

AMI Front End User Manual

Version 3.0

Contact Information

3Forge Support

646-490-3733

Support@3forge.com

Table of Contents

Welcome to AMI	6
AMI Building Blocks.....	7
AMI Basics	8
Log-in & Dashboard	8
AMI Menu	9
Set Global Style	10
AMI Work Mode and Editor Mode	11
AMI Window	12
Minimizing, Maximizing, and Deleting Windows	14
Resizing and Moving Windows.....	14
Manage Windows	15
Dividers.....	16
Modifying Dividers.....	18
Using Tabs	20
Adding, Editing, & Deleting Tabs	24
Editing Names of Windows, Tabs, & Tables.....	26
Real-time Tables/Visualizations	27
Real-time Tables	29
Real-time Table Settings.....	30
Table Style	31
Aggregate Tables	32
Heat Maps	34
Heat Map Settings & Style.....	38
Static Tables/Visualizations.....	39
Data Sources	40
Data Models	41
Static Tables.....	42
Add Datamodel	43
Request Script Transform	44
Response Script Transform	45
Aggregation on Databases	51

Static HeatMap.....	55
Query Form.....	59
Adding Fields.....	62
Custom HTML.....	67
Table Search.....	69
Columns	73
Adding Columns	73
Cell Contents.....	75
Format.....	76
Column Header.....	77
Column Position.....	78
Formatting.....	78
Editing Columns	85
Copying Columns	86
Sorting Columns	87
Primary Sorting.....	87
Sort (Ascending/Descending) Once.....	88
Secondary Sort (Ascending/Descending)	88
Clear All Sorts.....	90
Search Columns	90
Arranging Columns.....	92
Deleting/Removing Columns.....	93
Filtering Columns	94
URL's.....	97
Cloud.....	99
Managing AMI Users	99
Adding New Users.....	101
Publishing Layouts to the Cloud.....	102
Functions.....	103
Using functions in <i>Cell Contents</i> of Columns	106
Using functions in <i>Actions</i> of Columns	108
Using functions in <i>Formatting</i> of Columns	109

Using the gradient function on a column	110
Relationships.....	112
Setting up a basic relationship	112
Editing & removing relationships	113
Using the Right-Click Menu to run queries (static relationships).....	114
Additional Filters	116
When nothing is selected	118
AMI Script	121
Relationship Matrix	122
Direct Query.....	123
Running Simultaneous Queries on Multiple Target Panels from One Source Panel.....	125
Cutting & Pasting.....	128
Exporting & Importing	130
Exporting/Importing Multiple Panels or Entire Windows	131
Exporting/Importing Dashboards.....	132
Data	134
Data Statistics.....	134
Schema.....	134
String Pool	135
Local Real Time Objects.....	135
Commands	136
Charts	139
Creating a new chart & adding a data model	140
Main Chart Options.....	143
Edit Underlying Datamodel	143
View Underlying Datamodel	143
Add Underlying Datamodel	144
Unlink Underlying Datamodel.....	144
Edit Global Style.....	144
Chart Plot Options.....	145
Adding & removing plots	146
Adding axes	150

Editing & deleting axes.....	154
Add Rendering Layer.....	159
Adding Series.....	164
Transparency of Layers.....	178
Zooming.....	179
Chart Legends	181
Setting up Relationships with Charts.....	185

Welcome to AMI

AMI from 3Forge is a data visualization platform to help users see across vast amounts of data, and then act on that data.

We help users who want to explore data, see relationships, and need to drill-down to details. And then take action.

AMI complements and enhances machine learning, monitoring tools, and alerting systems by ingesting data from those systems and correlating with other internal and external data.

Helping users is not just about charts, graphs, and 3D visualization. Large-scale challenges require charting, alerting, filtering, and correlation across many data sources that change on-the-fly and are often disparate. AMI does all this – without preparation of data before use. AMI grabs data at the source. There is no requirement for cubes, OLAPs, data warehouses, data marts, or data lakes. AMI taps live sources, customer databases, and transactional databases.

Collaboration at scale means helping users quickly build dashboards with simple wizards to access data sources, layout the dashboards, and setup relationships across data. For enterprises, collaboration often means that one group designs visualization for a larger group, shares these quickly, and still empowers users to modify as necessary. With AMI, a UI can be designed, connected to data sources, and tested in hours. AMI eliminates the need to write UIs from scratch.

Sign-on and authentication integrate with enterprise standards like LDAP. Access is granular to the field level, and controlled by user profiles, group membership, and roles.

Data is encrypted in-flight, and can also be controlled from the platform to avoid performance degradation over slow connections, and redistribution of data beyond agreements.

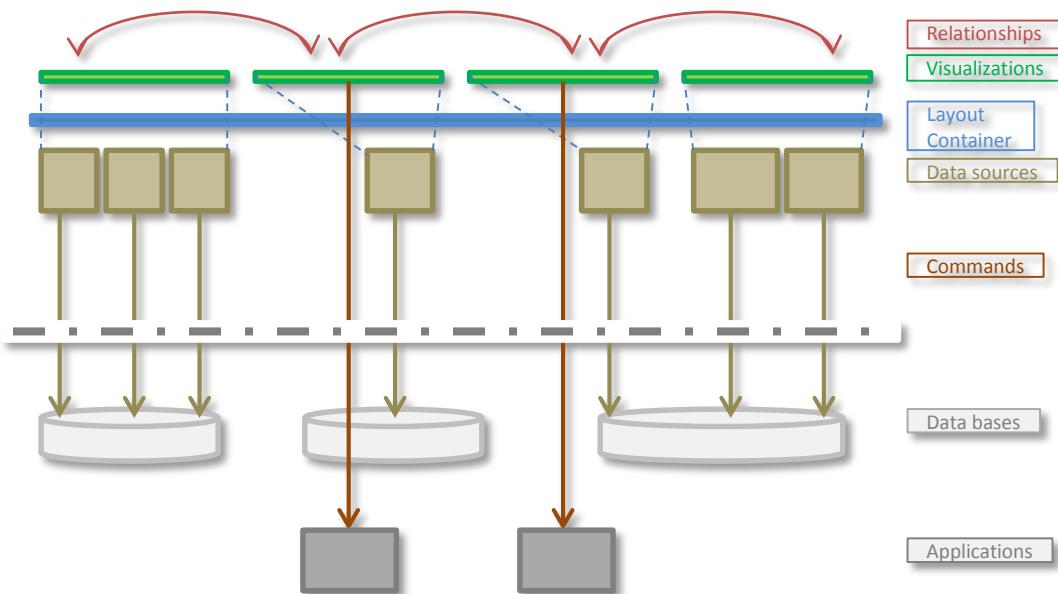
AMI helps people interact with complex systems in big data, cloud, mobile, IoT, and social applications.

This manual describes how to setup and customize dashboards and connect to data sources. Basic computer experience is all that's needed.

AMI Building Blocks

The front end of AMI can be broken down into 6 basic concepts, with a basic understanding of each of these you can immediately start to build dashboards.

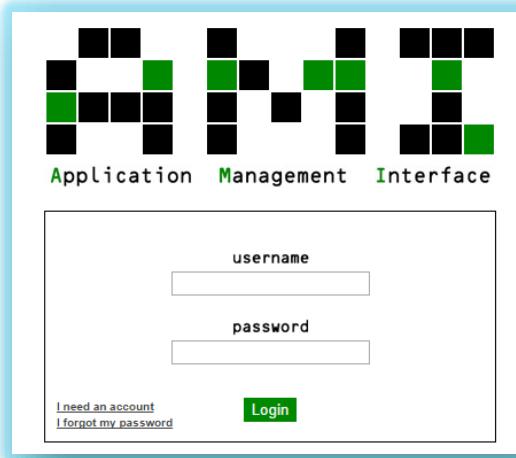
1. **Layout Containers** – These provide the framework for organizing your dashboard. They include Tabs, Horizontal Dividers, Vertical Dividers and Windows. These Containers can be nested within each other to ultimately house visualizations. In addition to visualizations you can also nest forms and custom html / plugins inside of layout containers.
2. **Visualizations** – These provide the final data view that the end user sees, which include charts and tables. Visualizations are interlinked to each other via **Relationships** and are backed by one or more **Data sources**. Visualizations can be configured to invoke **commands** on backend systems.
3. **Data sources** – This provides the schema used to represent physical underlying data bases and external data sources. Data sources accept "queries" from external relationships, interpret and transmit the queries to underlying data bases and then deliver the result sets back for display.
4. **Relationships** – Relationships are at the core of what makes AMI so powerful for drilling in and investigating data. Visualizations can be related to each other so that clicking on a visualization's data points will cause a query to be automatically run on a separate visualization's data sources.
5. **Commands** – Commands are registered by backend systems and can then be executed by users on the frontend via clicking on nodes or through context menus depending on how the visualizations are configured.



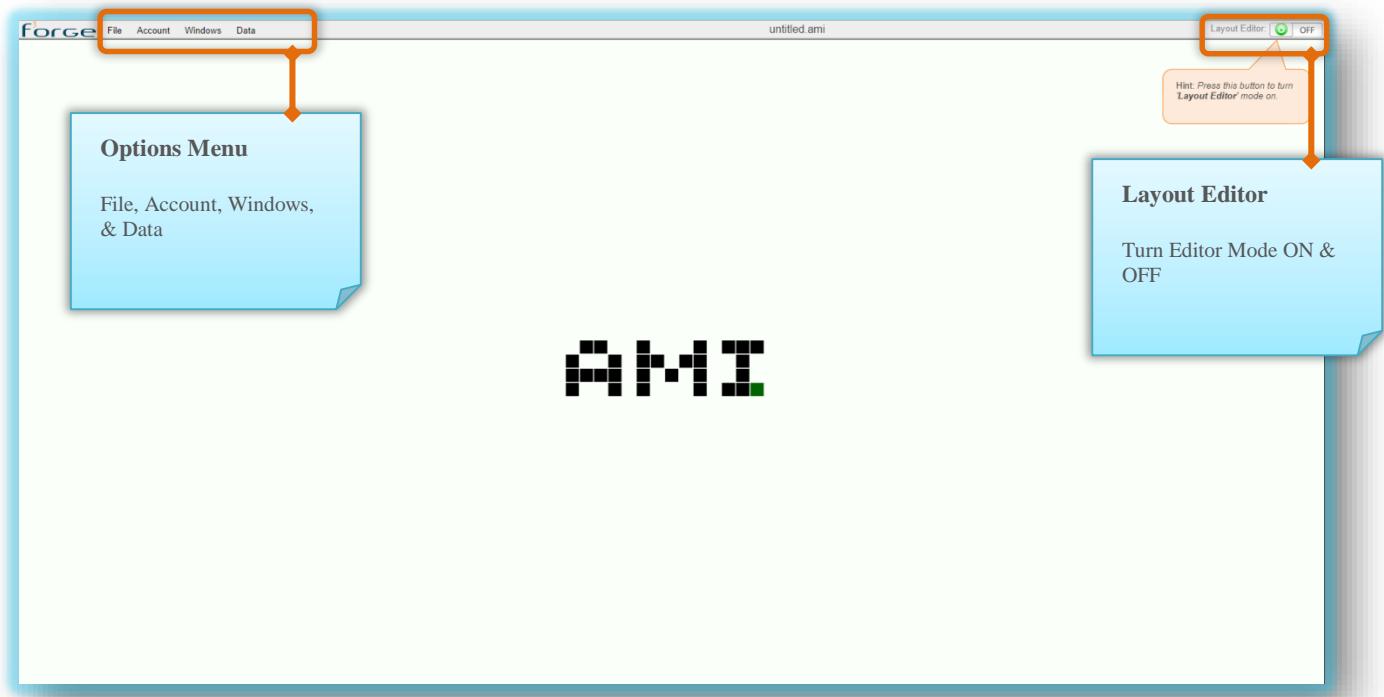
AMI Basics

Log-in & Dashboard

Log into AMI with the provided username and password. *Please contact your administrator if you have any trouble logging in.* See **AMI Configuration Manual** for details on adding user/passwords. DEFAULT user/password is **demo/demo123**



After logging into AMI, a blank dashboard will appear



AMI Menu

File

New	Clear the dashboard and start a new layout
Open	Open a saved layout
Save	Save the current layout
Save As	Save the current layout under a different name
Export	Export the current layout, as text, for use in another dashboard
Import	Import dashboard layout into the current dashboard
Publish To Cloud	Upload the current layout to the cloud
Import From Cloud	Load a layout from the cloud
Remove From Cloud	Delete a layout from the cloud
Manage Users	Permission functionality and layouts to specific users. This option is available to only Admins
Set Global Styles...	Establish a color scheme to be used in the layout. Use this option to apply a color to the background of the title (distinguish environments)

Account

Logout Safely log out of the current AMI session

Windows

[Name of Window]	Select different windows in the dashboard & bring minimized/hidden windows to the foreground
 New Window  Manage Windows...	Add a new window (Available in Editor Mode); Manage all windows in the layout. Able to maximize windows and take away the header (Available in Editor Mode)

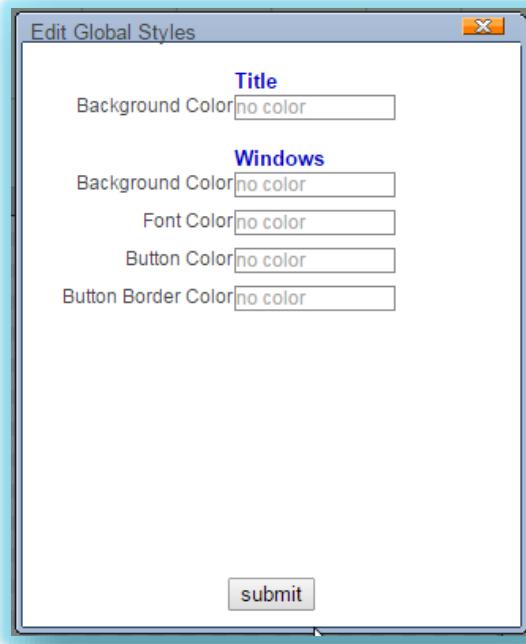
Data

Data Statistics	Overview of data currently in the backend
Datasources	Add, delete, or edit different datasources currently connected to AMI
Datamodels	Opens the data model platform
Upload Data	Upload data to the backend
Clear all Data	Clear all data from the backend

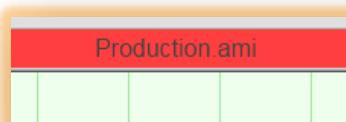
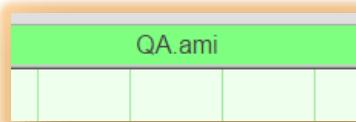
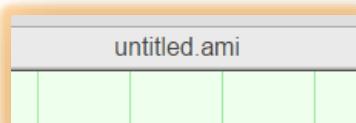
Note: Those with **User-level permission** are only able to open layouts made available to them in the cloud. Logging out and selecting windows are the only other options available to users in the menu

Set Global Style

In order to keep a consistent look and feel throughout the layout, use the **Set Global Style** option under the **File** menu in order to assign colors to the following fields:



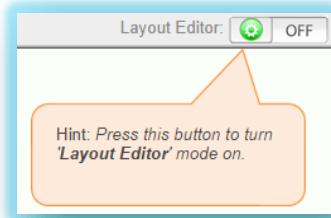
Example – using the Title Background Color to distinguish environments



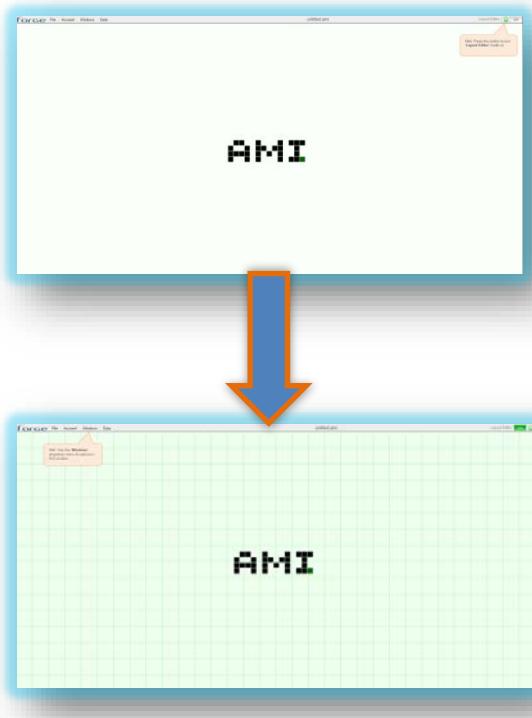
AMI Work Mode and Editor Mode

There are two main modes in AMI – the work mode and the editor mode. Editor mode is used when building and configuring dashboards. While in editor mode, new windows can be added and all relationship lines and configuration buttons are visible on the dashboard. Only those with at least **Developer-level permission** are able to enter the editor mode. Those with **User-level permission** are limited to the work mode. When in work mode, layouts can be used but cannot be edited.

- 1) To enter editor mode, turn on **Layout Editor** by clicking on the green button on the top right corner



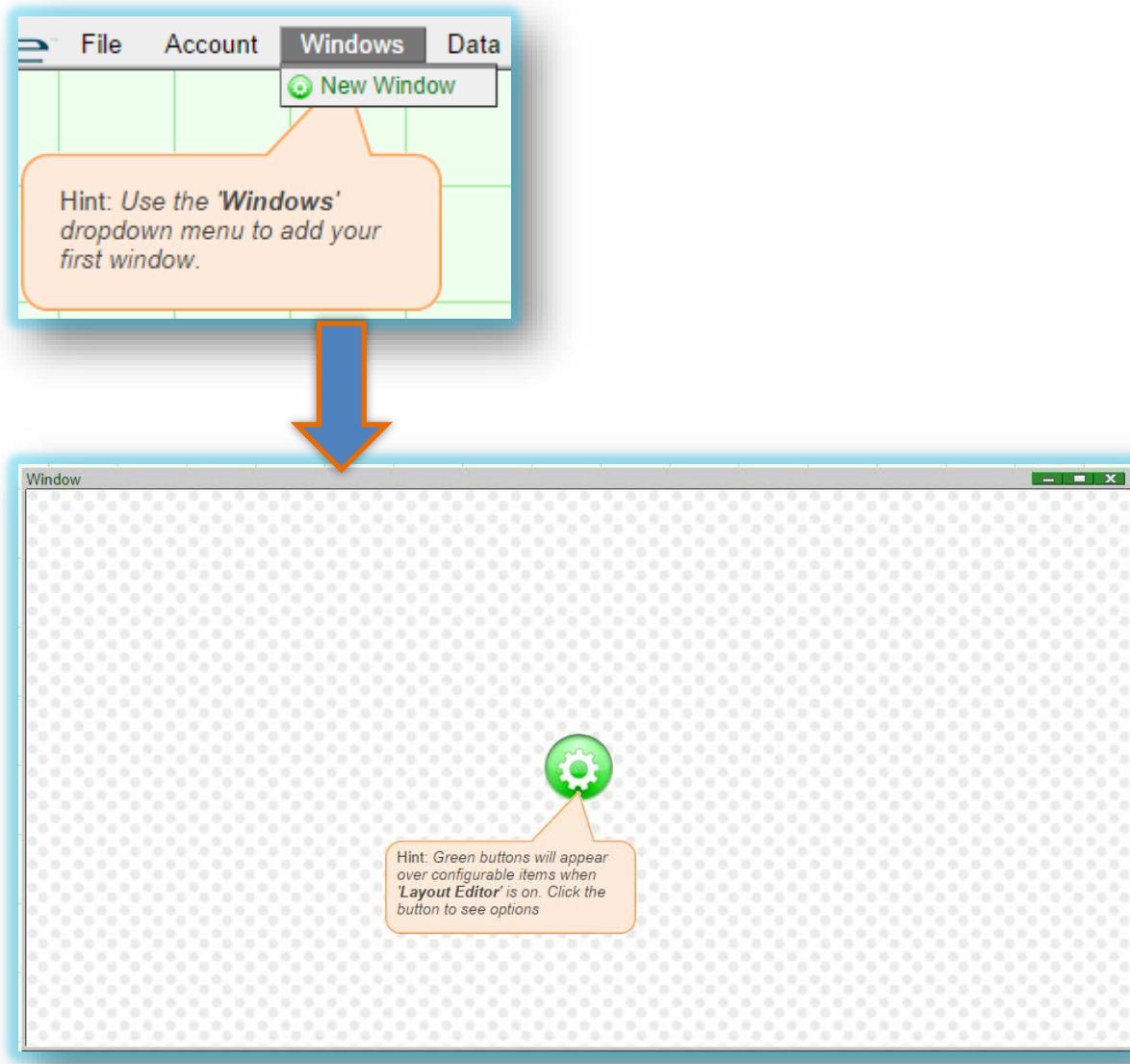
- 2) Turning on Layout Editor changes the dashboard from *Work Mode* to *Editor Mode*



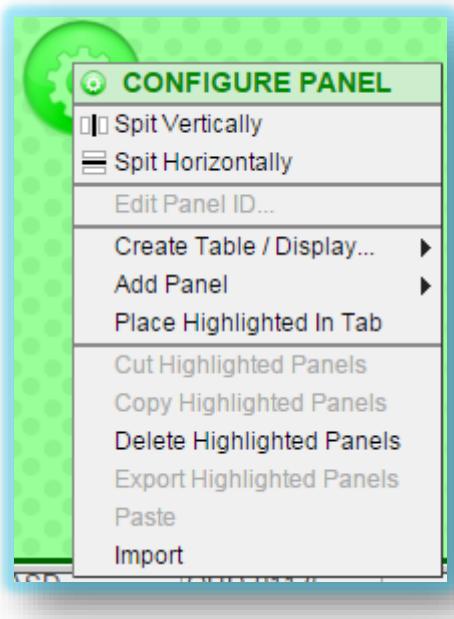
AMI Window

In AMI, windows are highly customizable and entire layouts can be built within one window.

- 1) In order to get started, turn on **Layout Editor** by clicking on the green button on the top right corner. Turning on Layout Editor allows the addition of new **windows**. Select **New Window** under the **Windows** menu to add a new window. The window will have a blank panel where you can customize the dashboard. Blank panels are indicated with the dotted pattern seen below.



- 2) Click on the **Green configuration button** in order to view the different options available for the new window



Split Vertically	Divide the panel vertically, creating 2 panels
Split Horizontally	Divide the panel horizontally, creating 2 panels
Edit Panel ID	Change the ID of the panel
Create (Recreate)	Select what kind of table/display the window will be used for
Table/Display	
Add Panel	Add a panel to the left, right, above, or below the selected panel
Place Highlighted In	Places the selected panel into a tab
Tab	
Cut Highlighted Panels	Cut the highlighted panel to paste into another panel in the same dashboard
Copy Highlighted Panels	Copy the highlighted panel to paste into another panel in the same dashboard
Delete Highlighted Panels	Delete the panel
Export Highlighted Panels	Export the highlighted panel, as text, for use in another dashboard
Paste	Paste the previously copied panel(s) using "Copy Highlighted Panels". Note this only appears if something was copied
Import	Import a panel that was exported using "Export Highlighted Panels". Note: to import whole layouts, use Import under the File menu

Note: Windows can be split into separate sections/panels. Dividers will appear when new panels are made and these dividers will also have their own menu (discussed in the following section)

Minimizing, Maximizing, and Deleting Windows

Windows in AMI can be minimized, maximized to fit the entire dashboard, or deleted entirely from the dashboard



: minimizes the window

Note: Minimized windows can be accessed by selecting them from the **Windows** menu



: maximize the window or restores a maximized window to its original size



: deletes the entire window

Note: In order to delete individual panels within windows, use the configuration button menu of the panel and select **Delete Highlighted Panels**. If the window does not have any panels, selecting this option will delete the window.

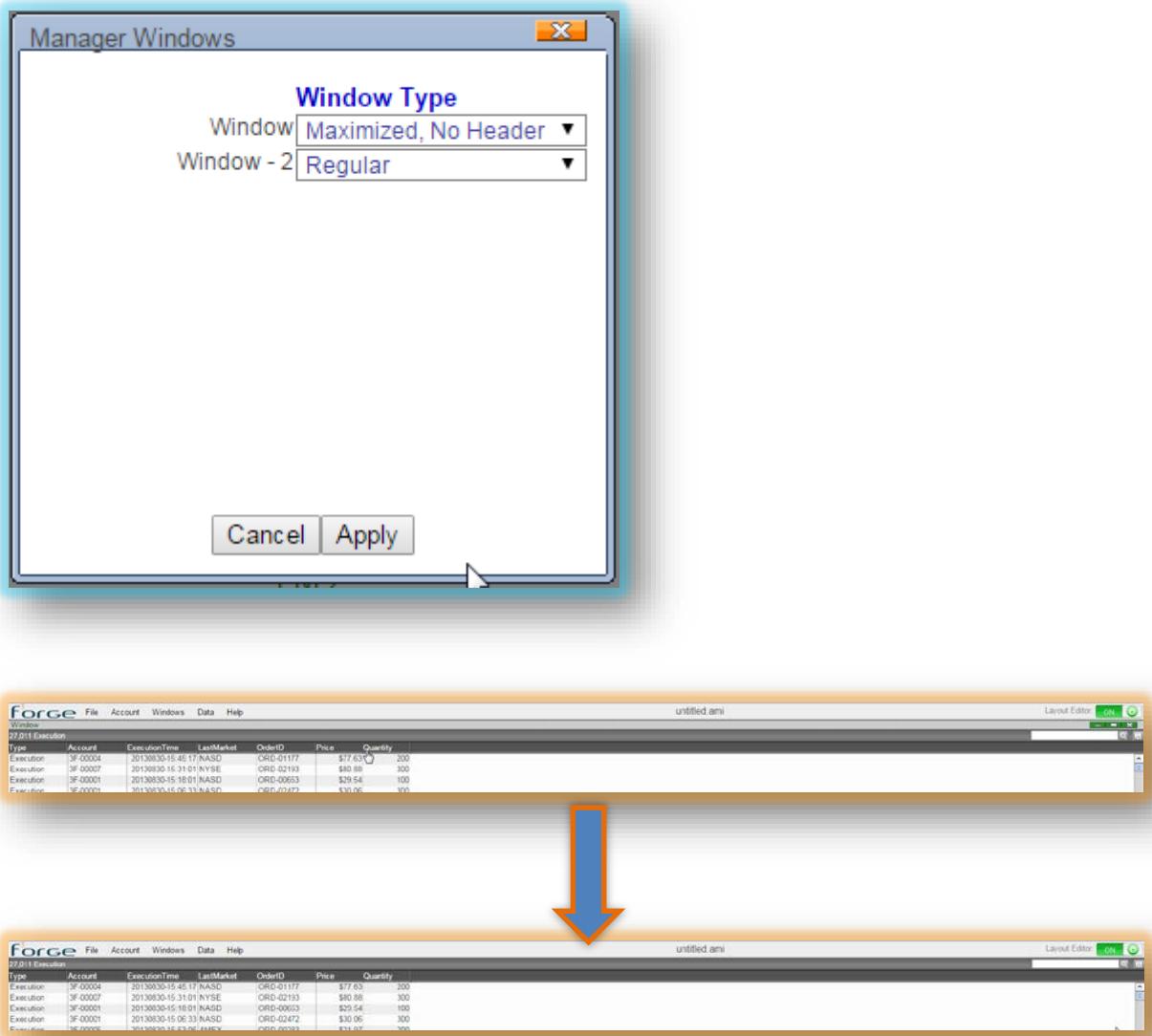
Resizing and Moving Windows

- In order to resize a window, click and hold any part of the edges or corners of the window. While holding the mouse button, drag the cursor to resize the window.
- In order to move a window around the dashboard, click and hold the grey header of the window. While holding the mouse button, move the window to any area of the dashboard. Note: saving a layout will save the position and size of windows as well.

Manage Windows

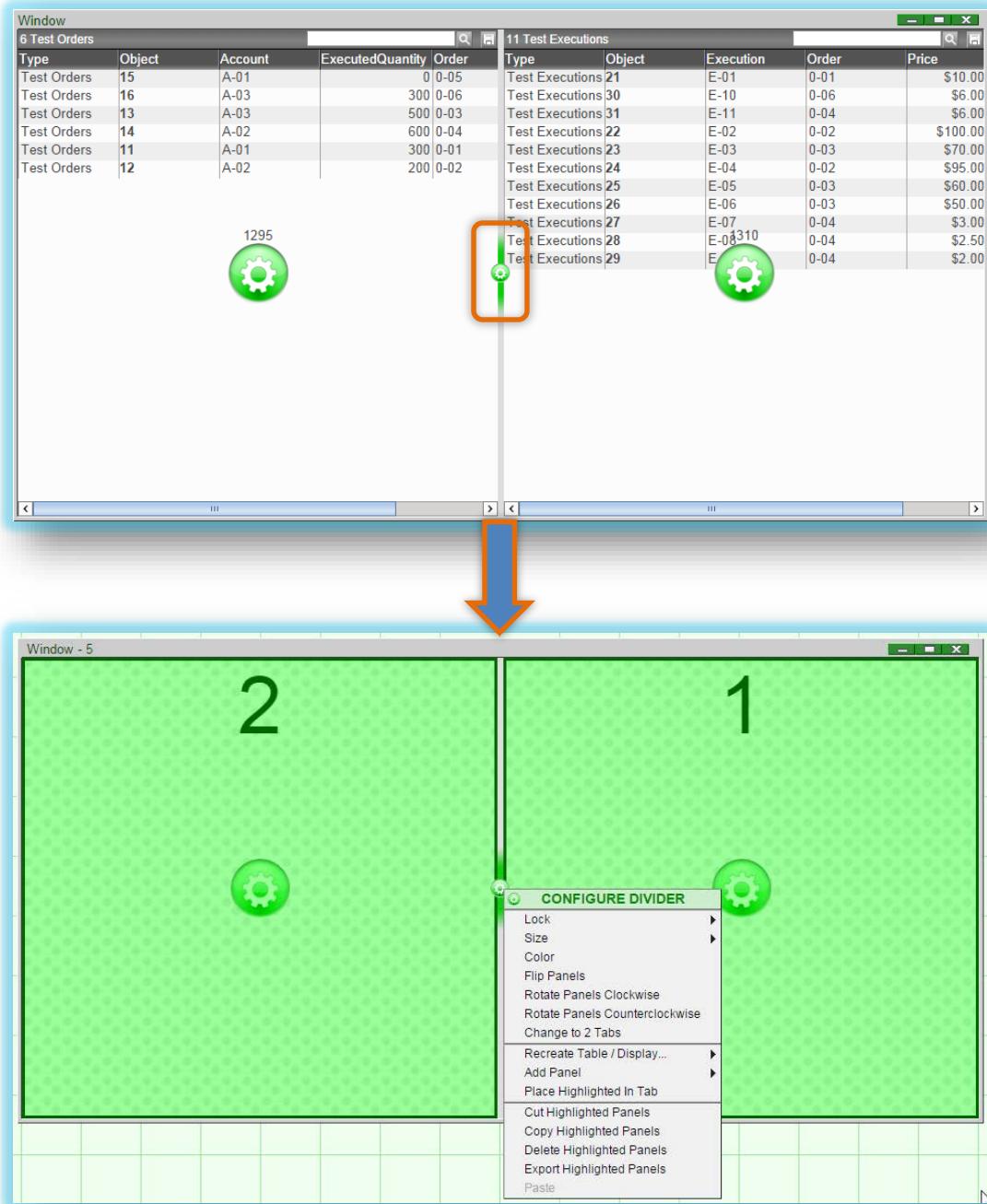
The Manage Windows option under the *Windows* menu lists all of the windows currently in the layout and allows the user to select the type of window:

- **Regular:** window may be minimized, maximized, deleted, moved, and resized
- **Maximized, No Header:** window is locked in a maximized position without the header



Dividers

When windows are split into panels (sections), **dividers** will appear and will have their own smaller configuration buttons. Dividers, by default, can be dragged by the user to adjust the real-estate of the two panels they divide. Click on this configuration button to further modify the window. Clicking on the configuration button will highlight the associated panels and numbers will be assigned to each panel.



Lock	Ability to lock the width of the panels
Size	Change the size of the divider (between the two panels)
Color	Apply a color to the divider
Flip Panels	Flip the arrangement of the two panels
Rotate Panels Clockwise	Rotate the panels in the clockwise direction
Rotate Panels	Rotate the panels in the counterclockwise direction
Counterclockwise	
Change to 2 Tabs	Places the panels into two separate tabs
Add Panel*	Add panels to the left, right, above, or below the panels
Place Highlighted In Tab	Places the highlighted panels or the entire window into a tab
Cut Highlighted Panels	Cuts the highlighted windows
Copy Highlighted Panels	Copies the highlighted windows
Delete Highlighted Panels	Deletes multiple panels
Export Highlighted Panels	Export the highlighted panel, as text, for use in another dashboard

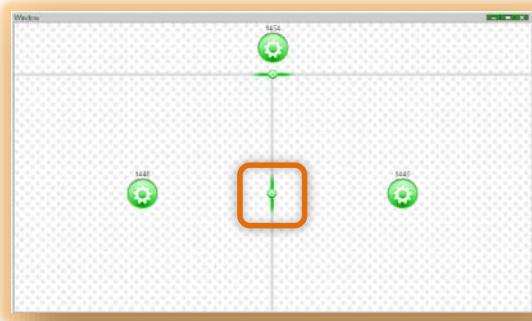
*Note: New panels will be added differently depending on the configuration button used

Example

Adding a panel **above** using the (right) **panel** configuration button adds a panel only above the right panel as seen below:

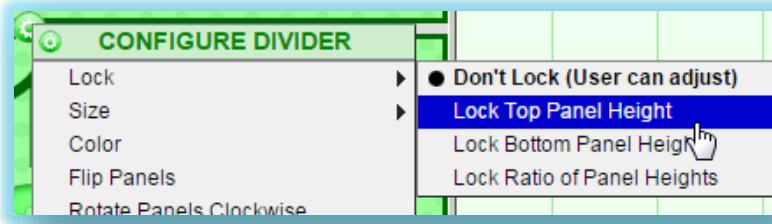


Adding a panel **above** using the **divider** configuration button adds a panel above both panels:



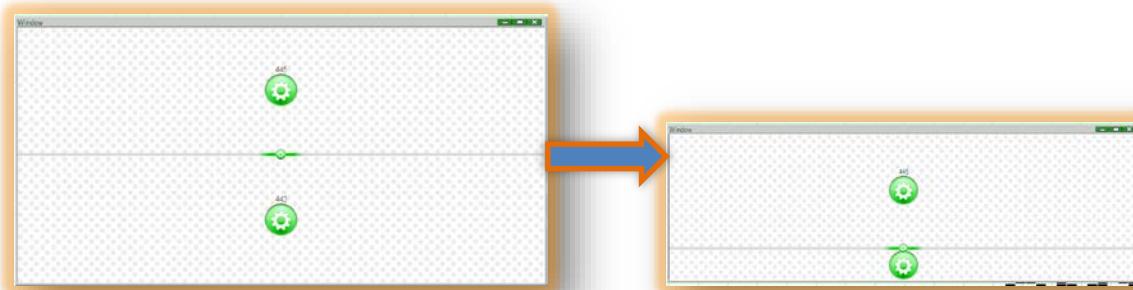
Modifying Dividers

Locking Dividers

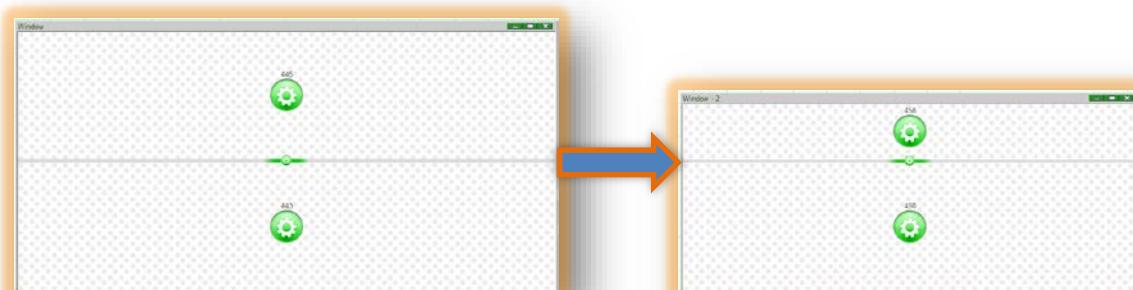


- **Dividers** can be locked in various ways in order to keep the size of the panels consistent throughout the resizing of windows. Below are examples of locking dividers and then resizing their parents' windows.

