Envision

Product Introduction

Akana Envision

Product Introduction
May 2015

Copyright

Trademarks

Akana, Inc.

Akana, Inc.
12100 Wilshire Blvd, Suite 1800
Los Angeles, CA 90025
(866) SOA-9876
www.akana.com
info@kana.com

Disclaimer

Contents

Chapter 1 | Product Overview

Envision is an analytics platform that allows you to slice and dice your data to analyze every aspect of your business and digital strategy. It can help you collect data and identify patterns, trends, violations, baselines, or historical context. The domain of the data you wish to analyze is open ended. You may wish to analyze:

API operations such as:

- API usage patterns by apps, by geographic region, users, etc.
- API performance patterns and trends by examining load, throughput, and latency
- SLA violations

Business specific transactions such as:

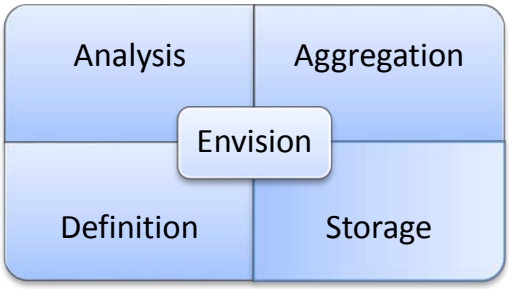
- Sales trends based on model in different regions
- Channel partner effectiveness by examining sales volumes
- Effectiveness of sales mediums
- Effectiveness of different apps

Or business operations such as:

- API/microservice lifecycle and bottlenecks
- Time to market trends for client apps
- Popularity of APIs
- Customer satisfaction trends

The Envision product does not require any prior understanding of what the data it analyzes represents. It does not require any business understanding. It can analyze data from any business domain because it works using general analytical concepts.

Envision's capabilities can be broken down into four categories, data definition, data enrichment & storage, data aggregation, and data analysis.



The foundation for analytics is understanding and defining what data you have to work with and what you want to do with that data. This is what is done with the data definition capabilities of Envision.

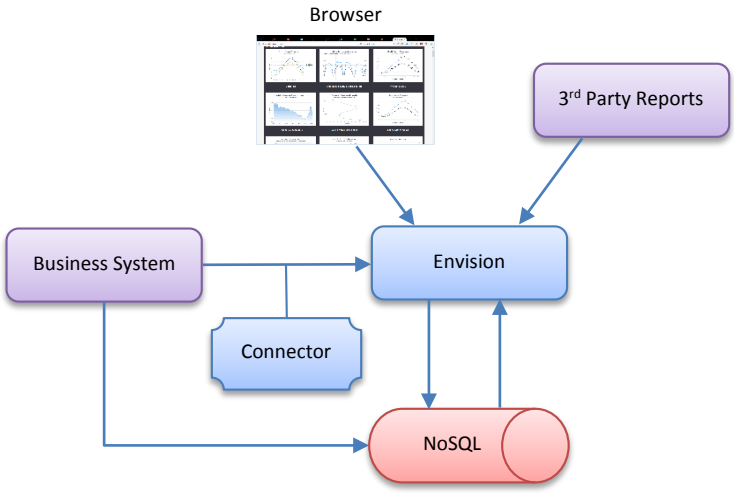
Then the data must be collected. Envision at its core does not reach out to different business systems to collect data. Instead it provides a data storage API for clients to push the data into a highly scalable Envision data store. Connectors are then provided that are domain specific to extract data from different business systems and use the Envision data storage API. One such connector is provided by Akana Policy Manager which enables the extraction of business specific data from payload and stores it for future analysis. The use of the Envision data storage API is optional. Alternatively data can be saved directly to the Envision data store if access is permitted.

Envision also supports data enrichment during the storage process. This is the amending, or altering, of data before it is persisted based on information obtained from a different source. For example the business system might be just collecting the user-agent field but for proper charting the corresponding browser/OS/device may be what is persisted instead. Enrichment is performed by plugins invoked by the data storage API prior to persisting the data or by scheduled jobs to manipulate data not persisted through the API.

Based on the data definition raw business data must be transformed into metrics that can be compared, charted, and trended. This is done by organizing, or grouping, the data in ways that can be queried with the proper granularity and purpose, and performing aggregating calculations on the data according to those groupings. For example calculating the average sales figures of each channel partner on a quarterly basis. This is what the Envision data aggregation capabilities provide.

Finally users of Envision want to view the aggregated data in ways that easily depict the trends, patterns, and violations they are looking for. Envision provides analysis capabilities for the user to do this. Envision does this through a highly tailorable charting and dashboard generation portal. It can also support the extraction of data through the use of an API so that third party visualization tools can be used.

The following illustration shows how Envision would interact with users and other systems.



Envision stores all data in a NoSQL database. Connectors extract data from business systems and call Envision API's that store the data in the NoSQL database. Envision performs the needed aggregations on the data in the NoSQL database. The data is then available for viewing in a browser using the Envision web application in user designed dashboards and charts. Alternatively a Third Party reporting product can query the data from Envision.

Chapter 2 | Introduction to Data Sets

A data set is a group of metrics that are collected so that they can be analyzed through queries based on different criteria, or dimensions. It is the foundation on top of which all analysis can be performed. Take a company named Acme that manufactures shoes, for example. Acme provides an API for merchants to order shoes that are sold to the public. Acme wishes to collect metrics on those sales such as:

- How many orders by each merchant
- Number of items ordered by style
- Number of items ordered total
- Total sales by each merchant
- Total sales by each style
- Total sale of all orders
- Fees collected by each merchant
- Total fees collected

Acme wants this information reported on a weekly and monthly basis. This group of metrics becomes the basis for a data set.

The data set consists of metrics and dimensions. The metrics themselves are simply numeric values, such as a price or quantity. The metrics are collected over one or more time intervals (i.e., one minute, one hour, and one day). Because multiple values for a metric will likely be gathered over a time interval, a calculation must be defined for combining the multiple values to a single value. This is called aggregation. Some examples are calculating an average, a sum, or recording the minimum or maximum values. The metrics are then partitioned, or grouped, by organization properties of the transactions called *dimensions*, such as merchant or style.

In Acme's case we have a data set named "Shoe Sales." The metrics in the data set are:

- Number of Orders (sum)
- Number Of Items Sold (sum)
- Sales Amount (sum)
- Fees Paid (sum)

For both metrics, they will be summed over the collection interval. The dimensions in the data set are:

- Merchant
- Style
- Order Date

Note there is no dimension for Order. This is because collecting metrics per order, or by transaction is the empty dimension. It is the single grouping of metrics without a dimension and is always collected. The collection intervals in the data set are:

- Week

- Month

By defining the data set in this way each week the number of orders made, number of items sold, sales amount, and fees paid, will be calculated. The numbers will be calculated by creating their sum. There will be a separate sum for each combination of merchant and style, and a sum independent of any dimension. The following table lists a set of orders that will be used to illustrate Shoe Sales Data Set.

Order Number	Date	Merchant	# Items	Style	Price	Trans Fee
1001	5/11	Walmart	100	High Top	1000	20
1002	5/11	Costco	50	High Top	500	10
1003	5/11	Costco	50	Mid Top	500	10
1004	5/11	Target	100	Low Top	1000	20
1005	5/12	Walmart	50	High Top	500	10
1006	5/12	Walmart	50	Low Top	500	10
1007	5/13	Costco	50	Low Top	500	10
1008	5/13	Target	100	Low Top	1000	20
1009	5/14	Walmart	100	High Top	1000	20
1010	5/15	Walmart	50	Low Top	500	10

For the week of 5/11 the data set will collect the following sums.

Merchant	Style	# Orders	# Items	Sales Amount	Fees Paid
Walmart	High Top	3	250	2500	50
Walmart	Low Top	2	100	1000	20
Costco	High Top	1	50	500	10
Costco	Mid Top	1	50	500	10
Costco	Low Top	1	50	500	10
Target	Low Top	2	100	2000	40

With the metrics and these combinations of dimensions the original requested queries can be performed:

- **Orders made by Merchant** – Sum the # Orders column for each unique Merchant: Walmart 5, Costco 3, Target 2
- **Items Sold by Merchant** – Sum the # Items column for each unique Merchant: Walmart 350, Costco 150, Target 100
- **Items Sold by Style** – Sum the # Items column for each unique Style: High Top 300, Mid Top 50, Low Top 250
- **Total Items Sold** – Sum the # Items column: 600
- **Total Sales by Merchant** – Sum the Total Sale column for each unique Merchant: Walmart 3500, Costco 1500, Target 2000
- **Total Sales by Style** – Sum the Total Sale column for each unique Style: High Top 4, Mid Top 1, Low Top 2500
- **Total Sales of all Orders** – Sum the Total Sale column: 6000
- **Fees Collected by Merchant** – Sum the Trans Fee column for each unique Merchant: Walmart 70, Costco 30, Target, 40
- **Total Fees of all Orders** – Sum the Trans Fee column: 140

Now let's say that Acme changes their ordering system so that multiple styles of shoes can be ordered at the same time by a Merchant and each order has only one transaction fee. The orders would then look like:

Order Number	Date	Merchant	# Items	Style	Price	Trans Fee
1001	5/11	Walmart	100	High Top	1000	20
1002	5/11	Costco	50	High Top	500	20
			50	Mid Top	500	
1003	5/11	Target	100	Low Top	1000	20
1004	5/12	Walmart	50	High Top	500	20
			50	Low Top	500	
1005	5/13	Costco	50	Low Top	500	10
1006	5/13	Target	100	Low Top	1000	20
1007	5/14	Walmart	100	High Top	1000	20
1008	5/15	Walmart	50	Low Top	500	10

The system can no longer simply create rows of data from unique combinations of the dimensions because if that were to be done the metrics for Number of Orders and Transaction Fee would be summed incorrectly. For example orders 1002 and 1004 would be counted twice for calculating the number of orders made and the transaction fees would be counted twice as well. To accommodate the differences between metrics general to a single transaction and metrics for multiple line items of a transaction a hierarchy of grouping is performed. The table below shows how the sums would be calculated for this new ordering system.

Merchant	# Orders	Fees Paid	Style	# Items	Sales Amount
Walmart	4	70	High Top	250	2500
			Low Top	100	1000
Costco	2	30	High Top	50	500
			Mid Top	50	500
			Low Top	50	500
Target	2	40	Low Top	100	2000

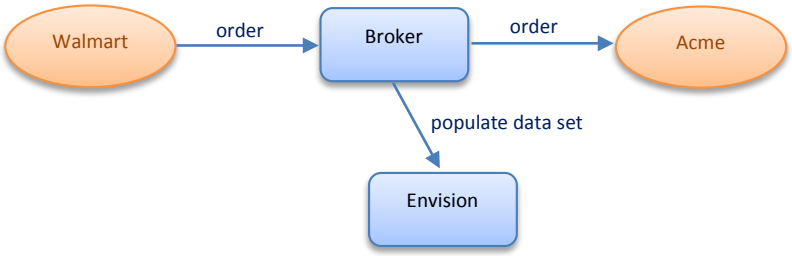
The queries that can be performed against the sums for the new ordering system are all the same as the old ordering system. The data set is defined the same way. The system just performs a different level of grouping. The columns can still be summed the same way to perform the queries, there just may be less rows for the transaction level metrics to sum.

Chapter 3 | Data Set Collection and Aggregation

In the previous chapter we introduced the concept of a data set, how metrics are defined, their aggregation, and collection intervals. In this chapter we will describe how the data in a data set is actually collected and how the aggregation is performed.

