract not only improves performance but also makes more functionality available to you. Note that if you connect Tableau to a large text file, you will be prompted to extract the data if Tableau discovers that the file is too large to perform well.

Avoid using custom SQL

In most cases, custom SQL runs slower than queries created by Tableau. Tableau cannot perform query optimizations on custom SQL. But in cases where you must use custom SQL, use an extract so that the query runs only once. For more information, see [Connect to a Custom SQL Query on page 404](#).

Create Efficient Joins and Blends

Now that you've made the initial connection to your data, consider how you set up your data source—especially joins and blends—to make it efficient.

If your analysis requires data from different databases, including different workbooks or files, you should consider the impact of joining or blending your data. Joins and blends can improve performance, but they also require more processing power, and so take more time to execute.

A key factor in efficiency is reducing the amount of data you analyze. We'll outline a few ways you can do this.

Should I join or blend my data?

When you consider joining tables or blending tables, think about where the data is coming from, the number of connections, and the number of records you have. If the workbook uses data from more than one database, you must either blend data or add another connection to the existing data source and create a cross-database join. If the workbook uses multiple tables from the same database, joining the tables can improve performance and filtering control.

For more information, see [Blend Your Data](#) on page 369.

Use blending and joins carefully

You should restrict joins to the fewest number of tables possible. In cases where you need access to many tables in a workbook, you may want to create separate data connections with joins that are tailored to a particular worksheet.

Blending queries the data from both data sources at the level of the linking fields, and then merges the results of both queries together in memory. For this reason, best practice is to avoid data blending on dimensions with many unique values (for example, Order ID, Customer ID, or exact date/time).

Reduce query time by aggregating, then blending data

If you have multiple connections to large amounts of data that take a long time to query, consider aggregating the tables, and then blending the data on the aggregate. For more information, see [Blend on Summary Data](#) on page 388. For information about aggregating data in an extract, see [Quick Start: Aggregated Extracts](#) on page 1647.

Create a primary group

If you must blend two data sources because one contains the “fact” records and the other contains dimensional attributes, you may be able to improve performance by creating a primary group or alias. *Primary groups* are used for one-to-many relationships and *primary aliases* are used for one-to-one relationships. Note that both primary groups and aliases are not dynamic and must be updated if the data changes. Consequently, they aren’t a great solution for frequently updated data, but if you need a quick mapping, they can potentially eliminate the need for costly blends. For an example, see [Bring a Field into the Primary Data Source](#) on page 379.

Design for Performance While You Build a View

You’re connected to your data, and you’ve built your data source. The next step is to begin building a view in Tableau. There are a few things to keep in mind as you build your workbook so that it will perform faster while you’re building it, as well as after it is created.

Use the **Describe field** to get to know your data

Rather than dragging a dimension out onto the rows shelf so that you can eyeball the data—which requires Tableau to render the data as a crosstab, one of the slowest visualization types—use the **Describe field** option. Right-click the field in the **Data** pane to view a description of the data type as reported by the data source, as well as a sample of the dimension member values.

Keep workbooks a reasonable size

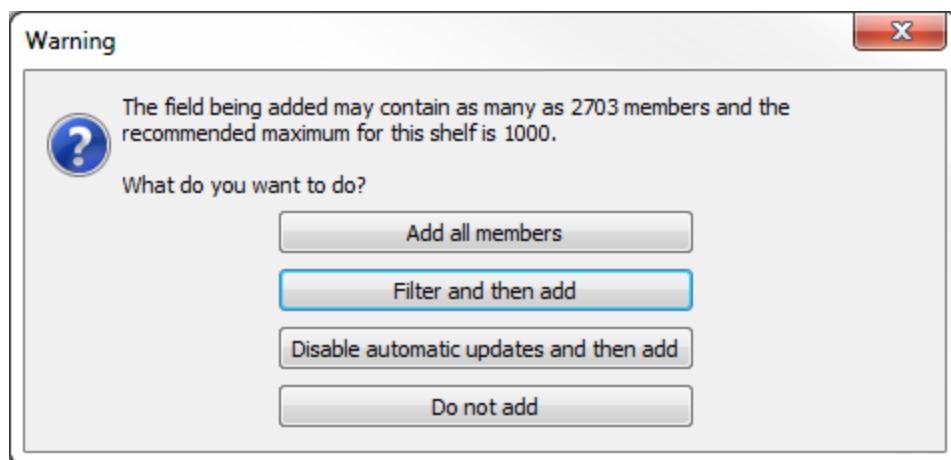
The fewer worksheets and data sources in a workbook, the faster it will perform. If you have a big topic to explore and are tempted to put everything into a single monolithic workbook, reconsider. Notice if your workbook starts to slow down with the addition of a new view and additional queries. If the performance starts to suffer as you enlarge the scope of your workbook, think about breaking your workbook into separate files.