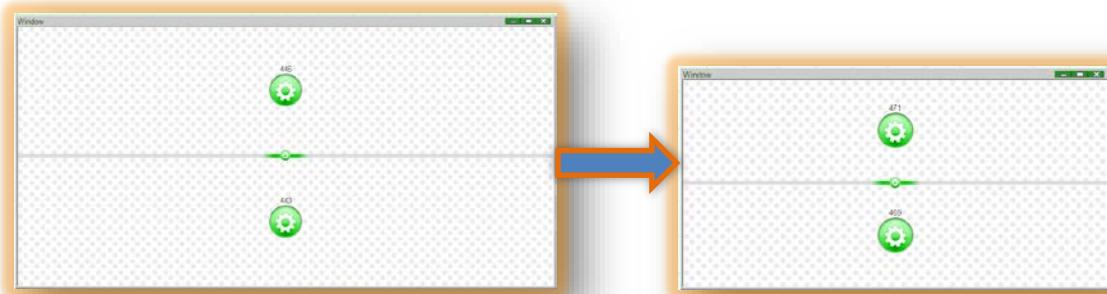
Lock Top Panel Height:



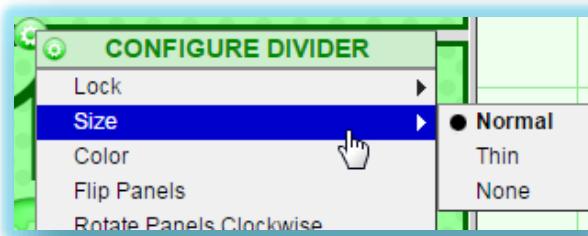
Lock Bottom Panel Height:



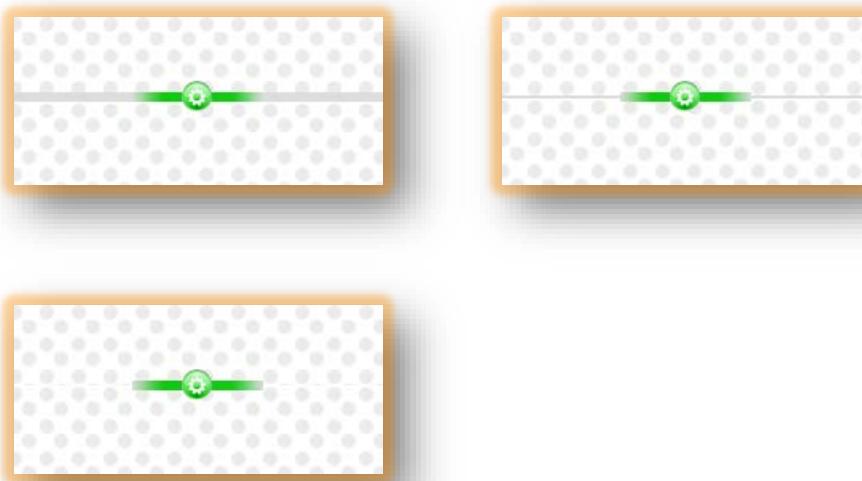
Lock Ratio of Panel Heights:



Size of the Dividers



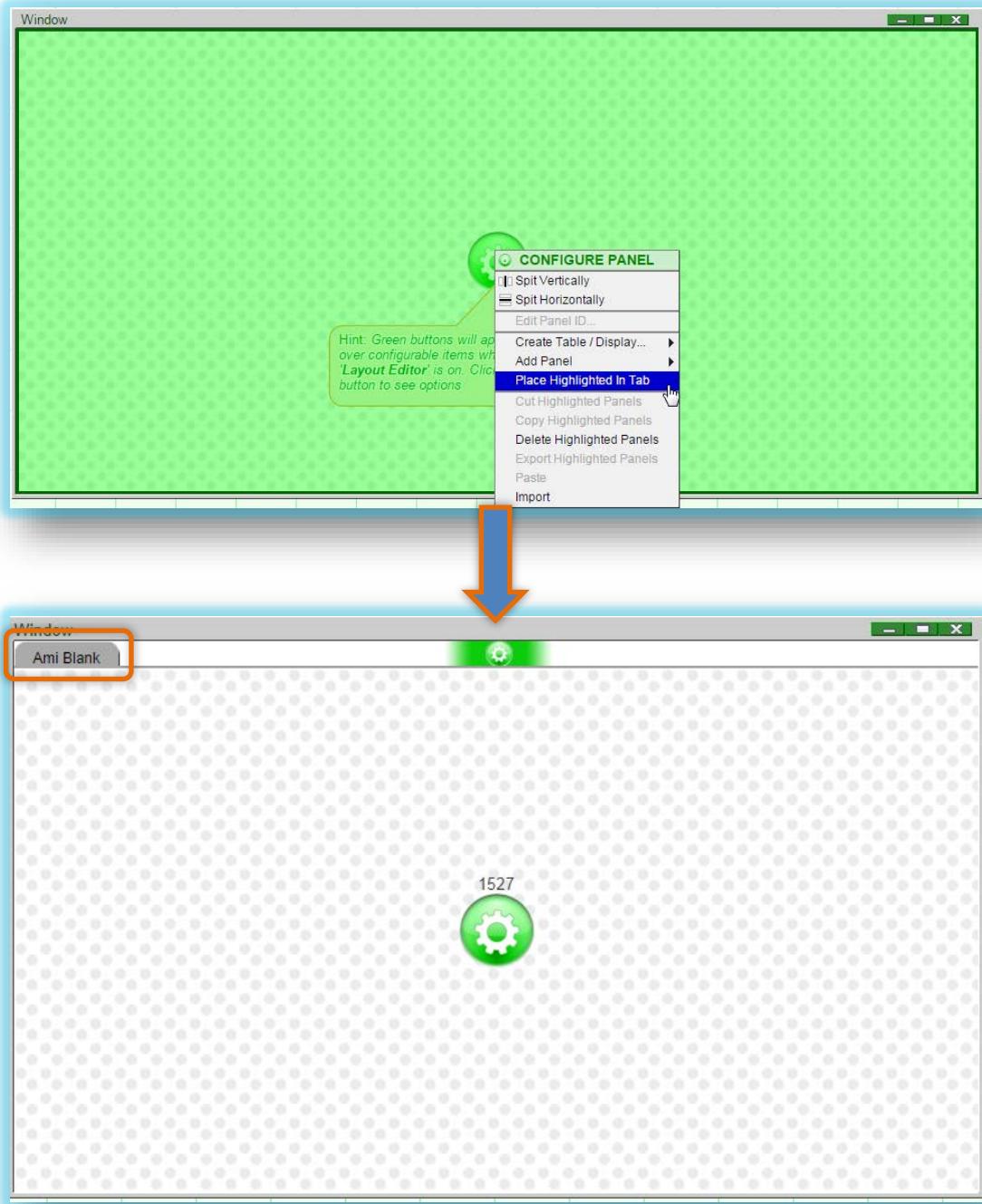
- The Divider lines can be resized in order to appear normal, thinner, or invisible.
Note: hidden dividers cannot be adjusted by the user in work mode.



Using Tabs

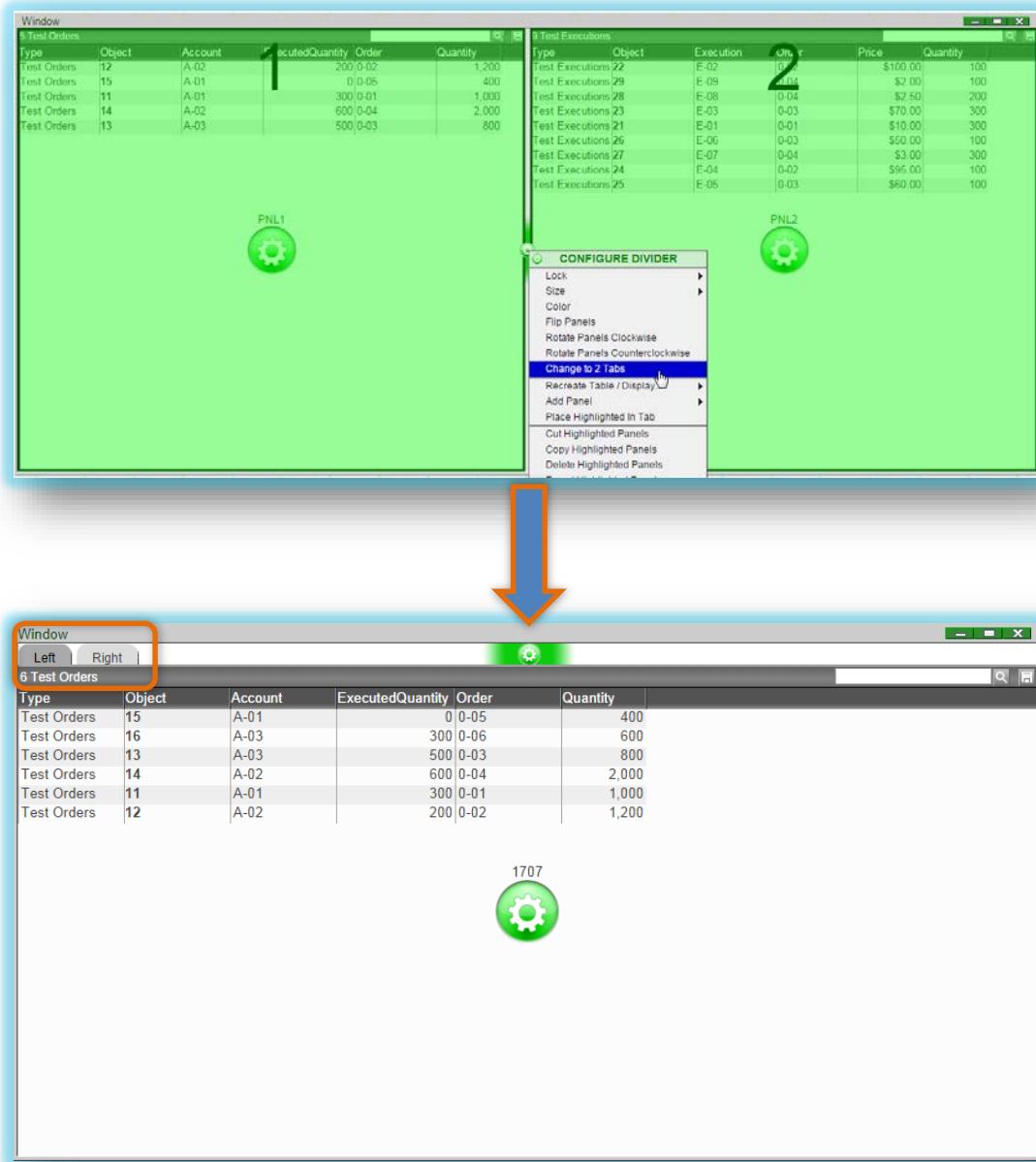
Individual panels, groups of panels, and entire windows can be placed into tabs

- In order to place individual panels into tabs, click on the configuration button of the panel and select **Place Highlighted in Tab** from the menu

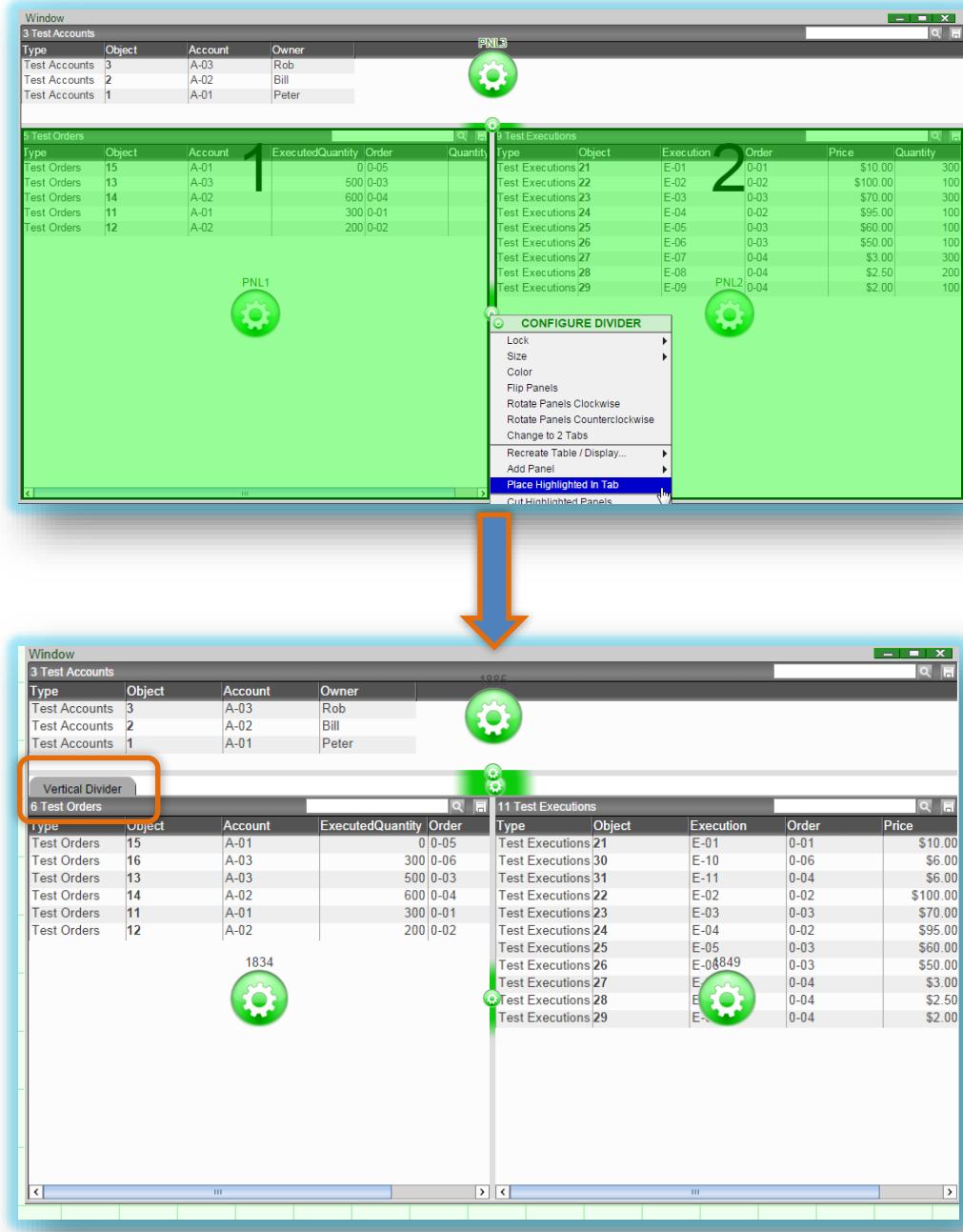


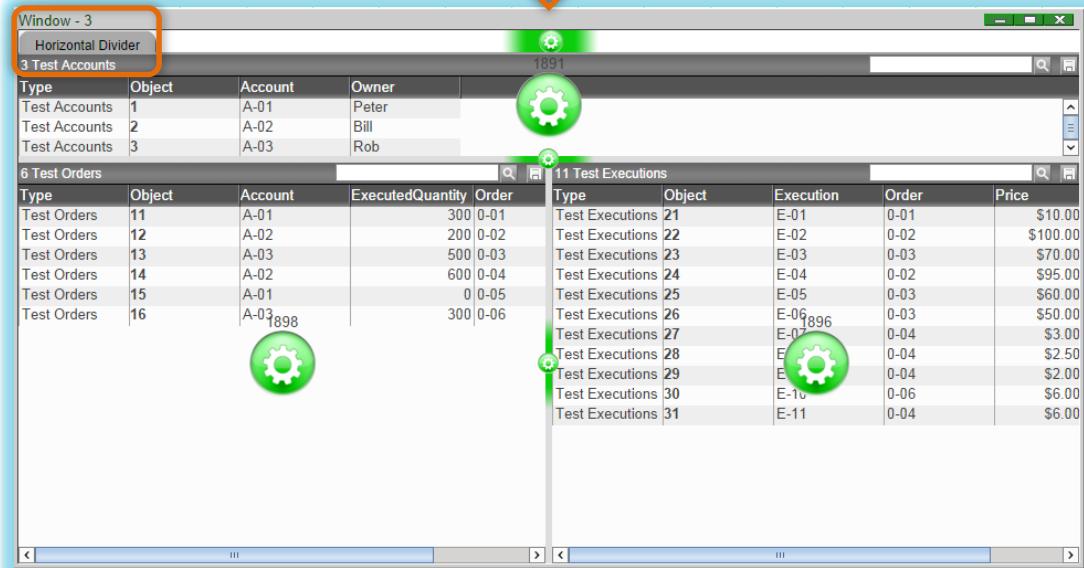
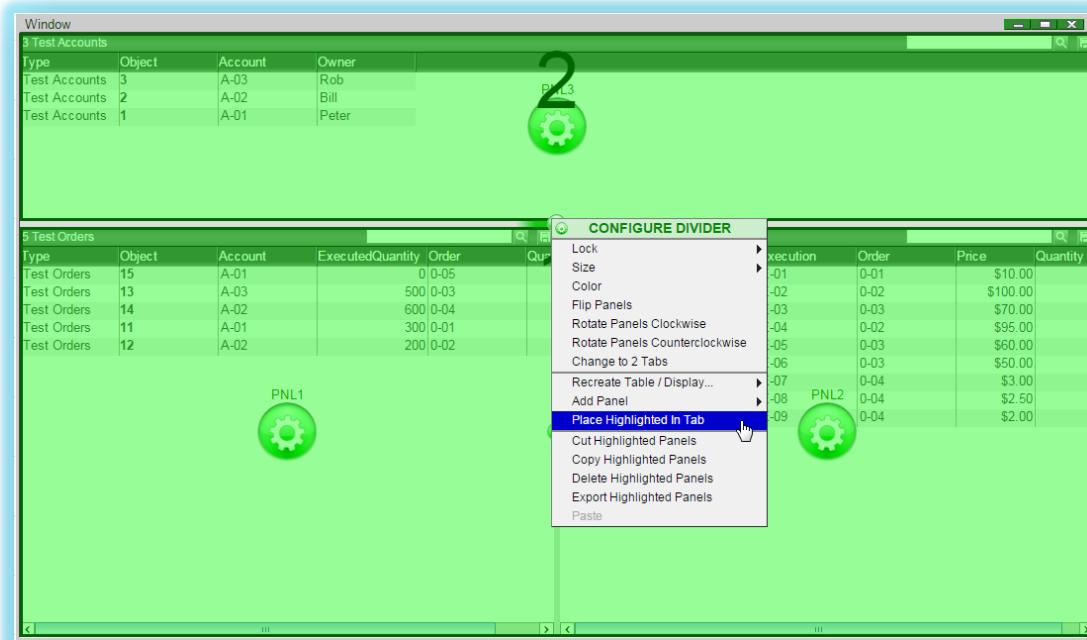
If the window consists of 2 or more panels, they can be separated into 2 tabs or can be placed into one tab in its entirety.

- To place the panels into 2 tabs, select **Change to 2 Tabs** from the divider menu



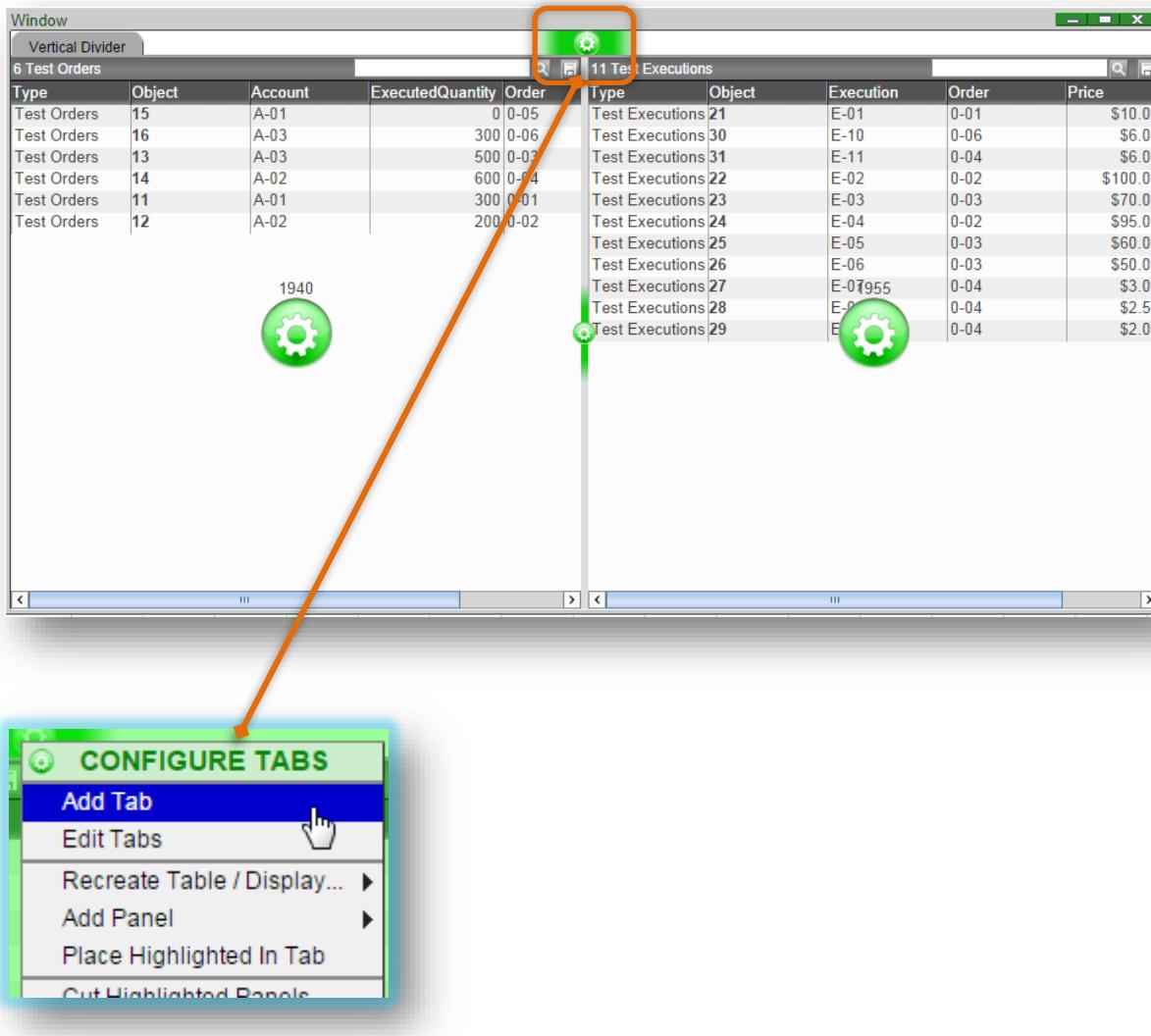
- To place multiple panels or the entire window into a tab, choose the divider configuration button which highlights the panels or the entire window and choose **Place Highlighted In Tab** in the menu





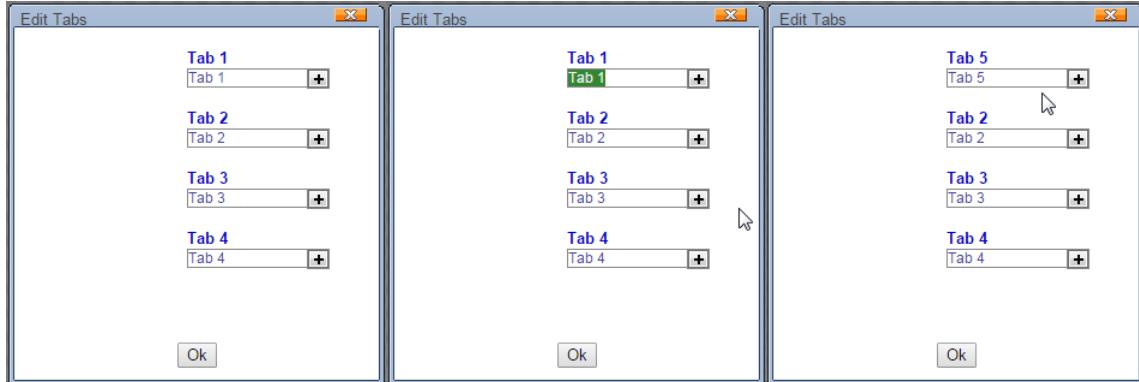
Adding, Editing, & Deleting Tabs

To add additional tabs, click on the configuration button in the same heading as the tabs in order to bring up the **Configure Tabs** menu. Select **Add Tab**

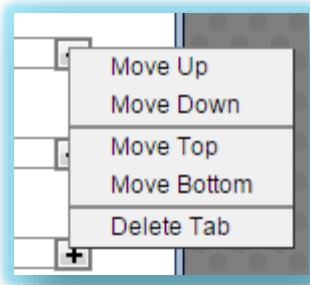


The **Edit Tabs** menu makes it easy to change the name and order of the tabs. Tabs may also be deleted through this option.

To change the name of a tab, highlight the name field and assign a new name:



To rearrange the position of the tabs or to delete a tab, click on the  icon for the following options:



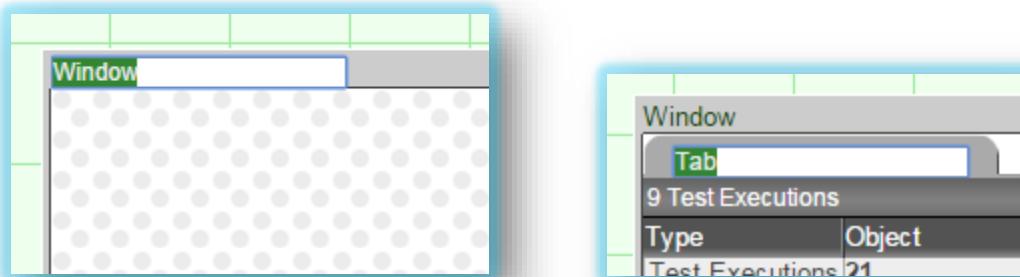
Another way to delete tabs is to click on the divider button which will highlight all of the panels in the tab. Then in the menu, select **Delete Highlighted Panels**.

Editing Names of Windows, Tabs, & Tables

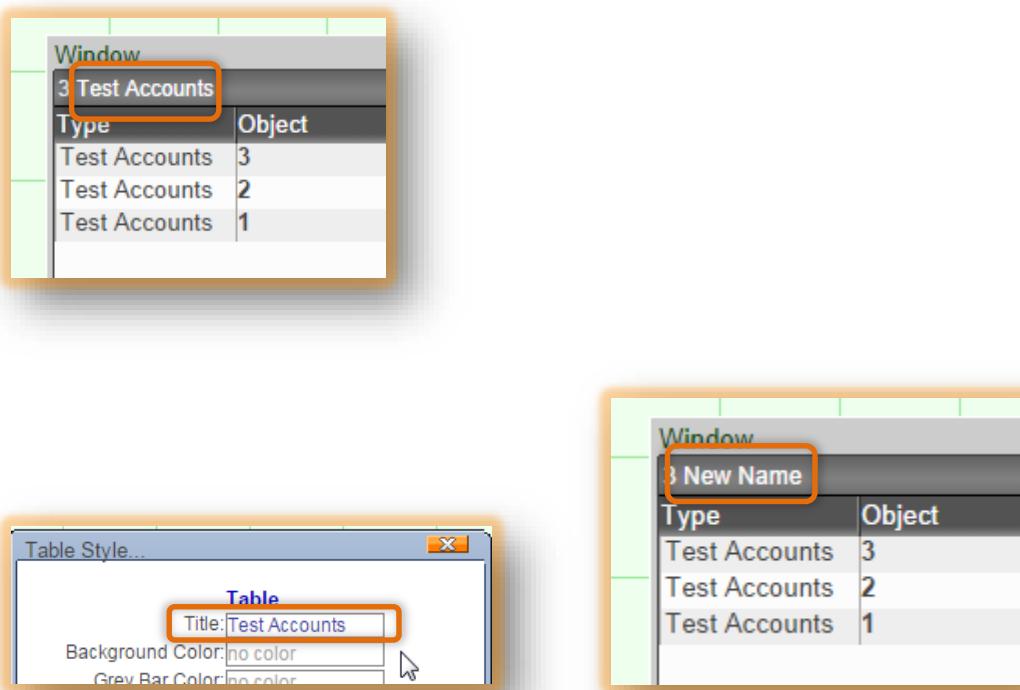
Names of windows and tabs can be changed easily (Note: **Layout Editor** must be ON).

To change the name of a window or tab, double click on the current name (default: *Window*).

Change the name and press enter or click away from the name



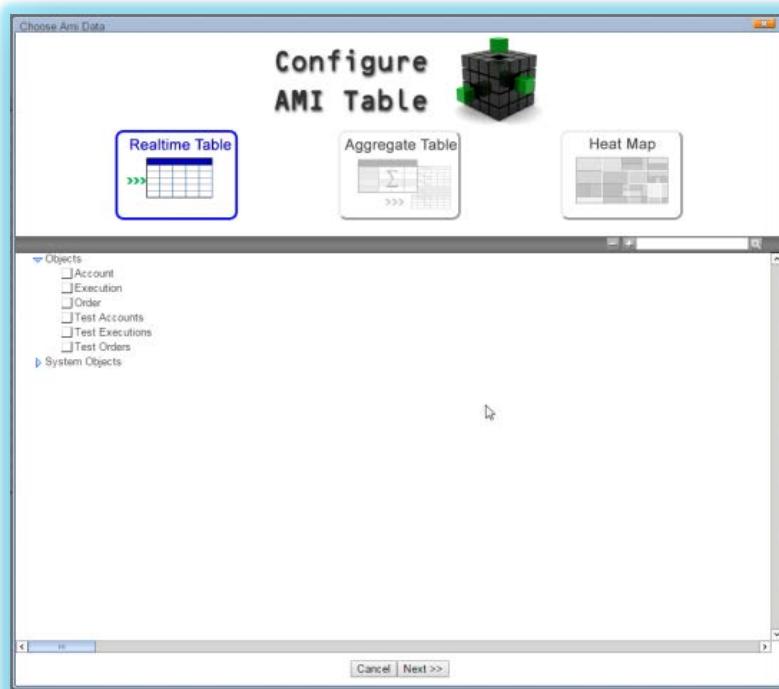
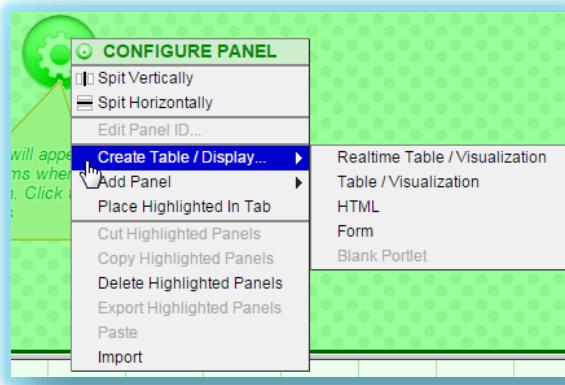
To change the name of a table, click on the configuration button of the panel and select **Style** from the menu



In the style window, change the name under **Table Title** and click on **submit**

Real-time Tables/Visualizations

In AMI, there are many table and display types available for use in the windows. When a new window or panel is created, it is a blank portlet. To display real-time data, a table or display type must first be chosen through the wizard. Once established, windows/panels can later be recreated to other table/display types. To access the real-time wizard, click on the **green configuration button** and select **Realtime Table/Visualization** under the **Create Table/Display** menu (Note: **Layout Editor** must be ON)



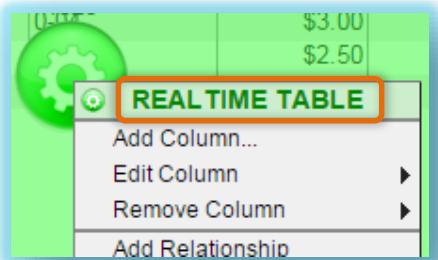
Realtime Table/Visualization

Real-time Table Data is displayed and updated in real-time

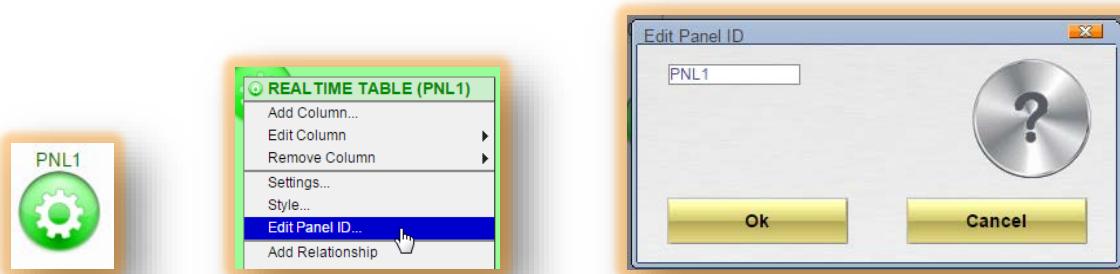
Aggregate Table Displays a summary of the data in a table

Heat Map Visual representation of data using colors

Clicking on the configuration button of a panel will tell the user what type of table is being displayed.

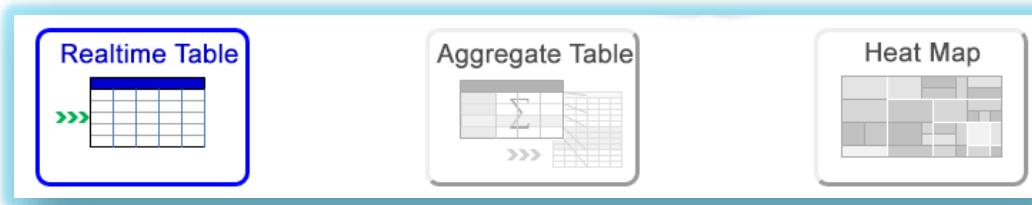


Panels are also given unique ID's, which can be found above the configuration button. **Panel ID**'s can be edited through the configuration button.



Real-time Tables

Real-time tables display data in real-time; adding, deleting, and updating (changing) data as it is happening in the backend. With real-time tables, there is no need to refresh the table as data is pushed to the front end.

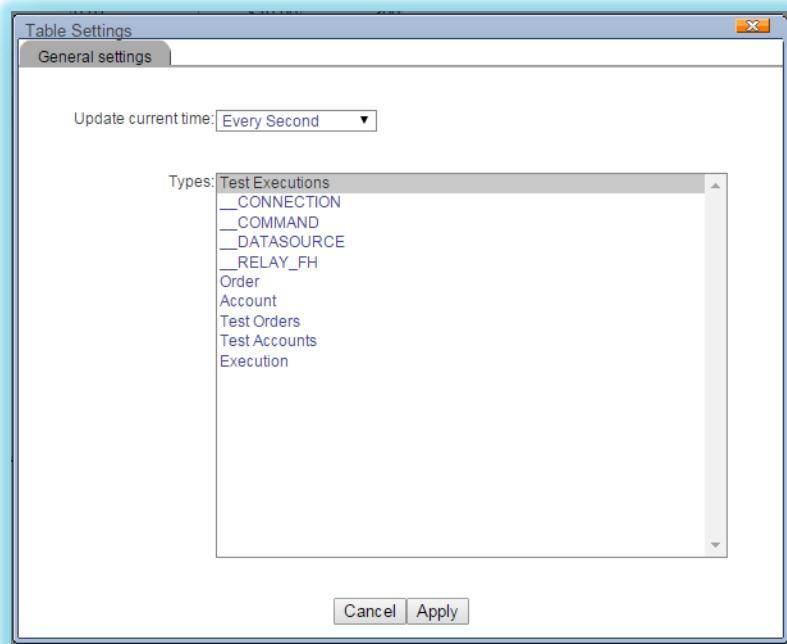


- Real-time tables are useful when it is necessary to quickly view new data that is coming into the system or to view the changes that are being made to existing data
 - Setting up a relationship to a real-time table will display related data in real-time (as objects are highlighted in the **Source** table)
 - Note: Refer to the **Relationship** section for the steps in setting up a relationship
- 1) In order to create a real-time table, click on the green configuration button and select **Realtime Table/Visualization** under **Create Table / Display**
 - 2) In the *Configure AMI Table* wizard, select **Realtime Table** and choose the data to be displayed in real-time. Click Next to configure the real-time table
 - 3) In the second part of the wizard, assign a table name. The names of the columns and formatting can be customized in this part of the wizard
 - Note: AMI will automatically detect the proper formatting of the variables but these can be changed under **Formatting**

The screenshot shows a configuration interface for a table. At the top, there is a 'Table name' field containing 'Test Executions'. Below this, there are two sections: 'Column Names' and 'Formatting'. The 'Column Names' section lists four columns: 'Order', 'Quantity', 'Execution', and 'Price', each with a checked checkbox next to it. The 'Formatting' section lists four corresponding dropdown menus: 'Text', 'Numeric', 'Text', and 'Price'.