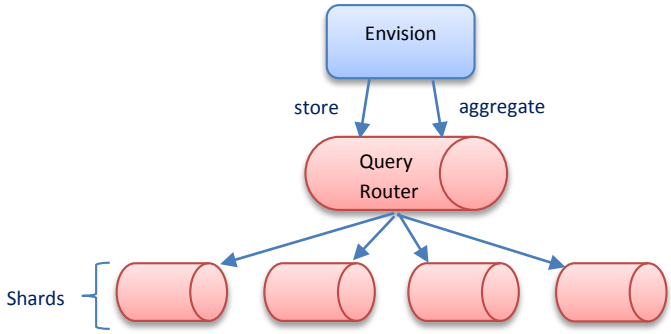
Envision on its own does not collect data. It only aggregates data based on the data set definition. It does, however, provide an API that other systems and products can use to report the data they have collected that belong to a data set. Envision also provides an API for defining the data set itself so a system or product can create a data set in Envision and then start reporting data that belongs to that data set.

Other Akana products, when integrated with Envision, will install their own data sets and will collect data and feed it to Envision. The Akana API Gateway product will allow users to define metrics collection policies that are configured to extract data from business transactions that are processed by the API Broker. The policy defines instructions to the broker on how to extract information from a transaction and map it to a metric or dimension of a data set in Envision.

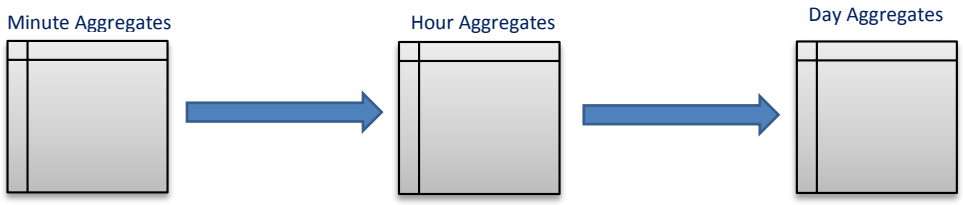


The Envision API will accept data from individual transactions (“raw” data) or pre-aggregated data. The API will persist the data in a MongoDB document data store. From here Envision will instruct the data store to aggregate the stored data on intervals that match those defined in the data set. For example, if the data set is defined to provide daily and weekly metrics then Envision will instruct the data store to aggregate the metrics on a daily and weekly basis.

The calculations performed by the data store to aggregate the metrics are dictated by Envision. Envision will consult the data set definition and instruct the data store to perform the correct aggregation calculations. If the data store is clustered, or sharded, these calculations can span multiple instances providing the scalability that may be required for large volumes of data.



Envision has predefined time intervals that it supports for aggregating data, minute, hour, day, week, month, and year. Any and all of these can be used for a particular data set. All smaller intervals are used in the aggregation of the larger intervals, so aggregation may be performed more often than what is defined in the data set. For example, let's say the data set is defined to collect metrics on a daily basis. Envision will first aggregate the data on a minute basis. Then it will use the output of that aggregation over the course of an hour to create an hourly aggregation. It will then use the output of the hourly aggregations over the course of a day to create the daily aggregation.

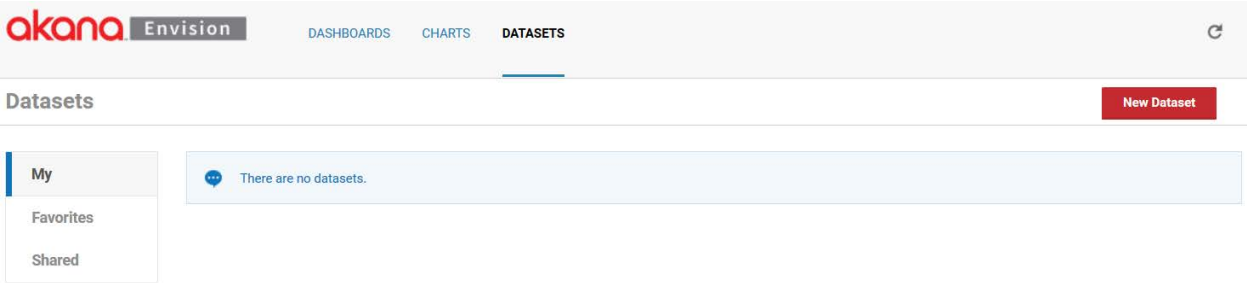


When defining a data set you can specify how long the aggregate numbers should be kept in the data store. If a smaller aggregate interval is not part of the data set but needed for a larger aggregate interval all the smaller aggregate data will be deleted once it is used for the larger aggregate calculation. In the example above if only daily numbers are required, hourly aggregate data will be deleted as soon as the daily numbers are calculated. Minute aggregate data is deleted as soon as hourly aggregate data is calculated.

Chapter 4 | Creating a Data Set

In the previous chapters we described the concept of a data set. In this chapter we will describe how a data set is created in the Envision product.

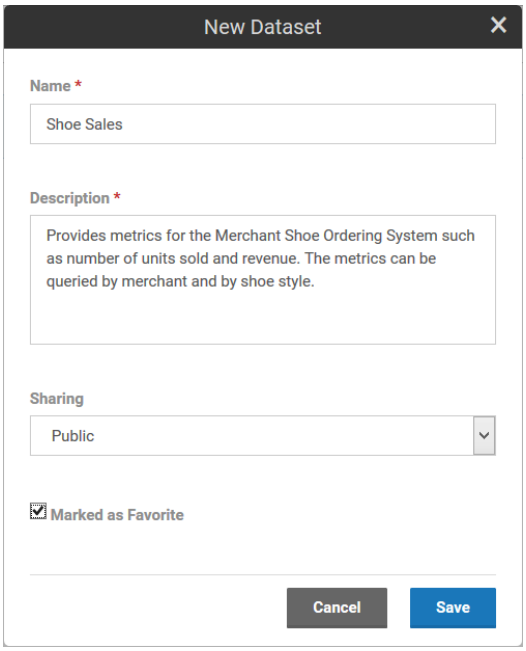
When a user logs into Envision they are presented with a top level menu consisting of *DASHBOARDS*, *CHARTS*, and *DATASETS*. Select the **DATASETS** menu item.



You are then presented three lists of data sets, *My*, *Favorites*, and *Shared*.

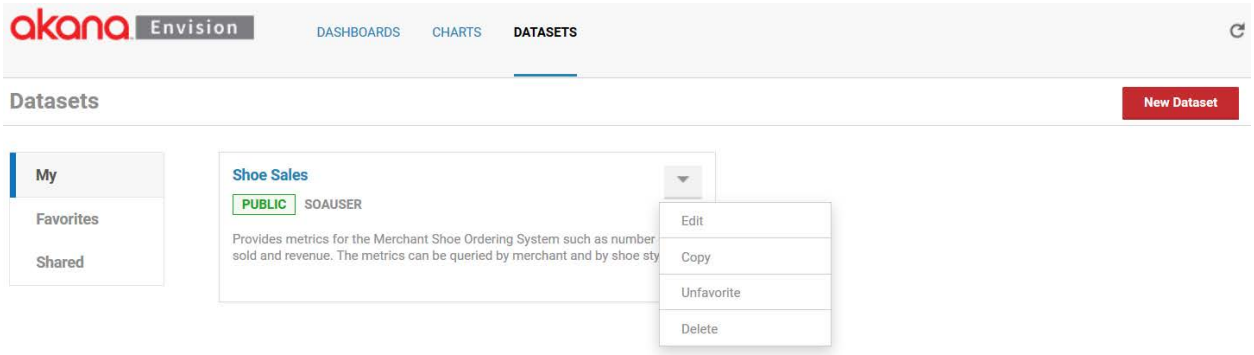
- **My** data sets are the data sets that the logged in user has created.
- **Shared** data sets are data sets that others have created and made available to the user.
- **Favorites** data sets are those selected by the user to (either their own or someone else's) to be in the list. It provides a faster lookup of commonly used items by giving them their own list.

Select **New Dataset** to start the process of creating a new data set. A pop-up will be displayed where you can enter a name, a description, and designate how it will be shared.



A data set can be shared other users. When a data set is shared with a user that user can view the data set and build charts against it. They cannot change the data set. A data set can be shared with nobody, everybody, or individual users and/or groups. The data set will automatically be placed in the *My* data sets list. If the **Marked as Favorite** checkbox is checked it will also be placed in the *Favorites* data sets list.

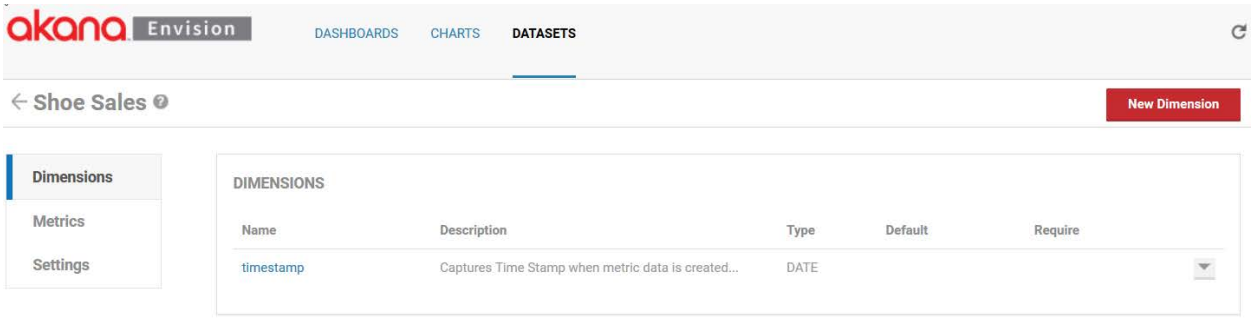
Each data set card in a list has a pull-down menu of options that can be performed on the data set.



- **Edit** – Displays the pop-up used to create the data set initially so that changes can be made.
- **Copy** - Creates a copy of the data set. The same pop-up will be displayed once again but this time it will before a new data set and all the information from the copied data set is filled in.
- **Favorite /Unfavorite** - Can be used to toggle whether the data set should be placed in the *Favorites* list.
- **Delete** - Removes the data set from the system. If any charts exist that use the data set an error will be displayed to avoid breaking any dependent charts.

To define the details of a data set such as metrics and dimensions, select the name of the data set on the data set card. The data set details page will be displayed. The page is divided into three sections, *Dimensions*, *Metrics*, and *Settings*.

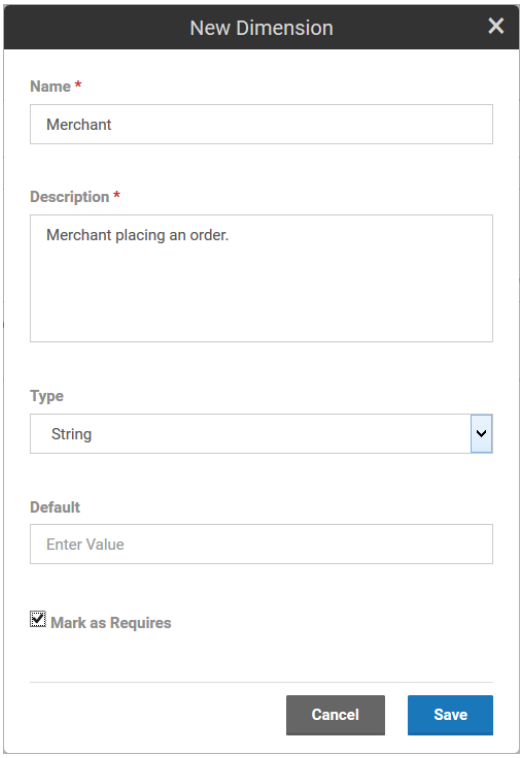
The *Dimensions* section lists all the dimensions for the data set. Dimensions are the properties of a data set that can be used to query, or organized, metrics by and provide the basis for how many combinations of aggregations will be made.



Each data set is created initially with a single default dimension, *timestamp*. The timestamp dimension will group all metrics collected with a timestamp within the same collection time interval. The dimension

can be deleted if you are not interested in time based aggregations but there must be at least one Date dimension for aggregation over time to function properly.

To add another dimension select **New Dimension**. A pop-up displays where you can enter a name, a description, a type, a default value (if any), and if the dimension is required or not.