Turn off Automatic Updates

When you place a field on a shelf, Tableau generates the view by automatically querying the data. If you are creating a dense data view, the queries might be time-consuming and significantly degrade system performance. In this case, you can turn off queries in Tableau while you build the view. You can then turn queries back on when you are ready to see the result. For more information, see [Turn off Automatic Updates to Boost Performance](#) on page 1832.

Look for warnings

Tableau displays a performance warning dialog box when you attempt to place a large dimension (with many members) on any shelf. The dialog box provides four choices, as shown below. If you choose to add all members, then you might experience a significant degradation in performance.



You might also see a warning when you attempt to create too many panes in a table. In this case, Tableau warns you that the requested table "contains more than the recommended maximum number of panes." It is best not to display more than the recommended number of panes, in part because you won't get a useful view. For more information, see [Panes on page 166](#).

Filter Your Data Carefully

Filters are an effective way to restrict the amount of data for analysis. The less data to analyze, the faster the query executes. In Tableau, there are a variety of ways to filter your data. You can create a filter on a data source, you can implement a filter when you create an extract, you can filter dimensions, measures, and dates. You can create a context filter that acts as an independent filter—then any other filters you set are dependent filters because they process only the data that passes through the context filter. You can also show a filter in a view that users can interact with. For information about when filters are executed, see [Tableau's Order of Operations on page 285](#).

Filters provide enormous flexibility, but you should use them carefully. They can be computationally expensive, so reducing the number of filters can improve workbook performance.

Filter your data in the data source

When you create a filter on a data source, you reduce the amount of data in the data source. For systems that rely heavily on partitions or indexing, data source filters may yield tremendous control over the performance of queries issued by Tableau. For more information, see [Filter Data from Data Sources on page 434](#).

Use Keep Only filters instead of Exclude filters

Because **Exclude** filters load all of the data for a dimension, they run more slowly than **Keep Only** filters. For more information, see [Select to keep or exclude data points in your view on page 673](#).

Add filters first

If you are working with a large amount of data and have automatic updates turned off, you can inadvertently create a very slow query when you add filters to the view. Rather than building the view first, and then specifying filters, specify the filters first and then drag fields to the view. That way, when you run the update or turn on automatic updates, the filters are evaluated first.

Use context filters sparingly

Think of a context filter as an independent filter; any other filters you set process only the data that passes through the context filter. Only use context filters when they limit the size of the data set significantly. If you are setting filters that significantly reduce the data set size, and that will be used for more than several data views, you should set those filters as context filters. For

more information, see [Improve View Performance with Context Filters](#) on page 694. For more information about performance improvement with context filters, see [Speed up Context Filters](#) on page 696.

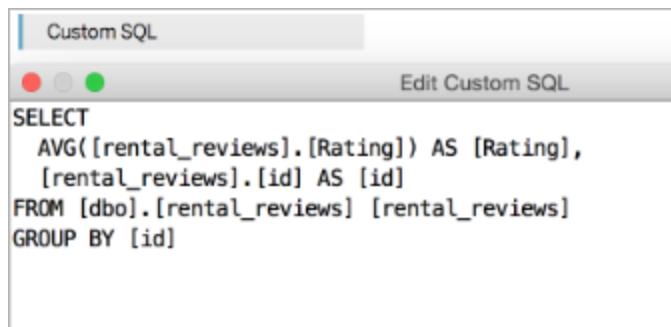
Use a set to filter data

If you want to filter a dimension to remove members based on a range of measure values, you should create a set rather than using a quantitative filter. For instance, you can create a set that only returns the Top 50 items in a dimension, rather than all of the items in a dimension.

When you create a group from a selection as described in [Creating Groups](#) on page 497, make sure you include only the columns of interest. Each additional column in the set decreases performance.