Column Names		Formatting
Order:	<input checked="" type="checkbox"/> Order	Text
Quantity:	<input checked="" type="checkbox"/> Quantity	Numeric
Execution:	<input checked="" type="checkbox"/> Execution	Text
Price:	<input checked="" type="checkbox"/> Price	Price

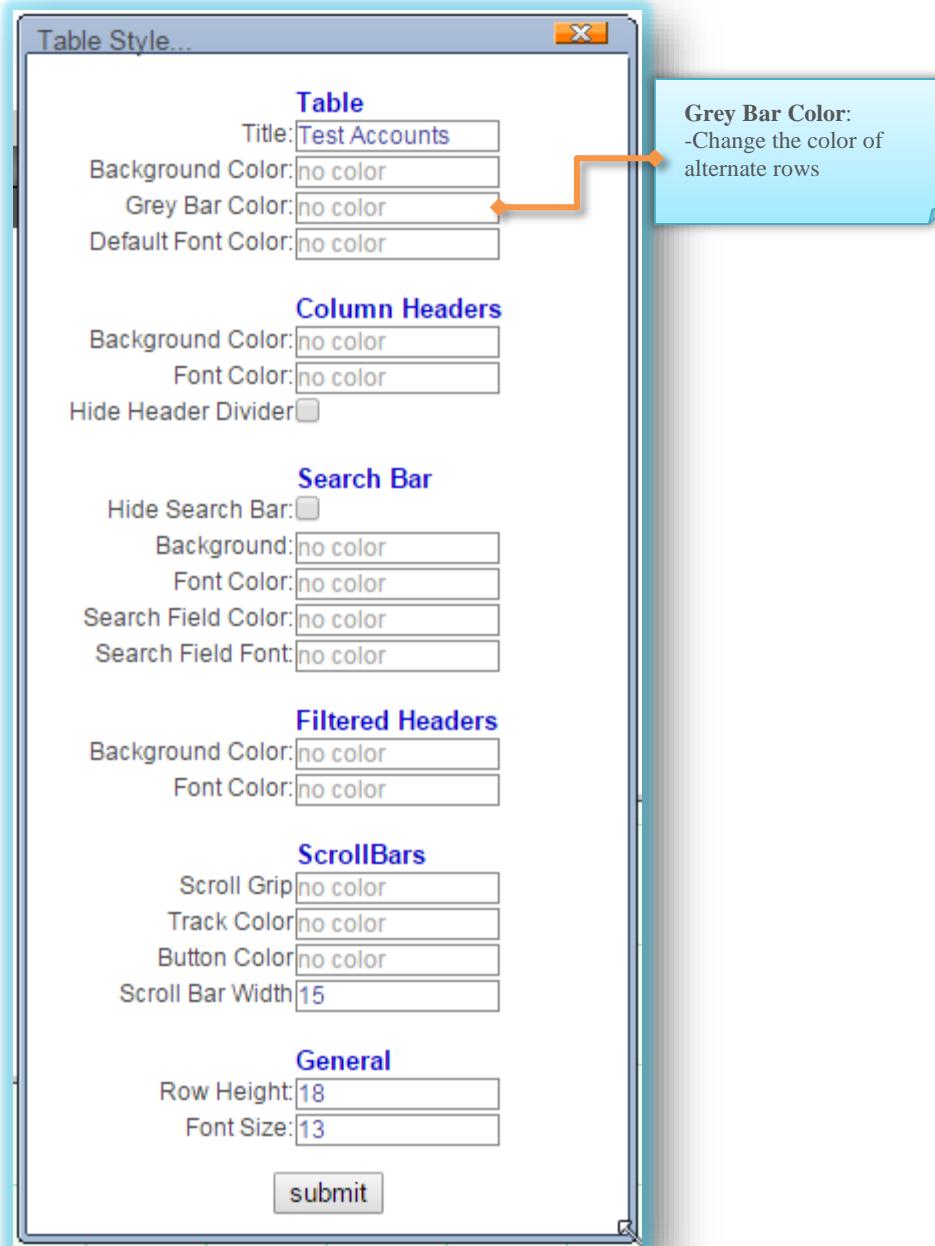
Real-time Table Settings



Update current time	Modify how frequently the Current Time is updated
Types	Data Types , the type of data included in the table can be changed. Any new data selected will use the columns of the <u>original</u> data. To change the table type, use the Recreate Table/Display option instead.

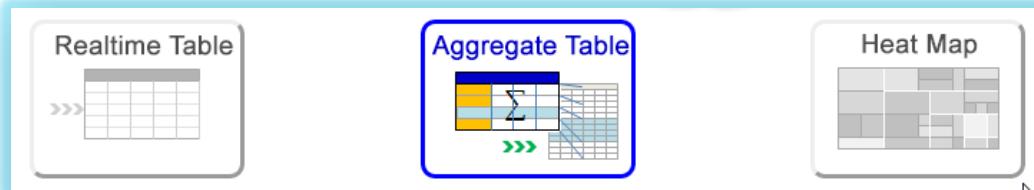
Table Style

The look and feel of tables in AMI is highly customizable through the **Style** option in the configuration button menu.



Aggregate Tables

Aggregate tables are real-time tables that can be used to obtain a summary of the data grouped by certain variables. Aggregate tables will integrate changes to the data in real-time.



- 1) In order to create an aggregate table, click on the green configuration button and select **Realtime Table/Visualization** under **Create Table / Display**
- 2) In the *Configure AMI Table* wizard, select **Aggregate Table** and choose the data to be summarized. Click Next to configure the aggregate table
- 3) In the second part of the wizard, assign a table name and choose the variable(s) that the data is to be grouped under in addition to the types of aggregations to be performed. Variables can also be hidden and not used in aggregation. Note: Aggregation in AMI follows the Group By paradigm in relational databases
 - Types of aggregations: **Count, Sum, Max, Min, & Average**
 - Note: Advanced groupings and aggregations on derived values can be done after the table is created (not in the wizard)

Table name		
Execution		
	Column Names	Formatting
OrderID:	Hidden	Text
Account:	Group By	Text
ExecutionTime:	Hidden	Date & Time
Quantity:	Sum	Numeric
LastMarket:	Group By	Text
Price:	Average	Price

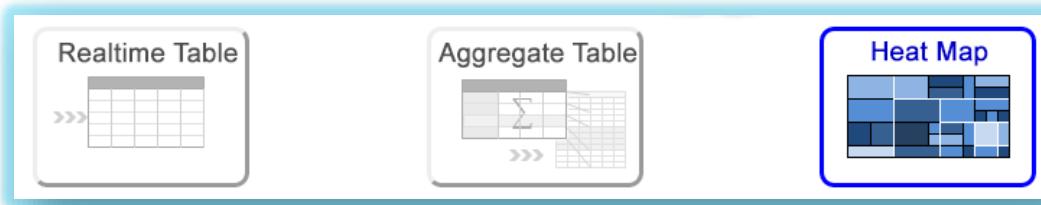
Window - 2

48 Execution

▲ Account	LastMarket	Count	Avg Price	▲ Total Quantity
3F-00000	BATS	215	\$48.57	42,700
3F-00000	Euro	214	\$47.52	43,400
3F-00000	AMEX	214	\$45.38	44,400
3F-00000	NITE	227	\$45.89	45,500
3F-00000	NYSE	678	\$46.92	132,100
3F-00000	NASD	1,951	\$47.11	391,500
3F-00001	BATS	206	\$46.05	42,100
3F-00001	AMEX	214	\$46.56	44,600
3F-00001	Euro	217	\$44.91	44,600

Heat Maps

Heat maps are real-time graphs which allow the user to visualize complex data using different colors and size; enabling a quick interpretation of large amounts of data. Various colors can be assigned to a range of values and data is displayed in real-time.

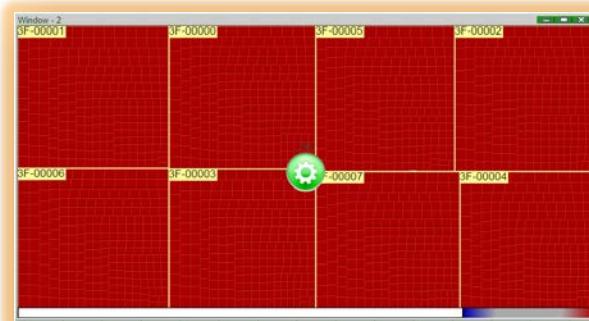


- 1) In order to create a heat map, click on the green configuration button and select **Realtime Table/Visualization** under **Create Table / Display**
- 2) In the *Configure AMI Table* wizard, select **Heat Map** and choose the data to be displayed in the heat map. Click Next to configure the heat map
 - In the second part of the wizard, assign a name and the variables to the configurable items of the heat map – *Top Level Grouping, Grouping, Size, & Heat*.
 - Note: Advanced groupings, size and heat formulas on derived values can be done after the heat map is created (not in the wizard)

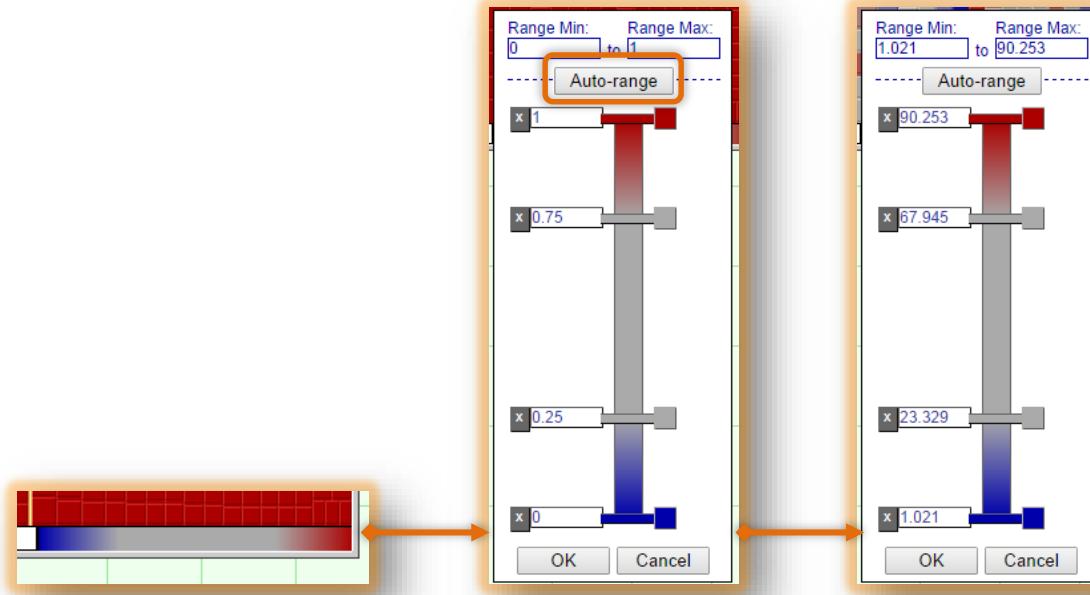
Example – setting up a heat map of *Executions*

This screenshot shows the 'Configure AMI Table' wizard. At the top, there is a 'Table name' input field containing the value 'Executions'. Below this, there are four dropdown menus for configuring the heat map:

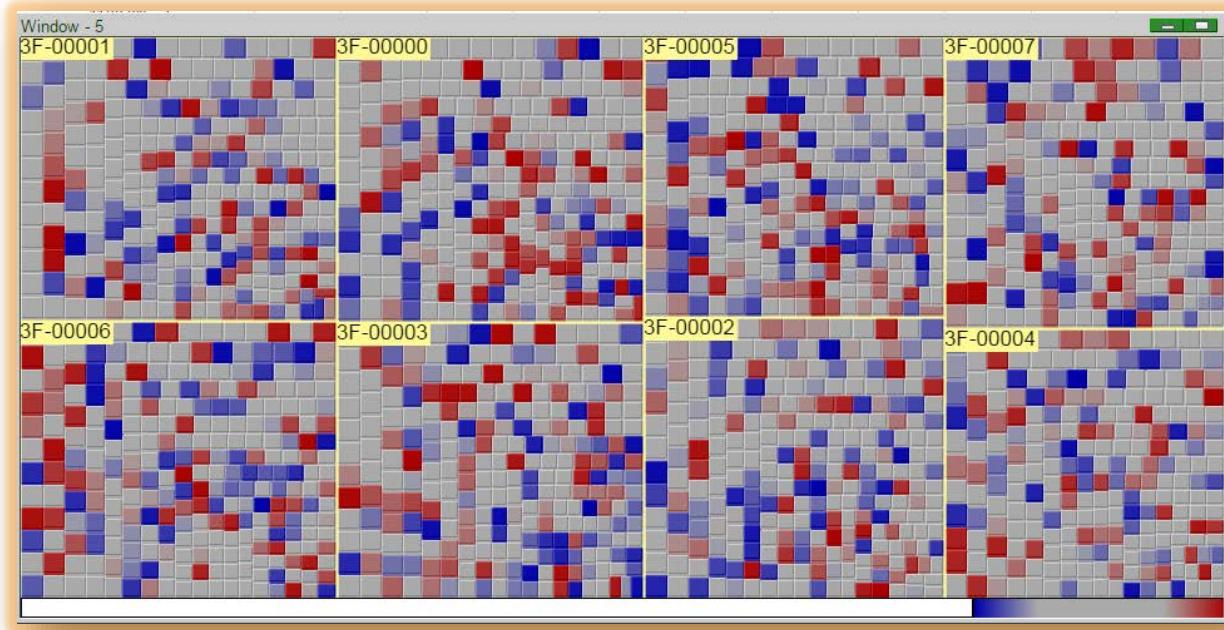
- Top Level Grouping:** Set to 'Group By' with 'Account' selected.
- Grouping:** Set to 'Group By' with 'OrderId' selected.
- Size:** Set to 'Sum' with 'Quantity' selected.
- Heat:** Set to 'Average' with 'Price' selected.



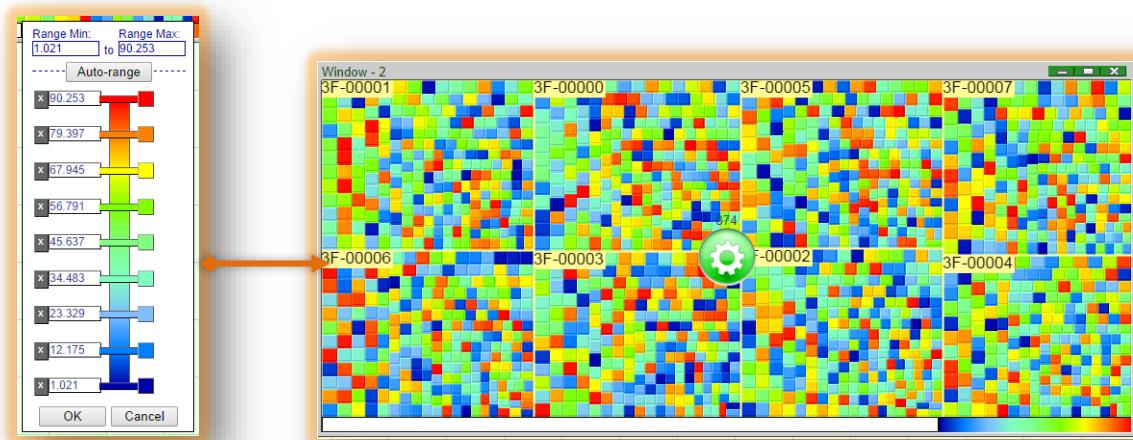
- 3) Click on the color bar on the lower right corner in order to choose the numbers and colors to be used in the heat map.



- Selecting **Auto-range** will automatically determine the values to be assigned to the colors, along with the **Range Min & Range Max**



- Additional points can be assigned along the gradient to represent a smoother transition between the values. In order to add points, click on the gradient and assign numbers and colors



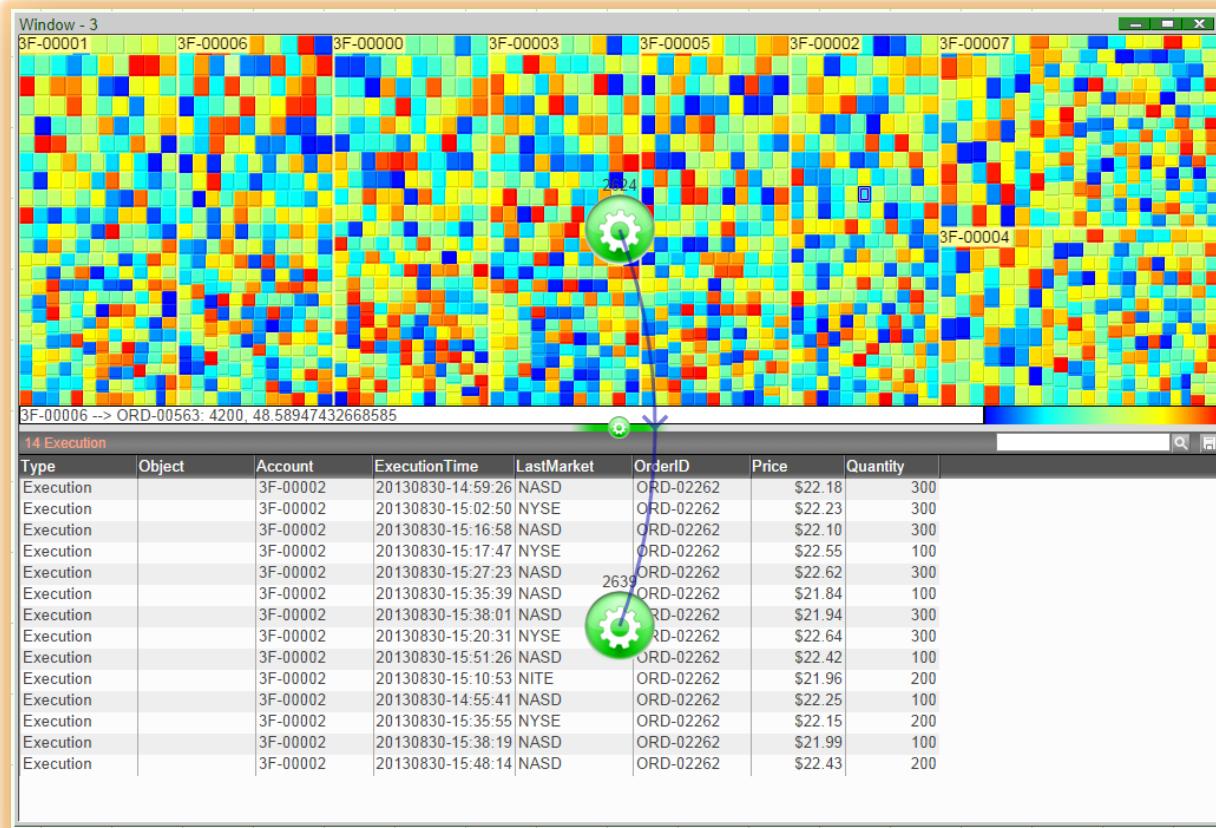
The formula used for a heat map can be modified after it has been established.

- To modify the formula of a heat map, click on the configuration button of the heat map and select **HeatMap Formulas** from the menu. This will bring up the menu to modify the formulas being used for the current heat map



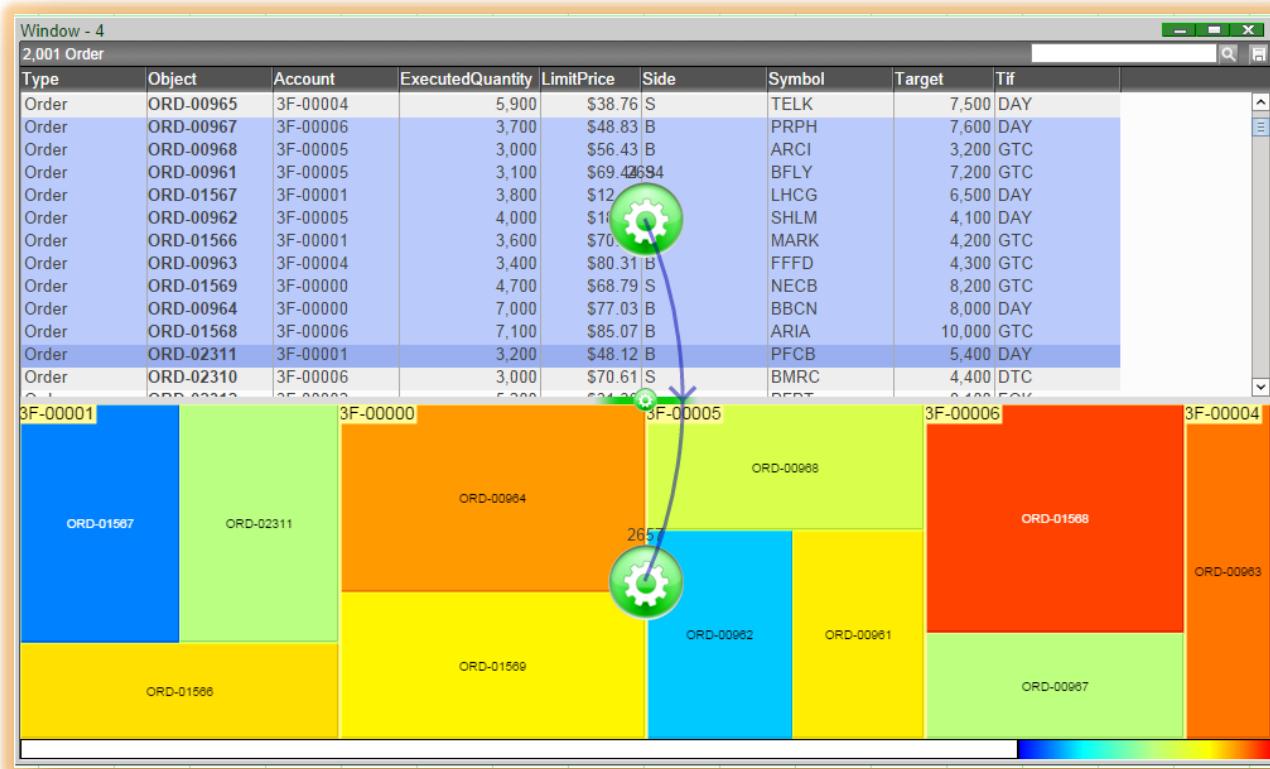
Example – linking heat maps to real-time tables/linking real-time tables to heat maps

- Linking a heat map to a real-time table will display the related data on the real-time table; enabling the user to view the exact objects that are being represented in the heat map.



Note: holding **Ctrl** while clicking on the objects in the heat-map will allow the selection of multiple objects

- Linking a real-time table to a heat map will display a heat map in real time as objects are selected in the real-time table



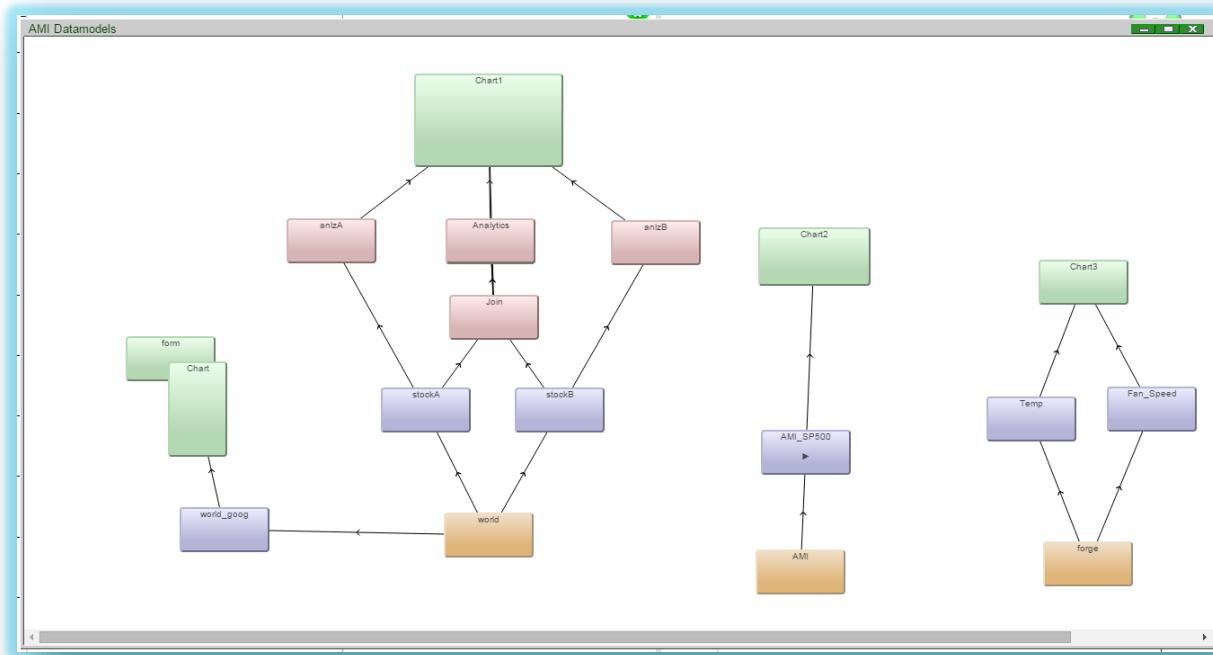
Heat Map Settings & Style

- Heat Map Settings:** modify the stickiness of the cells
 - Stickiness determines how frequently the position of the cells changes with new incoming data. The higher the stickiness, the less frequently the positions are changed. The option ranges from 0 (no stickiness) to 9 (very sticky)
- Heat Map Style:** modify the font size

Static Tables/Visualizations

The static visualizations can source both real time AMI data as well as external data sources. This is accomplished using the Data modeling platform, depicted below. This is comprised of three distinct layers:

- Data sources (orange) – Interface to external databases, etc. Note there is one reserved data source called "AMI" which represents AMI's internal real-time database
- Data Models (purple and pink) – Retrieves data from data sources for normalization and transformation. Then the resulting data is sent to the visualization layer
- Static Visualizations (green) – All static "front end" components.



In the following sections we will discuss all three of the concepts in detail.

Data Sources

AMI allows users to connect to databases from the front end. Connecting to a database is done from the managed datasources option under the data drop down menu. Note: Datasources are global to the entire AMI environment, meaning all users and layouts will be affected by changes made to datasources.

When the datasources window is opened, it will display all of AMI's existing connections with databases. This window will be blank before any datasources are added. Right clicking in the blank space (if there are datasources, then the empty space below the existing datasources) will give the user the option to add a database connection. Once a database connection is established, the data will be available for use in the dashboard. The form for adding a database has 6 fields:

1. Name - enter an alias used to identify the database. This is the name that is referenced by tables and other visualizations. NOTE: When copying visualizations between instances of AMI (ex your QA and PROD environment) it is important that the aliases are the same (case sensitive), otherwise you must update the datasource name in the visualization.
2. Adapter - choose the type of database from the list of available connections. See the Appropriate Adapter's AMI manual for details
3. URL - this is the network path where the database exists
4. Username - credentials for logging into the database.
5. Password - credentials for logging into the database.
6. Options - if the database requires some other options to be specified for the connection, this is where those options would be added.

Right clicking on an existing datasource will give the user two options:

1. Delete Datasource - this will remove the connection with the database for all users of that instance of AMI. Note: visualizations that depend on the deleted datasource will no longer be valid.
2. Edit Datasource - this brings up the same form as adding a new datasource allowing the user to edit the original fields from the creation of the connection.

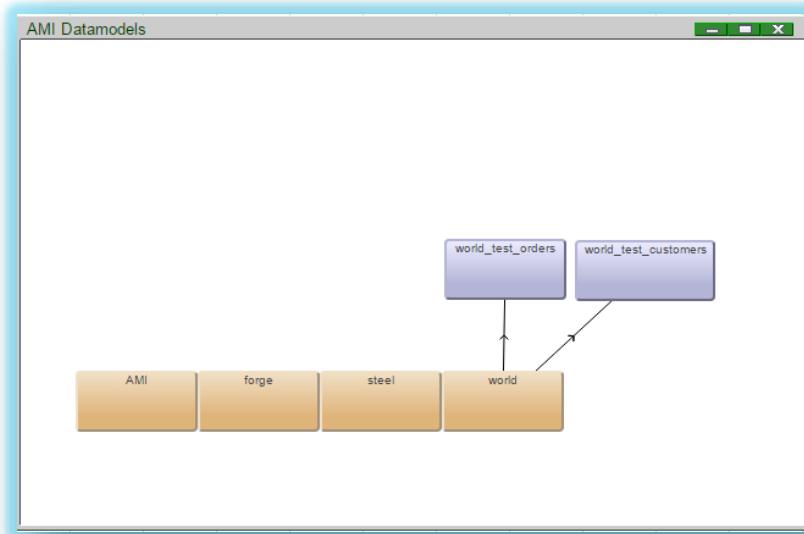
Data Models

Data models are essentially layers that are created with data obtained from databases. All static tables/visualizations will be based on a datamodel. The **datamodel platform** will show all of the datamodels being used in the current layout.

Datamodels are created in two ways:

- Creation of a new static table/visualization using the **configuration button**
- The **Datamodels** option under **Data** in the menu bar
 - This option may be used to create datamodels in preparation for their later use in tables/charts

This is the **datamodel platform** – it is a map containing all of the datamodels created for use in the layout:

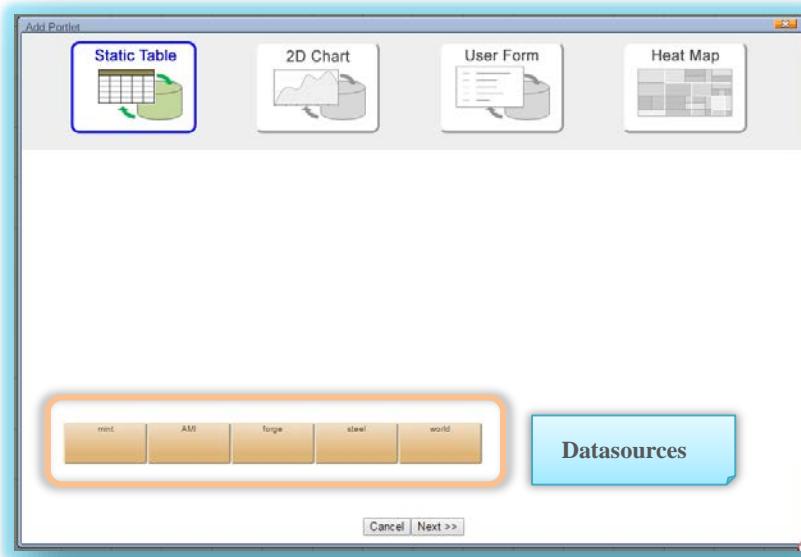


To create a new datamodel, right click on a datasource or datamodel icon and select [**Add Datamodel to...**]



Static Tables

Static tables are used to display the data from **Data Sources** and **Objects**. Static tables display data as it is at the moment of the query and is updated upon a new query.

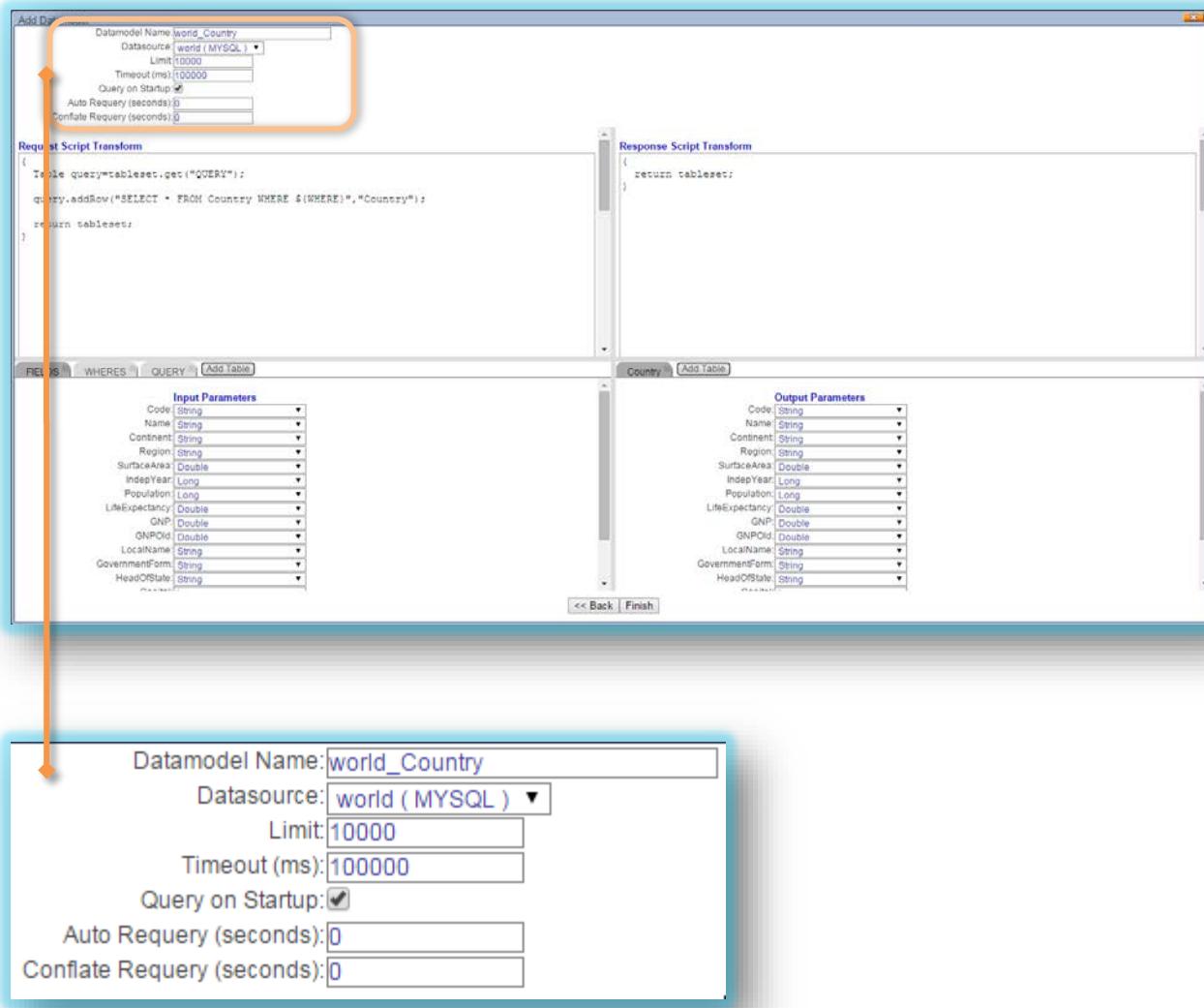


- 1) In order to create a static map, click on the green configuration button and select **Table/Visualization** under **Create Table / Display**
- 2) In the datamodel platform, select **Static Table** and the datasource from which the data will be retrieved. Click Next to choose a specific table from the datasource.



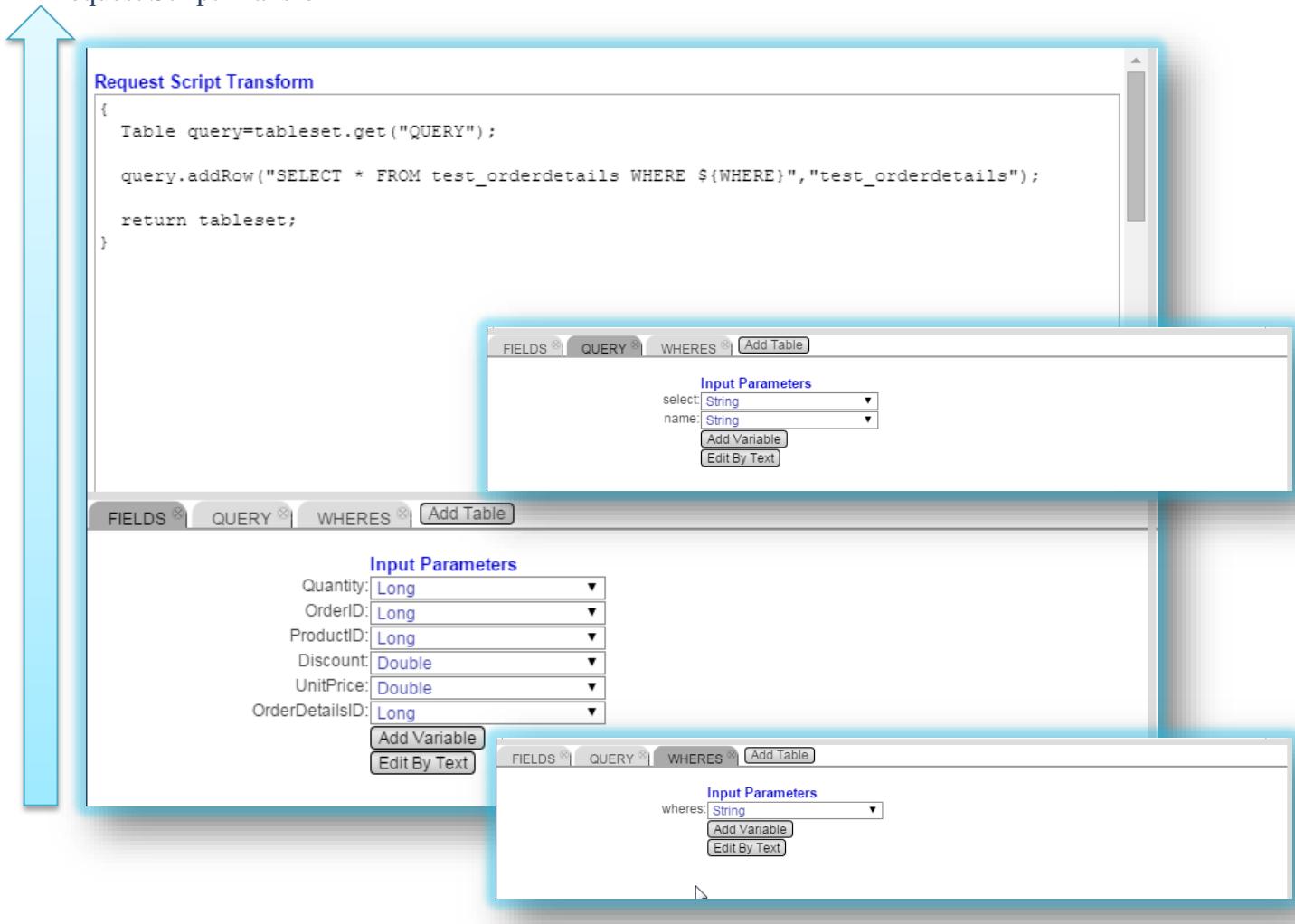
- 3) Click Next after selecting a table to view the **Add Datamodel** window (*please refer to the following sections for details on the Add Datamodel window*)
- 4) After creating the datamodel, it will appear in the datamodel platform (already selected). Click Next to view all of the variables and make any necessary changes to their parameters. Click Finish to create the static table.