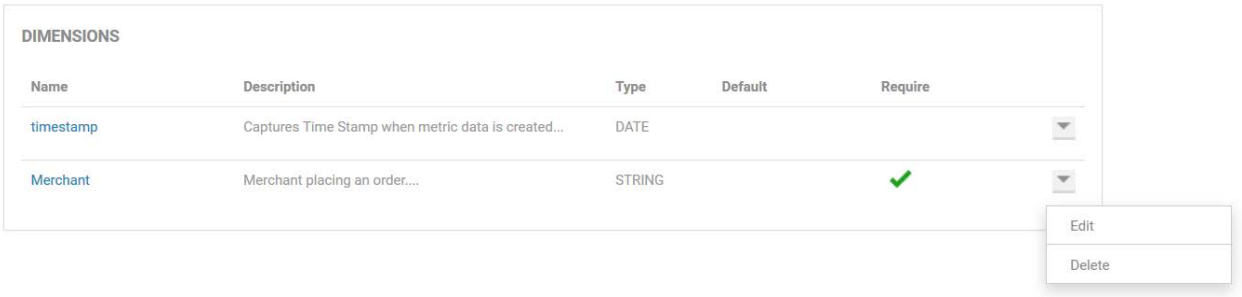


Note: At least one Dimension must be defined per Dataset

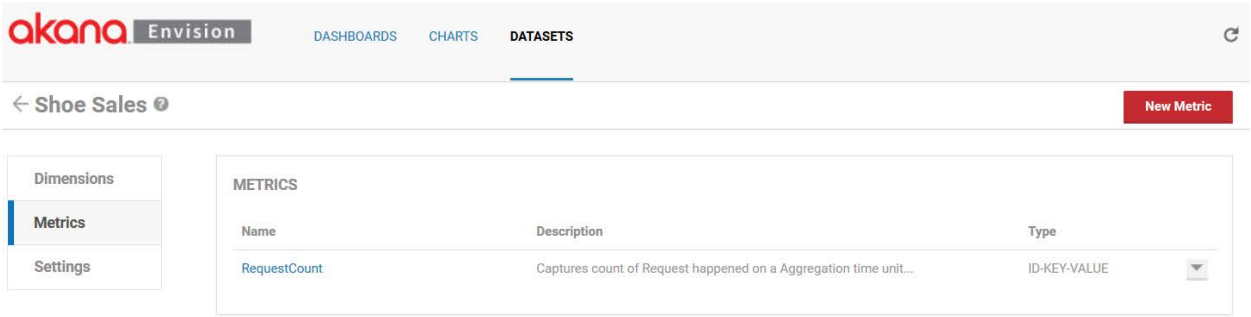
Dimensions can be one of the following types: Date, String, ID-Name, IP-Address, and Location. The ID-Name type is a dual valued property with both an ID and a name. For example, if integrating with Policy Manager a dimension may be an organization. An organization in Policy Manager has an ID and a descriptive name. The engine will group metrics using the organization's ID. But when displaying choices of organizations in the Envision UI the organization names will be displayed to the user since they won't likely know the IDs. The Location type is a property that can reference a city, state, zip code, country, or geolocation (latitude/longitude).

If a Default is specified, any data set row that does not have the dimension will have the dimension added with the default value. The default value must be of the same type as the dimension. If the dimension is classified as *required* any collector of data MUST include the dimension or the data will not aggregate correctly and charts will likely not work well.

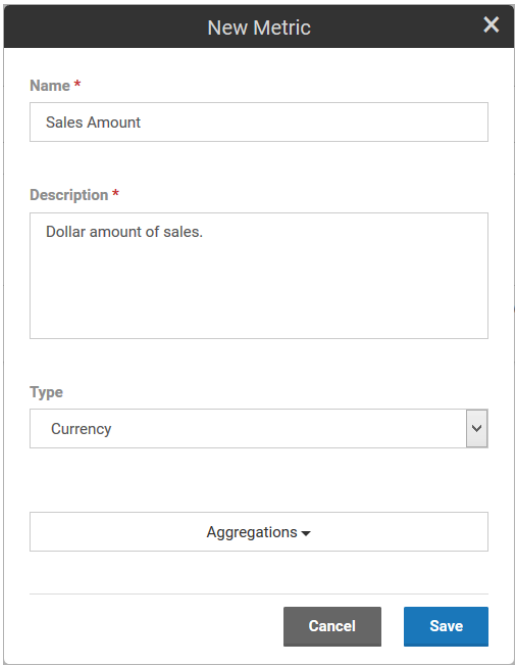
Each dimension in the list has a pull-down menu of options that can be performed on the dimension.



- **Edit** – Displays the pop-up used to create the dimension initially so that changes can be made (excluding the name which cannot be changed).
- **Delete** - Removes the dimension from the data set.
- **Metrics** - Lists all the metrics for the data set. Metrics are the properties of a data set that can be measured, aggregated, and compared.

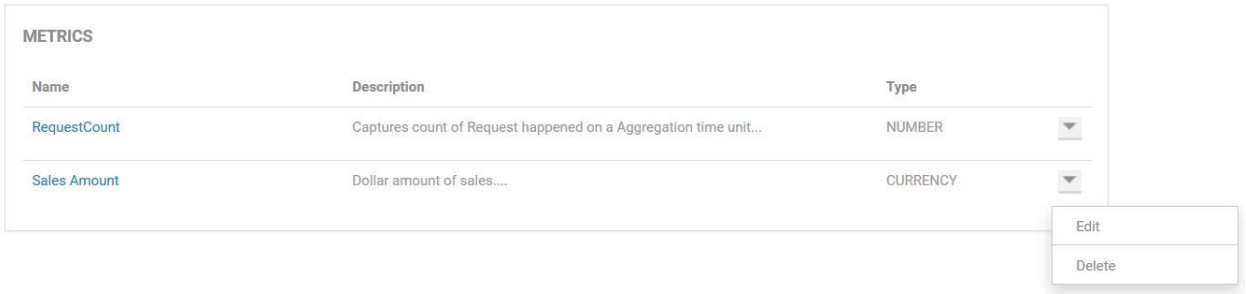


Each data set is created initially with a single default metric, **RequestCount**. The RequestCount metric is the measure for the number of transactions, or orders in this example. The metric can be deleted if you do not wish to collect it. To add another metric select **New Metric**. A pop-up displays where you can enter a name, a description, a type, and a set of aggregation calculations to perform.

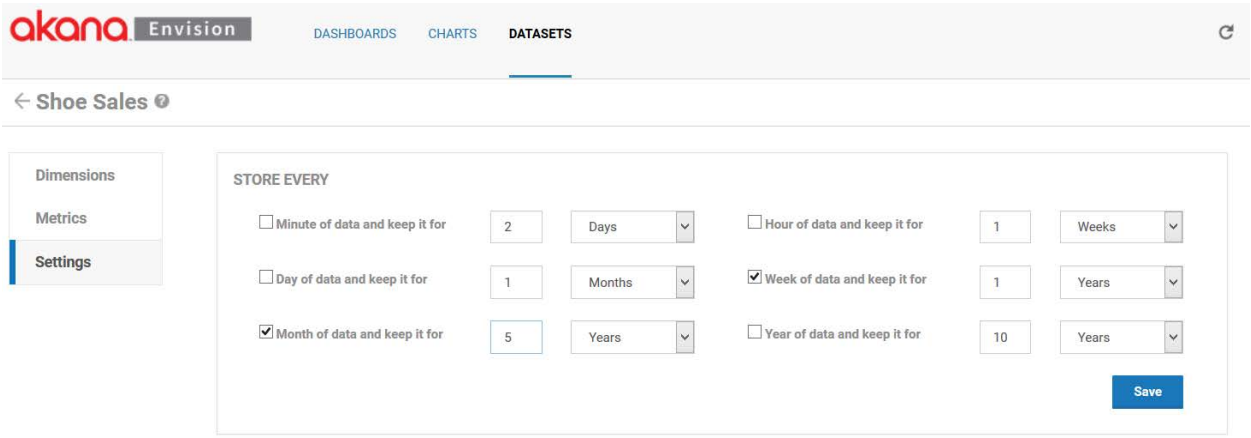


Metrics can be one of the following types: Count, Time, Size, Number, and Currency. The aggregation choices are average, sum, minimum, maximum, first, and last.

Each metric in the list has a pull-down menu of options that can be performed on the metric.



- **Edit** – Displays the pop-up used to create the metric initially so that changes can be made (excluding the name which cannot be changed).
- **Delete** – Removes the metric from the data set.
- The *Settings* section lists all the aggregation and storage intervals for the data set.

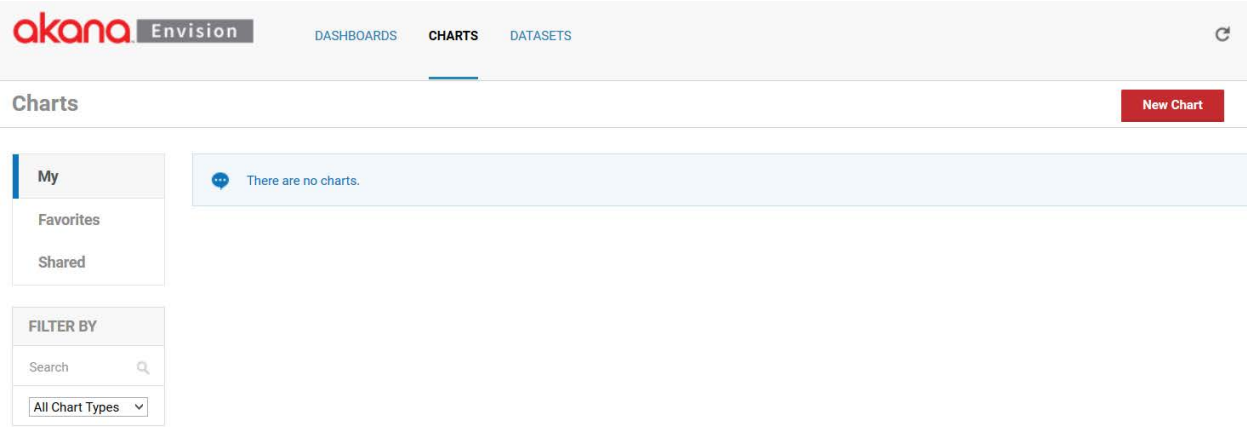


Envision supports aggregating metrics every minute, hour, day, week, month, and/or year. For each of these aggregation sets you can also specify how long the results should be held in the data store. This is done by selecting the a unit of time (minutes, hours, days, weeks, months, years) and a number of units. For example in the image above the metrics will be aggregated on a weekly and monthly basis. The weekly results will be held in the data store for 1 year and the monthly results will be held in the data store for 5 years. It is important to think about how these intervals will limit the ability to make charts. It is not possible to create a chart with data points more granular than the aggregations calculated. Once again in our example you could not create a chart that shows metrics for shoe orders per minute or day.

Chapter 5 | Creating a Chart

Once a data set has been created a Chart can be created that will display the metrics collected for the data set. There are several kinds of Charts that are available in Envision. This chapter will describe the common concepts of all charts. The options specific to each chart and examples of their use can be found in Appendix A.

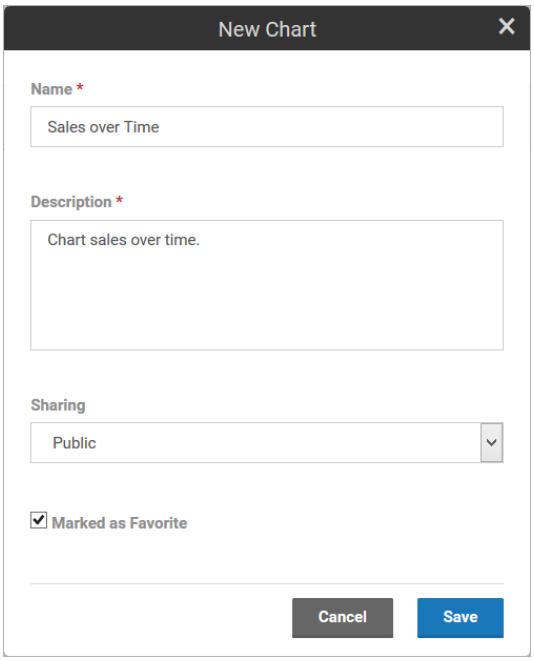
From the Envision top level menu select the *Charts* menu item.