Aggregate your data using custom SQL

As an alternative to filters, another effective way to restrict the amount of data for analysis is to use custom SQL to aggregate measures before you bring the data into Tableau. Suppose your reviews data is stored in a SQL Server table, and for your analysis, you're only interested in the data from the "rating" and "reviewer ID" fields. To pre-aggregate the data so that you bring into Tableau only the fields you're interested in, you might use the following custom SQL:



The screenshot shows the 'Custom SQL' tab in the Tableau Data Source editor. The query is as follows:

```
SELECT
    AVG([rental_reviews].[Rating]) AS [Rating],
    [rental_reviews].[id] AS [id]
FROM [dbo].[rental_reviews] [rental_reviews]
GROUP BY [id]
```

This custom SQL query reduces the amount of data brought into Tableau because it pre-aggregates the reviews and averages the values in the review ratings.

Filter your cube data

Here are some specific tips that apply to filtering cube data.

Filter on a specific dimension

If your cube has a single large dimension, you should set a filter directly on that dimension rather than setting a filter on another dimension or measure. For example, suppose you want to reduce the numbers of products being displayed in a view. It is much more efficient to set the filter directly on Products or to create a computed set based on Products (such as Top 10) rather than to filter other fields such as Location or Profit.

Also, avoid selecting large numbers of members from a large dimension. When a dimension is large, keep the size of the filter to less than a thousand members.

Create a set

When creating a set from a selection as described in [Creating a Set on page 513](#), make sure you've included only the columns of interest in the Create Set dialog box. Each additional column in the group decreases performance. For example, if you create a set that contains all regions with sales between 8,000 and 15,000, but you include a column that doesn't affect the members of the set, you might notice a performance decrease. To remove extra columns, right-click the column and select **Remove This Column** from the context menu.

Don't apply sorts to levels

Avoid applying sorts to levels within a very large hierarchy in a cube.

Use the Exclude command

If you are working with a dimension whose root level is greater than 1000 but less than 100,000, avoid using the Filter dialog box to filter the data. Instead, drag the dimension to a shelf and use the **Exclude** command in the headers context menus to limit the data that is displayed in the view.

Create Efficient Calculations

When your data doesn't provide all the information you need to answer your questions, you can create calculated fields to help with your analysis.

Within a calculated field you can define a hardcoded constant (such as a tax rate), do very simple mathematical operations like subtraction or multiplication (for example, revenues minus cost), use more complex mathematical formulas, perform logical tests (IF/THEN, CASE), do type conversions, send expressions to external services such as R, and much more.

There are different calculation types in Tableau:

Basic and aggregate calculations: These types of calculations are generated as part of the query to the underlying data source and are calculated in the database. In general, basic and aggregate calculations scale very well, and there are many database tuning techniques that can improve their performance.

Table calculations: These calculations are calculated by Tableau on the query result set. While this means more work for Tableau, table calculations are generally done over a much smaller set of records than are in the original data source. If table calculation performance is a problem (possibly because the result set returned to Tableau is very large) consider pushing some aspects of the calculation back to the data source layer. One way to do this is to aggregate the data and then perform the calculation on the aggregated data.

Level of detail (LOD) expressions: LOD expressions are generated as part of the query to the underlying data source and are calculated in the database. They are expressed as a *nested*

select, so they are dependent on database performance. A table calculation or blending might perform better than a LOD expression, or vice versa.

If you suspect performance is slow due to a LOD expression, you might try replacing it with a table calculation or a data blend to see if performance improves. For an example, see Example 2 in [Tableau's Order of Operations on page 285](#).

LOD expressions can be affected by join culling, so look at [Assuming Referential Integrity on page 367](#) if your queries run slowly when you use LOD expressions.

For more information, see the Tableau whitepaper [Understanding Level of Detail \(LOD\) Expressions](#).

Booleans and integers are faster

When you create calculated fields, the data type you use has a significant impact on the calculation speed. Integers and Booleans are generally much faster than strings. If your calculation produces a binary result (for example, yes/no, pass/fail, over/under), be sure to return a Boolean result rather than a string.

Use parameters for conditional calculations

A common technique in Tableau is to show a parameter control so users can select a value that determines how a calculation is performed. Typically, to give the user easy-to-understand options, it makes sense to create the parameter as a string type. But numerical calculations are much faster than string calculations, so take advantage of the **Display As** feature of parameters: that is, show text labels but use underlying integer values for the calculation logic. For more information, see [Creating Parameters on page 989](#).