Add Datamodel



Datamodel Name	The name of the data model – the name of the database followed by the data set name
Datasource	Drop down menu listing all of the datasources that are currently connected to AMI
Limit	Modify the maximum number of data returned in response to a query
Timeout (milliseconds)	Change allowed time for a query to run
Query on startup	Option to have the data appear from the start
Auto-requery (seconds)	Change the frequency (in milliseconds) of automatic re-queries, using a high-frequency re-query will update the table/chart with current data
Conflate Requery (Seconds)	Specify the amount of time that must pass before another query can be made to the datamodel

Request Script Transform



The request script is the message that is sent to the database to retrieve the data to be used in the datamodel.

FIELDS

- A column is created per parameter in the fields tab
- Field mapping (configured in the relationship)

QUERY

- Uses the query established in the **Request Script Transform**

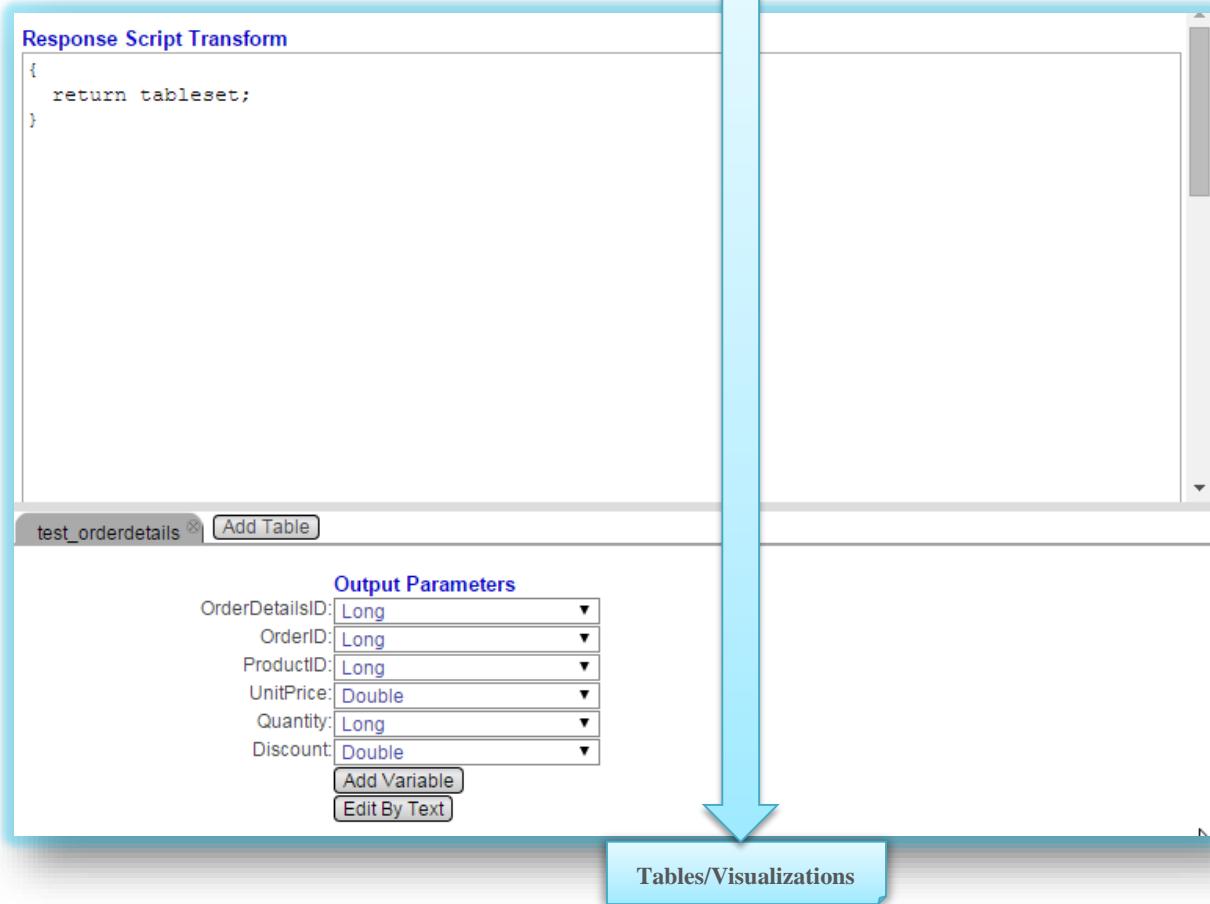
WHEREES

- References the **Query Clause** in relationships

Please refer to the **AMI-SQL** manual for further details

Response Script Transform

Request Script Transform



In the response side, the script takes the data that was retrieved and pushes it out to the static tables/visualizations. AMI SQL can be used in the response script transform side to perform such actions as joining multiple tables.

Relationships in AMI require a **fields** table and a **wheres** table. Fields are adapter specific and are mainly used for custom adapters.

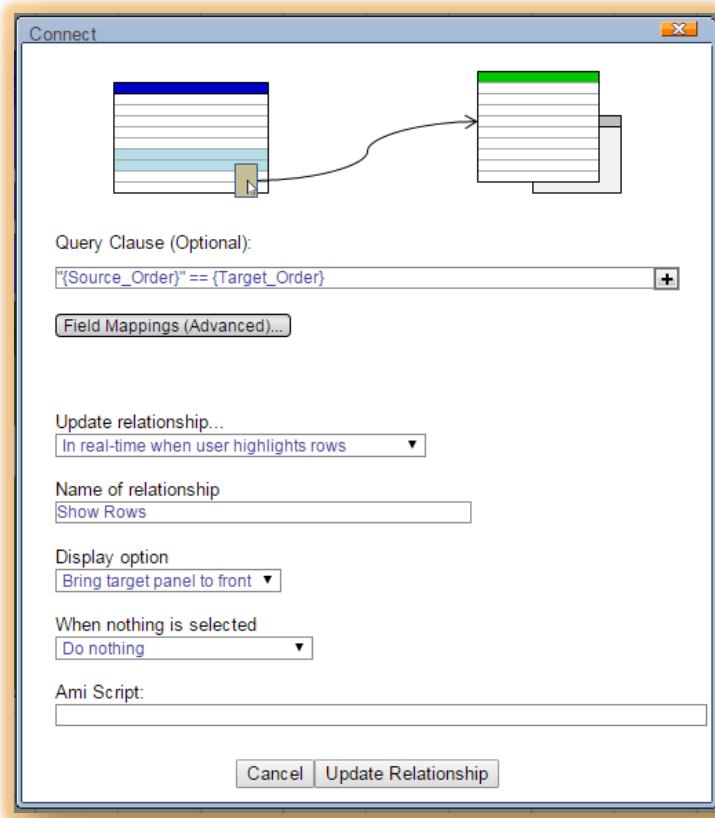
Adapters take the **\${WHERE}** and replace it with the statement in the relationship's **Query Clause** and returns a table (e.g., "test_orderdetails"). This name should match with the name seen below the Response Script Transform (by default and without changes)

Please refer to the [AMI-SQL](#) manual for further details

Creating Relationships to Tables Based on Datamodels

With the new data model system, the way relationships are established has changed. When creating a link to a static table, variables of the source table must be **enclosed in quotations and {}**

Example – creating a relationship from Orders (RT) to Executions (static/data model based) based on the variable **Order**

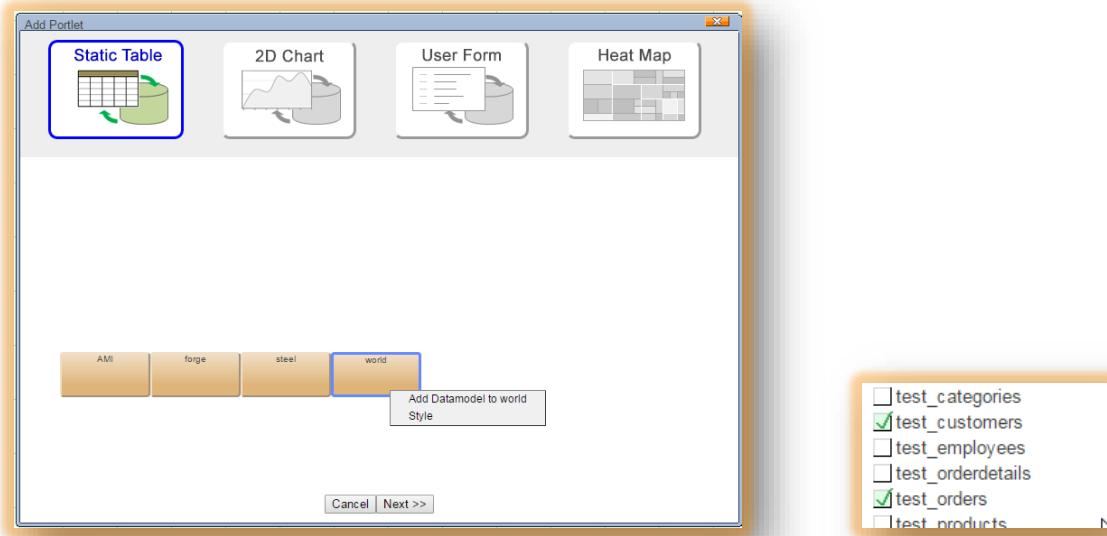


Relationships to static tables may also be updated in real-time – i.e., the target table will update as the user clicks on objects in the source table (without the need for right-click menus)



Example – creating a static table that joins two tables using datamodels

- 1) Click on the panel configuration button and select **Table/Visualization** under Create Table/Display in order to bring up the datamodel platform
- 2) In the datamodel platform, select **Static Table** and right click on the database in order to bring up the option to add a datamodel
- 3) Select **Add Datamodel to [database]** and select the two tables you would like to join.



- 4) In the **Add Datamodel** window, make any necessary changes to the datamodel options (e.g., query on startup etc.). Enter the script for the join in the **Response Script Transform** box (Please refer to the AMI SQL documentation for further details)

```

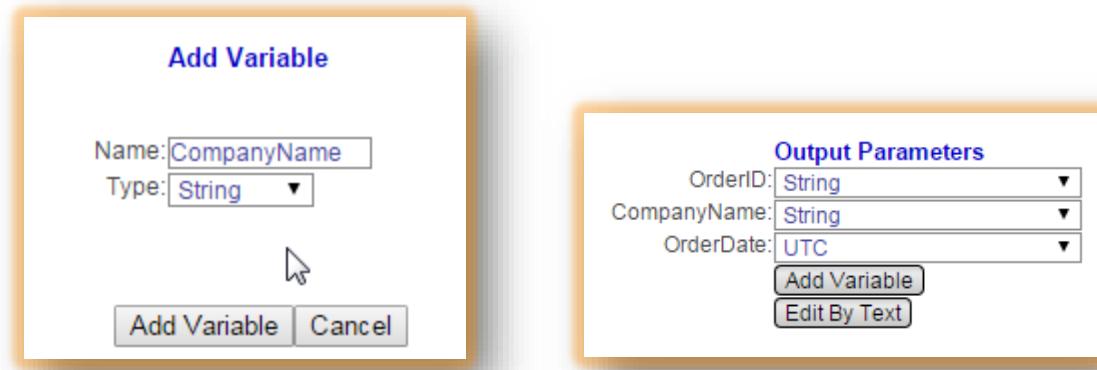
{
  tableset.query
  ("CREATE TABLE join AS
  SELECT test_orders.OrderID AS OrderID, test_customers.CompanyName AS CompanyName,
  test_orders.OrderDate AS OrderDate
  FROM test_orders, test_customers
  WHERE test_orders.CustomerID==test_customers.CustomerID");
  return tableset;
}

```

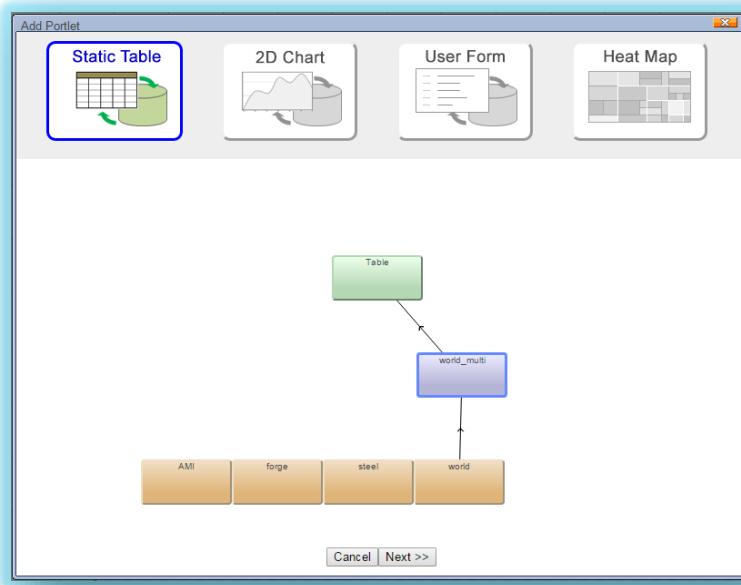
- 5) Create a new Table Schema by clicking on **Add Table**. Input the same name from the JOIN statement (in this example – “join”). Once the table is created, add the variables from the script by clicking on Add Variable under Output Parameters



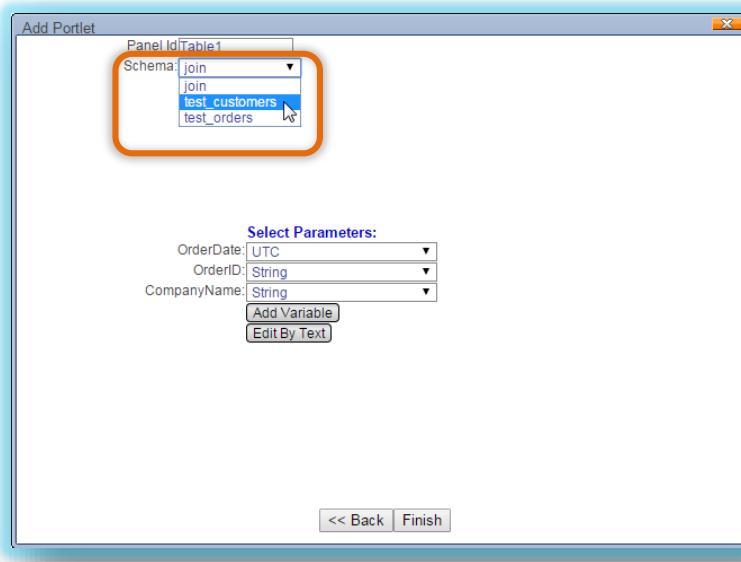
- 6) In the Add Variable window, enter the name of the variable and select the appropriate data format (e.g., Boolean, Integer, Float, etc.)



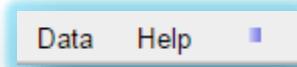
- Once created, datamodels can be accessed for use in other tables and visualizations.
- Different tables and visualizations can be based on the same datamodel – running a query on one table will affect any other table/visualization that is based on the same datamodel



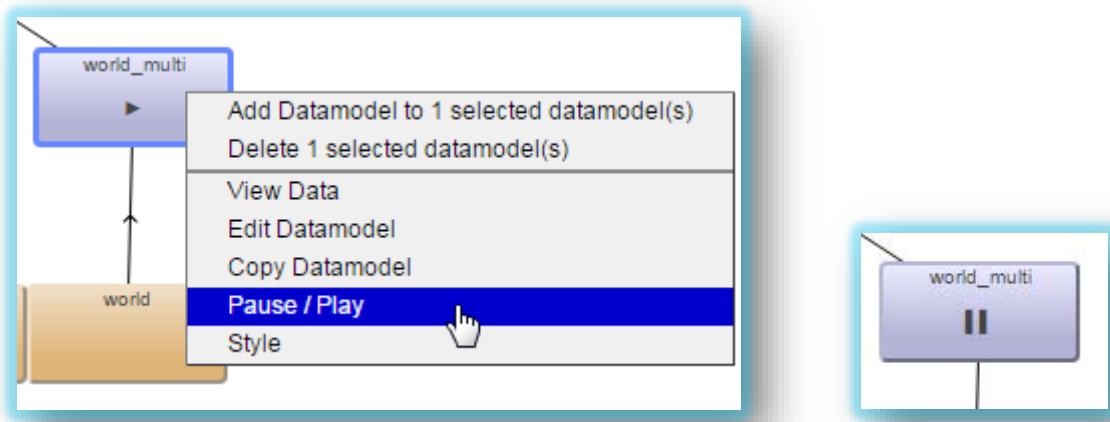
- When selecting an existing data model, you have the option to choose any of the schemas that are available in that data model



- Deleting any table/chart will not affect the data model – they will remain for use in the creation of another table/chart
- Auto re-queries being ran by data models will be indicated by a status icon on the menu bar (next to Help):



Auto re-queries applied to data models may be paused easily in the data model platform. This is done by right-clicking on the data model to bring up the options and choosing Pause/Play



Add Datamodel to [#] selected datamodel(s)	Add a new datamodel to the selected datamodel(s).
Delete [#] selected datamodel(s)	Delete the selected datamodel(s)
View Data	Displays the data that is being returned for each query to the database
Edit Datamodel	Change the settings of the datamodel
Copy Datamodel	Copy the selected datamodel (copied datamodels will be automatically created)
Pause/Play	Pause/resume the re-query
Style	Change the look and feel of the datamodels

Aggregation on Databases

With the AMI datamodel system, aggregations can be performed on databases.

- 1) In order to create a custom aggregate table, click on the green configuration button and select **Table/Visualization** under Create Table/Display
- 2) In the data model platform, right click on the datasource and select **Add Datamodel to [datasource]**



Add DMS

Datamodel Name: world_Country
 Datasource: world (MYSQL)
 Limit: 10000
 Timeout (ms): 100000
 Query On startup:
 Auto Request (seconds): 0

Request Script Transform

```
{
  Table query=tableset.get("QUERY");
  query.addRow("SELECT * FROM Country WHERE ${WHERE}", "Country");
  return tableset;
}
```

Response Script Transform

```
{
  return tableset;
}
```

FIELDS **WHERES** **QUERY** **Add Table**

Input Parameters		Output Parameters	
Code	String	Code	String
Name	String	Name	String
Continent	String	Continent	String
Region	String	Region	String
SurfaceArea	Double	SurfaceArea	Double
IndepYear	Long	IndepYear	Long
Population	Long	Population	Long
LifeExpectancy	Double	LifeExpectancy	Double
GNP	Double	GNP	Double
GNPOld	Double	GNPOld	Double
LocalName	String	LocalName	String
GovernmentForm	String	GovernmentForm	String
HeadOfState	String	HeadOfState	String
Capital	Long	Capital	Long
Code2	String	Code2	String

<< Back | Finish

3) Under the **Response Script Transform**, enter the aggregate SQL statement

Example – creating an aggregate table using sample Country data stored in a database

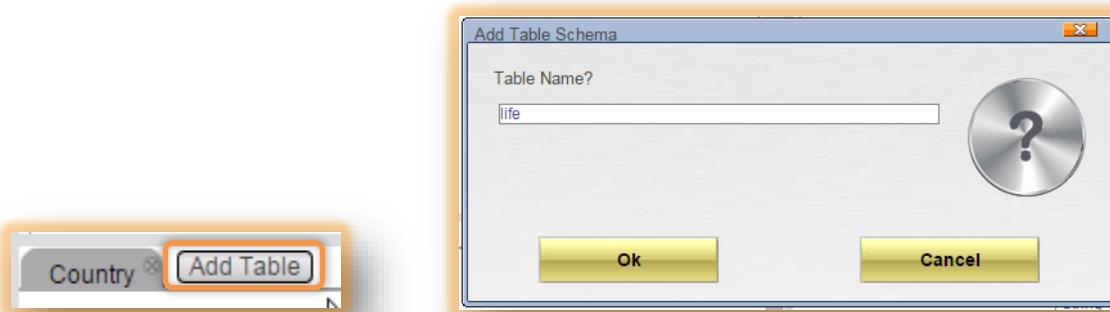
The following script will create an aggregate table showing the *average life expectancy* grouped by *region* and *continent*:

```

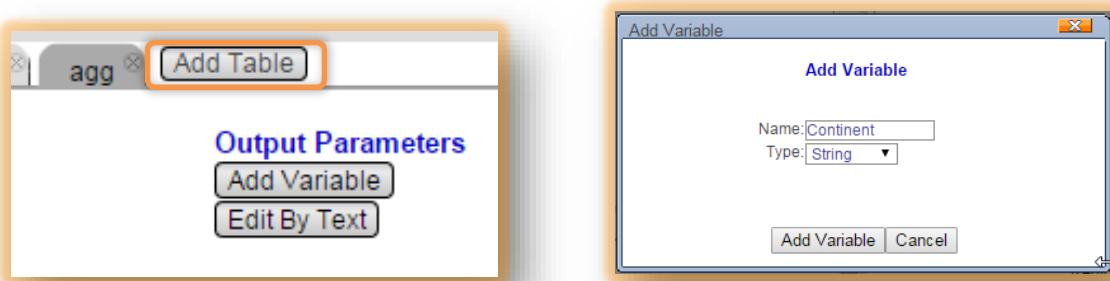
Response Script Transform
{
    tableset.query("CREATE TABLE life AS SELECT avg(LifeExpectancy) AS Life, Continent,
Region FROM Country GROUP BY Region,Continent");
    return tableset;
}

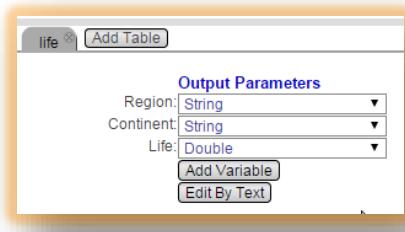
```

After inputting the SQL statement, click on the **Add Table** button below the **Response Script Transform** box and put in the name used in the SQL statement (e.g. Life).



Once the table is created, the variables selected in the SQL statement must be added to the table. To add variables, click on the **Add Variable** button found under Output Parameters

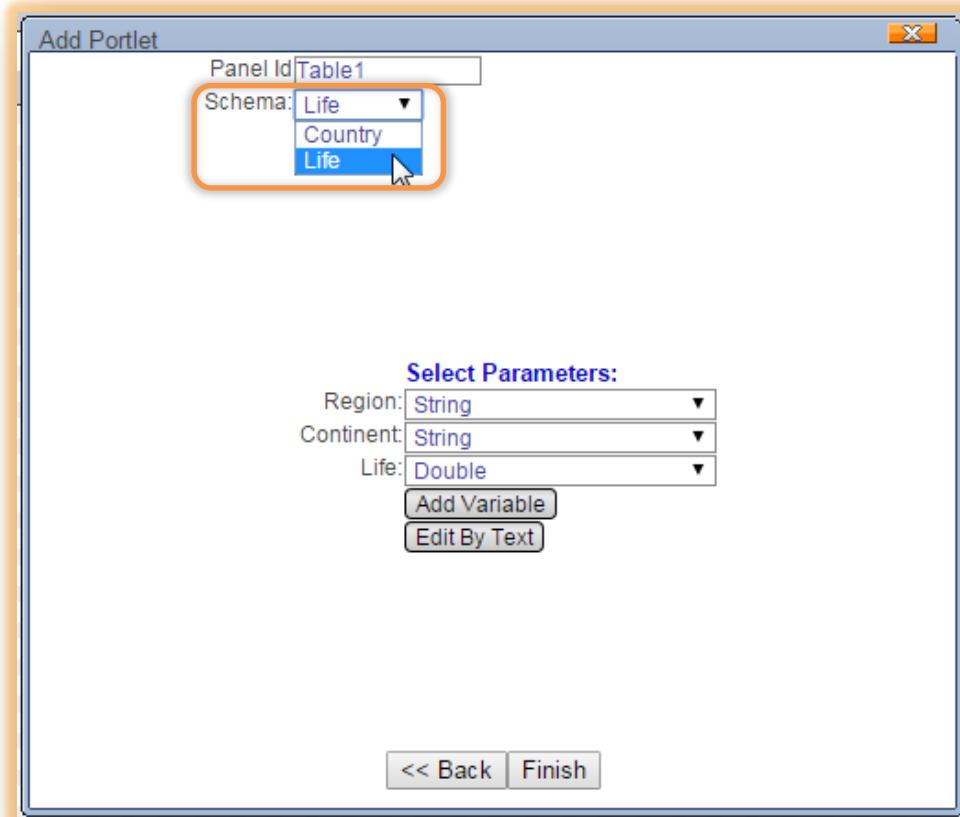




Note: Another way to add variables to the new table is to use the **Edit by Text** window.

Once all of the variables are added, click on **Finish** to move onto the next part of the wizard

In the last part of the wizard, select the new table from the **Schema** drop down list and click **Finish**



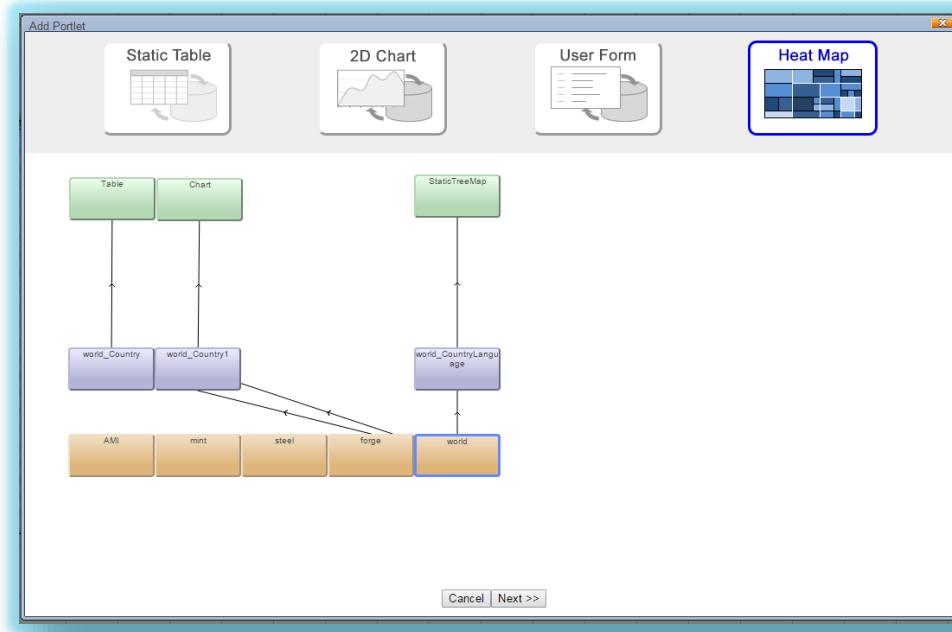
Window
25 Rows

Continent	Region	Life
Africa	Southern Africa	45
Africa	Central Africa	50
Africa	Eastern Africa	51
Africa	Western Africa	53
Asia	Southern and Central Asia	61
Asia	Southeast Asia	64
Africa	Northern Africa	65
Oceania	Melanesia	67
Oceania	Micronesia	68
Europe	Baltic Countries	69
Europe	Eastern Europe	70
Asia	Middle East	71
Oceania	Polynesia	71
South America	South America	71
North America	Central America	71
North America	Caribbean	73
Asia	Eastern Asia	75
North America	North America	76
Europe	Southern Europe	77
Europe	British Islands	77
Europe	Western Europe	78
Europe	Nordic Countries	78
Oceania	Australia and New Zealand	79

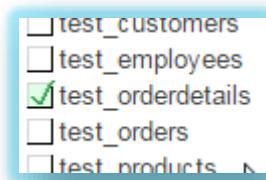
Static HeatMap

HeatMaps may be created using database (static) data.

- 1) In order to create a static heat map, click on the green configuration button of a blank panel and select **Table/Visualization** under Create Table / Display
- 2) This will bring up the Add Portlet wizard (**datamodel platform**). Select **Heat Map** and the datasource from which the data will be retrieved.

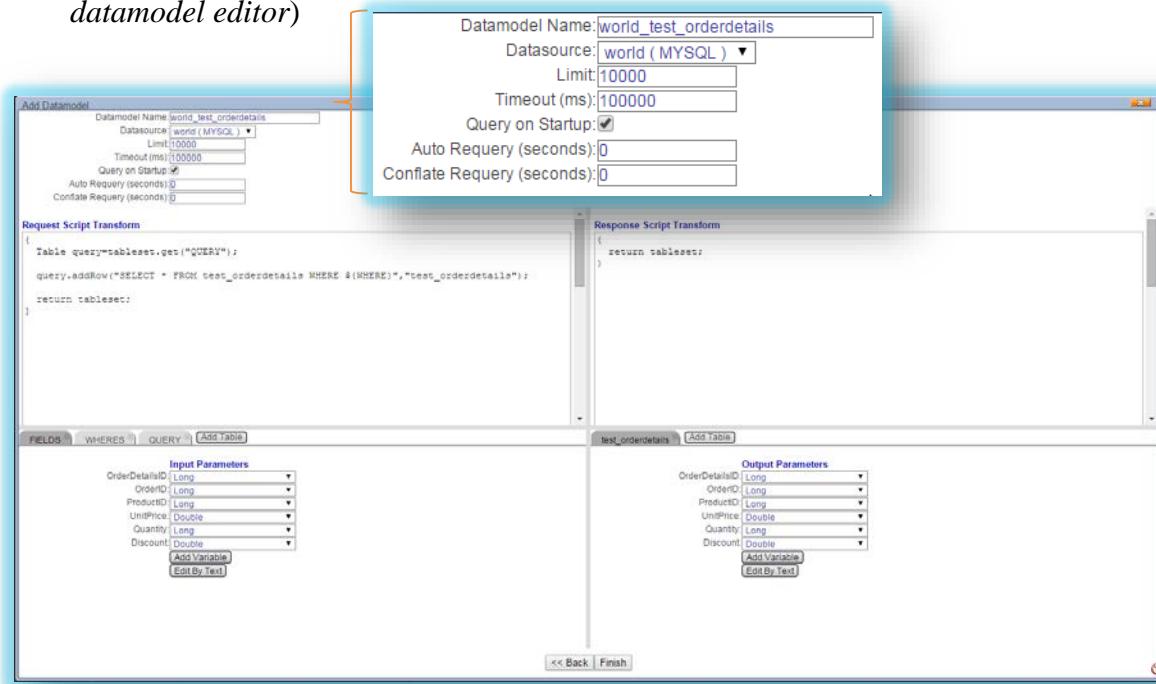


- 3) In the next window, select the data to be used and click **Next**

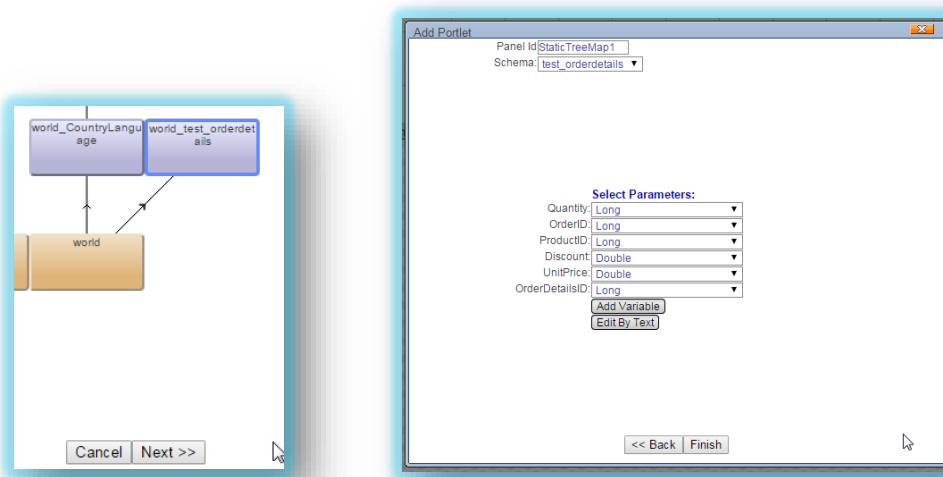


- 4) In the Add Datamodel window (datamodel editor), make any necessary changes to the options (e.g., turning on **Query on Startup** or applying an **Auto Requery**) and click **Finish**.

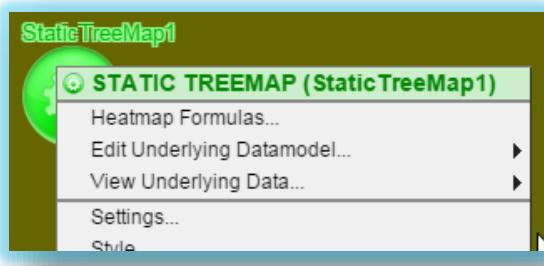
(Note: Please refer to the Data Model section for more information on the layout of the datamodel editor)



- 5) The newly created datamodel will appear in the datamodel platform. Click **Next** for the final review window. After checking to see that all of the columns are returned as the correct type, click **Finish** to create the chart.



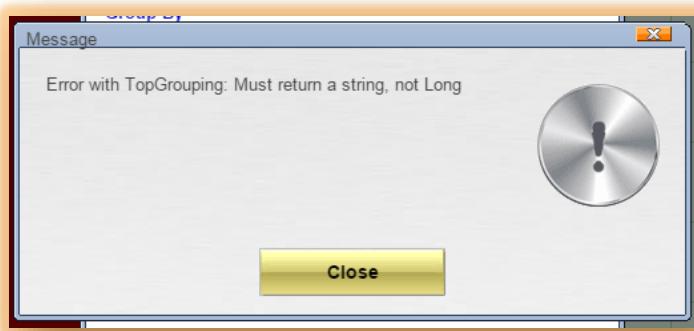
- 6) Unlike real-time heat maps, static heat maps require manual inputs of the formulas to be used for *Top Level Grouping*, *Group By*, *Size*, and *Heat*. Click on the configuration button of the newly created heat map and select **HeatMap Formulas**



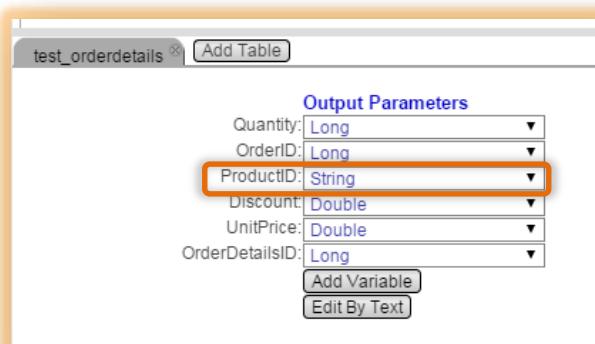
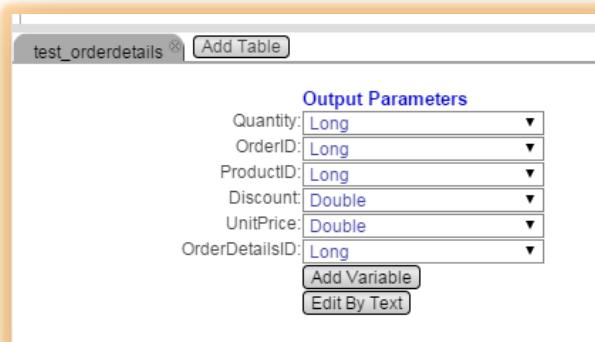
- 7) In the HeatMap Formulas window, input the variables to be used for the heat map and click **Update**. The rest of the steps are the same as real-time heat maps.

Example of using the datamodel editor to reconfigure the data

When working with certain variables and visualizations, it may be necessary to reconfigure the data in order for it to work. For example – in the initial configuration of the **OrderDetails** datamodel, **ProductID** was set as a Long. However, the heat map requires a String to be used for the **Top Level Grouping**.



- To change the data type, click the configuration button and select the datamodel under **Edit Underlying Datamodel**.
- Locate the list of variables and their types under the **Response Script Transform** section of the datamodel editor. Change the type of the variable by selecting the appropriate type from the dropdown menu.