You are then presented three lists of charts:

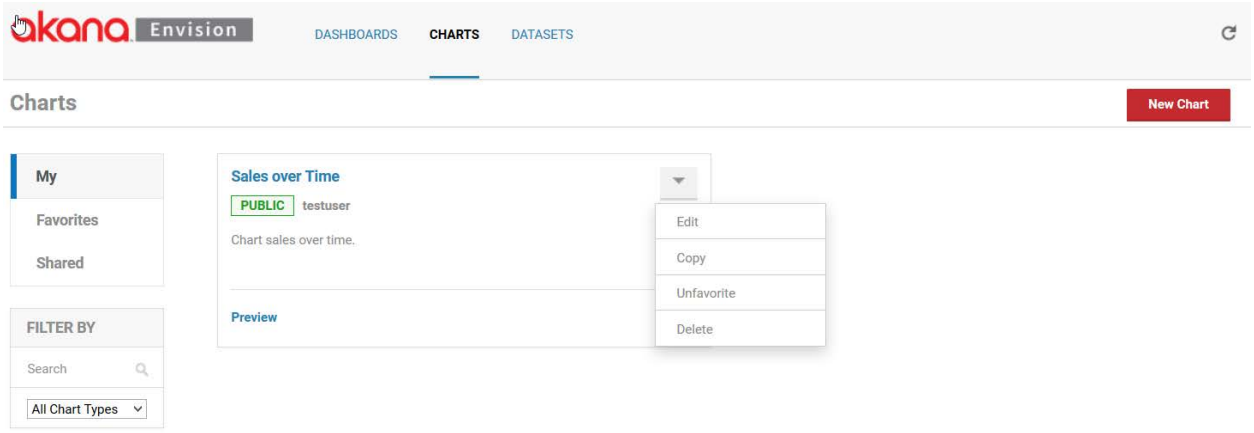
- **My** - Charts that the logged in user has created.
- **Shared** - Charts that others have created and made available to the user.
- **Favorites** - Those selected by the user to (either their own or someone else's) to be in the list. This option provides a faster lookup of commonly used items by giving them their own list.

Select **New Chart** to start the process of creating a new chart. A pop-up displays where you can enter a name, a description, and designate how it will be shared.



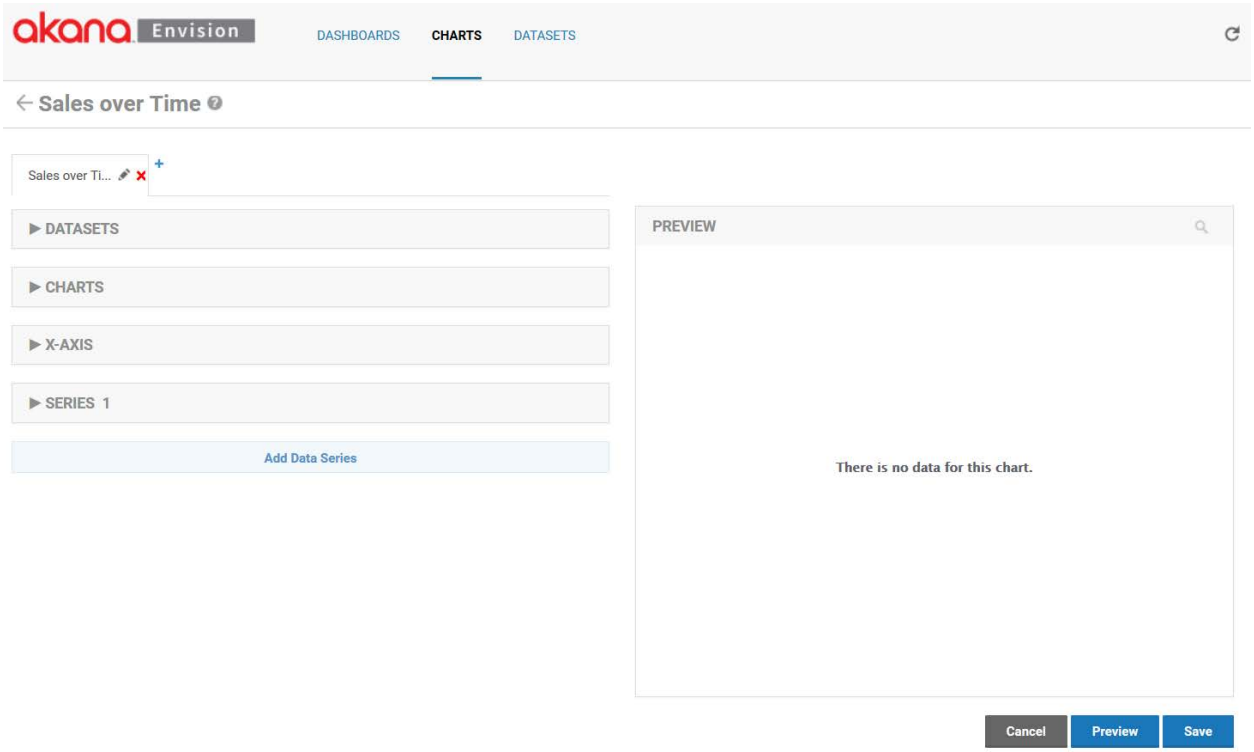
A chart can be shared other users. When a chart is shared with a user that user can view the chart and place them on dashboards. They cannot change the chart. A chart can be shared with nobody, everybody, or individual users and/or groups. The chart will automatically be placed in the *My* charts list. If the **Marked as Favorite** checkbox is checked it will also be placed in the *Favorites* charts list.

Each chart card in a list has a pull-down menu of options that can be performed on the chart.

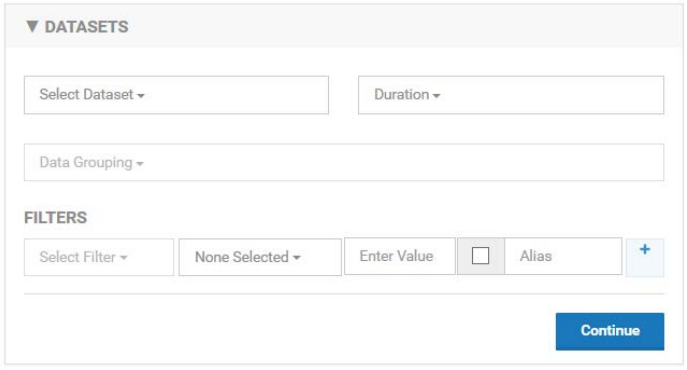


- **Edit** - Displays the pop-up used to create the chart initially so that changes can be made.
- **Copy** - Create a copy of the chart. The same pop-up will be displayed once again but this time it will be for a new chart and all the information from the copied chart will be filled in.
- **Favorite / Unfavorite** - Used to toggle whether the chart should be placed in the *Favorites* list.
- **Delete** - Remove the chart from the system. If any dashboards exist that use the chart a warning will be displayed. If the user chooses to continue the chart will be removed from the dashboards.

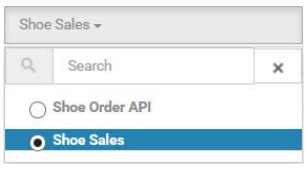
To configure a chart, select the name of the chart on the chart card. The chart configuration page displays. The page is divided into two columns. The left column contains the configuration settings for the chart. The right column has a preview area to test the chart configuration.



There are several configuration settings for a chart. Those settings are organized into multiple sections in the configuration column. The *DATASETS* section defines what data from a data set will be queried to fill the chart.



You select a data set using the *Select Dataset* pulldown. All data sets you have defined or are defined by others and are public will be listed in the pulldown.

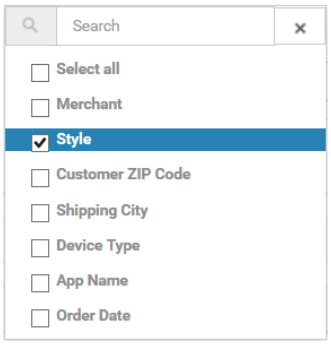


Because there may be a large number of data sets a search box is displayed above the pulldown options to narrow the options down. In general all pulldowns of potentially large option sets will have this capability.

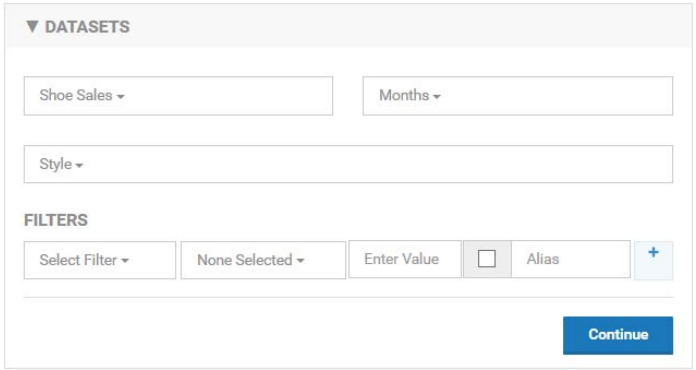
As described in the data sets section each data set can be defined to have different aggregation intervals. You must choose which interval you want to populate the chart with using the *Duration* pulldown. Only the intervals defined for the selected data set will be displayed in the pulldown.

The interval is important because the data points in the chart can be no more granular than the interval chosen. For example, if you choose the Months duration the data points in the chart cannot be plotted per day or per hour, only per month or year.

If the selected data set has more than one dimension you must select one or more dimensions that you want the charted data to be organized by using the Group By pulldown. For example if you want to chart sales by shoe style you would select the *Style* dimension. The dimensions of the data set are displayed as pulldown options. The pulldown supports the selection of multiple options.



The data from the data set that will be charted can also be filtered. For example, if you want to chart only data from a particular period of time or if you want to only chart certain Styles.



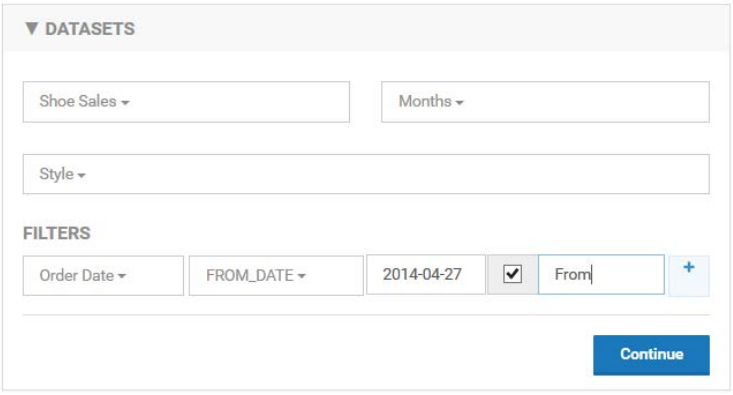
A filter is a constructed using one or more expressions represented as rows in the *FILTERS* section. To create a filter you work from left to right starting by selecting a dimension to filter by using the *Select Filter* pulldown. Then, moving to the right you select an operator in the operator pulldown. The operator choices will be different based on the type of dimension being filtered by. The following table shows what choices you have.

Dimension Type	Operators
String	equals, not-equals, starts with, ends with, contains, search
Date	before, after, between
ID-Name	equals, not-equals, starts with, ends with, contains, search
IP-Address	equals, not-equals, starts with, ends with, contains, search
Location	equals, not-equals, starts with, ends with, contains, search

The search operator allows matching using regular expressions.