Convert date fields

Users often have date data that is not stored in native date formats—for example, a date might be a string or a numeric timestamp. You can use the DATEPARSE function if your data supports it—this function is available for non-legacy Microsoft Excel and text file connections, MySQL, Oracle, PostgreSQL, and Tableau data extract data sources. Otherwise, parse the field into a date string such as “2012-01-01”. ISO strings are preferred because they are not locale-specific. Then pass the value to the DATE function. If the originating data is a numeric field, converting it first to a string and then to a date is very inefficient. It is much better to keep the data as numeric and use DATEADD and date literal values to perform the calculation. The performance gains can be significant with large data sets. See [Date Functions on page 1565](#) for more information about these functions.

Use ELSEIF logic statements

When working with complex logic statements, remember that ELSEIF is faster than ELSE IF, because a nested IF computes a second IF statement rather than being computed as part of the first.

Aggregate measures

If the views you create are slow, make sure you are working with aggregated measures. With disaggregated data, you might be trying to view many rows of data at once. You can reduce the number of rows by aggregating the data. To do this, select **Analysis > Aggregate Measures**.

Calculation tips

There are lots of little things you can do that can improve calculation performance.

- Distinct counting values is one of the slowest aggregation types in almost all data sources. Use the COUNTD aggregation sparingly.
- Using parameters with a wide scope of impact (for example, in a custom SQL statement) can affect cache performance.
- Filtering on complex calculations can potentially cause indexes to be missed in the underlying data.
- Script functions like RAWSQL and SCRIPT_* for integrating with external services can be slow, particularly if there are lots of values that need to be passed back and forth from the DBMS/R server.
- Use NOW only if you need the time stamp level of detail. Use TODAY for date level calculations.
- Remember that all basic calculations are passed through to the underlying data—even literal calculations like label strings. If you need to create labels (for example, for column headers) and your data is very large, create a simple text/Excel file data source with just one record to hold them so they don't add overhead on the big data source.

Make Visualizations Faster

You've put in a lot of work to make your visualization—view, dashboard, or story—make a point or tell a story. You don't want to lose your audience while you wait for the visualization to load. By following the guidelines in this topic, you can improve the speed of your visualizations.

Reduce the scope

Whether you're creating a view, dashboard, or story, it's tempting to pack a lot of information into your visualization because it's so easy to add more fields and calculations to the view and more sheets to the workbook. But the result can be that the visualization becomes slower and slower to render.

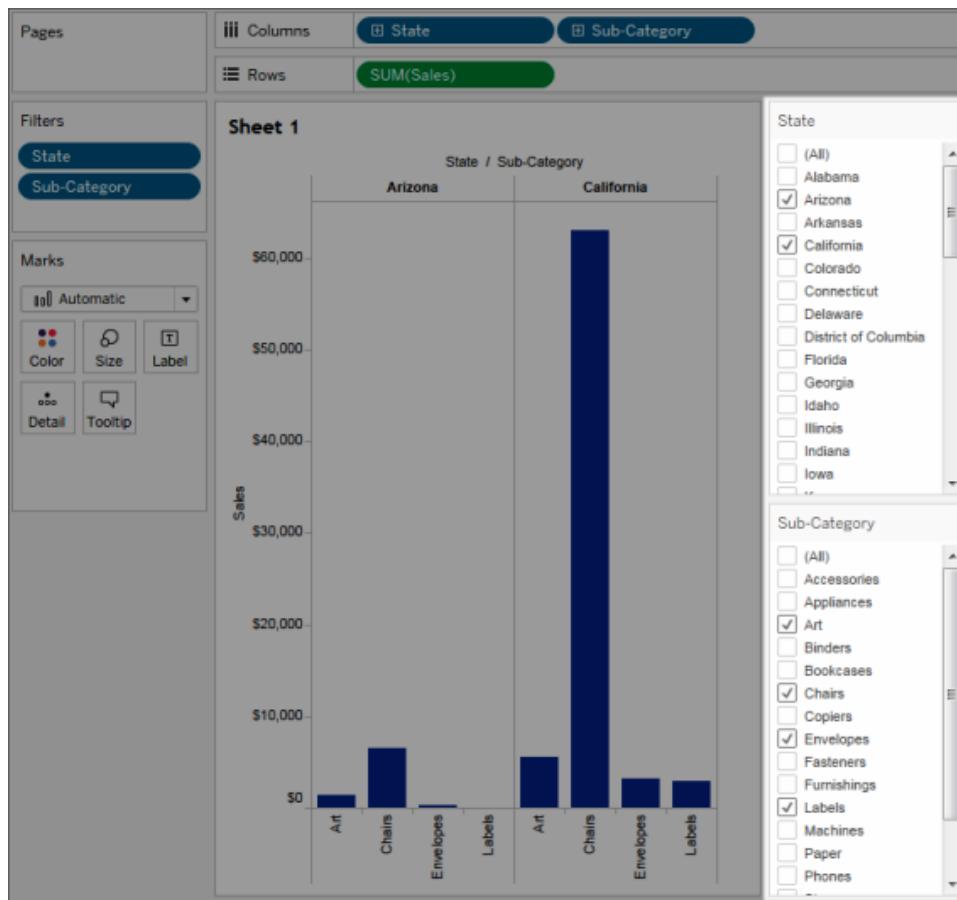
Remember that each worksheet runs one or more queries against the data, so the more sheets, the longer it takes to render the visualization.