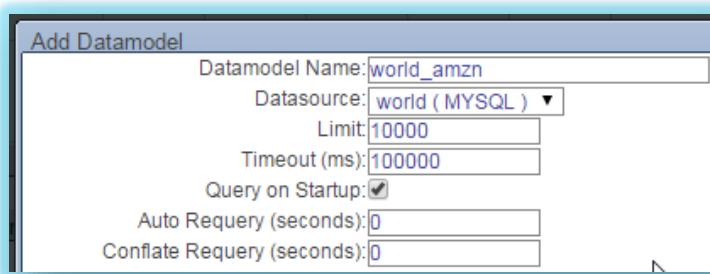


Changing the type of a variable under the **Response Script Transform** means AMI will apply the type before it presents it to the table/visualization but will not make any changes to the underlying data

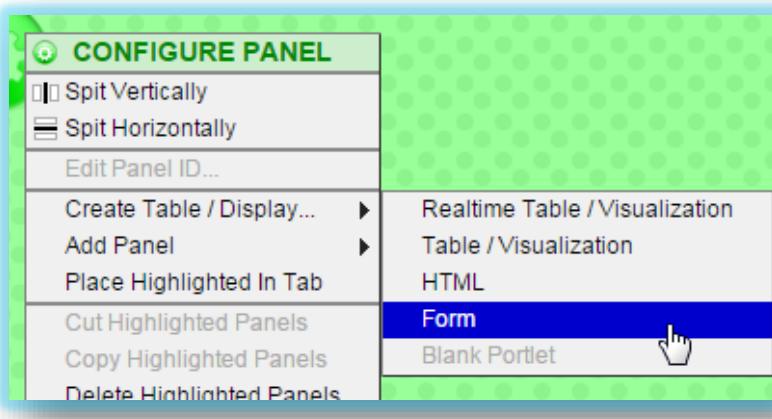
Query Form

Query forms can be used to run queries on static tables and charts. With the addition of **fields**, query forms can be used to retrieve specific information from the data.

- When new static tables/charts are created through the datamodel, the option to **Query on Startup** can be used to have the table/chart display the data as soon it is created. Once made, query forms can be used to run queries on tables/charts when an auto-requery is not required.



- In order to create a query form, click on the green configuration button of a new window or blank panel and select **Form** under **Create Table/Display**. This will create a simple query form without any fields.



- Forms based on datamodels may also be created by selecting **Table/Visualization** under **Create Table/Display**. In the datamodel platform, select **User Form** and a datasource from which the data will be retrieved (*please refer to the datamodel section for further details on the use of datamodels*).

- 2) Create a relationship between the query form and the static table/chart by clicking on the configuration button of the table/chart and selecting Add Relationship (*Please refer to the Relationship section for further details on linking panels*).

The screenshot shows a window titled "PNL1" containing a table with 4,079 rows. The columns are labeled: CountryCode, District, Name, and Population. A context menu is open over the table, and the "Add Relationship" option is highlighted.

CountryCode	District	Name	Population
AFG	Kabol	Kabul	1,780,000
AFG	Qandahar	Qandahar	237,500
AFG	Herat	Herat	186,800
AFG	Balkh	Mazar-e-Sharif	127,800
NLD	Noord-Holland	Amsterdam	731,200
NLD	Zuid-Holland	Rotterdam	593,321
NLD	Zuid-Holland	Haag	440,900
NLD	Utrecht	Utrecht	234,323
NLD	Noord-Brabant	Eindhoven	201,843
NLD	Noord-Brabant	Tilburg	193,238
NLD	Groningen	Groningen	172,701
NLD	Noord-Brabant	Breda	160,398
NLD	Gelderland	Apeldoorn	153,491
NLD	Gelderland	Nijmegen	152,463
NLD	Overijssel	Enschede	149,544
NLD	Noord-Holland	Haarlem	148,772

- 3) Use the Connect menu to add any filters or to change the name of the relationship. The query button on the form uses the **name of relationship** field. Clicking on the query button will then generate data on the table/chart.

The screenshot shows the "Connect" dialog box. It displays two tables connected by a relationship line. The dialog includes fields for "Query Clause (Optional)", "Field Mappings (Advanced)...", "Update relationship...", "Name of relationship" (set to "Query City Table"), "Display option" (set to "Bring target panel to front"), "When nothing is selected" (set to "Do nothing"), and "Ami Script".

Window

Query City Table

4,079 Rows

CountryCode	District	Name	Population
AFG	Kabul	Kabul	1,780,000
AFG	Qandahar	Qandahar	237,500
AFG	Herat	Herat	186,800
AFG	Balkh	Mazar-e-Sharif	127,800
NLD	Noord-Holland	Amsterdam	731,200
NLD	Zuid-Holland	Rotterdam	593,321
NLD	Zuid-Holland	Haag	440,900
NLD	Utrecht	Utrecht	234,323
NLD	Noord-Brabant	Eindhoven	201,843
NLD	Noord-Brabant	Tilburg	193,238
NLD	Groningen	Groningen	172,701
NLD	Noord-Brabant	Breda	160,398
NLD	Gelderland	Apeldoorn	153,491
NLD	Gelderland	Nijmegen	152,463
NLD	Overijssel	Enschede	149,544

Additional query buttons will be added to the query form as more relationships between the query form and other tables/charts are established. Note: Multiple relationships with the same name will share a single button that will run the queries

3 Test Accounts. Evaluated 3 items in 0.004 seconds.

Type	Object	Account	Owner
Test Accounts	3	A-03	Rob
Test Accounts	2	A-02	Bill
Test Accounts	1	A-01	Peter

5 Test Orders. Evaluated 5 items in 0.006 seconds.

Type	Object	Account
Test Orders	15	A-01
Test Orders	13	A-03
Test Orders	14	A-02
Test Orders	11	A-01
Test Orders	12	A-03

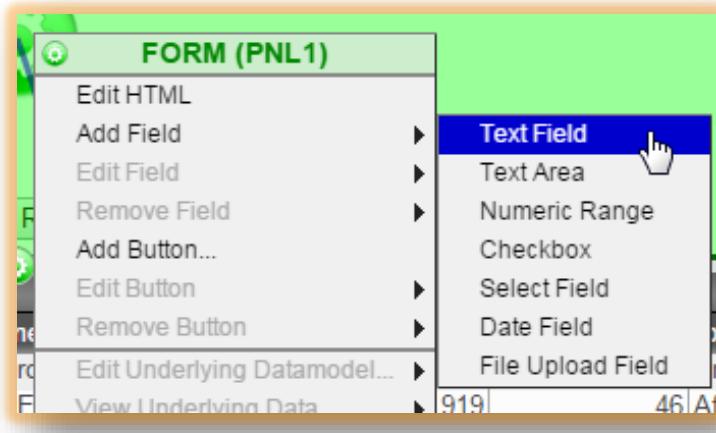
9 Test Executions. Evaluated 9 items in 0.009 seconds.

Type	Object	Execution	Order	Price	Quantity
Test Executions	21	E-01	0-01	\$10.00	300
Test Executions	22	E-02	0-02	\$100.00	100
Test Executions	23	E-03	0-03	\$70.00	300
Test Executions	24	E-04	0-02	\$9500	100
Test Executions	25	E-05	0-03	\$100.00	100
Test Executions	26	E-06	0-03	100	
Test Executions	27	E-07	0-04	300	
Test Executions	28	E-08	0-04	\$2.50	200
Test Executions	29	E-09	0-04	\$2.00	100

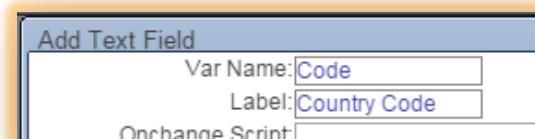
Adding Fields

In order to look for specific information in a table, **input fields** can be added to the query form and a relationship established between the input field and a variable in the table.

- 1) To add a field, such as a Text Field, click on the configuration button of the query form and select the Text Field option under **Add Field**. Other field types include Text Area, Numeric Range, Checkbox, Select Field, Date Field, and File Upload Field.



- 2) In the **Add Text Field** wizard, input the name of the variable and a label and press submit.



- 3) Edit the relationship between the query form and table/chart in order to link the field to the variable on the table/chart.
 - **Note:** Ensure the source variable (field variable) is enclosed in quotes since the relationship is to a static table/chart

Window
Country Code:

Query Country Table

239 Rows

Code	Name	LocalName	Region	Continent	Capital	GNP	GNPOld	GovernmentForm	Head
ATF	French Southern Territories	Territoires australes	Antarctica	Antarctica		63	612	Nonmetropolitan	
ATG	Antigua and Barbuda	Antigua and Barbuda	Caribbean	North America		135	351,182	Constitutional Monarchy	
AUS	Australia	Australia	Australia and New Zealand	Oceania		1,523	211,860	Constitutional Monarchy	
AUT	Austria	Österreich	Western Europe	Europe		144	4,127	Federal Republic	
AZE	Azerbaijan	Azərbaycan	Middle East	Asia		552	903	Federal Republic	
BDI	Burundi	Burundi/Uburundi	Eastern Africa	Africa		179	249,704	Constitutional Monarchy	
BEL	Belgium	België/Belgique	Western Europe	Europe		187	2,357	Republic	
BEN	Benin	Bénin	Western Africa	Africa		549	2,425	Republic	
BFA	Burkina Faso	Burkina Faso	Western Africa	Africa		150	32,852	Republic	
BGD	Bangladesh	Bangladesh	Southern and Central Asia	Asia		539	12,178	Republic	
BGR	Bulgaria	Balgarija	Eastern Europe	Europe		149	6,366	Republic	
BHR	Bahrain	Al-Bahrayn	Middle East	Asia		148	3,527	Monarchy (Emirate)	
BHS	Bahamas	The Bahamas	Caribbean	North America		201	2,841	Constitutional Monarchy	
BIH	Bosnia and Herzegovina	Bosna i Hercegovina	Southern Europe	Europe				Federal Republic	

Window
Country Code: CZE

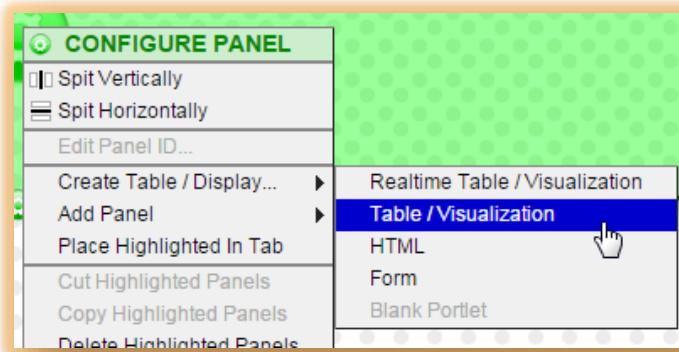
Query Country Table

1 Rows

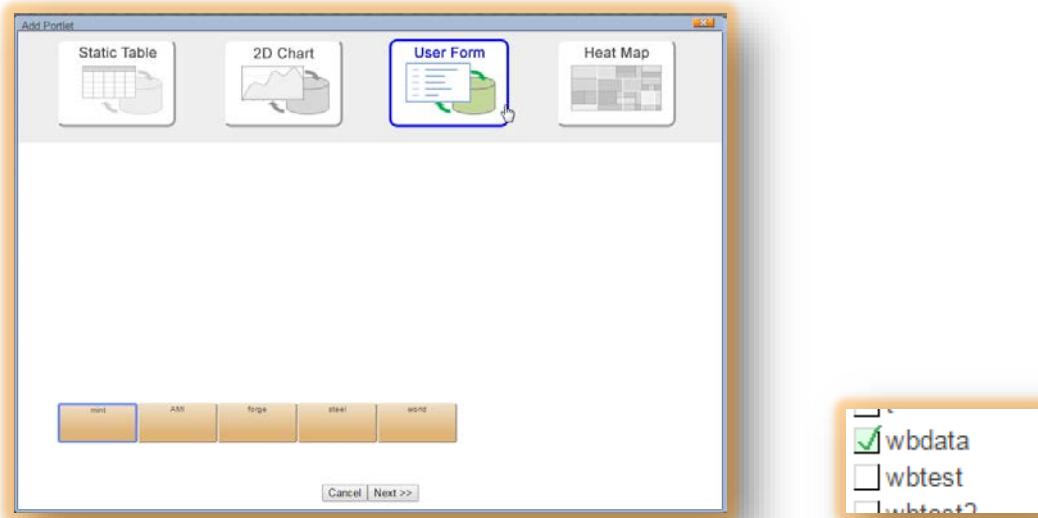
Code	Name	LocalName	Region	Continent	Capital	GNP	GNPOld	GovernmentForm	Head
CZE	Czech Republic	Česká republika	Eastern Europe	Europe	3,339	55,017	52,037	Republic	Václav Klaus

Example – creating select fields on a datamodel-backed query form

- 1) To create a query form based on a datamodel, click on the green configuration button of a new window or blank panel and select **Table/Visualization** under Create Table/Display



- 2) In the datamodel platform, select **User Form** and the datasource where the data is located. Click Next to choose the table.



- 3) In the Add Datamodel window (datamodel editor), check the Query on Startup option and click Finish.
- 4) After creating the datamodel, it will appear in the datamodel platform (already selected). Click Next to view all of the variables and make any necessary changes to their parameters. Click Finish to create the query form.
- 5) Click on the configuration button of the newly query form and select '**Select Field**' under **Add Field** to create select fields

- 6) In the first part of the **Add Select Field** window, input the name of the variable and a label into the ‘Var Name’ and ‘Label’ fields.



- 7) In the bottom half of the window, select the **datasource** (Note: Although the form and table will use the same dataset, it is best to create two separate datamodels) and the **columns** to be used. Once a datamodel is chosen, the column options will list all of the columns available in that datamodel. Click ‘**submit**’ to create the select field.

Datasource	mint_wbdata -> wbdata
Id Column	CountryCode
Value Column	CountryCode CountryName SeriesName YR2000 YR2001 YR2002 YR2003 YR2004 YR2005 YR2006 YR2007 YR2008 YR2009 YR2010 YR2011 YR2012 YR2013 YR2014 wbdata_id

- 8) Once the select fields have been created, establish a link between the table and the form. Use the Query Clause to link the fields and the columns in the table.

Query Clause (Optional):

Window

Country Name: United States

Series Name: Foreign direct investment, net (BoP, current US\$)

7,626 Rows

wbdata_id | CountryCode | SeriesName

			YR2000	YR2001	YR2002	YR2003	YR2004	YR
1 AFG	AFG	Internet users (per 100 people)				8	1	
2 AFG	AFG	Life expectancy at birth, female (years)				2,461,666,315	4,128,818,042	4,583,648,925
3 AFG	AFG	Life expectancy at birth, male (years)		115	186	198	220	
4 AFG	AFG	Life expectancy at birth, total (years)						210
5 AFG	AFG	Afghanistan	Exports of goods and services (% of GDP)			32	44	34
6 AFG	AFG	Afghanistan	Foreign direct investment, net (BoP, current US\$)					
7 AFG	AFG	Afghanistan	GNI per capita, PPP (current international \$)			860	920	920
8 AFG	AFG	Afghanistan	GINI index (World Bank estimate)					
9 AFG	AFG	Afghanistan	Inflation, consumer prices (annual %)					
10 AFG	AFG	Afghanistan	Inflation, GDP deflator (annual %)				12	11
11 AFG	AFG	Afghanistan	Internet users (per 100 people)		0	0	0	0
12 AFG	AFG	Afghanistan	Imports of goods and services (% of GDP)			65	94	87
13 AFG	AFG	Afghanistan	Life expectancy at birth, total (years)	55	55	56	56	57
14 AFG	AFG	Afghanistan	Literacy rate, adult total (% of people ages 15 and above)					
15 AFG	AFG	Afghanistan	Unemployment, total (% of total labor force) (modeled ILO estimate)	5	5	5	5	4
16 AFG	AFG	Afghanistan	Poverty headcount ratio at national poverty lines (% of population)					
17 AFG	AFG	Afghanistan	Agriculture, value added (% of GDP)			38	38	30
18 AFG	AFG	Afghanistan	CO2 emissions (metric tons per capita)	0	0	0	0	0
19 AFG	AFG	Afghanistan	Central government debt, total (% of GDP)					
20 AFG	AFG	Afghanistan	Population, total	20,595,360	21,347,782	22,202,806	23,116,142	24,018,682
21 AFG	AFG	Afghanistan	Urban population (% of total)		21	22	22	23
22 AFG	AFG	Afghanistan	Urban population	4,383,105	4,610,267	4,865,745	5,140,336	5,419,095
23 AFG	AFG	Afghanistan	Urban population growth (annual %)		4	5	5	5
24 AFG	AFG	Afghanistan	Unemployment, female (% of female labor force) (modeled ILO estimate)	11	10	10	11	10
25 AFG	AFG	Afghanistan	Unemployment, male (% of male labor force) (modeled ILO estimate)		4	4	4	4
26 AFG	AFG	Afghanistan	Access to electricity (% of population)		37			
27 AFG	AFG	Afghanistan	Life expectancy at birth, female (years)		56	56	57	58
28 AFG	AFG	Afghanistan	Life expectancy at birth, male (years)		54	54	55	55

Window

Country Name: United States

Series Name: Internet users (per 100 people)

1 Rows

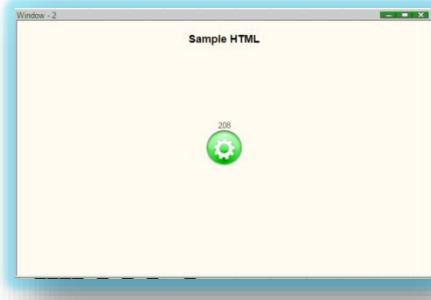
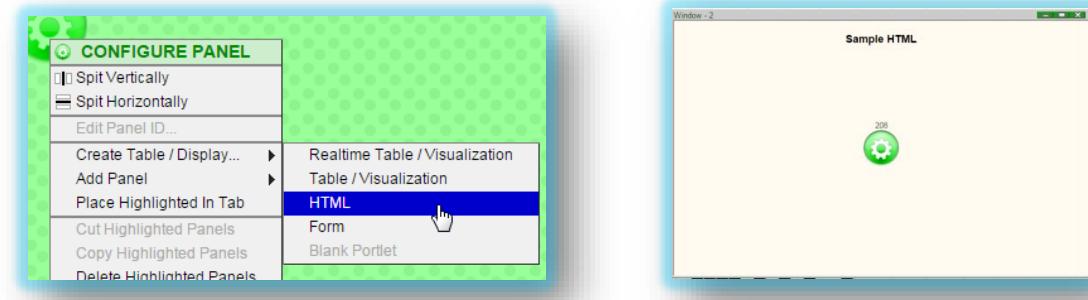
wbdata_id | CountryCode | CountryName | SeriesName

			YR2000	YR2001	YR2002	YR2003	YR2004	YR
6,304	USA	United States	Internet users (per 100 people)	43	49	59	62	65

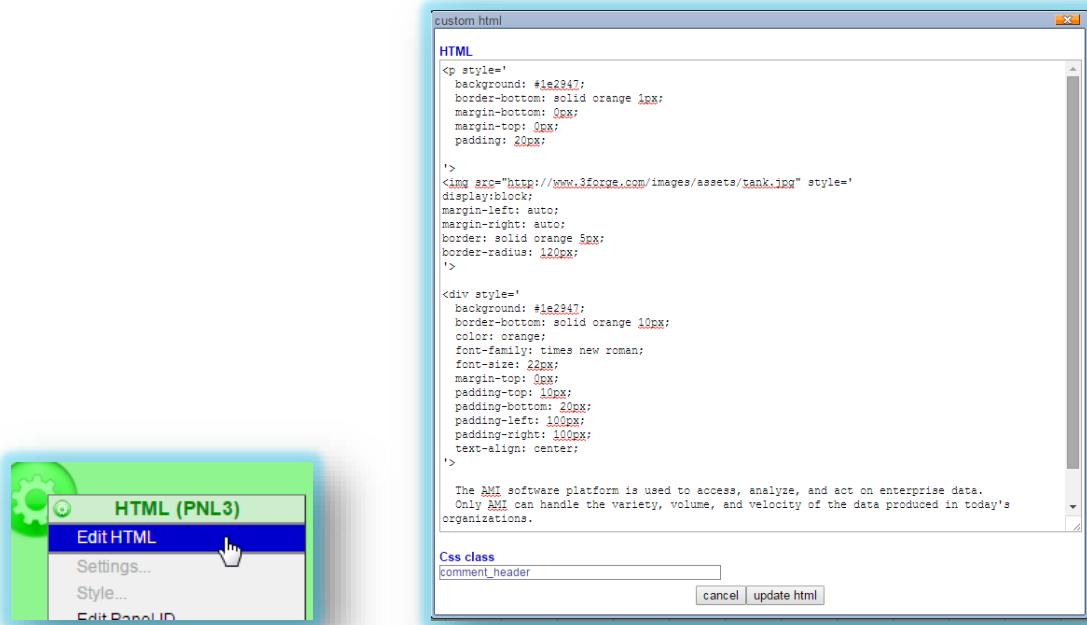
Custom HTML

Windows or blank panels can be set aside for custom HTML. These HTML panels can be utilized to make a title panel for the entire layout.

In order to create a Custom HTML panel, click on the green configuration button of a new window or blank panel and select **HTML** from the **Create Table/Display** menu. This will create a new HTML window with the title **Sample HTML**



To apply a custom HTML, click on the configuration button of the HTML panel, and select **Edit HTML**. This will bring up the **custom html** window. Input the custom html into the **HTML** field and click on **preview html** or the **submit** button.



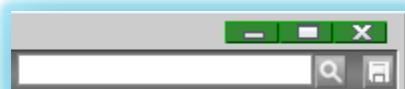
Window - 2

The AMI software platform is used to access, analyze, and act on enterprise data. Only AMI can handle the variety, volume, and velocity of the data produced in today's organizations.

WARNING: Contents will be placed inside a div and an invalid HTML can have adverse effects on the rest of the dashboard

Table Search

AMI's search field is a powerful filtering tool which allows the user to quickly search for interested records while providing a rich language for advanced searching. Searches can simply be keyed into the search field and the filter is applied once the enter key is pressed. For real-time tables, the search is actively evaluated in real-time as records are added, updated, and removed.



Searches are applied on a row by row basis, meaning either an entire row meets the search criteria or the row will be filtered out. Hidden columns are not included in the search. Each row is represented as a single string with all of the cells in the row concatenated together from left to right and tab delimited, with an additional tab at the beginning and end

Example

Type	Object	Account	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif
Order	ORD-00132	3F-00000	4,600	\$47.36	B	CGNX	6,800	DAY

Would be evaluated as: \tOrder\tORD-00132\t3F-00000\t4,600\t\\$47.36\tB\tCGNX\t6,800\tDAY\t

By default, the search does partial matching – so if any part of the row matches the search text, the row is considered a match

17 Order								
Type	Object	Account	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif
Order	ORD-00138	3F-00001	3,100	\$55.00	B	BEAT	4,200	GTC
Order	ORD-00310	3F-00000	6,100	\$49.05	B	BEAV	6,300	GTC
Order	ORD-00638	3F-00004	5,200	\$79.70	B	EAC	7,500	DAY
Order	ORD-00724	3F-00007	3,900	\$62.90	S	RDEA	8,300	DAY
Order	ORD-00848	3F-00001	3,700	\$35.06	B	ESEA	4,300	FOK
Order	ORD-01075	3F-00006	3,200	\$17.94	B	ATEA	3,800	DAY

Searching for ea

\t can be used to make the search more specific:

5 Order									
Type	Object	Account	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif	
Order	ORD-00638	3F-00004	5,200	\$79.70	B	EAC	7,500	DAY	
Order	ORD-02213	3F-00004	6,300	\$48.80	B	EAGL	8,000	DAY	
Order	ORD-02364	3F-00005	3,500	\$15.87	B	EA	5,500	IOC	
Order	ORD-02378	3F-00002	4,100	\$44.75	B	EAGLW	6,000	DAY	
Order	ORD-02742	3F-00002	6,400	\$76.07	S	EAGLU	6,700	GTC	

\tea will show any matches that start with ea

6 Order									
Type	Object	Account	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif	
Order	ORD-00724	3F-00007	3,900	\$62.90	S	RDEA	8,300	DAY	
Order	ORD-00848	3F-00001	3,700	\$35.06	B	ESEA	4,300	FOK	
Order	ORD-01075	3F-00006	3,200	\$17.94	B	ATEA	3,800	DAY	
Order	ORD-01293	3F-00002	5,400	\$35.24	S	VLGEA	9,000	DAY	
Order	ORD-02293	3F-00007	4,600	\$16.77	B	SENEA	9,400	GTC	
Order	ORD-02364	3F-00005	3,500	\$15.87	B	EA	5,500	IOC	

ea\t will show any matches that end with ea

1 Order									
Type	Object	Account	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif	
Order	ORD-02364	3F-00005	3,500	\$15.87	B	EA	5,500	IOC	

\tea\t returns an exact match

By default, the search is case insensitive, but this can be changed by prefixing the search with a tilde (~)

2 Order								
Type	Object	Account	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif
Order	ORD-02766	3F-00003	750	\$45.97	B	gt	500	DAY
Order	ORD-02765	3F-00005	500	\$58.93	S	GT	1,000	DTC

Case insensitive search – \tgt\t

1 Order								
Type	Object	Account	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif
Order	ORD-02765	3F-00005	500	\$58.93	S	GT	1,000	DTC

Case sensitive search – ~\tGT\t

Note: Using special characters – when your search will include special characters (not numbers and letters), it is best practice to escape them using a backslash (\).

.62								
Type	Object	Account	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif
Order	ORD-01991	3F-00001	3,600	\$18.62	S	AACOU	4,300	GTD
Order	ORD-01627	3F-00002	4,500	\$87.11	S	AAXJ	9,500	GTC
Order	ORD-01696	3F-00006	4,600	\$62.00	B	AGNCP	5,200	OPG
Order	ORD-00126	3F-00001	5,700	\$62.70	S	AMSF	8,000	DAY
Order	ORD-00629	3F-00003	5,000	\$52.99	S	ANAC	9,800	FOK
Order	ORD-01036	3F-00004	6,400	\$62.78	B	BMTC	9,400	DAY
Order	ORD-01623	3F-00005	4,000	\$30.54	B	CBRX	6,400	GTD
Order	ORD-00125	3F-00004	4,800	\$76.62	S	CHCO	5,400	DAY
Order	ORD-01804	3F-00000	3,000	\$62.98	B	CHI	3,000	OPG

Without escape

.62								
Type	Object	Account	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif
Order	ORD-01991	3F-00001	3,600	\$18.62	S	AACOU	4,300	GTD
Order	ORD-00125	3F-00004	4,800	\$76.62	S	CHCO	5,400	DAY
Order	ORD-00648	3F-00003	7,700	\$36.62	S	CTAS	7,800	GTC
Order	ORD-00714	3F-00006	6,900	\$41.62	B	DEST	9,400	IOC
Order	ORD-02036	3F-00005	3,100	\$43.62	S	DWA	4,600	DAY

With escape

For complex searching, it is possible to use a combination of conditional clauses which include and (...&...) or (...|...) along with the conditional stanza (... ? ... : ...)

2 Order									
Type	Object	Account	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif	
Order	ORD-00272	3F-00007	3,500	\$3.97	B	GIFI	6,500	IOC	
Order	ORD-02364	3F-00005	3,500	\$15.87	B	EA	5,500	IOC	

(\tbt)&(\tioc)&(\t3,500)

2 Order									
Type	Object	Account	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif	
Order	ORD-02364	3F-00005	3,500	\$15.87	B	EA	5,500	IOC	
Order	ORD-00482	3F-00006	8,700	\$24.71	B	AMZN	9,100	IOC	

(\tea)|(\tamzn)

Columns

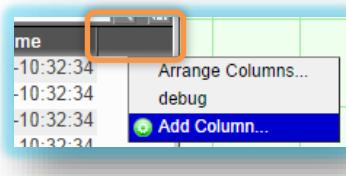
Adding Columns

There are three ways to add columns in tables (Note: **Layout Editor** must be ON)

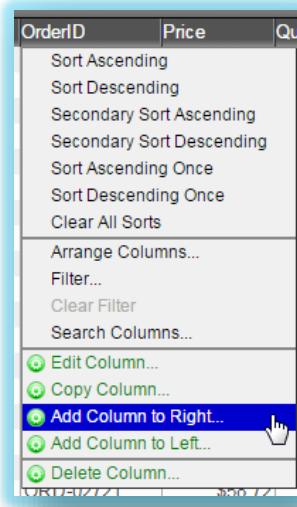
- The first way to add a column is to click on the configuration button of the panel to bring up the menu. In the menu, select the **Add Column** option



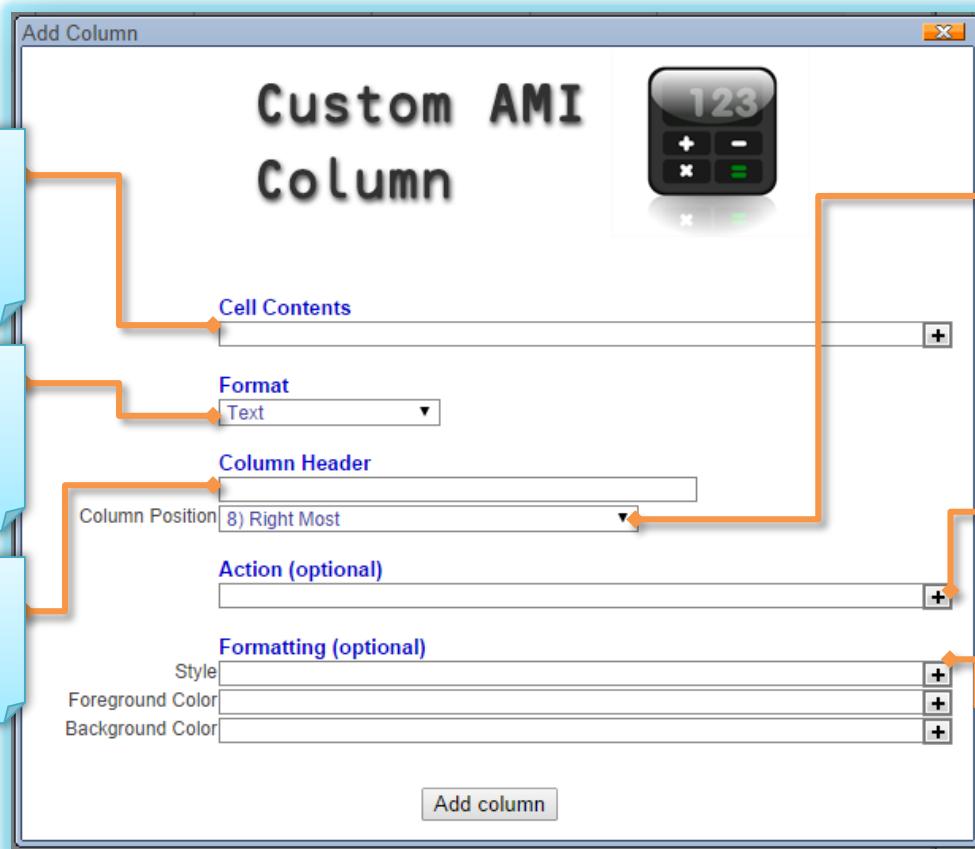
- The second way to add columns is to click on an open area of the header in order to bring up the menu. In the menu, select **Add Column**



- The third way to add columns is to click on a column to bring up the menu. This method allows you to place the new column in between existing columns. In the menu, select whether to add the new column to the **right** or **left** of the selected column.



- Selecting the **Add Column** option will bring up the *Add Column* window

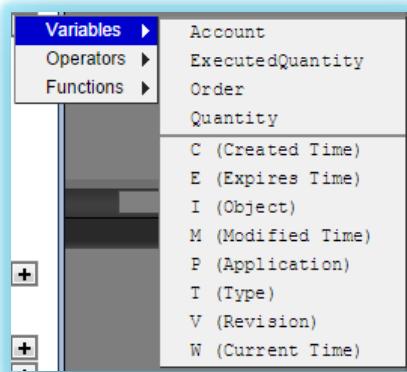


Cell Contents

- Input the variables to be used in the new column in **Cell Contents**. Click the icon to view the various options available for use in **Cell Contents**.

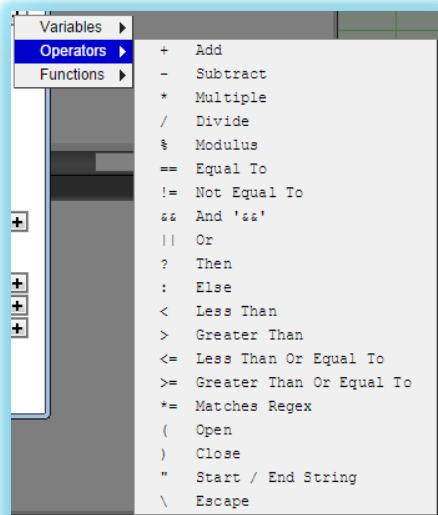
Variables

- The top portion of the variable list displays the columns that are in the panel
- The bottom portion of the list remains consistent throughout all **Variables** lists



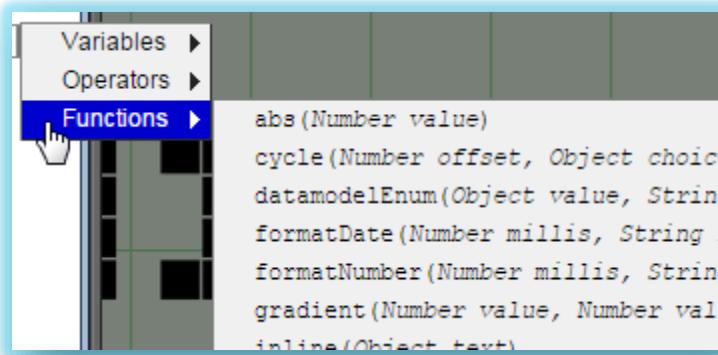
Operators

- Operators can be used in conjunction with **variables** &/or **functions** in order to build custom expressions. *Boolean expressions will return true or false.* Note: These operators can be manually typed in



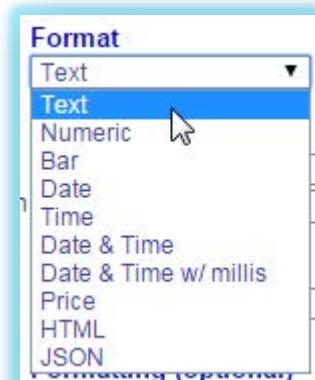
Functions

- Refer to the Functions section



Format

- Choose an appropriate format for the variables to be shown in the new column



Example – using the HTML format in a column



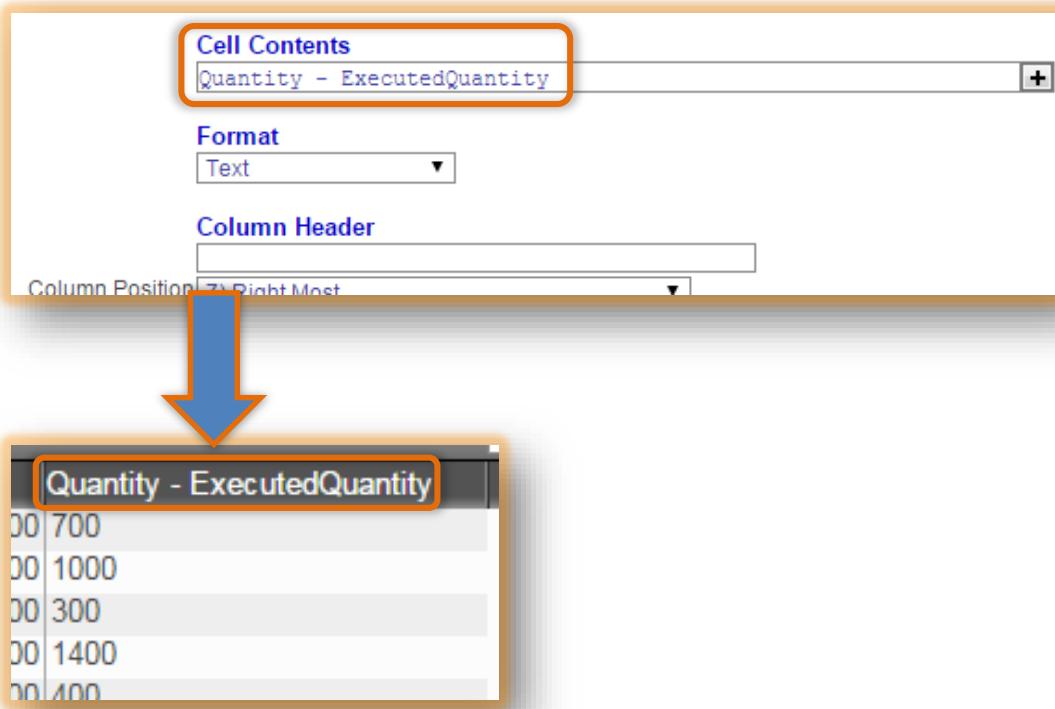


Column Header

- Name to be shown at the head of the new column. These names must be unique from existing columns and will be filled in automatically with what is stated in **Cell Contents** if nothing is provided.