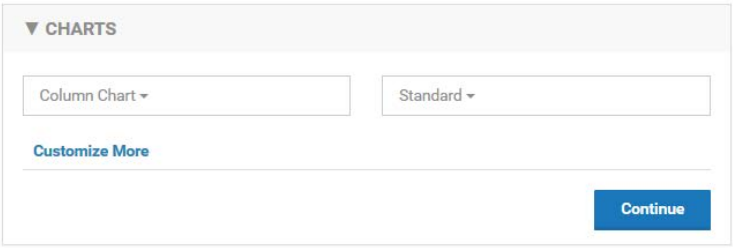
Next you enter a value to filter by. The type of value you enter must be of the same type as the dimension being filtered by. The checkbox to the right allows you to show the filter to the user when viewing the chart in a dashboard. This means the user can change the filter value and change the data being charted. If this option is chosen the value you entered to the left becomes the default value. When displaying the filter you can then specify the name of the filter you want to display to the user in the Alias field. Finally, if you want to add additional expressions to your filter you can select the '+' button and a new row will be displayed.

The following illustrates the *DATASETS* section using our Shoe Sales data set.

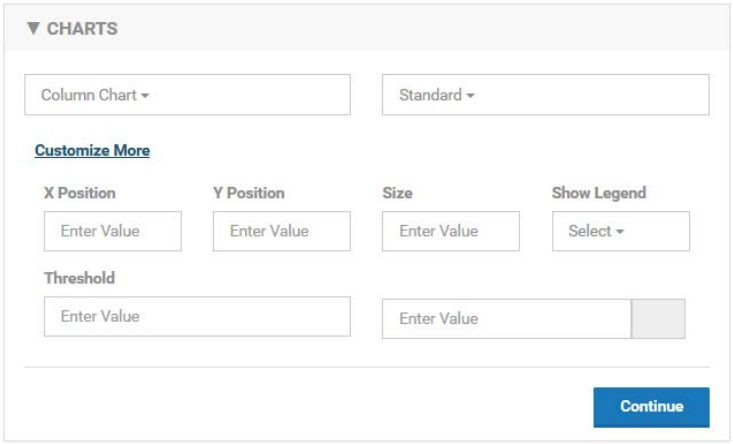


In this example we will be plotting monthly data points from the Shoe Sales data by Style. Only data points starting from 2014-04-27 will be plotted. The user will be able to change the start date of the data on the chart with a field named "From".

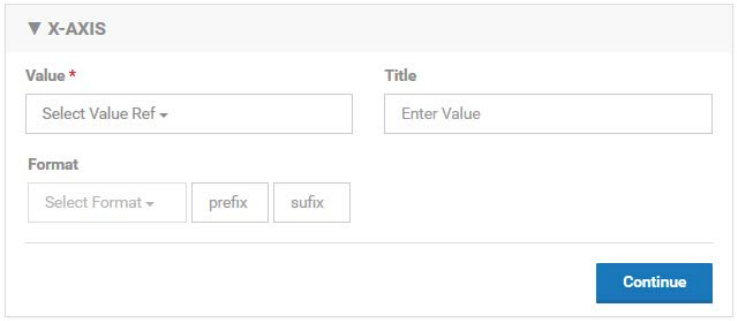
The *CHARTS* section identifies the type of chart to visually display the data with. There are several choices of charts, such as line chart, bar chart, or pie chart. Each of those charts may have a set of variations, or sub-types, such as standard, or stacked. The types of charts and their variations are listed in *Appendix A*. The following illustrates the selection of a Standard Column Chart.



Each chart type and variation may have specific options. These options are also listed in *Appendix A* and can be accessed in the user interface by selecting the Customize More link. The following illustrates these options for the Standard Column Chart.



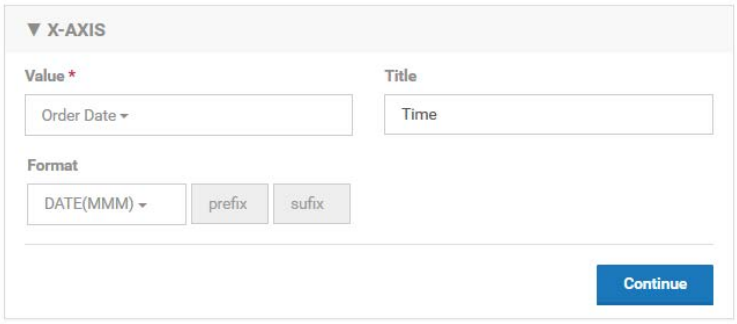
The X-AXIS section defines the x axis of the chart, what data will be used for tick marks, the title of the axis and the format of the tick mark labels.



To select the data to use for the x axis tick marks select a data set dimension from the Value pulldown. The x axis title can optionally be entered in the Title text box. The format of the tick marks can be specified in the Format section which has a pulldown of formats that are specific to the type of dimension chosen and optionally a prefix and suffix to the label. The following table shows the formats available for each type of dimension.

Dimension Type	Operators
String	uUpper case, lower case, camel case (i.e., myFirstDimension)
Date	mm (01 – 59), HH (01 – 23), dd (01 – 31), WW (01 – 52), MM (01 – 12), MMM (Jan – Dec), yyyy (1999, 2015), HH:mm (12:15), MM-dd (05-21), MMM-dd (May-21), MM-dd-yy (05-21-15), MM-dd-yyyy (05-21-2015), MM-yy (05-15), MM-yyyy (05-2015), MMM-yy (May-15), MMM-yyyy (May-22015), dd HH:mm (21 12:15), dd-MM-yy (21-05-15), dd-MM-yyyy (21-05-2015)
ID-Name	upper case, lower case, camel case (i.e., myFirstDimension)
IP-Address	upper case, lower case, camel case (i.e., myFirstDimension)
Location	upper case, lower case, camel case (i.e., myFirstDimension)

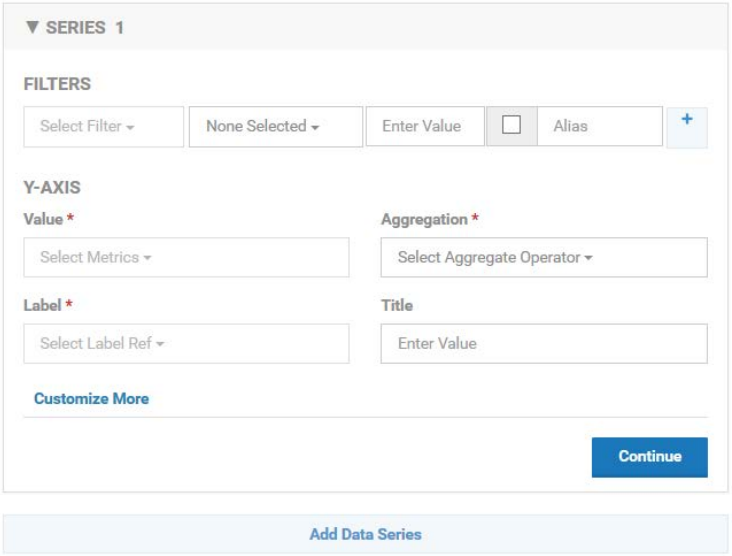
The following illustrates the X-AXIS section using our Shoe Sales data set.



In this example our x axis will consist of tick marks representing the order dates. Each tick mark will be labeled with the month abbreviation (Jan, Feb, March, etc.). The x axis title will be **Time**.

The *SERIES* section defines data series of the chart. There may be more than one data series for a chart which can be defined in a single *SERIES* section if they all share the same definition. However you can also define multiple data series with unique definitions by using the Add Data Series button at the bottom of the column. This will add a new *SERIES* section to the column.

Each data series is plotted using the y axis of the chart. For example a data series of a column chart may be the sales totals of the high top shoe style which would be drawn as a single bar. Each shoe style may have its own data series which would result in multiple bars on the column chart.



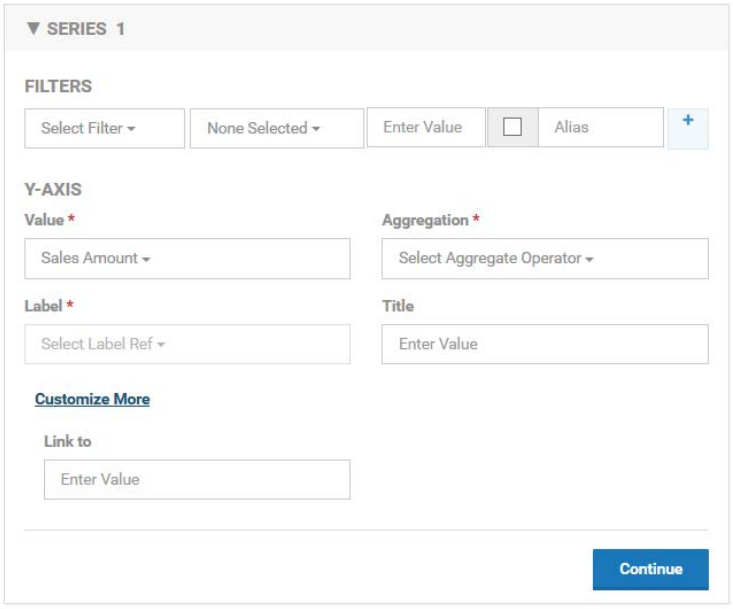
You select the metric you want plotted along the y axis using the Value pulldown. The Value pulldown will list all metrics from the data set. You then choose the aggregation method (sum, average, etc.) for the metrics using the Aggregation pulldown. The Aggregation pulldown will list the aggregations that were chosen for the metric in the data set definition. The y axis title can optionally be entered in the Title text box.

The Label pulldown provides options for how each series will be labeled. In a line chart this would be the label for each line. For a bar chart it would be the label for each bar. Since there may be multiple series the label will be derived from the dimensions used in the Group By field of the *DATASET* section which identifies how the data points are organized. For example if we are grouping by shoe style then we would want each line to be labeled with the shoe style. Since we can group by multiple dimensions the

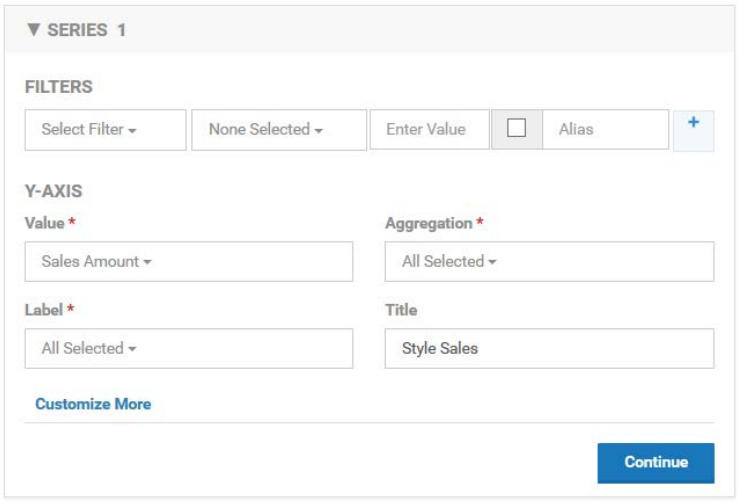
label can be derived using multiple dimensions chosen in the pulldown. The format will be *dim1 – dim2 – dim3*.

Just as we can filter the data in the *DATASET* section we can filter the data in a series. For example if we want to only have a single bar in the column chart for high top shoes we would add a filter for Style equal to **high top**. The filter section works the same as the filter section in the *DATASET* section described previously. By using the filter in conjunction with multiple *SERIES* sections we can define individual data series with their own settings.

Each chart type and variation may have specific *SERIES* options. These options are also listed in *Appendix A* and can be accessed in the user interface by selecting the Customize More link. The following illustrates these options for the Standard Column Chart.