Take advantage of the fact that Tableau is designed to deliver interactive visualizations to your users, and spread the data out across multiple visualizations, if you can. Be strategic when

designing your visualization—the fewer sheets and data sources, the faster your visualization will perform.

Limit the number of filters you show in the view

Filters that you show in your view (formerly called quick filters) are a very powerful feature of Tableau that you can use to create rich, interactive visualizations for your users. See the filters highlighted on the right in the following screenshot:



When you add an interactive filter to a view, each filter in the view requires a query in order to populate the options. If you add a lot of interactive filters to your dashboard, it can cause the dashboard to take a long time to render.

Also, when you use “show relevant values” on a filter, it requires a query to update the shown values each time other filters are changed. Use this feature sparingly.

Reduce the number of marks on the view

While there is no hard and fast rule on what defines “too many marks,” be aware that more marks means that more processing power and memory is required to render them. You can find the number of marks by looking at the lower left of the Tableau Desktop window in the

status bar. Watch out for large crosstabs and for maps with complex custom polygons. Keep in mind that too many data points on a view can also reduce the visual analytics value by causing information overload.

To avoid this problem, compile related views and connect them with action filters so that you can go from an overview to a more granular view as you explore the data. Make sure that you remove any unneeded dimensions from the **Detail** shelf. You can also try displaying your data in different types of views to see what's most effective.

Zoom without filtering

When users zoom in on a visualization with a large number of marks, Tableau doesn't filter out the marks you can't see. What changes is the view of the data, not the total number of marks. If you only need a subset of the data, filter out the data you don't need.

Record and Analyze Workbook Performance

Your workbook is done and you're wondering if its performance is taking a little longer than it should. To find out what's slowing it down, you can use a performance recording to evaluate your workbook. This is an especially good idea if you plan to share or publish the workbook.

The Performance Recording feature in Tableau records performance information about key events as you interact with a workbook. You can then view performance metrics in a workbook that Tableau creates to analyze and troubleshoot different events that are known to affect performance:

- Query execution
- Geocoding
- Connections to data sources
- Layout computations
- Extract generation
- Blending data
- Server blending (Tableau Server only)

Tableau support may ask that you create a performance workbook as they work with you to diagnose performance issues.

Create a performance recording in Tableau Desktop

To start recording performance, follow this step:

Help > Settings and Performance > Start Performance Recording

To stop recording, and then view a temporary workbook containing results from the recording session, follow this step:

Help > Settings and Performance > Stop Performance Recording

You can now view the performance workbook and begin your analysis.

If you are sending the recording to Tableau Support, save this workbook as a packaged workbook (.twbx) file, and then send it.