Example – Leaving Column Header blank



Cell Contents
Quantity - ExecutedQuantity

Format
Text

Column Header

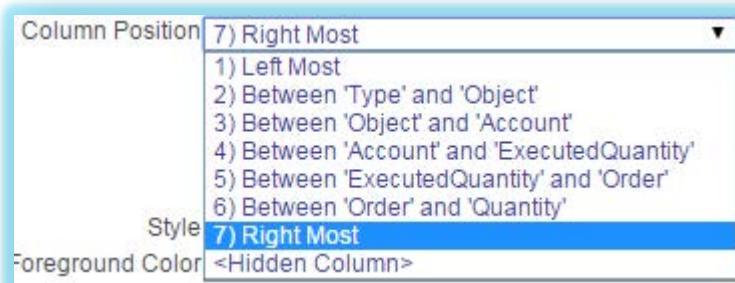
Column Position: 8) Right Most

Quantity - ExecutedQuantity

00	700
00	1000
00	300
00	1400
00	400

Column Position

- Choose from a drop down list of available column positions in the panel.
- Note: Column positions can be re-arranged in work mode by selecting **Arrange Columns** in the column dropdown

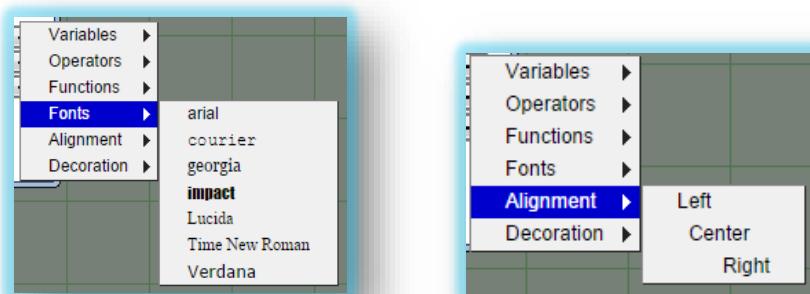


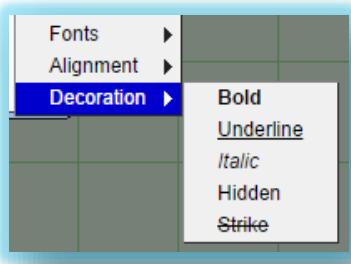
Formatting

- Formatting can be used to change the way the data is displayed in the tables
- **Conditional formatting** – formatting can be used to highlight data in specific formats when certain parameters are met (*please refer to the end of the section for an example*)

Style

- In addition to the normal options available under the icon (*Variables, Operators, & Functions*); **Style** has *Fonts, Alignment, & Decoration* available for use in customizing the view of the data in the column.



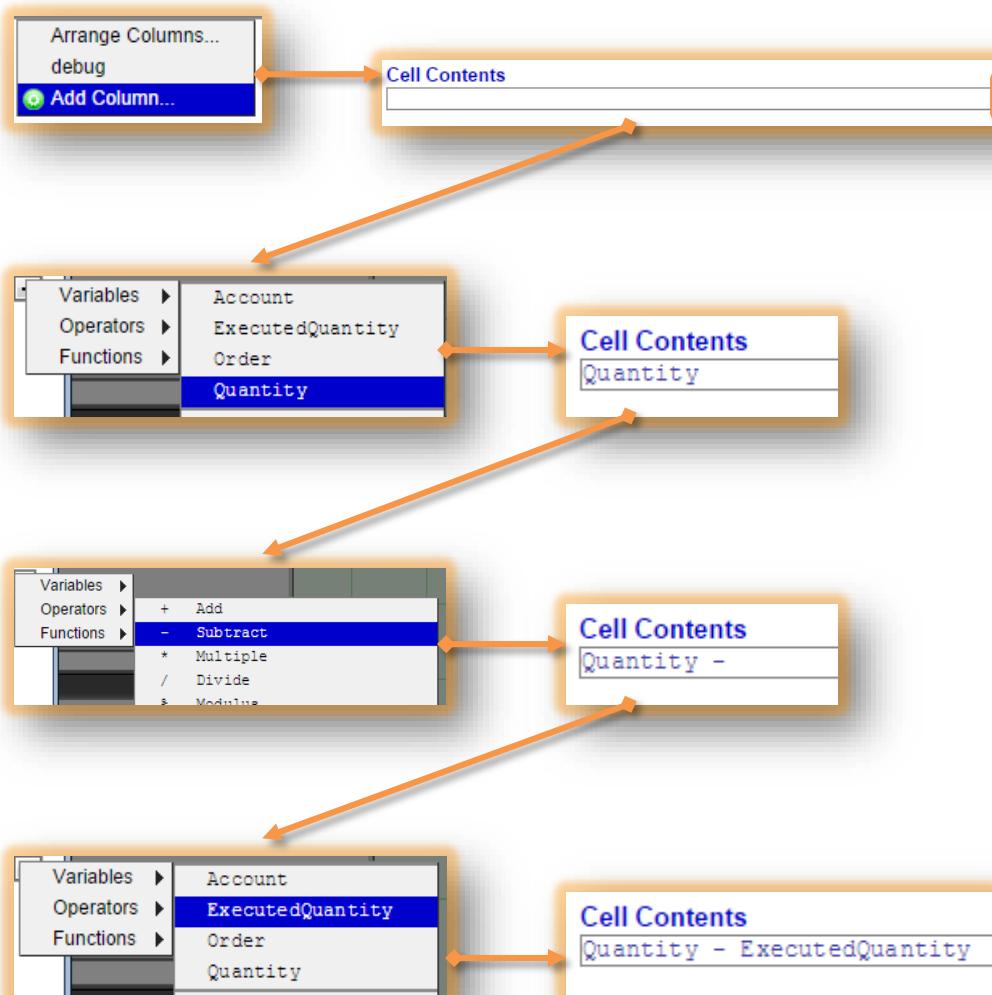


Foreground Color and Background Color

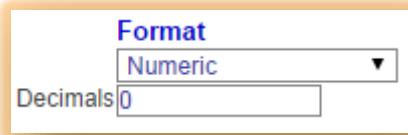
- The **Foreground Color** option is used to select the color of the variables in the column
- The **Background Color** option is used to select the color of the background of the cells of the column
- Note: *Color codes can be used in place of the available options*

Example – creating a column to display the remaining quantity to be executed

Order	Quantity	ExecutedQuantity
0-01	1,000	300
0-02	1,200	200
0-03	800	500
0-04	2,000	600
0-05	400	0



- Assign the correct **Format** (in this example, *Numeric*)



- Assign a header in **Column Header**



Quantity	ExecutedQuantity	Remaining
1,000	300	700
1,200	200	1,000
800	500	300
2,000	600	1,400
400	0	400

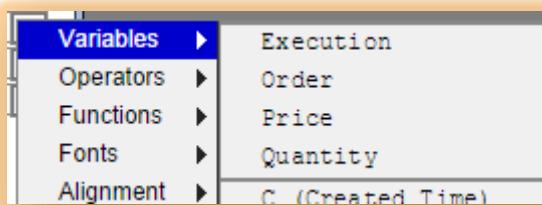
Example – using expressions in formatting to highlight certain data

Formatting can be used to highlight data in specific formats when certain parameters are met.

- In the first part of this example, the **Price** is to be in bold and highlighted if it is greater than or equal to \$50

Order	Execution	Price	Quantity
0-01	E-01	\$10.00	300
0-02	E-02	\$100.00	100
0-03	E-03	\$70.00	300
0-02	E-04	\$95.00	100
0-03	E-05	\$60.00	100
0-03	E-06	\$50.00	100
0-04	E-07	\$3.00	300
0-04	E-08	\$2.50	200
0-04	E-09	\$2.00	100

- 1) Click on the header of the *Price* column and select **Edit Column** from the menu. This will bring up the **Edit Column** window, which is the same as the **Add Column** window
- 2) In the **Style** field of the Formatting section, click on the icon to bring up the menu
- 3) In the menu, find and select **Price** under **Variables**. *Price* should now appear in the **Style** field



- 4) Click on the icon and select *Greater Than or Equal To* [] under **Operators**. [] should appear after *Price* in the **Style** field. Type in 50 after the \geq (the \$ symbol is not necessary)
- 5) Click on the icon and select *Then* [] under **Operators**. So far, the field states:

“If the Price is greater than or equal to \$50, then”

Formatting (optional)
 Style

- 6) Click on the icon and select **Bold** under **Decoration**. Since the price is to be in bold only if it is ≥ 50 and nothing else, type in : to indicate “else, do nothing.” Altogether, the field will now state:

“If the Price is greater than or equal to \$50, then make it bold. Else [otherwise], do nothing”

Formatting (optional)
 Style

Note: *Else* [] can also be found under **Operators**. Operators can be typed in and do not have to be put in through the icon.

- 7) In order to highlight the Prices which are greater than or equal to \$50, repeat steps 2 – 5 in the **Background Color** field. Click on the icon and choose a color (Yellow will be used for the example). Similar to step 6, type in : to indicate “else, do nothing.”

Background Color

- 8) Click on **Update column** to view the changes

Order	Execution	Price	Quantity
0-01	E-01	\$10.00	300
0-02	E-02	\$100.00	100
0-03	E-03	\$70.00	300
0-02	E-04	\$95.00	100
0-03	E-05	\$60.00	100
0-03	E-06	\$50.00	100
0-04	E-07	\$3.00	300
0-04	E-08	\$2.50	200
0-04	E-09	\$2.00	100

- In the second part of this example, *Orders* with *executions* of \$50 or greater will be highlighted

- 9) Click on the header of the *Order* column and select **Edit Column** from the menu
- 10) In the **Background Color** field, input the same statement from the end of step 7

Background Color `Price >= 50 ? "yellow" : ""`

- 11) Click on **Update column** to view the changes to **Order**

Order	Execution	Price	Quantity
0-01	E-01	\$10.00	300
0-02	E-02	\$100.00	100
0-02	E-04	\$95.00	100
0-03	E-03	\$70.00	300
0-03	E-05	\$60.00	100
0-03	E-06	\$50.00	100
0-04	E-07	\$3.00	300
0-04	E-08	\$2.50	200
0-04	E-09	\$2.00	100

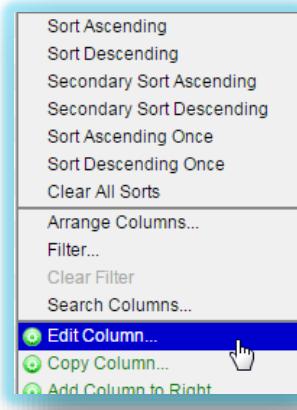
252

Editing Columns

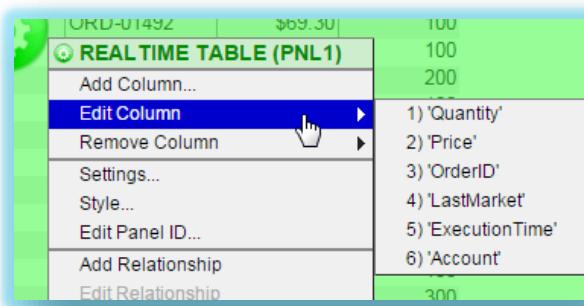
After columns have been added, changes can be made to the columns unless they are being used in relationships and sorting (**Note: Layout Editor** must be ON)

There are two ways to edit columns in panels

- The first way to edit columns is to click on the **header** of the column to bring up the menu.
- In the menu, select **Edit Column**



- The second way to edit columns is to click on the **configuration button** of the panel to bring up the menu. In the menu, highlight **Edit Column** in order to view the columns available for editing

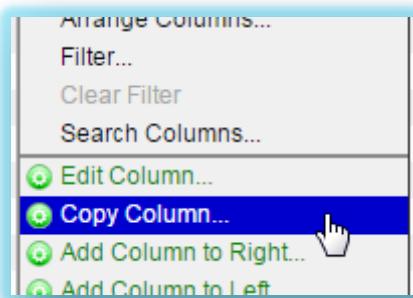


- Both methods will open a **Edit Column** window, which is the same as the **Add Column** window
- After making the necessary changes, click on **Update column**

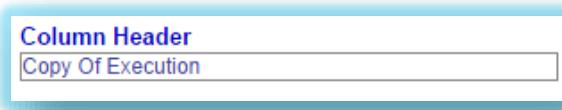
Copying Columns

Columns may be copied in panels (Note: **Layout Editor** must be ON)

- 1) Click on the header of the column to be copied
- 2) In the menu, select **Copy Column** & the **Edit Column** window will pop up



- 3) In the Edit Column window, the default name in the **Column Header** field will be:
Copy of [column name]



- Note: If the column to be copied has any inputs in its fields (**Action & Formatting**), those fields will also be copied [side by side]

Copy Of Order	Order
0-01	0-01
0-02	0-02
0-03	0-03
0-02	0-02
0-03	0-03
0-03	0-03
0-04	0-04
0-04	0-04
0-04	0-04

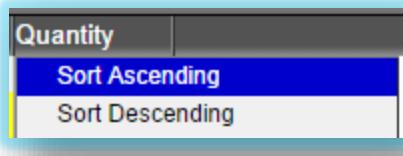
Sorting Columns

Variables in columns may be sorted in multiple ways

Primary Sorting

- Primary sorting will remain in place until it is cancelled. Any new data that is added will automatically be sorted
- There are two ways to add a primary sort to a column:

One way to add a primary sort is through the menu of the column. To do this, click on the header of the column and select either **Sort Ascending** or **Sort Descending** from the menu.



Note: Sort ascending sorts the data from the least to the greatest [0-9; A-Z]. Sort descending sorts the data from the greatest to the least [9-0; Z-A]

Another way to add a primary sort is by moving the cursor over the right side of the column header to bring up the following:



- Clicking the **up** arrow will apply a sort in **ascending** order & clicking the **down** arrow will apply a sort in **descending** order

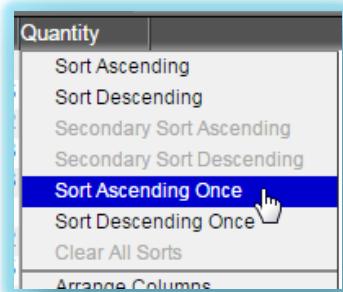
The following symbols will appear on the left side of the column header to indicate whether a primary sort (ascending or descending) is in place



Note: In order to *cancel* a primary sort on a single column, select **Sort Ascending Once** or **Sort Descending Once** from the column menu. This will perform a sort which will **not** remain in place & the primary sort will no longer be in place. The **Clear All Sorts** option will clear sorting applied to all of the columns in the table.

Sort (Ascending/Descending) Once

- **Sorting** (ascending/descending) **once** will be applied one time and will not remain in place. Any new data that is added will be added to the bottom of the table. Note: Sorting is stable; meaning two records that are equivalent based on the sort will maintain their relative ordering prior to the sort



- In order to include the new data in the sorting, either apply a primary sort or a new one-time sort

Secondary Sort (Ascending/Descending)

- Secondary sort can be used to sort real-time data on multiple columns in a panel at the same time
- The **Secondary Sort** (Ascending/Descending) option will not be available unless there is a primary sort in place on one of the columns

Example

- 1) Apply a primary sort on a column (a primary sort is applied to *Account*)

The diagram shows two tables side-by-side. The left table has 15 rows of data with columns: Account, LastMarket, Price, and Quantity. The right table has 15 rows of data with the same columns. Both tables have an orange border around them. An orange arrow points from the left table to the right table, indicating a transformation or result of applying a primary sort.

Account	LastMarket	Price	Quantity
3F-00006	NASD	\$75.91	200
3F-00000	NITE	\$38.32	300
3F-00006	Euro	\$64.03	300
3F-00005	NASD	\$19.89	300
3F-00007	NYSE	\$28.92	200
3F-00000	AMEX	\$77.08	100
3F-00004	NYSE	\$60.97	200
3F-00006	NITE	\$21.60	200
3F-00006	NYSE	\$74.42	200
3F-00005	NASD	\$11.42	300
3F-00004	NYSE	\$55.09	100
3F-00003	NASD	\$6.32	200
3F-00001	NASD	\$15.17	100
3F-00007	NASD	\$53.23	100

Account	LastMarket	Price	Quantity
3F-00000	NITE	\$38.32	300
3F-00000	AMEX	\$77.08	100
3F-00000	AMEX	\$33.64	100
3F-00000	NITE	\$89.77	100
3F-00000	NASD	\$26.39	300
3F-00000	NYSE	\$80.72	100
3F-00000	NASD	\$35.29	100
3F-00000	NASD	\$31.70	100
3F-00000	NASD	\$77.86	200
3F-00000	NYSE	\$5.82	200
3F-00000	NASD	\$45.55	300
3F-00000	NASD	\$71.22	200
3F-00000	NASD	\$79.09	100
3F-00000	NITE	\$79.50	300

- 2) In order to apply a secondary sort, click on the header of another column and select secondary ascending/descending

The screenshot shows a data grid with columns: Account, LastMarket, Price, and Quantity. A context menu is open over the grid, listing options: Sort Ascending, Sort Descending, Secondary Sort Ascending (highlighted with a blue background and a cursor), Secondary Sort Descending, Sort Ascending Once, Sort Descending Once, Clear All Sorts, and Arrange Columns.

Account	LastMarket	Price	Quantity
3F-00000	NITE	\$38.32	300
3F-00000	AMEX	\$77.08	100
3F-00000	AMEX	\$33.64	100
3F-00000	NITE	\$89.77	100
3F-00000	NASD	\$26.39	300
3F-00000	NYSE	\$80.72	100
3F-00000	NASD	\$35.29	100
3F-00000	NASD	\$31.70	100
3F-00000	NASD	\$77.86	200
3F-00000	NYSE	\$5.82	200
3F-00000	NASD	\$45.55	300
3F-00000	NASD	\$71.22	200
3F-00000	NASD	\$79.09	100
3F-00000	NITE	\$79.50	300

Account	LastMarket	Price	Quantity
3F-00000	AMEX	\$77.08	100
3F-00000	AMEX	\$33.64	100
3F-00000	AMEX	\$29.56	100
3F-00000	AMEX	\$72.00	300
3F-00000	AMEX	\$67.43	100
3F-00000	AMEX	\$61.11	100
3F-00000	AMEX	\$87.32	200
3F-00000	AMEX	\$64.13	100
3F-00000	AMEX	\$65.03	100
3F-00000	AMEX	\$88.77	300
3F-00000	AMEX	\$29.19	300
3F-00000	AMEX	\$56.98	300
3F-00000	AMEX	\$30.75	300
3F-00000	AMEX	\$1.85	100

- 3) Apply further secondary sorts to other columns in order to further organize the data

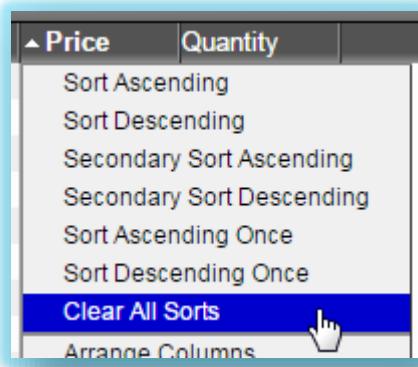
The screenshot shows two data grids. The left grid has its columns: Account, LastMarket, Price, and Quantity. The 'Quantity' column is highlighted with an orange border. The right grid has its columns: Account, LastMarket, Price, and Quantity. The 'Price' column is highlighted with an orange border.

Account	LastMarket	Price	Quantity
3F-00000	AMEX	\$77.08	100
3F-00000	AMEX	\$33.64	100
3F-00000	AMEX	\$29.56	100
3F-00000	AMEX	\$67.43	100
3F-00000	AMEX	\$61.11	100
3F-00000	AMEX	\$64.13	100
3F-00000	AMEX	\$65.03	100
3F-00000	AMEX	\$88.77	300
3F-00000	AMEX	\$29.19	300
3F-00000	AMEX	\$56.98	300
3F-00000	AMEX	\$30.75	300
3F-00000	AMEX	\$1.85	100

Account	LastMarket	Price	Quantity
3F-00000	AMEX	\$1.85	100
3F-00000	AMEX	\$2.24	100
3F-00000	AMEX	\$3.70	100
3F-00000	AMEX	\$4.15	100
3F-00000	AMEX	\$6.71	100
3F-00000	AMEX	\$7.74	100
3F-00000	AMEX	\$7.80	100
3F-00000	AMEX	\$10.18	100
3F-00000	AMEX	\$10.54	100
3F-00000	AMEX	\$15.90	100
3F-00000	AMEX	\$19.90	100
3F-00000	AMEX	\$22.64	100
3F-00000	AMEX	\$24.82	100
3F-00000	AMEX	\$25.02	100

Clear All Sorts

Sorts applied to columns can be easily cleared by selecting **Clear All Sorts** from the column menu. Note: This will clear sorting applied to **ALL** columns in the table



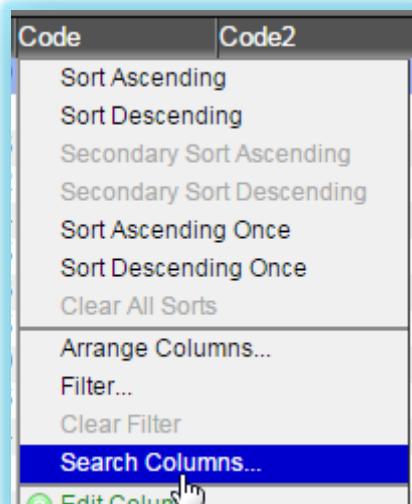
Search Columns

When working with tables with many columns, using the **Search Columns** option from the column menu can be used to find and move specific columns easily

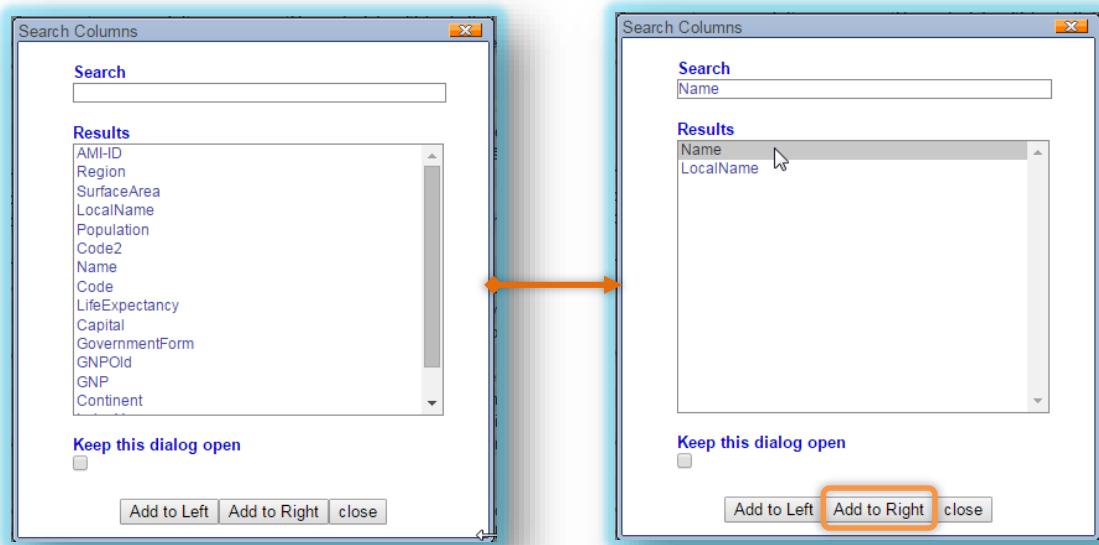
Example

Search for the *Name* column and add it to the right of the *Code* column. Note: Columns will be moved relative to the selected column

- 1) Click on the header of the *Code* column and select **Search Columns**



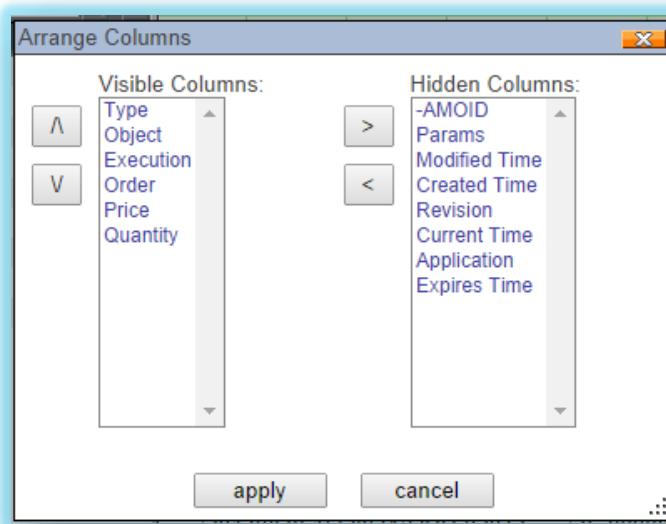
- 2) In the **Search Columns** menu, type in *Name* in the **Search** bar. Select *Name* from the **Results** and select **Add to Right**



Code	Name	Code2
ABW	Aruba	AW
AFG	Afghanistan	AF
AGO	Angola	AO

Arranging Columns

- Columns in panels can be rearranged easily.
 - **Hidden** columns can be moved to become visible in the panels & visible columns can also be hidden. Hidden columns with sorting and filters will still affect the table.
- 1) Access the **Arrange Columns** window by clicking on a column's header (or on an empty space in the header) and selecting **Arrange Columns** from the menu



- 2) Arrange the columns by selecting one or multiple objects in the **Visible Columns** field and using the   icons to move the objects up or down

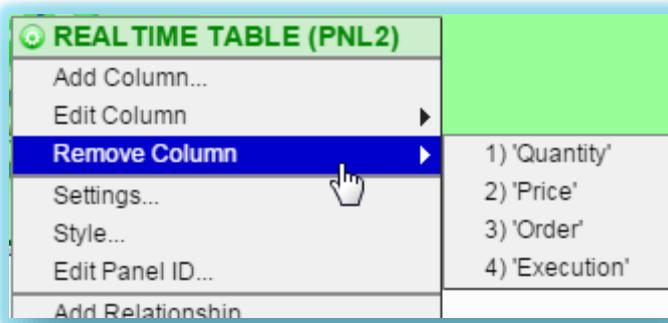
- 3) To move objects between the **Hidden Columns** field and the **Visible Columns** field, select one or multiple objects and use the icons

Deleting/Removing Columns

- Columns can be removed directly from the **column header** or from the **configuration button**'s menu (Note: Layout Editor must be ON)
- 1) To remove a column directly, click on its header and select **Delete Column** from the menu



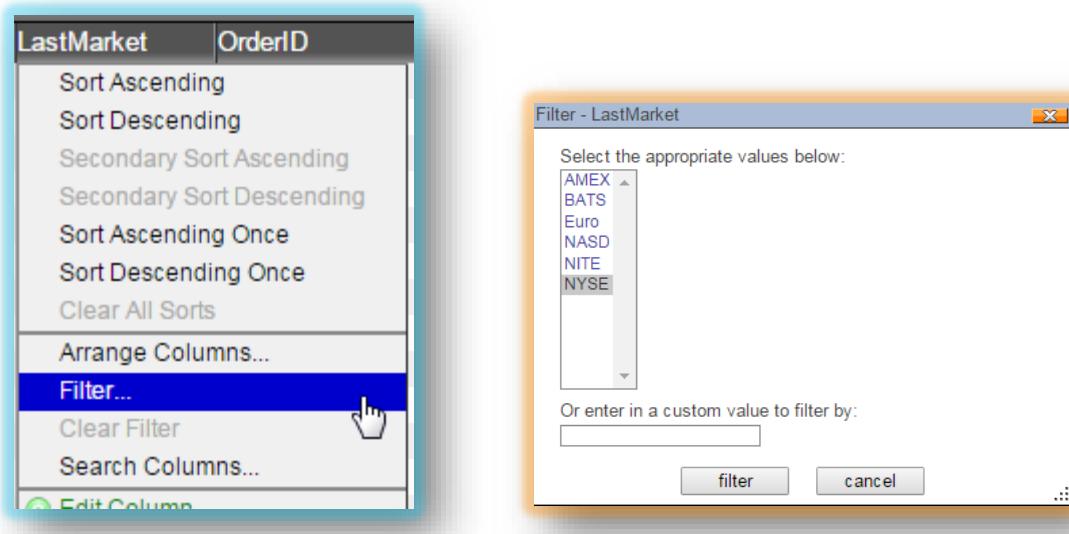
- 2) To remove a column using the configuration button, click on the configuration button and select the column to be deleted under the **Remove Column** sub-menu



Filtering Columns

Filters can be applied directly to columns in order to limit the data being displayed on the tables.

- 1) To add a filter to a column, click on the column's header and select **Filter** from the drop down menu. This will open up the filter window. All existing values are available in the upper selection list. To filter on a value that currently does not exist, type it in the lower edit box.



- 2) In the filter window, all of the distinct values present in the column will be listed. Select one or multiple values to filter the column by. For example, the **Last Market** column can be filtered to show only the NYSE. Any column with a filter will have its header highlighted in orange. To clear any filters, simply select **Clear Filter** from the menu

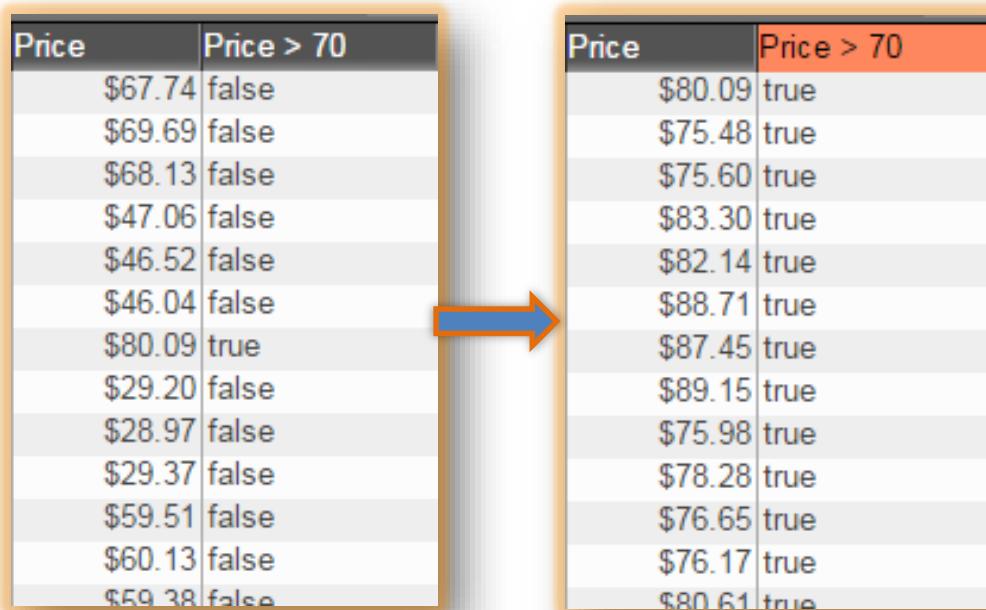
Time	LastMarket	OrderID
-15:05:26	NASD	ORD-01549
-15:08:24	NYSE	ORD-01549
-15:13:42	NITE	ORD-01549
-15:14:13	NITE	ORD-01549
-15:22:09	NASD	ORD-01549
-15:28:55	NASD	ORD-01549
-15:30:46	NASD	ORD-01549
-15:47:29	NASD	ORD-01549
-15:50:39	NITE	ORD-01549
-15:57:30	NYSE	ORD-01549
-14:52:41	Euro	ORD-01551
-14:57:13	NASD	ORD-01551
-15:10:40	NASD	ORD-01551
-15:11:18	NASD	ORD-01551
-15:21:51	NYSE	ORD-01551
-15:24:29	NASD	ORD-01551

Time	LastMarket	OrderID
-15:08:24	NYSE	ORD-01549
-15:57:30	NYSE	ORD-01549
-15:21:51	NYSE	ORD-01551
-15:54:00	NYSE	ORD-01551
-15:13:07	NYSE	ORD-01552
-15:16:32	NYSE	ORD-01552
-15:30:14	NYSE	ORD-01552
-15:31:31	NYSE	ORD-01552
-15:31:43	NYSE	ORD-01552
-15:45:02	NYSE	ORD-01552
-14:55:49	NYSE	ORD-01554
-15:21:08	NYSE	ORD-01554
-15:26:46	NYSE	ORD-01554
-15:55:24	NYSE	ORD-01554
-14:57:44	NYSE	ORD-01555
-15:05:48	NYSE	ORD-01555

Applying filters to other columns in the table can narrow down the data even further. Columns made with **Boolean** expressions can also be used to filter the data in the tables. A Boolean expression returns the value **true** or **false** when it is evaluated.

- 1) To use a column with a Boolean expression as a filter, add a new column using a Boolean expression in the **Cell Contents** field. Once added, the new column will display either the value **true** or **false** based on the statement used
- 2) Apply a filter to the Boolean column, selecting either the true or false value in the **Filter menu**. This filter will narrow down the data based on the Boolean expression. The header of the Boolean column will be highlighted in orange to indicate a filter has been applied. This Boolean column can be hidden from the table but its filter will remain in place (hidden columns with sorting and filters continue to affect the table)



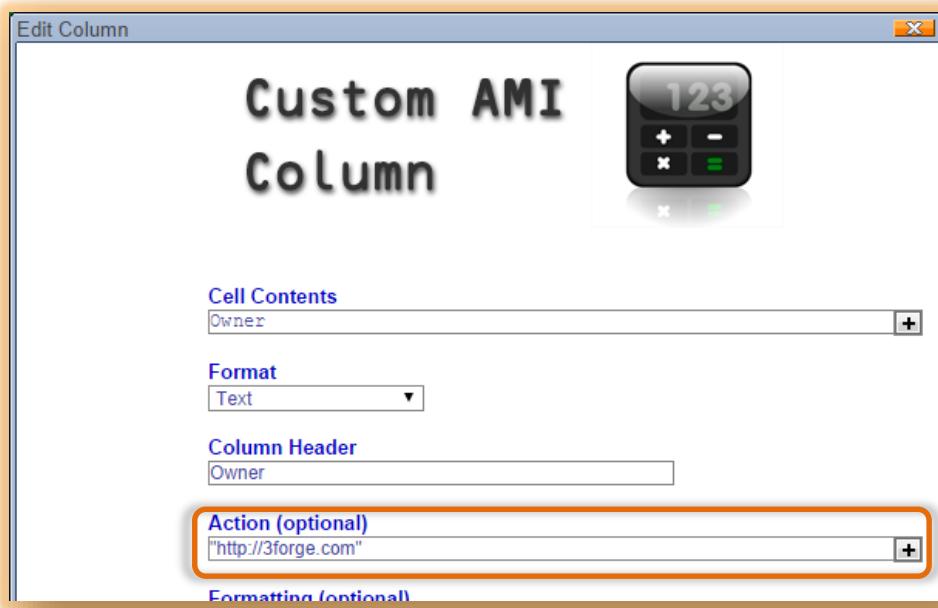


Price	Price > 70
\$67.74	false
\$69.69	false
\$68.13	false
\$47.06	false
\$46.52	false
\$46.04	false
\$80.09	true
\$29.20	false
\$28.97	false
\$29.37	false
\$59.51	false
\$60.13	false
\$59.38	false

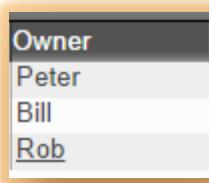
Price	Price > 70
\$80.09	true
\$75.48	true
\$75.60	true
\$83.30	true
\$82.14	true
\$88.71	true
\$87.45	true
\$89.15	true
\$75.98	true
\$78.28	true
\$76.65	true
\$76.17	true
\$80.61	true

URL's

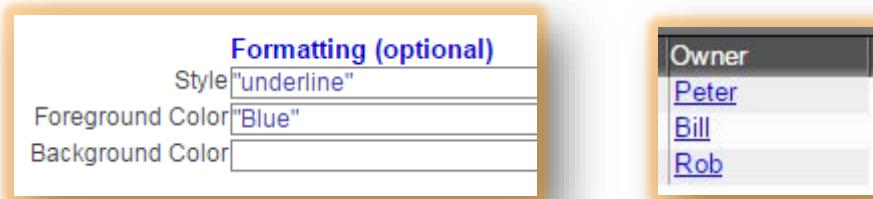
- URL's can be added to the **Action** field of column configurations. Clicking on an object in the column will then bring up the URL in a separate window
- 1) Add a new column or edit an existing column to bring up the **Custom AMI Column** window
 - 2) In the **Action** field of the **Custom AMI Column** window, add the URL
 - Note: The URL must evaluate to a string and must start with `http://` or `https://`
 - Example – <http://3forge.com>



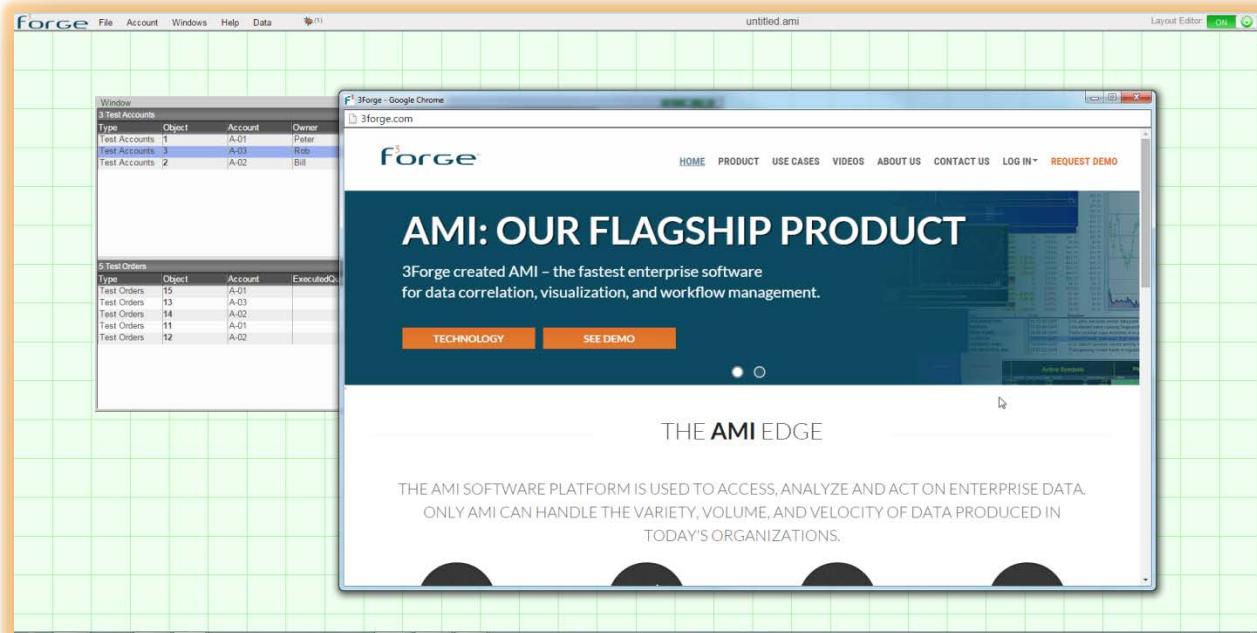
- 3) Moving a cursor over an object in the column will now underline the object – indicating that an action is available



- 4) The formatting of the column can be modified to make it easier to show that URL's are available



- 5) Clicking on an object in the column will now bring up the specified URL in a new window

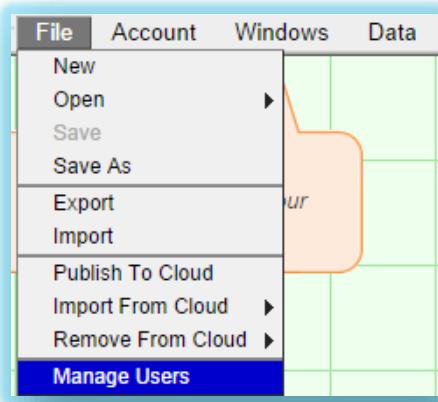


Cloud

AMI features a cloud system of sharing customized layouts between developers and users. This system allows for easy sharing of layouts between developers and users. Access to layouts can be managed by **administrators** who have access to the **Manage Users** option (*Please refer to the Access Matrix*).