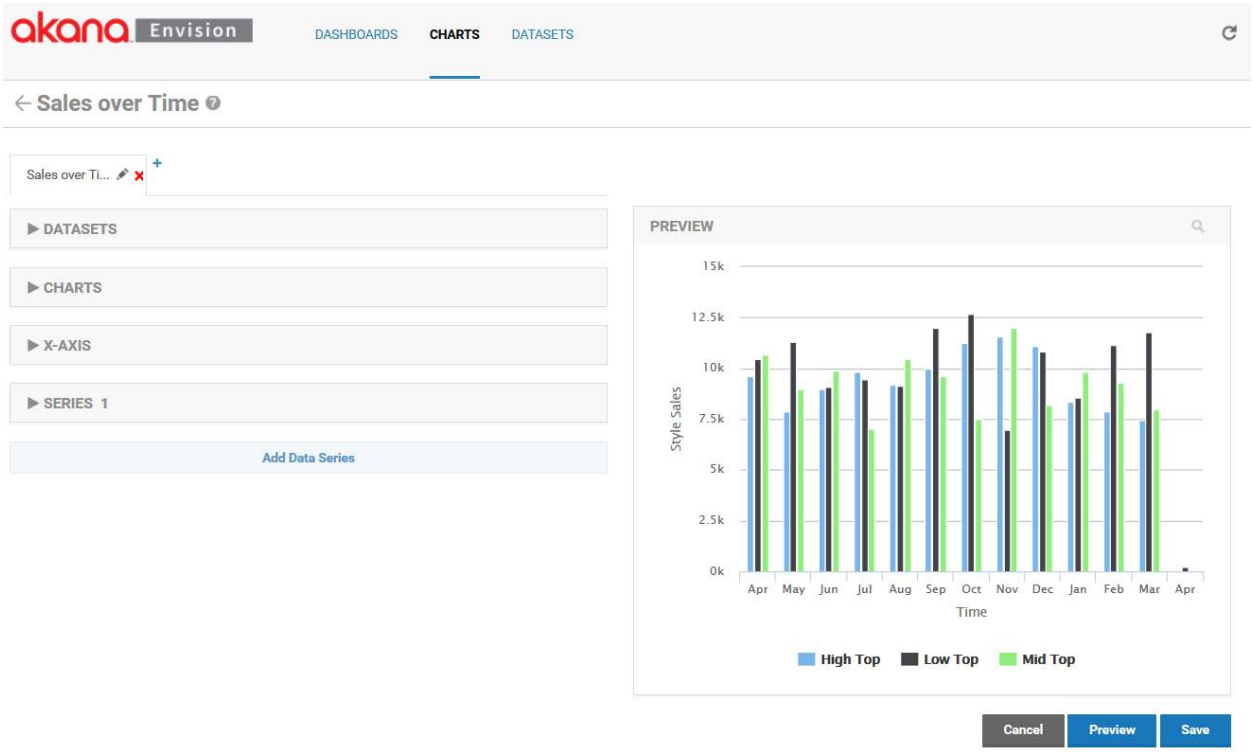


The following illustrates the single SERIES section using our Shoe Sales data set.



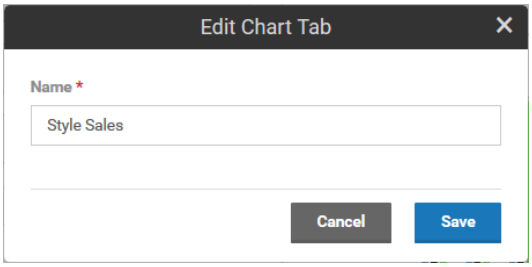
In this example we will have a vertical bar representing the sum of all orders' sales amounts for a given shoe style for a given month on the x axis. The label of each bar will be the name of the corresponding shoe style. The y axis title will be **Style Sales**.

At any time while modifying the settings of a chart you can preview what the chart will look like when placed on a dashboard by using the *PREVIEW* section in the right column. To see the preview you must select **Preview**. If you make changes on the left column, to see those changes on the right you must select **Preview** again. The changes are not saved until you select **Save**. The following illustrates the preview of our chart at this point.



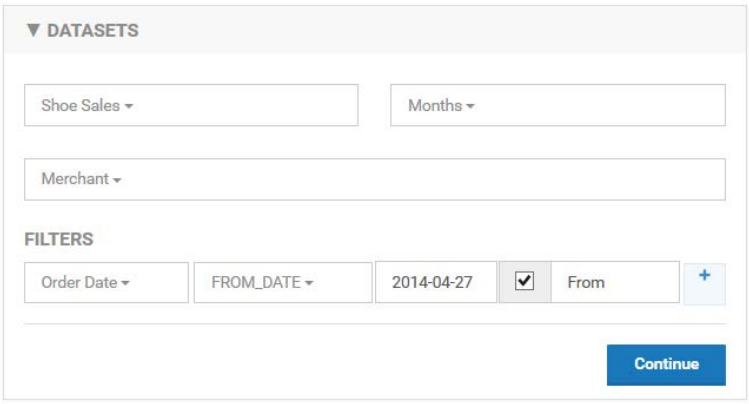
To this point we have only created a single chart although multiple charts can be grouped together, overlaying on one another, to create a combination chart. You'll notice the name of the chart we've created thus far is displayed on a tab. A combination chart will have multiple tabs with different chart configurations.

Each tab, or chart, in a combination chart can be renamed. To rename a chart click on the pencil icon on the tab. The *Edit Chart Tab* pop-up will be displayed where a new name can be entered.

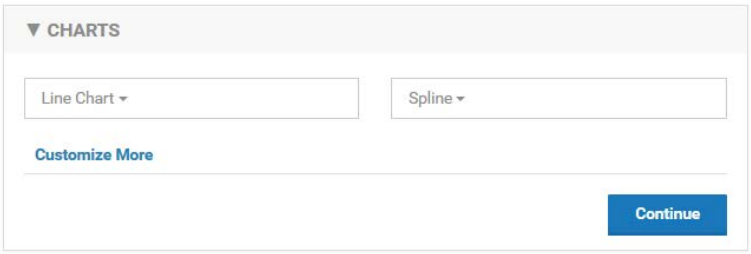


Next to the last tab there is a '+' symbol. Another chart can be overlaid on top of the current chart by clicking on the '+' symbol. When this is done a new tab will be displayed.

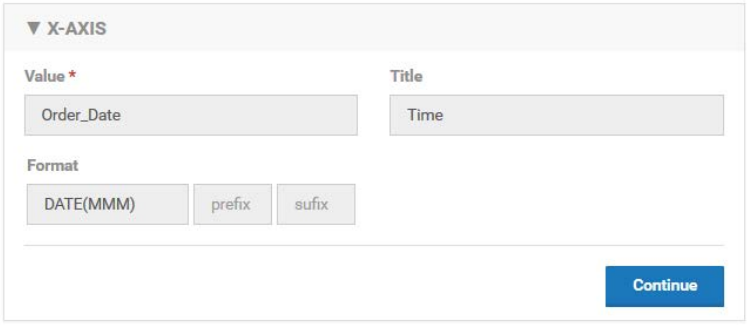
From the new tab we can configure the second chart just as we did with the first.



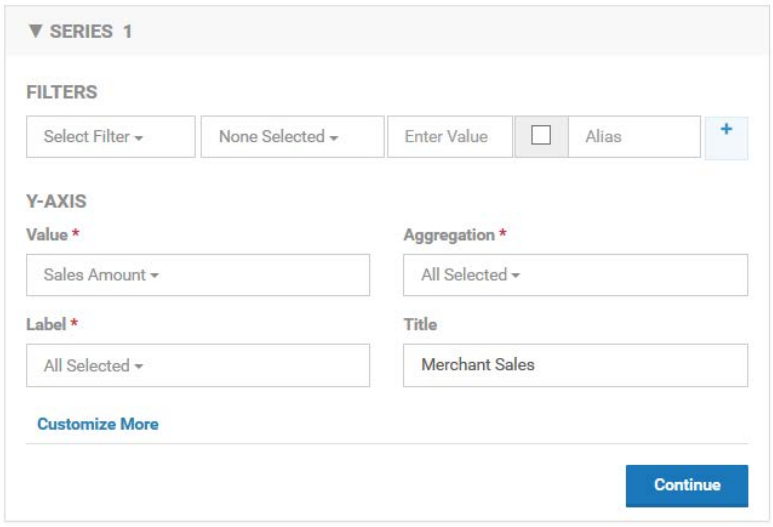
In this example we will be plotting monthly data points from the Shoe Sales data by Merchant instead of Style. Only data points starting from 2014-04-27 will be plotted. The user will be able to change the start date of the data on the chart with a field named *From*.



The type of chart used to display the Merchant sales will be a Spline Line Chart.

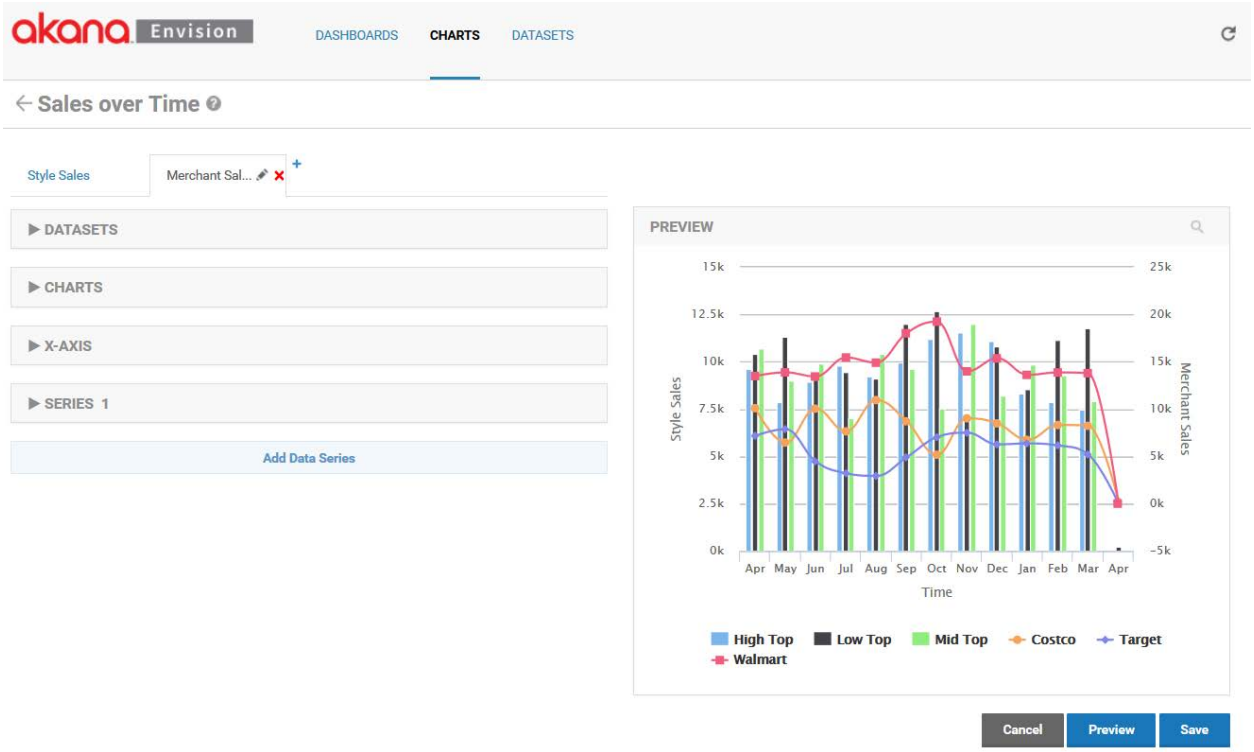


Note that the X-AXIS section is not editable for any charts that are not the initial one. All charts in a combination chart will share the x axis.



In the data series we will continue to chart the sum of the orders' sales amounts but because the data is grouped by Merchant the data points will be different than in the original chart.

You can see the overlay effect of the charts when you select Preview.