Interpret a performance recording workbook

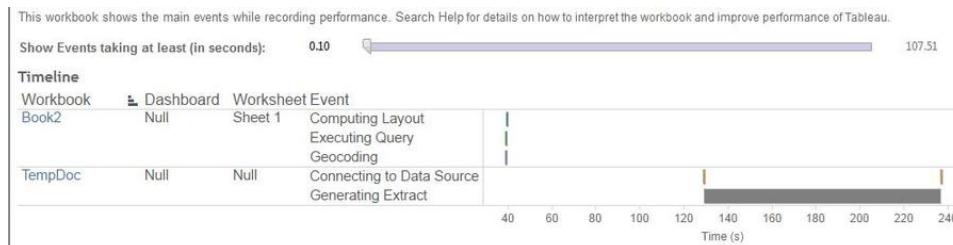
A performance recording workbook is a Tableau dashboard that contains three views:

Timeline, Events, and Query.

Timeline

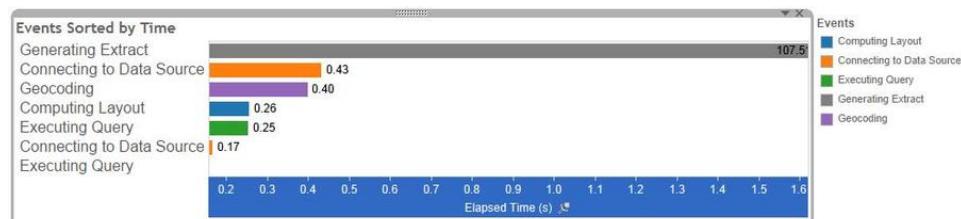
The uppermost view in a performance recording dashboard shows the events that occurred during recording, arranged chronologically from left to right. The bottom axis shows elapsed time since Tableau started, in seconds.

In the Timeline view, the **Workbook**, **Dashboard**, and **Worksheet** columns identify the context for events. The **Event** column identifies the nature of the event, and the final column shows each event's duration and how it compares chronologically to other recorded events:



Events

The middle view in a performance recording workbook shows the events, sorted by duration (greatest to least). Events with longer durations can help you identify where to look first if you want to speed up your workbook.



Different colors indicate different types of events. The range of events that can be recorded is:

- Computing layouts

If layouts are taking too long, consider simplifying your workbook.

- Connecting to data source

Slow connections could be due to network issues or issues with the database server.
- Executing query
 - For live connections, if queries are taking too long, it could be because the underlying data structure isn't optimized for Tableau. Consult your database server's documentation. As an alternative, consider using an extract to speed performance.
 - For extracts, if queries are taking too long, review your use of filters. If you have a lot of filters, would a context filter make more sense? If you have a dashboard that uses filters, consider using action filters, which can help with performance.
- Generating extract

To speed up extract generation, consider only importing some data from the original data source. For example, you can filter on specific data fields, or create a sample based on a specified number of rows or percentage of the data.
- Geocoding

To speed up geocoding performance, try using less data or filtering out data.
- Blending data

To speed up data blending, try using less data or filtering out data.
- Server rendering

You can speed up server rendering by running additional VizQL Server processes on additional machines.

Query

If you click on an **Executing Query** event in either the **Timeline** or **Events** section of a performance recording dashboard , the text for that query is displayed in the Query section. For example:

Query

```
SELECT "State"."ID" AS "ID",
       "StateSynonyms"."Name" AS "State_Name",
       "State"."ParentID" AS "State_ParentID"
  FROM "StateSynonyms"
 INNER JOIN "State" ON (( "State"."ID" = "StateSynonyms"."ParentID") AND ( "State"."MapCode" = "StateSynonyms"."MapCode")
```

If it makes sense, you can use the query text to work with your database team on optimizing at the database level. Sometimes the query is truncated and you'll need to look in the Tableau log to find the full query. Most database servers can give you advice about how to optimize a query by adding indexes or other techniques. See your database server documentation for details.

Sometimes for efficiency, Tableau combines multiple queries into a single query against the data. In this case, you may see an **Executing Query** event for the Null worksheet and zero queries being executed for your named worksheets.

Reduce Upload Times to Tableau Server

You've created your workbook and analyzed its performance recording to make sure that it's been optimized for speed. Now you're ready to publish the workbook to Tableau Server. It isn't always possible to reduce upload times to Tableau Server, but here are some considerations to keep in mind.