Managing AMI Users

- 1) In order to manage the access of various users to AMI, choose **Manage Users** under the **File** menu from the toolbar



- 2) This will open up the **Manager Users** window as seen below

12 Users						
		User Name	Is Admin	Can Edit	Default Layout	Layouts
delete	edit	copy	alex	true	true	FR/L1/Accounts.ami
delete	edit	copy	bcooke	true	true	Charts.ami
delete	edit	copy	david	true	true	FR/L1/Test 123.ami
delete	edit	copy	dev1	false	true	FR/L1/Accounts.ami
delete	edit	copy	dev2	false	true	FR/L2/Accounts & Orders.ami
delete	edit	copy	dev3	false	true	FR/L1/Accounts.ami
delete	edit	copy	iman1	true	false	FR/L1/Accounts.ami
delete	edit	copy	iman2	false	true	FR/L1/Accounts.ami
delete	edit	copy	test1	false	false	FR/L1/Accounts.ami
delete	edit	copy	test2	false	false	FR/L1/Accounts.ami
delete	edit	copy	test3	false	false	FR/L1/Accounts.ami

Note: The **Manager Users** window can only be accessed by Admins. If there are no Admins (as is the case when initially installing AMI) then ALL users have access to the Manager Users window.

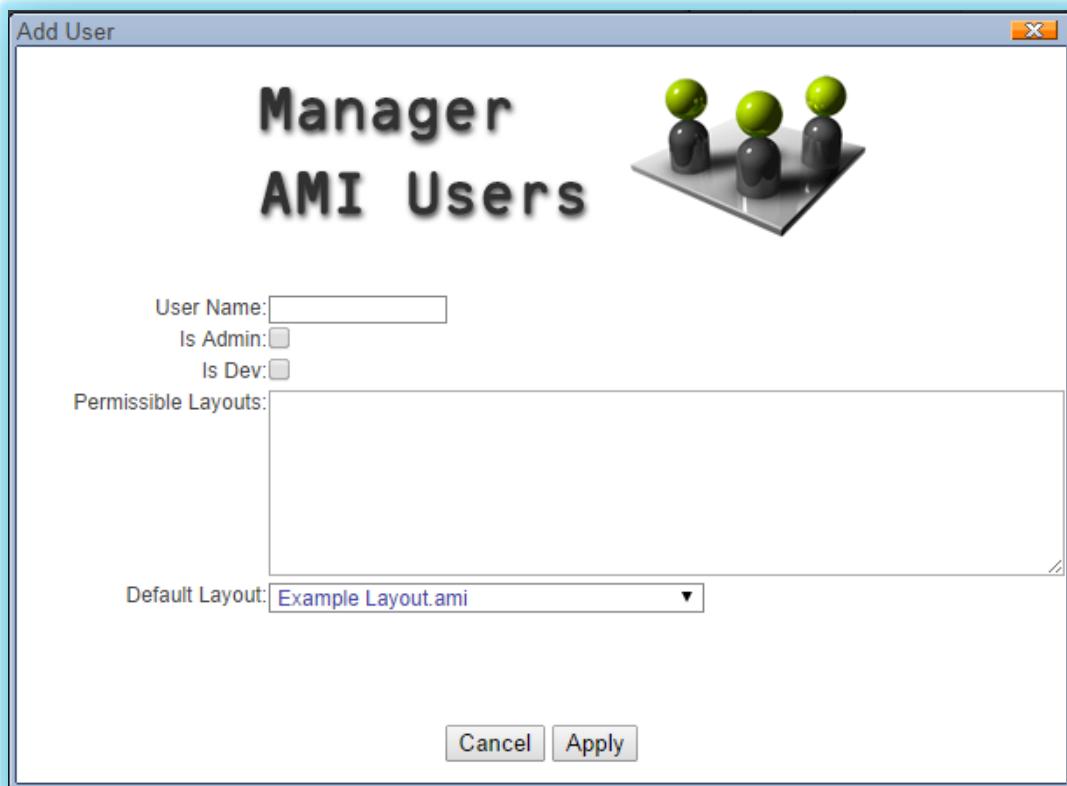
Access Matrix			
Feature	Admin	Developer	User
User Management	Yes	No	No
Build Layouts	Yes	Yes	No
Publish Layouts to Cloud	Yes	Yes	No
Import & Remove Layouts from Cloud	Yes	Yes	No
View Layouts	Yes	Yes	Yes*

*Only able to view layouts specified by admins

- Using the **Manager Users** window, admins will be able to **delete**, **edit**, or **copy** user profiles.
-

Adding New Users

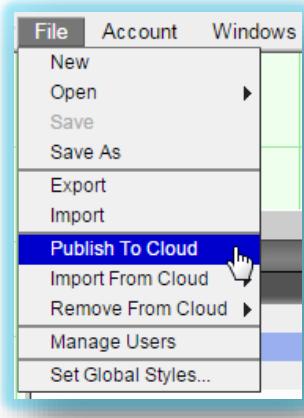
- 1) In the **Manager Users** Window, click on the **Add User** button in order to view the following window. (**Note:** Users must first be added through the configuration file)



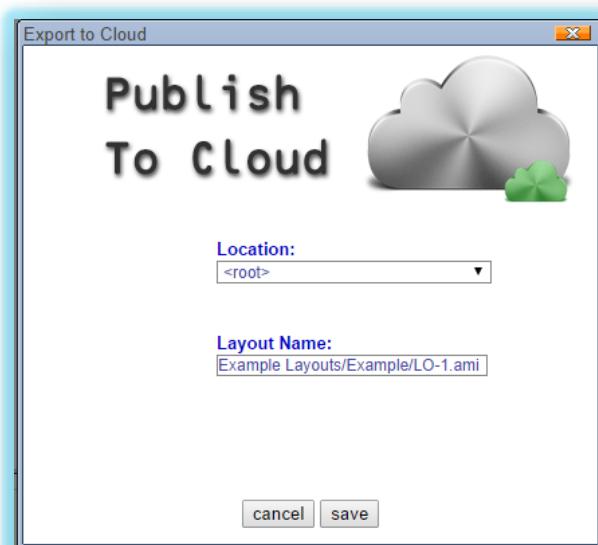
- 2) Enter the given **User Name** of the new user & click on either the “**Is Admin**” or “**Is Dev**” box in order to give the new user appropriate access to AMI
 - Refer to the **Access Matrix** for features available to Admins & Dev's
- 3) For non-Admin/Dev users, specify the layout(s) available to these users in the **Permissible Layouts** box
 - Each line should contain a single layout.
 - Layout names can be regular expressions (as defined by Java's regex standard).
Ex: `my_layout.*` would permit the user to access all layouts starting with `my_layout`
- 4) Admins may also choose a **Default Layout** for users – users will see this layout once they sign-in to AMI

Publishing Layouts to the Cloud

- 1) In order to publish new layouts to the cloud, choose **Publish To Cloud** under **File** from the toolbar



- 2) In order to save layouts to specific folders in the cloud, use / to separate folder names and add .ami to the layout name



Functions

Functions are very useful when building custom expressions in AMI. Functions are frequently used for (but not limited to) columns, additional filters in relationships, and charts.

Note: In the case of invalid inputs to a function (that cannot be caught at compile time) such as nulls, etc. the functions will return null.

```

abs(Number value)
cycle(Number offset, Object choices)
datamodelEnum(Object value, String datamodel, String dmTableName, String idColumn, String textColumn)
formatDate(Number millis, String format, String timezone)
formatNumber(Number millis, String format, String options)
gradient(Number value, Number value1, String color1, Number value2, String color2, Object moreNumbersAndColors)
inline(Object text)
power(Number base, Number exponent)
quote(Object text)
rand()
randGaussian()
round(Number value)
roundDown(Number value)
roundUp(Number value)
scale(Number value, Number source1, Number target1, Number source2, Number target2, Number moreSourcesAndTargets)
strLen(Object text)
timestamp()

```

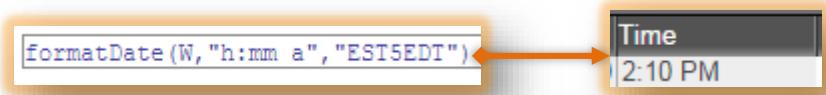
Some of the functions available for use are:

- **abs (Number value)**
 - Returns the absolute value
- **cycle (Number offset, Object... choices)**
 - Return the nth element from the array starting at zero (where offset is n and choices are the array). If n is larger than the number of choices or negative then the modulus of n will be used.
- **datamodelEnum (Object value, String datamodel, String dmTableName, String idColumn, String textcolumn)**
 - This function can be used in conjunction with the data model platform in order to create tables containing columns from multiple data models.
- **formatDate (Number millis, “String format”, “String timezone”)**
 - *Number millis:*
 - C (Created Time)
 - E (Expires Time)
 - M (Modified Time)
 - W (Current Time)
 - Any custom variable that contains a time stamp in milliseconds
 - *String format*

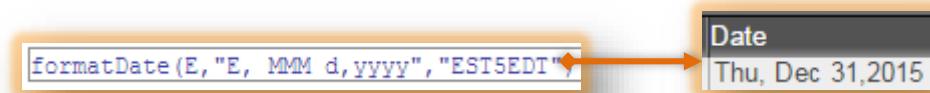
- Follows Java 1.5
 - *String time zone*
 - **formatNumber** (*Number millis, String format, String Options*)
 - Specifies the exact format of a numeric variable.
 - **gradient** (*Number value, Number value1, String color1, Number value2, String color2...*)
 - Used to apply color gradients to various visualizations inside AMI.
 - **inline**
 - This is an advanced function used for passing query syntax directly to a datasource (bypassing AMI's interpreter).
 - **power** (*Number base, Number exponent*)
 - Returns the exponential value
 - **quote**
 - Places quotes around the input string & escapes special characters within the input string. Can be used to quote a string for use in a query.
 - **rand**
 - Generates a random number
 - **randGaussian**
 - Generates a normally distributed random variable
 - **round** (*Number value*)
 - Rounds the value to the nearest whole number
 - **roundDown** (*Number value*)
 - Rounds the value down to the nearest whole number
 - **roundup** (*Number value*)
 - Rounds the value up to the nearest whole number
 - **scale** (*Number value, Number source1, Number target1, Number source2, Number target2...*)
 - Creates a numeric gradient; the value is the type on which the gradient will be made on.
 - **strLen** – Get string length
 - Returns the length of the given string
 - **timestamp**
 - Returns the current time
-

Example – using the formatDate function

Time:

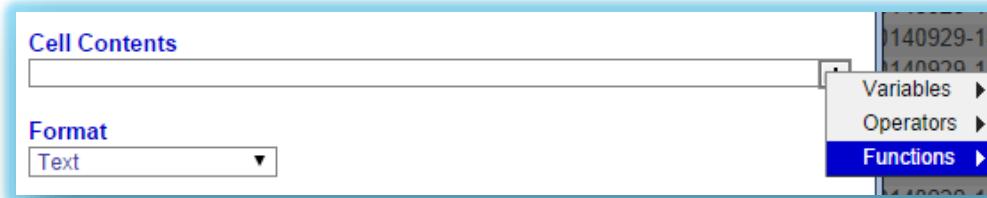


Date:

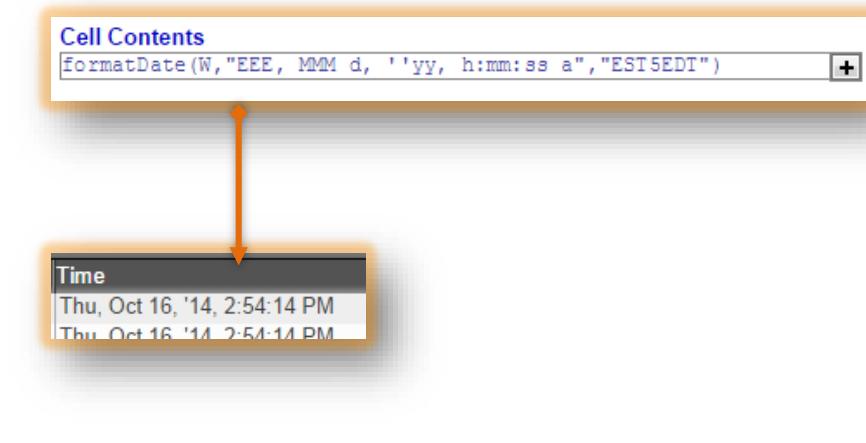


Using functions in *Cell Contents* of Columns

- Add Column
- Click on the icon in order to choose a **Function**

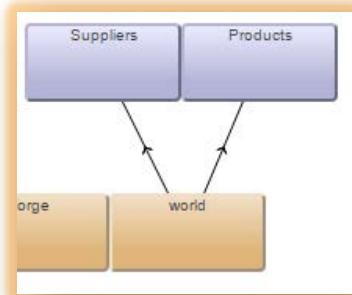


Example



Example – using the **datamodelEnum** function to add the *ContactName* column from the *Suppliers* table to the *Products* table

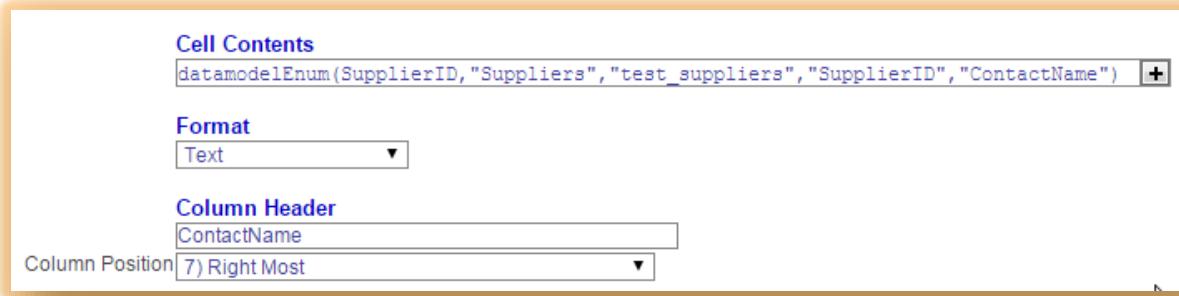
- 1) Add the *Suppliers* and *Products* data models to the data model platform (*please refer to the data model section for further details on using data models*)



- 2) Create a *Products* table based on the *Products* data model
- 3) Add a new column to the *Products* table. In the **Add Column** menu, input the **datamodelEnum** function in the **Cell Contents** section

```
datamodelEnum(SupplierID,"Suppliers","test_suppliers","SupplierID","ContactName")
```

- The function will use the common column between the two tables (*SupplierID*) in order to bring *ContactName* from the *Suppliers* table to the *Products* table



Using functions in *Actions of Columns*

Functions may be used to perform specific actions in columns when an object meets the specified requirements

Example

Action (optional)

```
strLen(Symbol) == 4 ? "ami_query:Query Order" : "ami_query:Query Execution" 
```

- If the string length of the symbol is 4, then Query Order; otherwise, Query Execution
 - Note: “ami_query:” is used to perform a **direct query**
-

Using functions in *Formatting of Columns*

Functions may be used to format columns to quickly interpret data

Example

Formatting (optional)

Style	Target > 4500 ? (strLen(Symbol) == 4 ? "bold" : "italic") : "strike"	<input type="button" value="+"/>
Foreground Color	Side == "B" ? "Blue" : "Green"	<input type="button" value="+"/>
Background Color	formatDate(M, "m", "EST5EDT") == 7 ? "yellow" : ""	<input type="button" value="+"/>

- **Style:**
 - If the Target is greater than 4,500, then follow the formatting (`strLen(Symbol) == 4 ? "bold" : "italic"`); otherwise, strike through
 - `strLen(symbol) == 4 ? "bold" : "italic"`
 - If the string length of the symbol is 4, then put it in bold; otherwise, put it in italics
- **Foreground color:**
 - If the side is B, then make it blue; otherwise, make it green
- **Background color:**
 - If the function `formatDate(M, "m", "EST5EDT")` returns a value of 7, make it yellow; otherwise, do nothing

Using the gradient function on a column

When using the gradient function, a variable is chosen, followed by a number at one end of the range of the variable and a color for that number. This is repeated until the full range of the variable is covered. When used in the formatting of columns, gradients can be very useful in quickly visualizing a trend in the data

Example

- A table with sample baseball data has a gradient applied to the *Losses* column. Low number of losses are green whereas high number of losses are red; a gradient between the two colors will be applied to the number of losses that lie between the low and high

Formatting (optional)

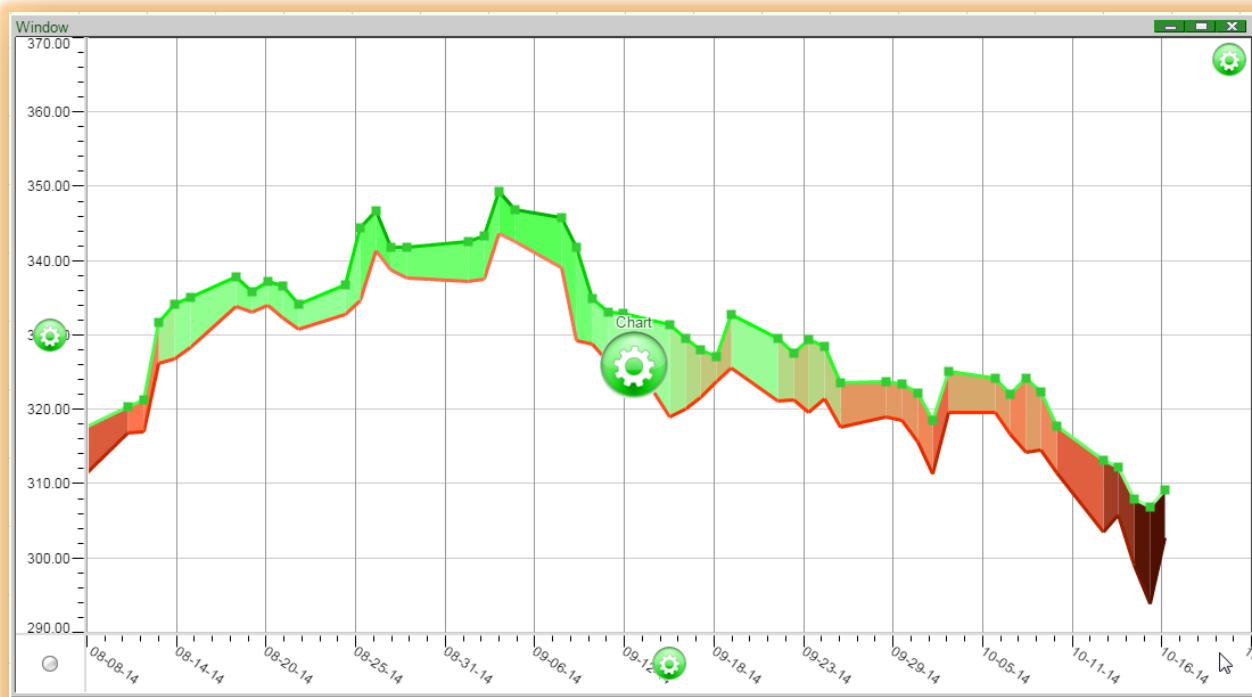
Style

Foreground Color

Background Color

Window - 2								
30 Rows								
HomeL	HomeW	League	Losses	Region	RoadL	RoadW	TeamID	Wins
28	53	AL	68	East	37	44	BOS	97
27	54	NL	68	Central	38	43	STL	97
29	52	AL	68	West	37	44	OAK	96
25	56	NL	69	East	41	40	ATL	96
31	50	NL	69	Central	37	44	PIT	94
30	51	AL	69	Central	39	42	DET	93
30	51	AL	70	Central	40	41	CLE	92
34	47	NL	70	West	36	45	LAN	92
30	51	AL	71	East	41	41	TBA	92
36	46	AL	72	West	36	45	TEX	91
31	49	NL	72	Central	41	41	CIN	90
37	44	AL	76	Central	39	42	KCA	86
34	47	NL	76	East	42	39	WAS	86
35	46	AL	77	East	42	39	NYA	85
35	46	AL	77	East	42	39	BAL	85
36	45	NL	81	West	45	36	ARI	81
42	39	AL	84	West	42	39	LAA	78
40	42	NL	86	West	46	34	SFN	76
36	45	NL	86	West	50	31	SDN	76
41	40	AL	88	East	47	34	TOR	74
48	33	NL	88	East	40	41	NYN	74
44	37	NL	88	Central	44	37	MIL	74
36	45	NL	88	West	52	29	COL	74
38	43	NL	89	East	51	30	PHI	73
45	36	AL	91	West	46	35	SEA	71
49	32	AL	96	Central	47	34	MIN	66
50	31	NL	96	Central	46	35	CHN	66
44	37	AL	99	Central	55	26	CHA	63
45	36	NL	100	East	55	26	MIA	62
57	24	AL	111	West	54	27	HOU	51

- Gradients can also be very useful in charts – a gradient applied to the area between two lines representing *high* and *low* prices of a stock in a chart can help users quickly pick up on a trend over time



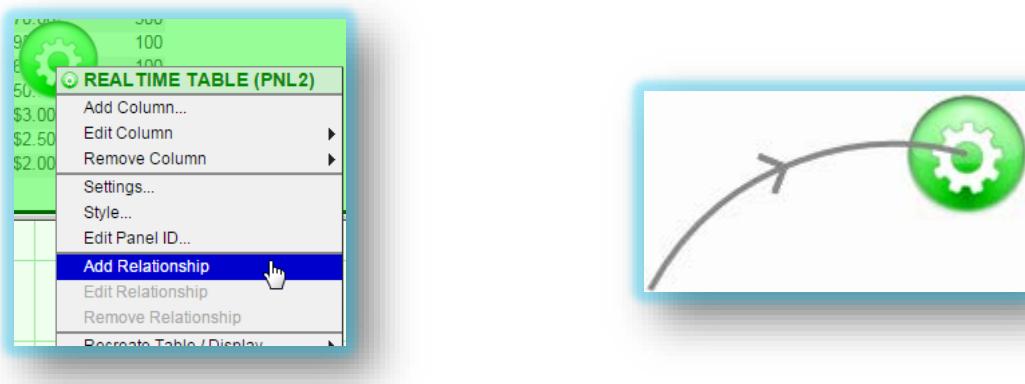
Please refer to the **Charts** section for further details on using functions in charts

Relationships

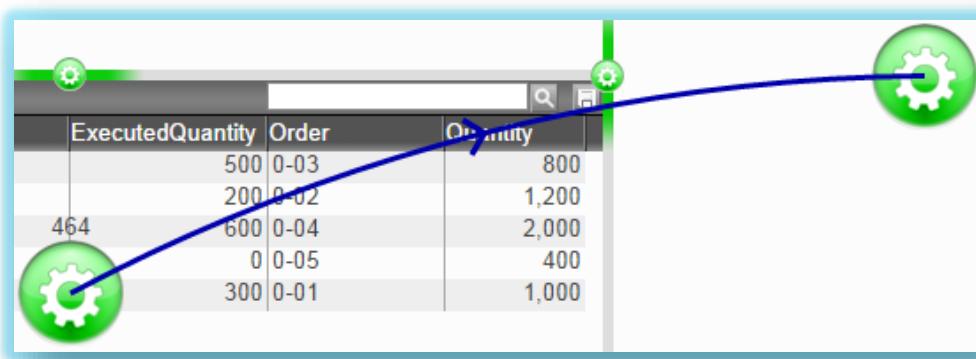
Relationships are used to establish links between the various panels in a layout. Establishing links between panels enables the drilling down of data from one table to another, the querying of static tables, and the rendering of displays in charts. When establishing a relationship, one panel will be considered the **Source** panel and the other as the **Target** panel. Relationships are established by first choosing the **Target** panel and then choosing the **Source** panel.

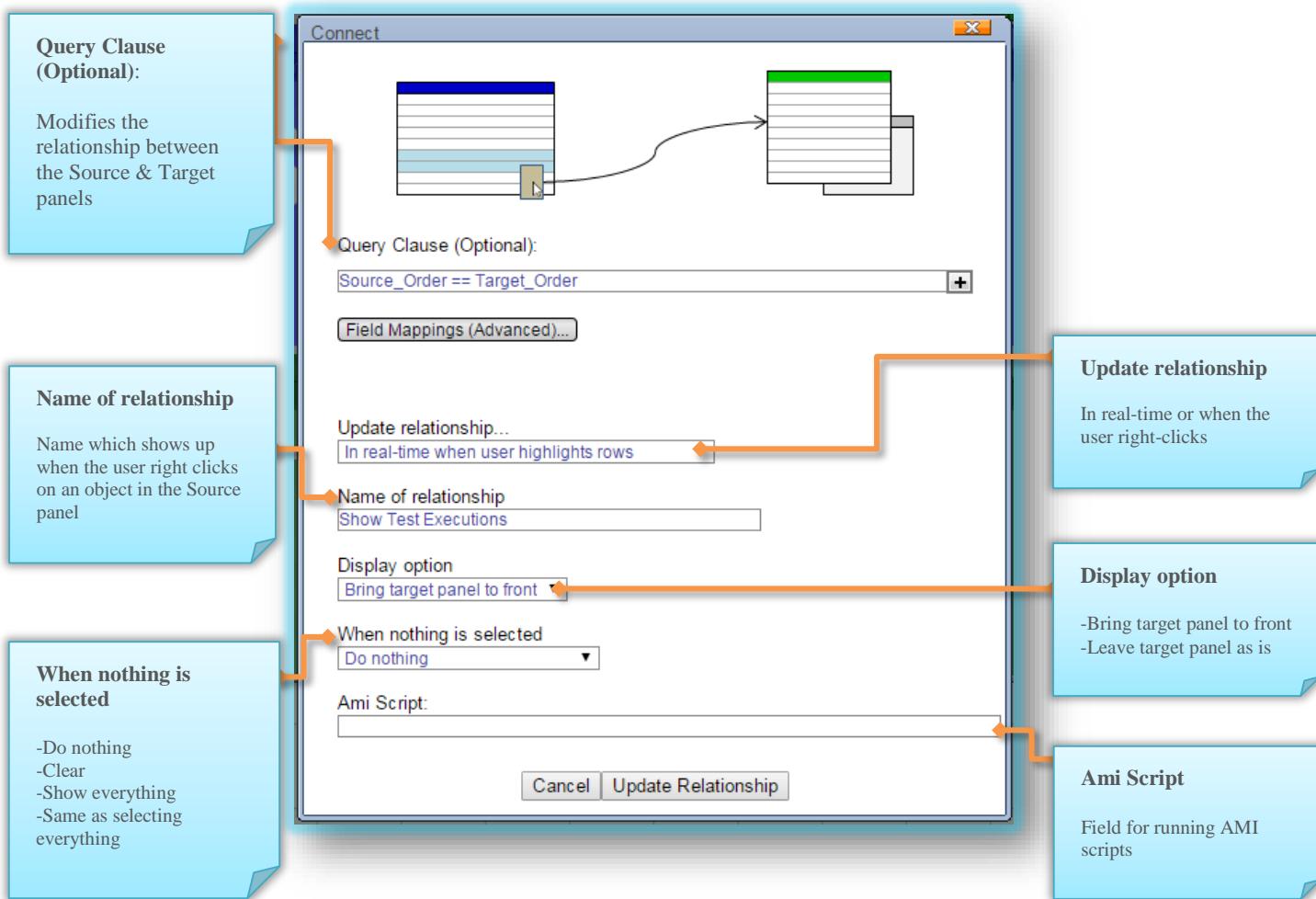
Setting up a basic relationship

- 1) In order to set up a relationship, click on the configuration button of the panel which you would like to be the **Target**.
- 2) From the menu, select **Add Relationship**. A relationship line with an arrow showing the direction of the relationship will appear and it will follow the mouse cursor.



- 3) Move the mouse cursor over to the configuration button of the panel you would like to establish the relationship to (this panel will be considered the **Source** panel). The relationship line will turn blue; click on the configuration button and the **Connect** (relationship) menu will now be displayed

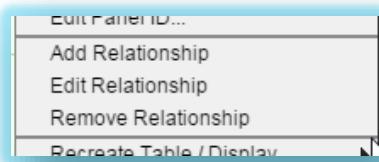




- 4) In order to establish a relationship between two panels, you must identify a commonality between the two panels.
 - In the example, **Order** is used in both the **Source** and **Target** panels.
- 5) Once all of the options have been configured, click on **Update Relationship** and the relationship will be established.

Editing & removing relationships

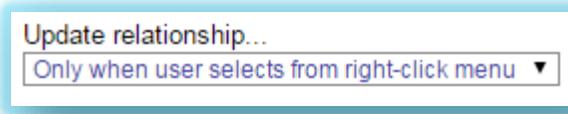
- 1) In order to edit or remove a relationship, click on the configuration button of the **Target** panel.



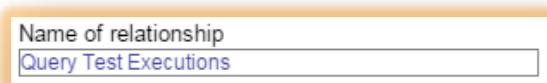
- 2) **Edit Relationship** will bring up the Connect menu. Once changes are made, click on **Update Relationship** in order to apply the changes made.

Using the Right-Click Menu to run queries (static relationships)

- The right-click menu is used to run queries which display data on static tables.
 - Note¹: The right-click menu can also be used to display data on real time tables (*covered in the following section*)
 - Note²: Although the right-click option is the default for static tables, static relationships may be updated in real-time
- 1) In order to set up a static relationship, click on the configuration button on a static table and choose **Add Relationship** from the menu.
 - Note: This static table will be considered the **Target** panel.
 - 2) A relationship line with an arrow showing the direction of the relationship will appear and it will follow the mouse cursor.
 - 3) Move the mouse cursor over the configuration button of the panel you would like to establish the relationship to (the **Source** panel). The relationship line will turn blue; click on the configuration button and the **Connect** menu will now be displayed.
 - 4) In the **Connect** menu, choose the “Only when user selects from right-click menu” option under **Update relationship**

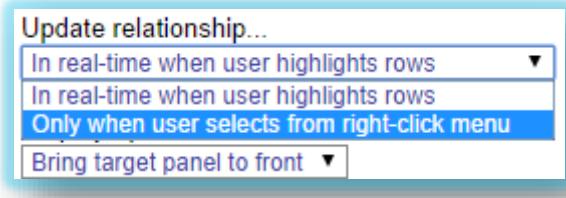


- 5) Rename the relationship (if necessary) in the **Name of relationship** field
 - This is the name which shows up when the user right clicks on an object in the **Source** panel



Quantity	Order	Quantity
500	0-03	800
200	0-02	100
600	0-04	2,000
0	0-05	400

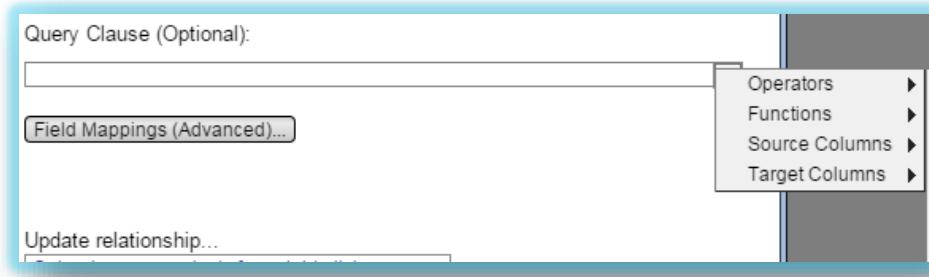
-
- Static relationships can be set up between two real-time tables by choosing the “Only when user selects from right-click menu” option under the **Update relationship** drop-down menu. Assign a name which will appear when the user right clicks an object in the **Source** panel. Click **Update Relationship** to finish and exit the menu.



Additional Filters

When working with tables or charts with large amounts of data, it may be necessary to limit the data being displayed. Additional filters are placed in the Query Clause in the **Connect** (relationship) menu and are applied along with the rest of Query between the **source** and **target** panels, helping to narrow down the data. Filters are highly customizable.

Additional filter statements can be made using all columns available in the **target** and **source** panels, operators, and functions. These options are available when clicking on the icon in the **Query Clause** field.