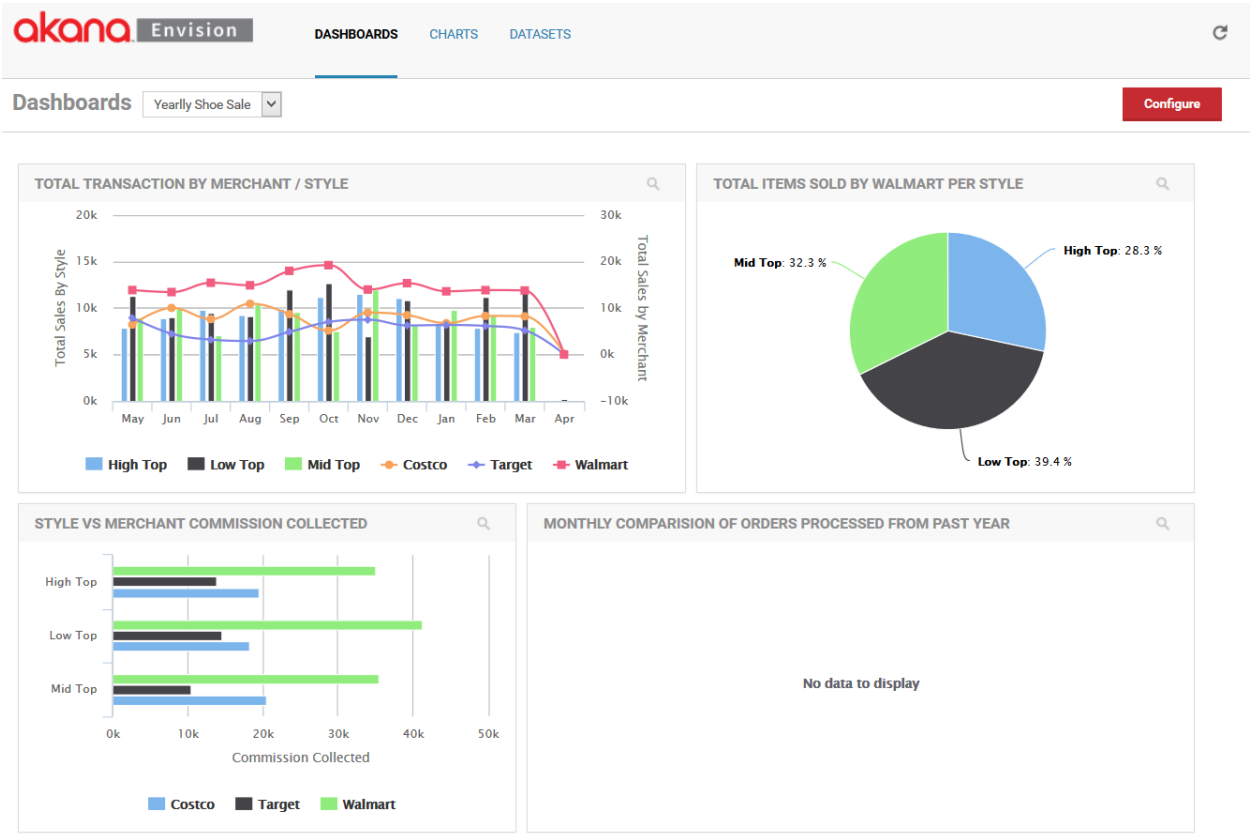


Notice the lines of the line chart are on top of the bars from the column chart. The charts are layered in order of the tabs. The first tab is the bottom layer. The last tab is the top layer.

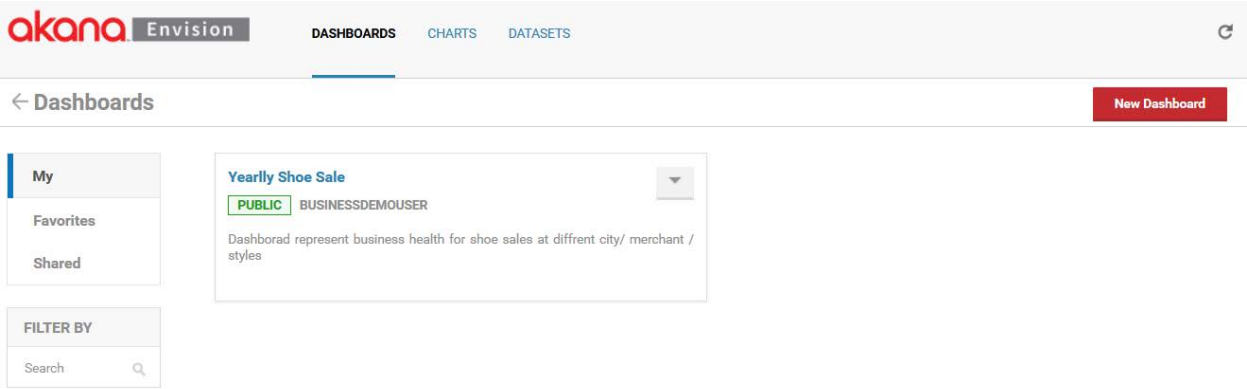
Chapter 6 | Creating a Dashboard

To present working Charts to Envision users those Charts need to be placed on a *Dashboard*. A Dashboard is simply a web page that will display 0 or more functional Charts. It is not for editing Charts but for displaying their results. If the Chart was designed to have interactive Filters then those would be functional on the Dashboard.

From the Envision top level menu select the *DASHBOARDS* menu item.

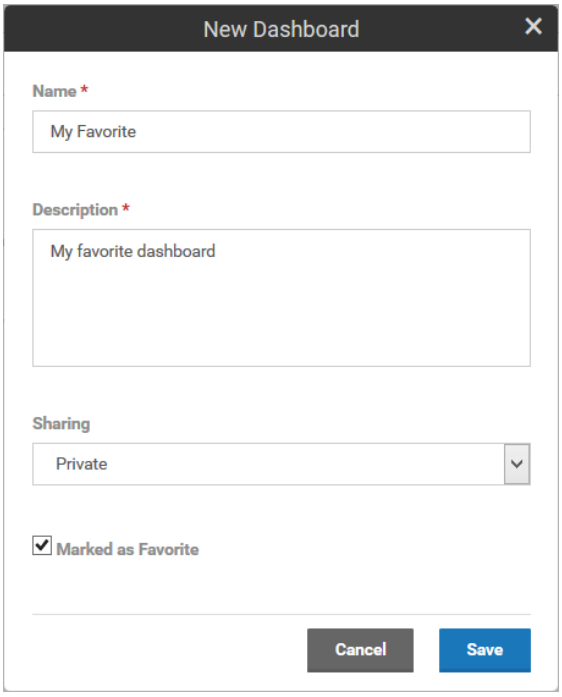


The default, or your most favorite, dashboard is displayed. It is a fully functional dashboard where you can view data. To switch to a configuration view of dashboards select *Configure* in the upper right.



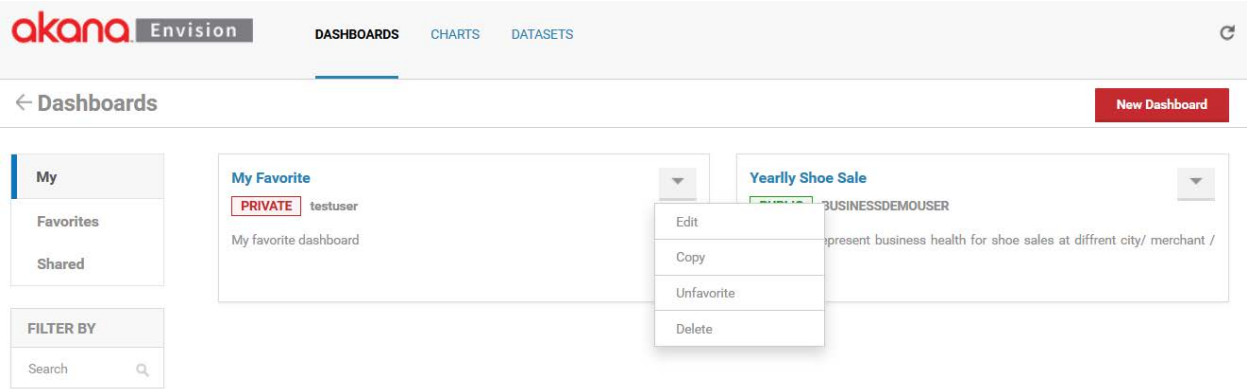
You are then presented three lists of dashboards, *My*, *Favorites*, and *Shared*. *My* dashboards are the dashboards that the logged in user has created. Shared dashboards are dashboards that others have created and made available to the user. *Favorites* dashboards are those selected by the user to (either their own or someone else's) to be in the list. It provides a faster lookup of commonly used items by giving them their own list.

Select **New Dashboard** to start the process of creating a new dashboard. A pop-up will be displayed where you can enter a name, a description, and designate how it will be shared.



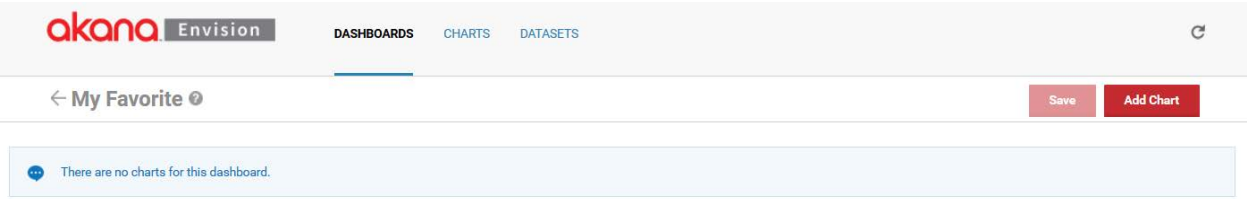
A dashboard can be shared other users. When a dashboard is shared with a user that user can view the chart and place them on dashboards. They cannot change the dashboard. A dashboard can be shared with nobody, everybody, or individual users and/or groups. The dashboard will automatically be placed in the *My* dashboards list. If the *Marked as Favorite* checkbox is checked it will also be placed in the *Favorites* dashboards list.

Each dashboard card in a list has a pull-down menu of options that can be performed on the dashboard.

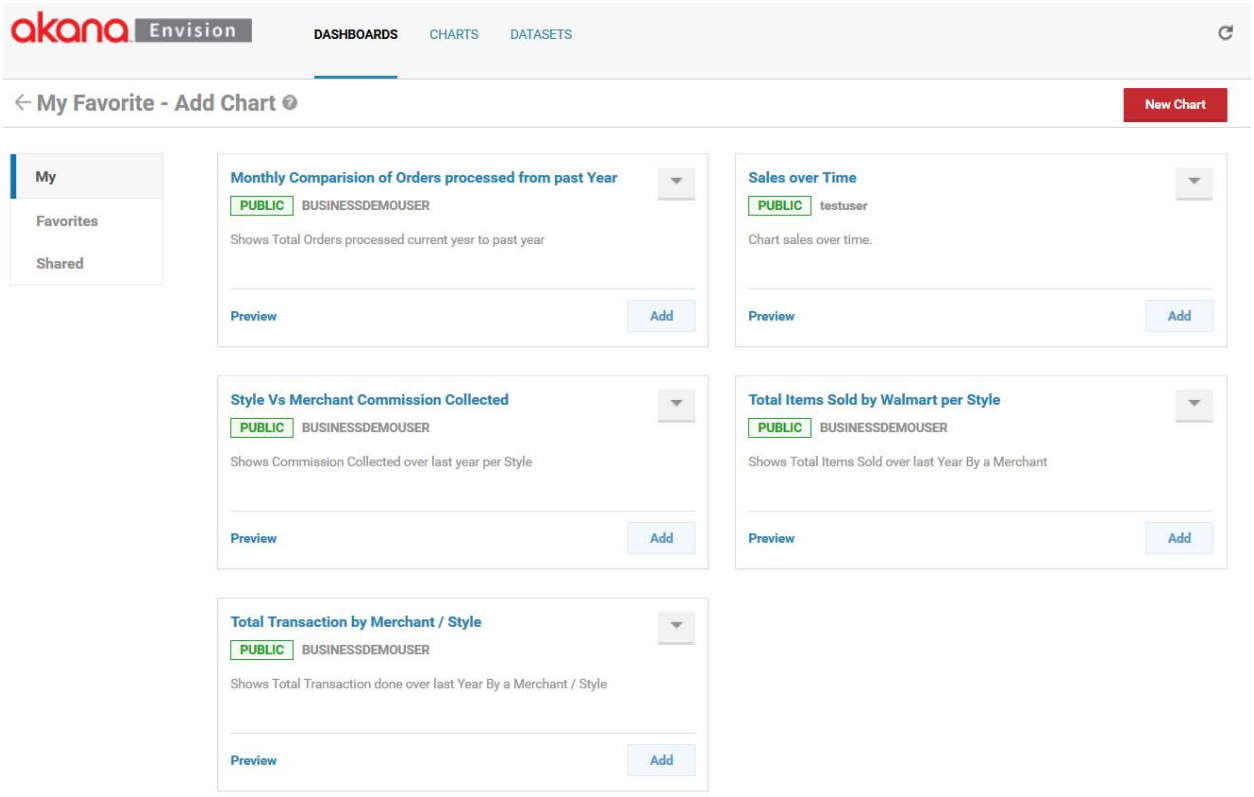


- **Edit** - Displays the pop-up used to create the dashboard initially so that changes can be made.
- **Copy** - Creates a copy of the dashbboard. The same pop-up will be displayed once again but this time it will be for a new dashboard and all the information from the copied dashboard will be filled in.
- **Favorite /Unfavorite** - Can be used to toggle whether the dashboard should be placed in the Favorites list.
- **Delete** - Removes the dashboard from the system.