Avoid publishing packaged workbooks

It is a best practice to publish workbooks (.twb files) and data sources separately, rather than publishing them together as packaged workbook (.twbx files). This may not make your initial uploads of workbooks and data sources any faster than publishing them together, but it makes republishing workbooks faster (if you don't have to also republish data sources), and it makes data sources available online for new workbooks. For data sources that are extracts, it also allows you to schedule automated refreshes.

Make extracts smaller

For Tableau extracts, you can help speed uploads by making your extract smaller. As you create extracts, consider:

- Removing unused fields from extracts.
- Making extracts smaller with sampling and filtering.
- Aggregating data for visible dimensions.

See also

[Extract Your Data](#) on page 440

[Publish a Data Source](#) on page 1234

Managing Queries

Queries are automatically generated every time you add a field to a shelf and interact with the view. Tableau offers several ways you can manage these queries once they are sent to the underlying data.

Turn off Automatic Updates to Boost Performance

When you place a field on a shelf, Tableau generates the resulting view by querying the data source. When you create a dense data view that involves many fields, these queries can be time-consuming. You can instruct Tableau to turn off automatic updates to improve performance.

Automatic updates for worksheets

By default, automatic updates are turned on and the toolbar button is highlighted . However, it is sometimes more efficient for Tableau to execute the queries you need only for your final view, rather than for every intermediate step required to compose that view. You can turn off updates for worksheets by clicking the **Pause Auto Updates** toolbar button.

You can also turn automatic updates on and off by pressing F10 (Option-Command-0 on a Mac) on your keyboard.

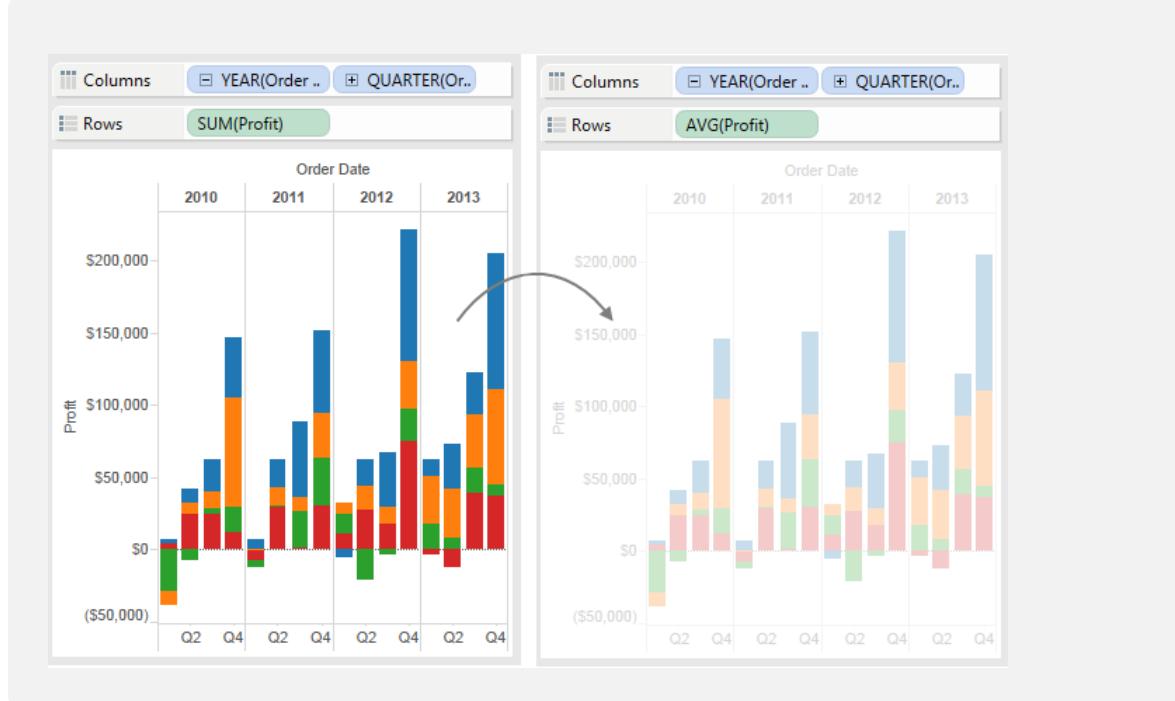
You can update the view at any time while automatic updates are off by pressing F9 (Shift-

Command-0 on a Mac) or clicking the **Run Update**  button on the toolbar.

Note: It is possible to enter an invalid state when automatic updates are turned off.

When this happens, the view is desaturated and invalid commands are disabled. The view and commands become available again when you click **Run Update** on the toolbar.

For example, the view below has automatic updates turned off. When the aggregation for Profit is changed from a summation to an average, the view is desaturated to let you know that you have made a change to the view that has made the current view invalid.



Automatic updates for filters

When you change a filter, even when you turn off automatic updates for worksheets, Tableau continues to query the data source to update the view.

When you work with a large data set, this continuous query action can degrade performance, so Tableau also gives you an option to pause automatic updates for filters.

If you pause automatic updates for filters, you can make as many changes as you need without having to wait for Tableau to update the filter each time.

To pause automatic updates for filters, click the drop-down arrow on the **Pause Auto Updates** button on the toolbar and select **Auto Update Filters** to clear the check mark .

To resume automatic updates for filters, click the drop-down arrow on the **Pause Auto Updates** button on the toolbar and select **Auto Update Filters** again. You can also click the

Run Update  button on the toolbar to manually update the view at any time.

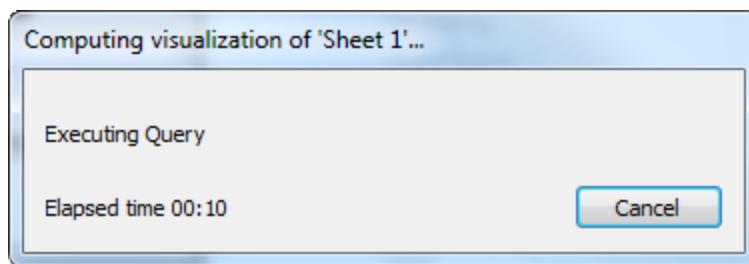
How Automatic Updates Affect Dashboards and Stories

You configure automatic updates on a per-view basis. This means that you can have a dashboard where some views update automatically, and others don't. Similarly, you can have a story where some story points update automatically, and others don't. But when dashboards or stories are published to Tableau Server, automatic updates affect either none of the contributing views in a story or dashboard, or all of the views.

Cancel Query

This command is used any time you want to stop a query that is in process. You may want to cancel a query that is taking a long time to complete due to the size of the data source. When a query is taking a long time to complete, a progress dialog box opens.

Click **Cancel** in the Processing Request dialog box.



After canceling a query the view becomes invalid because it is in an in-between state. The result is a blank view although all your fields are still on the shelves. To resume working with Tableau, alter the view in anyway and allow the query to complete.

Cancelling a large number of queries can result in performance degradation in the underlying database. Although the query has been abandoned by Tableau, it is still executing on the database.

Abandoned Queries

When you cancel a query in Tableau, the database is told to stop processing the query. However, some databases do not support cancel (MS Excel, MS Access, Essbase, Microsoft Analysis Services 2000). If you cancel a query using one of these types of data sources, the query is abandoned by Tableau but is still running in the background and using resources. When you have abandoned queries, an indicator appears in the bottom right corner of the workbook showing the number of queries still running  . As queries in the background complete, the number will go down. It is important to monitor the number of queries running and not let the number get too high, otherwise you will see performance degradation of both Tableau and the underlying database.

Note: Text, Microsoft Excel, and Microsoft Access data sources may be temporarily unavailable after canceling a query because of a lock performed internally. You may have to wait until the abandoned query has completed before re-connecting.

Precision Warnings

When you add a field to a view that contains values with more precision than Tableau can model, a warning icon  is displayed in the bottom right corner of the status bar. For example, a value in the database may have 22 decimal places but Tableau only supports up to 15. When you add that field to the view, you get a precision warning. If you click on the warning, you can read more details including the number of decimal places that have been truncated in the view. Remember that the precision of the data displayed in Tableau will always first be dependent on the data in your database. If the values in your database exceed 15 decimal places, when you add them to the view, the value is truncated and a precision warning appears.

Copyright

© 2016 Tableau Software, Incorporated and its licensors. All rights reserved.

Additional information regarding Tableau's copyrights, patents and trademarks can be found at www.tableau.com/ip.

For information about Third Party Code which may be distributed with the Software, please review the Tableau Software End User License Agreement.

Contact Us

Directory of worldwide offices

Sales

Contact

Support

Search our [support resources](#).