To configure a dashboard select the name of the dashboard on the dashboard card. The dashboard configuration page will be displayed.



Initially the dashboard is empty. To add a chart to the dashboard select the Add Chart button.

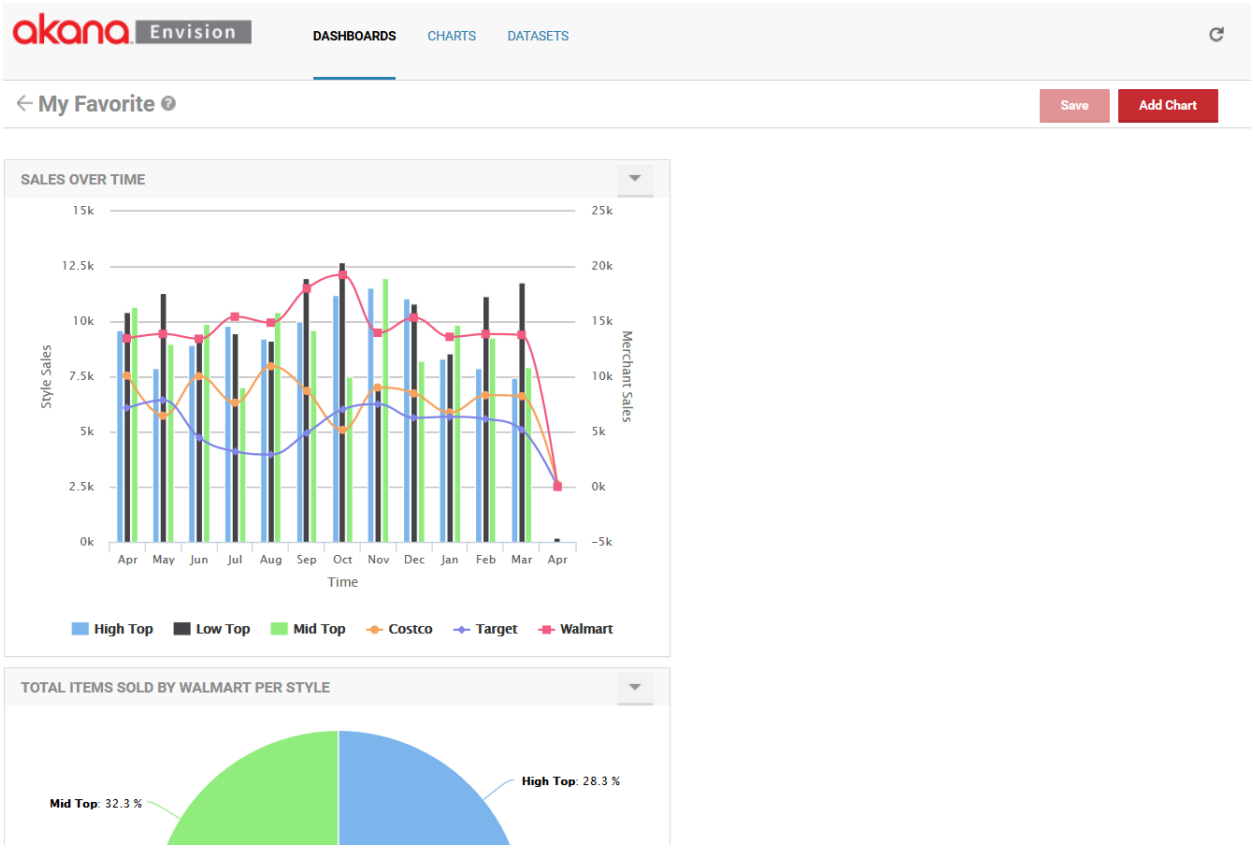


The *Add Chart* page looks very similar to the page you see when you click on the *CHARTS* link. There are three different sections for *My*, *Favorites*, and *Shared*. There are cards displayed for each chart. The difference here is that each chart card has an Add button and a Preview link. The Add button, when selected, will add the chart to the dashboard. You can continue to add charts to the dashboard without leaving this page. When you are finished click on the back arrow in front of the dashboard name.

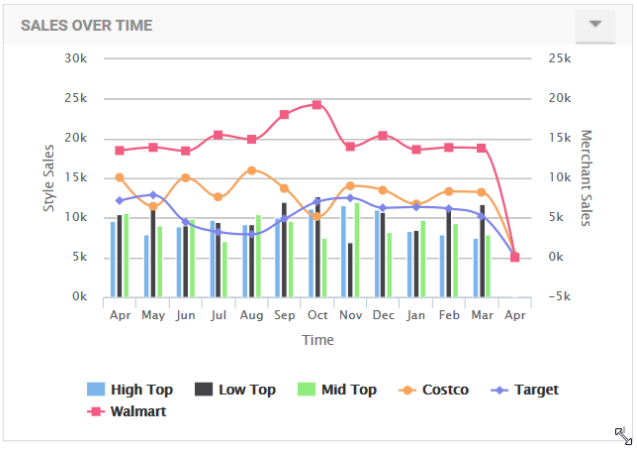
The Preview link will present a full page preview of the chart.

From the *Preview* page there is an **Add to Dashboard** button that you can select to add the chart to the dashboard immediately.

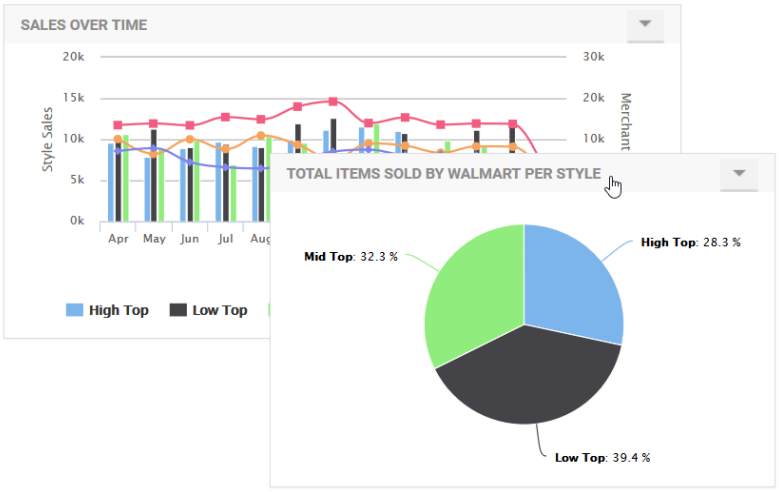
When returning to the dashboard page the charts are added but most likely may not be sized or positioned exactly as you want them.



A chart can be resized by grabbing its bottom right corner and dragging it to the desired size.

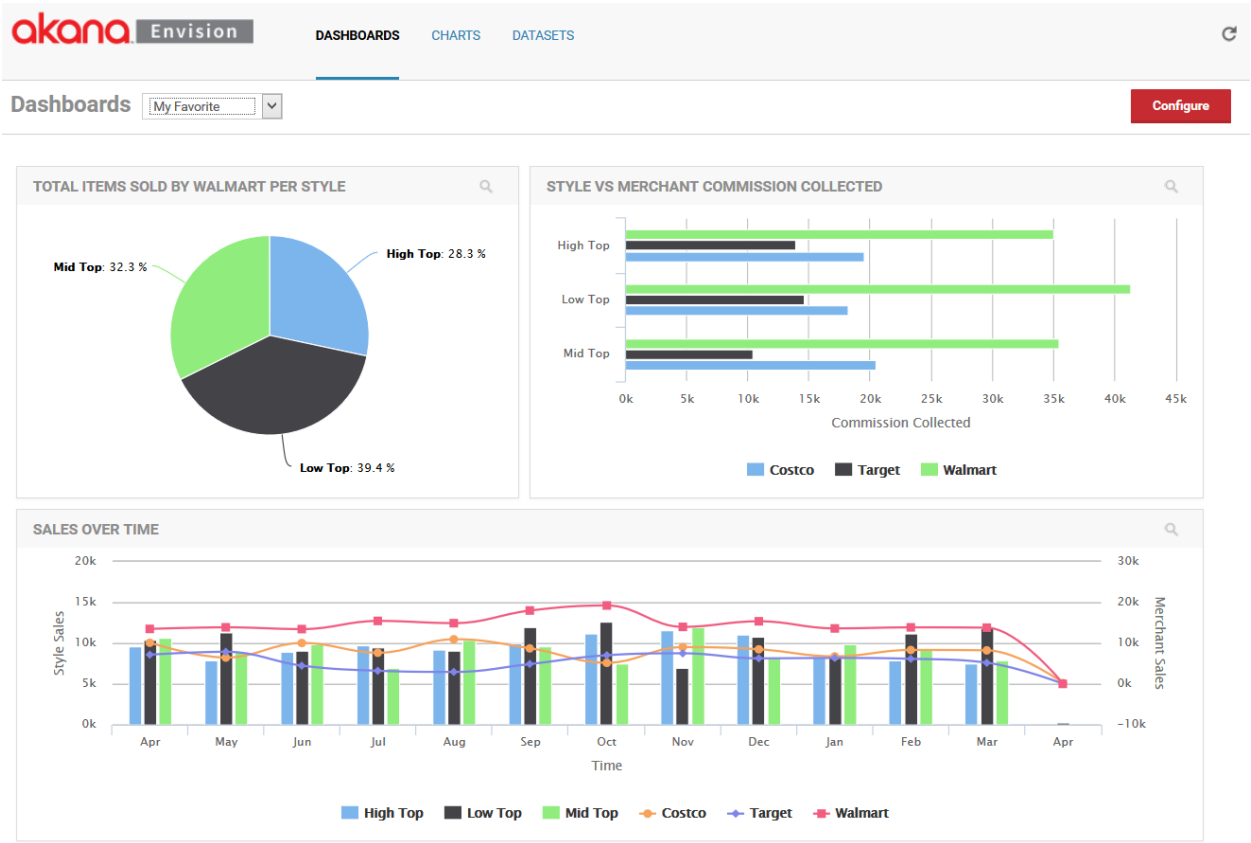


Charts can also be moved to different positions. To move a chart grab the title bar of the chart and drag it to the desired position on the dashboard.



Finally when the charts have been positioned and sized properly you can save the dashboard by selecting **Save**.

If the dashboard is saved as a favorite it will be listed in the *Dashboards* pulldown. When you select the dashboard from the pulldown it is displayed in its live mode.



In this mode the charts can no longer be repositioned or resized. However if a chart was configured to display filters to users the magnifying glass icon in the upper right corner of the chart can be selected to view those filters.

By changing the values in the filters and selecting **Update** the data charted will be changed.

Appendix A | Charts

<TBD>