In the following example, there are 27,011 *Executions (target)* under 2,001 *Orders (source)*. In order to narrow down the number of executions, an additional filter can be applied to the relationship

Type	Account	Object	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif
Order	3F-00004	ORD-01581	7,700	\$42.08	S	ISRG	9,500	GTD
Order	3F-00000	ORD-01580	4,700	\$28.92	B	ROIQU	4,800	DTC
Order	3F-00005	ORD-01585	3,700	\$15.67	S	CTDC	4,000	DAY
Order	3F-00002	ORD-00980	3,000	\$75.52	B	ALTI	3,600	DAY
Order	3F-00007	ORD-01584	4,700	\$56.44	S	ESIC	6,600	DAY
Order	3F-00003	ORD-01587	5,100	\$28.54	B	AMPE	8,400	GTC
Order	3F-00002	ORD-02710	5,400	\$14.86	B	HSOL	5,500	DAY
Order	3F-00002	ORD-00983	4,900	\$66.71	B	GOODO	4,900	GTC
Order	3F-00001	ORD-01589	6,200	\$25.40	S	OMEX	9,200	GTC
Order	3F-00000	ORD-01588	3,000	\$59.61	B	NVTL	3,200	FOK
Order	3F-00000	ORD-02712	8,000	\$23.48	B	PACOW	9,500	DAY
Order	3F-00003	ORD-02713	4,600	\$49.41	S	PSEM	9,600	IOC

Type	Object	ExecutionTime	LastMarket	OrderID	Price	Quantity
Execution	3F-00004	20130830-15:10:58	NASD	ORD-00177	\$25.19	300
Execution	3F-00004	20130830-15:14:00	NASD	ORD-00177	\$25.84	300
Execution	3F-00004	20130830-15:39:02	BATS	ORD-00177	\$25.56	300
Execution	3F-00004	20130830-15:39:55	NASD	ORD-00177	\$25.81	200
Execution	3F-00007	20130830-15:54:38	NASD	ORD-00176	\$6.68	100
Execution	3F-00004	20130830-15:05:00	AMEX	ORD-00177	\$25.09	200
Execution	3F-00004	20130830-15:09:19	NASD	ORD-00177	\$25.17	200
Execution	3F-00004	20130830-15:10:34	NASD	ORD-00177	\$25.92	200
Execution	3F-00002	20130830-14:58:14	BATS	ORD-00179	\$29.13	200
Execution	3F-00002	20130830-15:02:43	NASD	ORD-00179	\$28.47	300
Execution	3F-00002	20130830-15:04:50	NASD	ORD-00179	\$28.32	100
Execution	3F-00002	20130830-15:10:44	BATS	ORD-00179	\$28.60	100
Execution	3F-00004	20130830-15:40:55	NYSE	ORD-00177	\$25.41	200
Execution	3F-00004	20130830-15:43:05	NASD	ORD-00177	\$25.61	200
Execution	3F-00004	20130830-15:44:49	NYSE	ORD-00177	\$25.90	100
Execution	3F-00002	20130830-14:55:07	NYSE	ORD-00179	\$28.29	300
Execution	3F-00002	20130830-15:43:47	NASD	ORD-00179	\$28.31	300
Execution	3F-00002	20130830-15:43:13	NYSE	ORD-00179	\$28.71	300

The following statement is used in the **Query Clause**:

Query Clause (Optional):

('Source_Object' == 'Target_OrderID') && (Target_LastMarket == "NASD" && Target_Quantity == 300 && Target_Price > 80)

- This statement will display only those executions where the Last Market is **NASD**, quantity is **300**, and the price is greater than \$80 (Note: the first part of the statement establishes the relationship between the two tables based on a common variable)

2,001 Order

Type	Account	Object	ExecutedQuantity	LimitPrice	Side	Symbol	Target	Tif
Order	3F-00004	ORD-01581	7,700	\$42.08	S	ISRG	9,500	GTD
Order	3F-00000	ORD-01580	4,700	\$28.92	B	ROIQU	4,800	DTC
Order	3F-00005	ORD-01585	3,700	\$15.67	S	CTDC	4,000	DAY
Order	3F-00002	ORD-00980	3,000	\$75.52	B	ALTI	3,600	DAY
Order	3F-00007	ORD-01584	4,700	\$56.44	S	ESIC	6,600	DAY
Order	3F-00003	ORD-01587	5,100	\$28.54	B	AMPE	8,400	GTC
Order	3F-00002	ORD-02710	5,400	\$14.86	B	HSOL	5,500	DAY
Order	3F-00002	ORD-00983	4,900	\$66.71	B	GOODO	4,900	GTC
Order	3F-00001	ORD-01589	6,200	\$25.40	S	OMEX	9,200	GTC
Order	3F-00000	ORD-01588	3,000	\$59.61	B	NVTL	3,200	FOK
Order	3F-00000	ORD-02712	8,000	\$23.48	B	PACQW	9,500	DAY
Order	3F-00003	ORD-02713	4,600	\$49.41	S	PSEM	9,600	IOC

529 Execution

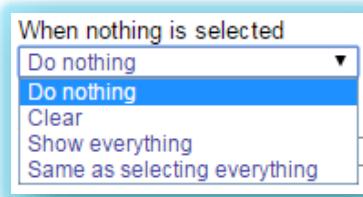
Type	Object	Account	ExecutionTime	LastMarket	OrderID	Price	Quantity
Execution	3F-00000		20130830-14:52:00	NASD	ORD-02714	\$89.98	300
Execution	3F-00000		20130830-15:04:25	NASD	ORD-02714	\$90.24	300
Execution	3F-00002		20130830-15:56:05	NASD	ORD-02723	\$83.39	300
Execution	3F-00007		20130830-15:27:55	NASD	ORD-02301	\$83.98	300
Execution	3F-00007		20130830-15:37:21	NASD	ORD-02301	\$82.61	300
Execution	3F-00007		20130830-15:08:14	NASD	ORD-02301	\$84.46	300
Execution	3F-00007		20130830-14:56:59	NASD	ORD-02301	\$82.08	300
Execution	3F-00000		20130830-15:37:36	NASD	ORD-02308	\$80.13	300
Execution	3F-00001		20130830-15:17:16	NASD	ORD-01564	\$80.34	300
Execution	3F-00004		20130830-15:48:28	NASD	ORD-00963	\$80.09	300
Execution	3F-00004		20130830-15:32:00	NASD	ORD-00963	\$81.66	300
Execution	3F-00006		20130830-15:52:21	NASD	ORD-01568	\$83.67	300
Execution	3F-00005		20130830-15:00:35	NASD	ORD-01577	\$84.45	300
Execution	3F-00005		20130830-15:22:51	NASD	ORD-01577	\$84.45	300
Execution	3F-00003		20130830-15:15:03	NASD	ORD-02338	\$90.03	300
Execution	3F-00003		20130830-15:15:47	NASD	ORD-02338	\$88.41	300
Execution	3F-00003		20130830-15:16:52	NASD	ORD-02338	\$88.84	300
Execution	3F-00003		20130830-15:28:29	NASD	ORD-02338	\$89.81	300

The amount of executions has been narrowed down to **529** from 27,011 executions. Filters can be applied directly to columns in order to narrow

*Please refer to the **Setting up Relationships with Charts** section for further details on using additional filters in chart relationships*

When nothing is selected

These options determine the behavior of target panels when the user does not select anything in the source panel. Note: the **Update relationship** field must be set to “**In real-time when user highlights rows**”

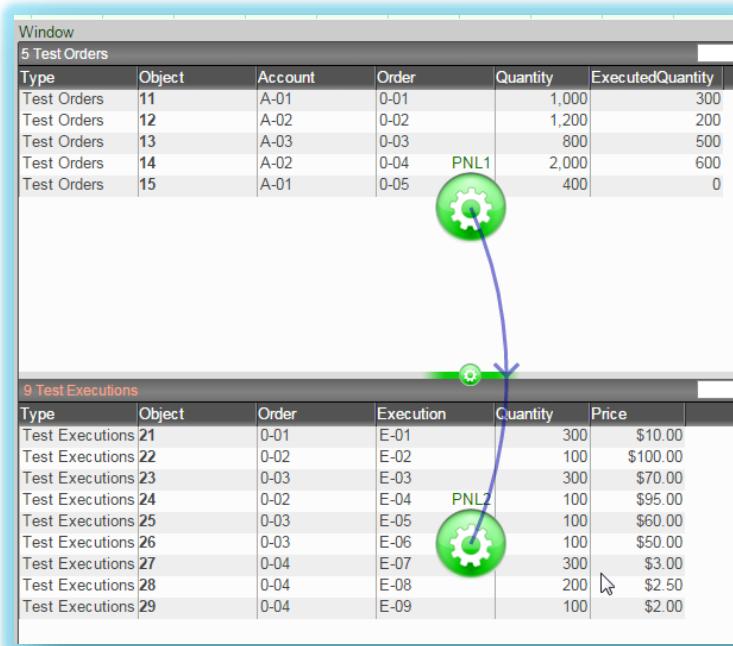


Do nothing	Results from the latest query will remain in the target panel
Clear	The target panel will clear of all results
Show everything	The target panel will display all of the records that belong to the data set chosen for the target panel
Same as selecting everything	The target panel will display all of the records that correspond to the records currently shown in the source panel

5 Test Orders					
Type	Object	Account	Order	Quantity	ExecutedQuantity
Test Orders	11	A-01	0-01	1,000	300
Test Orders	12	A-02	0-02	1,200	200
Test Orders	13	A-03	0-03	800	500
Test Orders	14	A-02	0-04	2,000	600
Test Orders	15	A-01	0-05	400	0

10 Test Executions					
Type	Object	Order	Execution	Quantity	Price
Test Executions	21	0-01	E-01	300	\$10.00
Test Executions	22	0-02	E-02	100	\$100.00
Test Executions	23	0-03	E-03	300	\$70.00
Test Executions	24	0-02	E-04	100	\$95.00
Test Executions	25	0-03	E-05	100	\$60.00
Test Executions	26	0-03	E-06	100	\$50.00
Test Executions	27	0-04	E-07	300	\$3.00
Test Executions	28	0-04	E-08	200	\$2.50
Test Executions	29	0-04	E-09	100	\$2.00
Test Executions	30	0-06	E-10	300	\$6.00

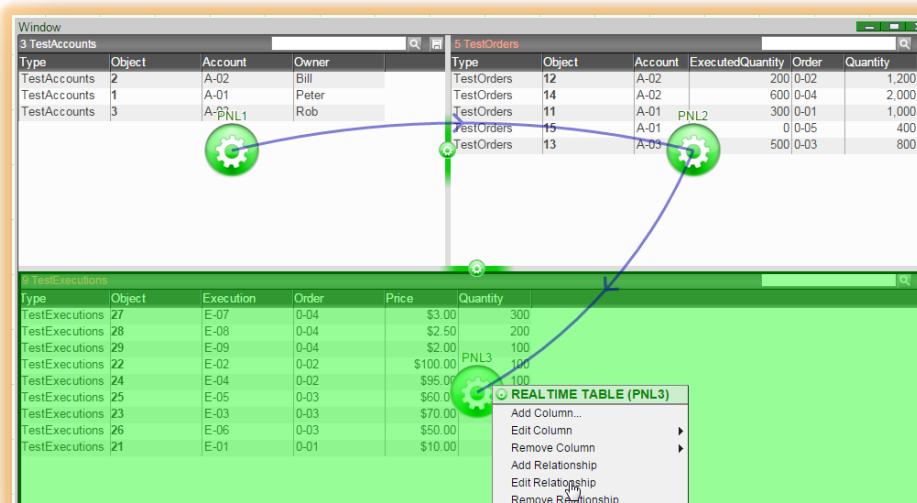
With the “**Show everything**” option, Object 30 shows up in the target panel although it does correspond to any source panel records



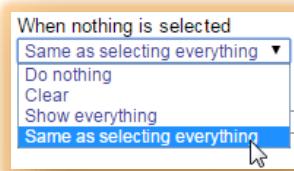
With the “Same as selecting everything” option, Object 30 does not show up in the target panel as it does not have any correlating records in the source panel

Example

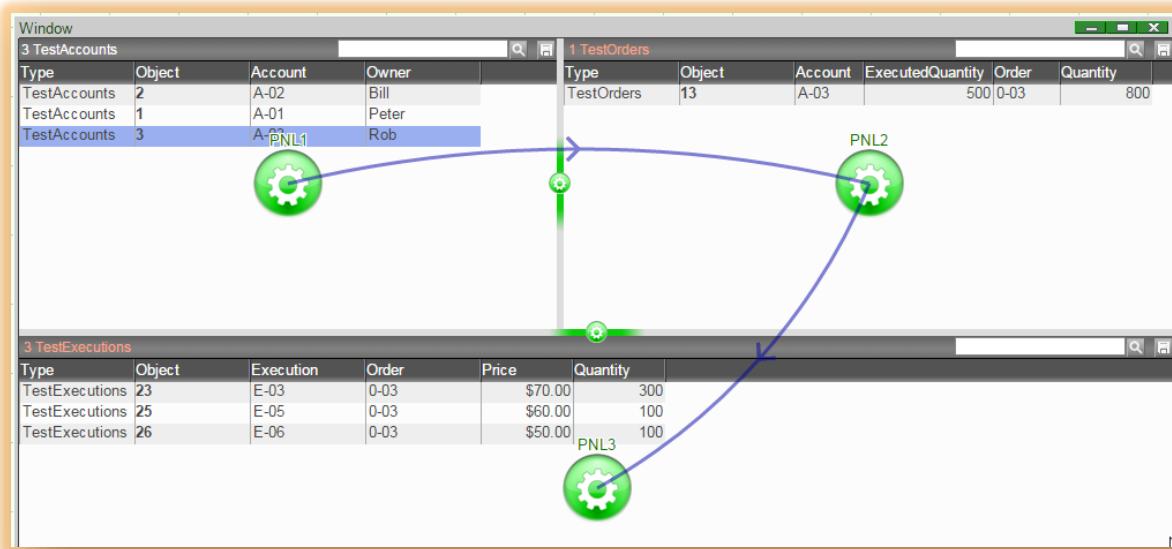
- Setting up relationships between 3 real-time tables (*Account-Orders-Executions*) in order to drive the display in the 3rd table when selecting records in the 1st table



- 1) When setting up the relationship between the 2nd (*Orders*) and 3rd (*Executions*) tables, select the “Same as selecting everything” option under **When nothing is selected**

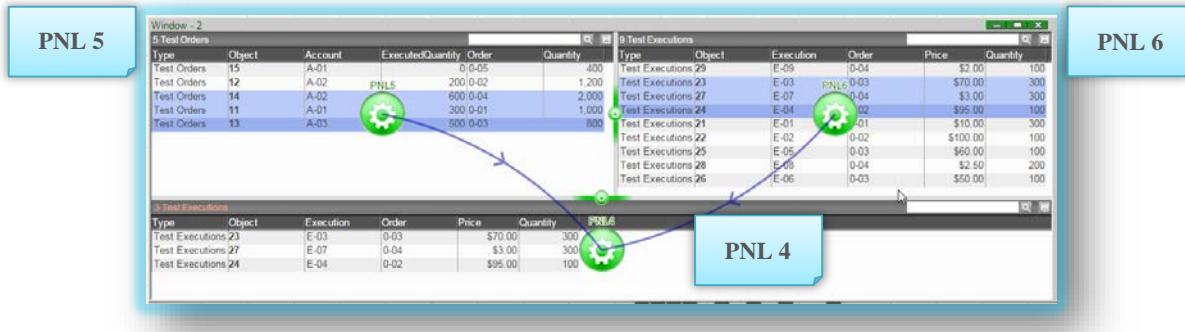


- 2) Selecting a record in the 1st (*Accounts*) table will now drive the display of records in the 3rd table



AMI Script

When making multiple relationships to a single panel, it may not be clear what records from which panels were used to drive the most recent query as seen in the example below



- The **Clear User Selection** script can be used to clear all other panel selections

Example

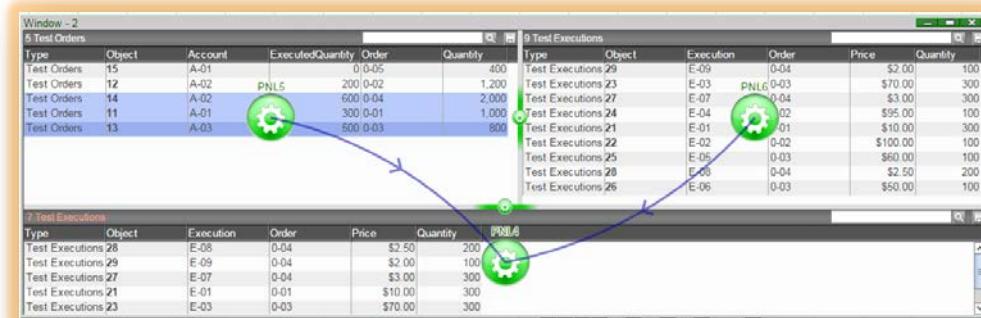
When creating/editing a relationship, place the following script into the **Ami Script** section of the relationship menu:

```
session.getPanel("Name of Panel").clearUserSelection()
```

Continuing from the example above, the following Ami script will be placed in the Ami Script section of the relationship menu between PNL4 & PNL5:

```
session.getPanel("PNL6").clearUserSelection()
```

This will clear all selections made in PNL6 when new selections are made in PNL5



Relationship Matrix

The following table lists all of the possible relationships that can exist between two tables (using real-time & static table types)

Key

RT	Real-Time
ST	Static

	Table 1	Table 2	Type of Relationship
1	RT	RT	RT
2	RT	RT	ST
3	RT	ST	ST
4	RT	ST	RT
5	ST	ST	ST
6	ST	ST	RT
7	ST	RT	ST
8	ST	RT	RT

Multiple Linking

Table/Display Type	Link to	Multiple links to	Link from	Multiple links from
Static (including Charts & Heat Maps)	✓	✓	✓	✓
Real-time (including Aggregate & Heat Maps)	✓	✓	✓	✓

Direct Query

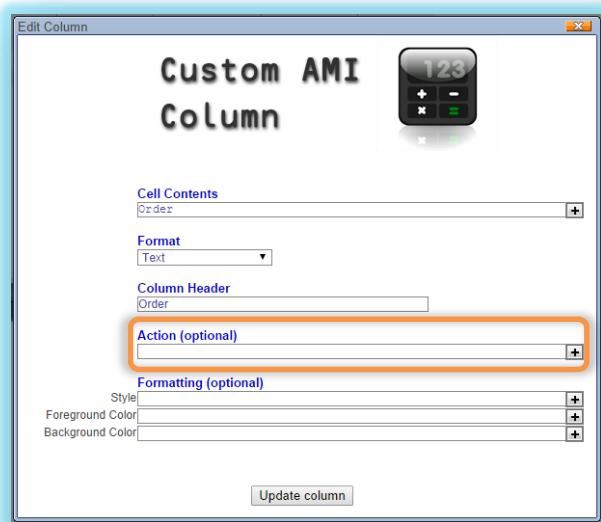
In addition to running queries by right-clicking on an object, users can directly run a query by clicking on an object in the **Source** panel. Note: This approach limits the user to running a query on a single row at a time, unlike the right click option which can be run after selecting multiple rows.

- 1) In order to set up a direct query, select a column in the **Source** panel to edit.



Example

- **Test Orders [Source – Real Time] and Test Executions [Target – Static]; edit the *Order* column in the **source** panel.**
- 2) In the **Edit Column** window, the command for the direct query will be placed in the **Action** field.



- 3) The statement for direct queries is:

- “`ami_query:[name of option in right click menu]`”

Example



- 1) *Query Test Executions* is what will be seen in the **Connect** menu when establishing a relationship between the two panels.



Note: The **Action** and **Name of relationship** must match in order for the direct query to work.

- 2) Establish a **static** relationship between the two panels.
- 3) Moving the cursor over an object in the **column** will now underline the object – indicating that it can be clicked in order to perform a query.

Order
0-01
<u>0-02</u>
0-03
0-04
0-05

Note¹: Use the right-click option to query multiple objects and direct query for single objects.

Note²: Direct queries may also be used in **aggregate tables**.

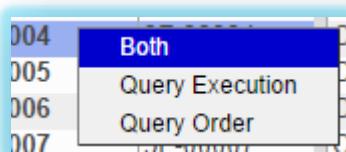
Running Simultaneous Queries on Multiple Target Panels from One Source Panel

When there is one table linked to multiple static tables, it is possible to run multiple queries to the static tables simultaneously. This is done by adding in a clause after the query (relationship) name

- 1) Add a new relationship or edit an existing relationship in order to bring up the **Connect** menu (the relationship options menu)
- 2) In the **Name of relationship** field under the Connect menu, add a [] and the name which you would like to group the different queries under
 - **Example** – if one real-time table is connected to two static tables and the user wants to run queries to both static tables at the same time, [|Both] would be added to the name of **both** relationships



- 3) **Both** would now appear as an option in the right click menu on the **real-time** table



Multiple simultaneous queries can also be set up with **direct queries**

- 1) Add or edit a column in the real-time table to bring up the **Custom AMI Column** menu
- 2) In the **Action** field of the menu, add the name of the group of queries after “ami_query”:
 - **Example** – “ami_query:Both”



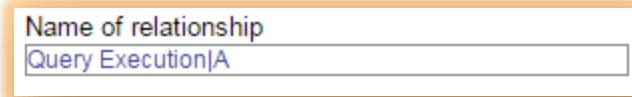
- 3) Clicking on an object in the column will now run both queries simultaneously

Multiple separators [|] can be used in the **Name of relationship** field under the Connect menu to establish different groups of queries.

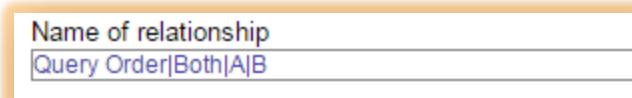
Note: This does not work with direct queries (i.e., “`ami_query:Both|A`” will not work)

Example

- Placing a query under group “A”:



- Placing a query under multiple groups in addition to “A” – “**Both**”, “**A**”, & “**B**”:

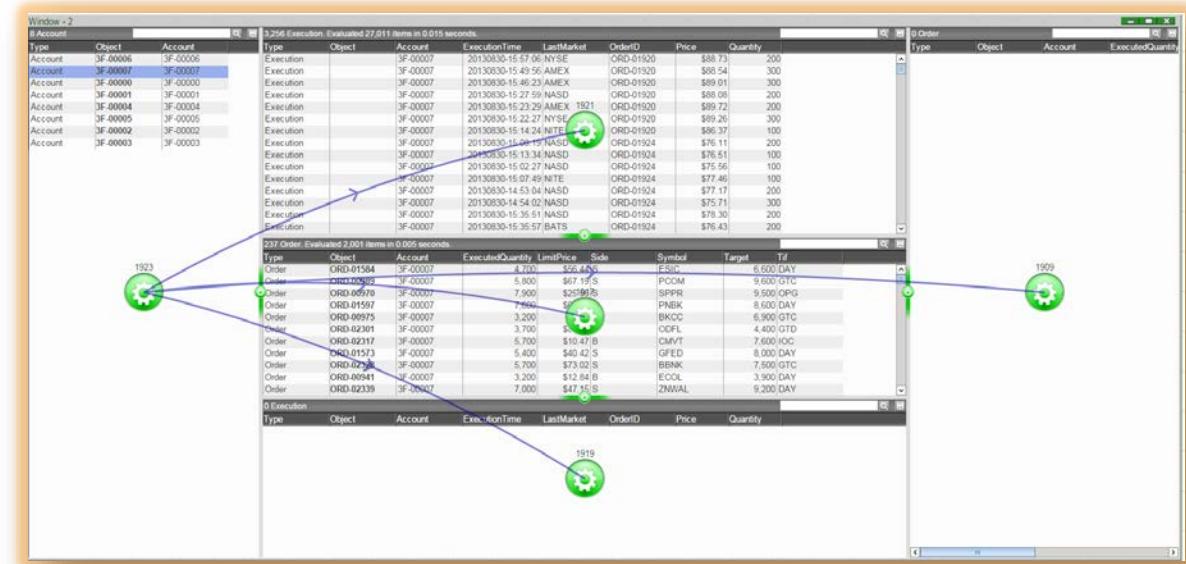
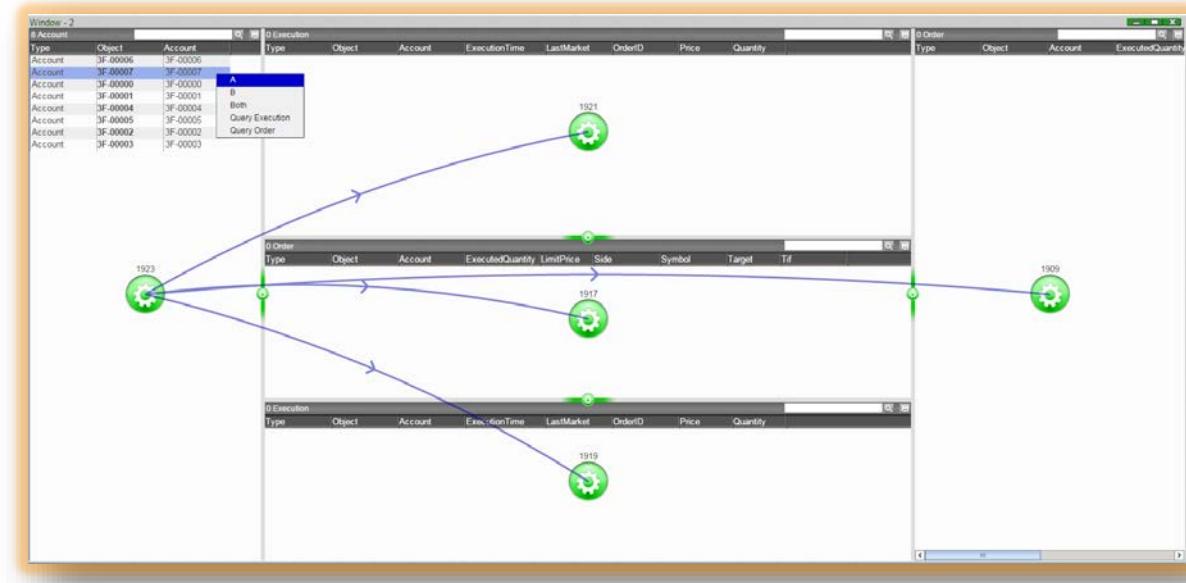


- Right clicking on an object in the **source** table will bring up all of the different query groups available

A screenshot of a software interface showing a table titled "8 Account". The table has three columns: "Type", "Object", and "Account". There are eight rows of data. The last row, which contains the value "3F-00005" in the "Object" column, is highlighted with a blue selection bar. A context menu is open over this row, listing five options: "A", "B", "Both", "Query Execution", and "Query Order". The "A" option is currently selected and highlighted with a blue background.

Type	Object	Account
Account	3F-00006	3F-00006
Account	3F-00007	3F-00007
Account	3F-00000	3F-00000
Account	3F-00001	3F-00001
Account	3F-00004	3F-00004
Account	3F-00005	A
Account	3F-00002	B
Account	3F-00003	Both

- Selecting A from the right click menu on the source table will run the queries grouped under A



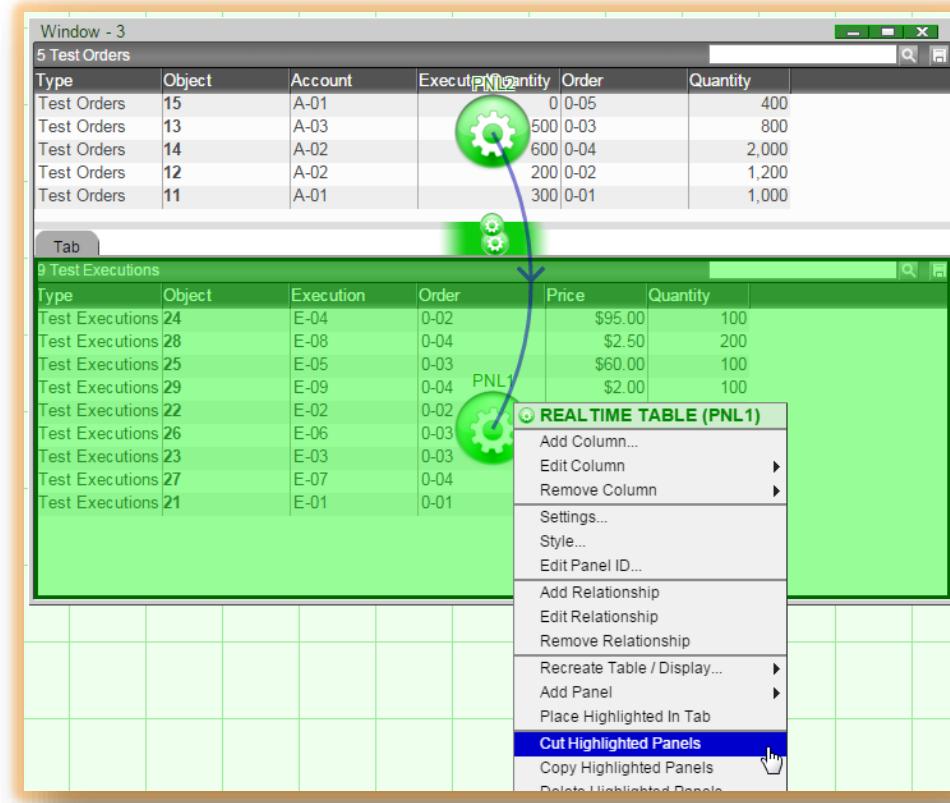
Cutting & Pasting

Tables/visualizations may be **CUT** out of a panel and pasted onto a blank panel. This is especially useful for tables/visualizations which are no longer required to be placed in tabs

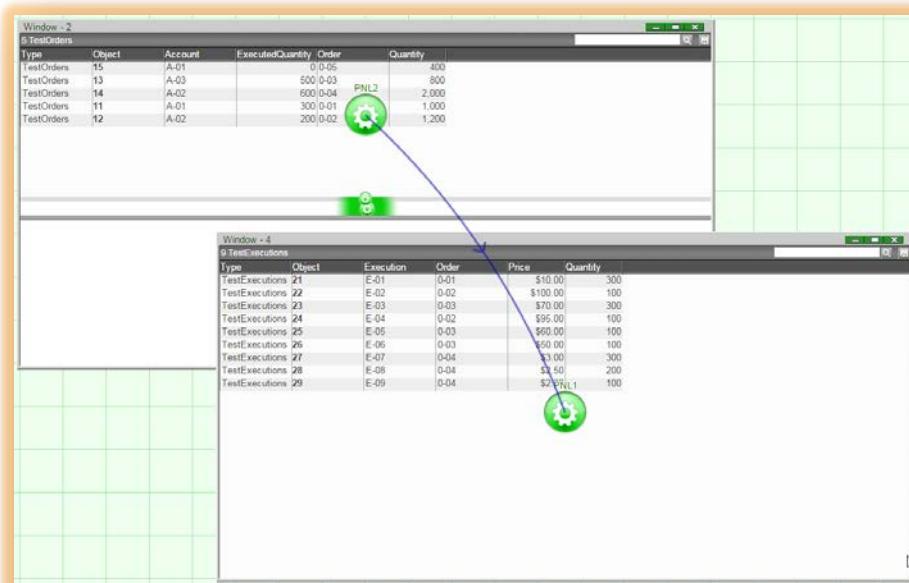
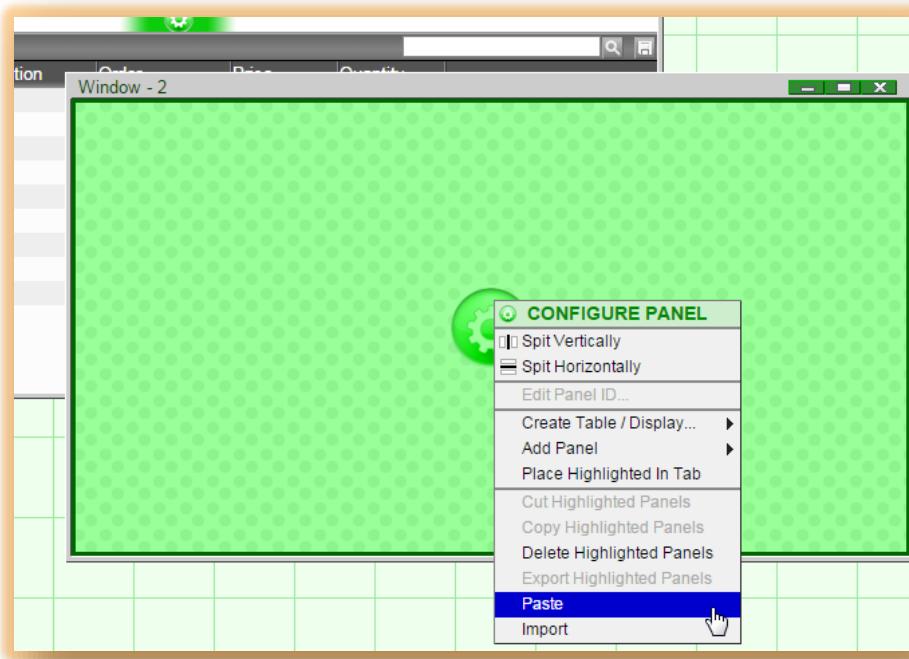
Note: all relationships established to the cut table/visualization will be maintained

Example – cutting a table of Executions out of a tab and placing it in a new blank panel

- 1) Click on the configuration button of panel to be cut. Select **Cut Highlighted Panels** from the list.



- 2) Add a new window (a blank portlet) and select **Paste** from the configuration button menu



A screenshot of the 3FORGE interface showing two windows. The top window, "Window - 2", contains a table titled "TestOrders" with columns: Type, Object, Account, ExecutedQuantity, Order, and Quantity. The bottom window, "Window - 4", contains a table titled "TestExecutions" with columns: Type, Object, Execution, Order, Price, and Quantity. A blue line connects the "Order" column in the "TestOrders" table to the "Order" column in the "TestExecutions" table, indicating a relationship or flow between the two data sets.

Type	Object	Account	ExecutedQuantity	Order	Quantity
TestOrders	15	A-01	0-0-05	400	
TestOrders	13	A-03	500-0-03	800	
TestOrders	14	A-02	600-0-04	PNL2	2,000
TestOrders	11	A-01	300-0-01		1,000
TestOrders	12	A-02	200-0-02		1,200

Type	Object	Execution	Order	Price	Quantity
TestExecutions	21	E-01	0-01	\$10.00	300
TestExecutions	22	E-02	0-02	\$100.00	100
TestExecutions	23	E-03	0-03	\$70.00	300
TestExecutions	24	E-04	0-02	\$95.00	100
TestExecutions	25	E-05	0-03	\$60.00	100
TestExecutions	26	E-06	0-03	\$50.00	100
TestExecutions	27	E-07	0-04	\$3.00	300
TestExecutions	28	E-08	0-04	\$1.60	200
TestExecutions	29	E-09	0-04	\$2.50	100

Exporting & Importing

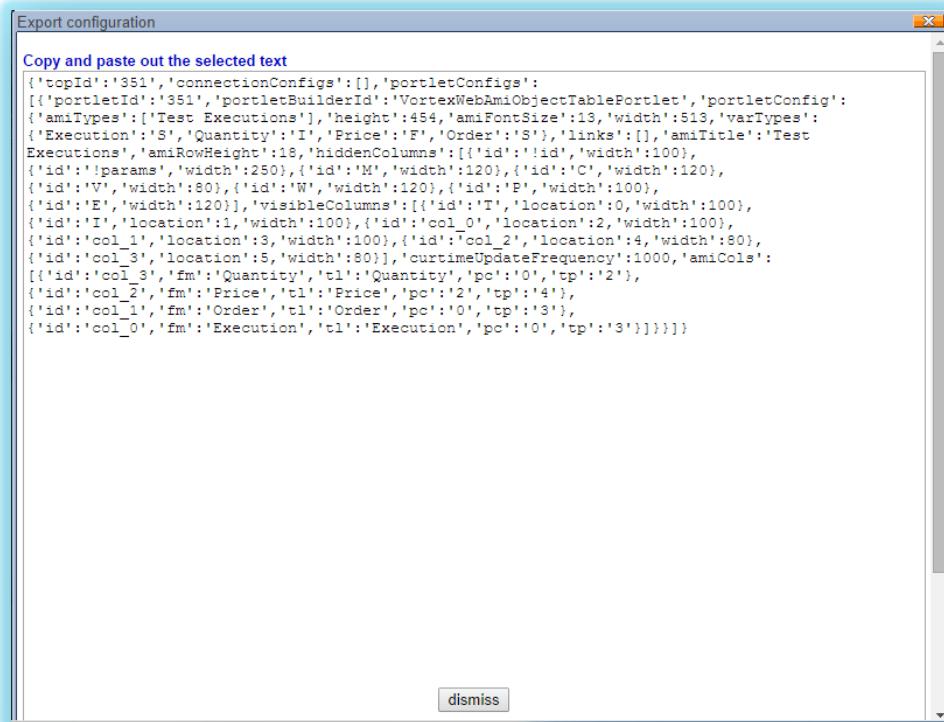
In AMI, individual panels, windows, and whole dashboards can be exported and imported to modify existing dashboards or to create new dashboards

Exporting/Importing Individual Panels

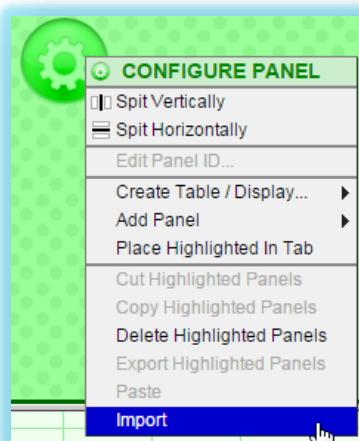
- 1) In order to export an individual panel, click on the configuration button of the panel and select **Export Highlighted Panels** from the menu



- 2) Copy the entire text that appears in the **Export configuration** window



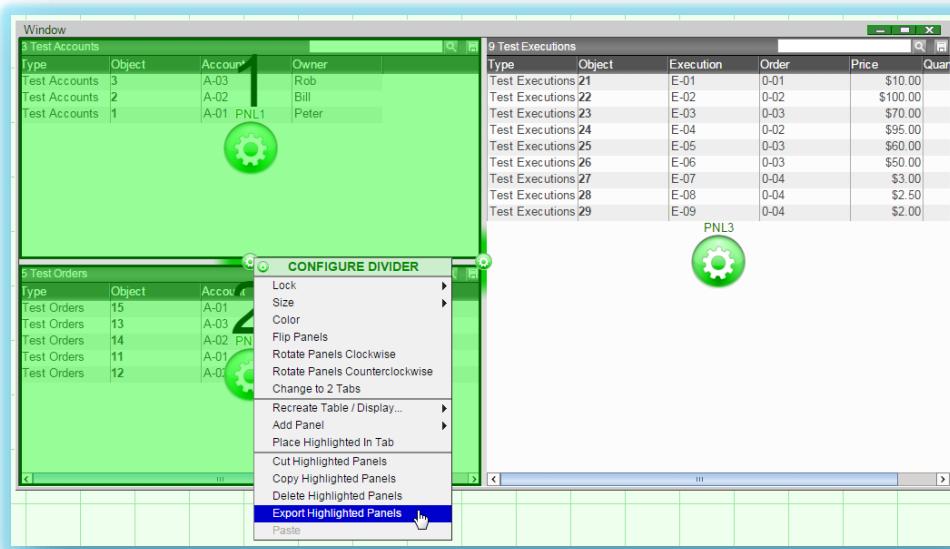
- 3) Click on the configuration button of a blank panel or window and select **Import** from the menu

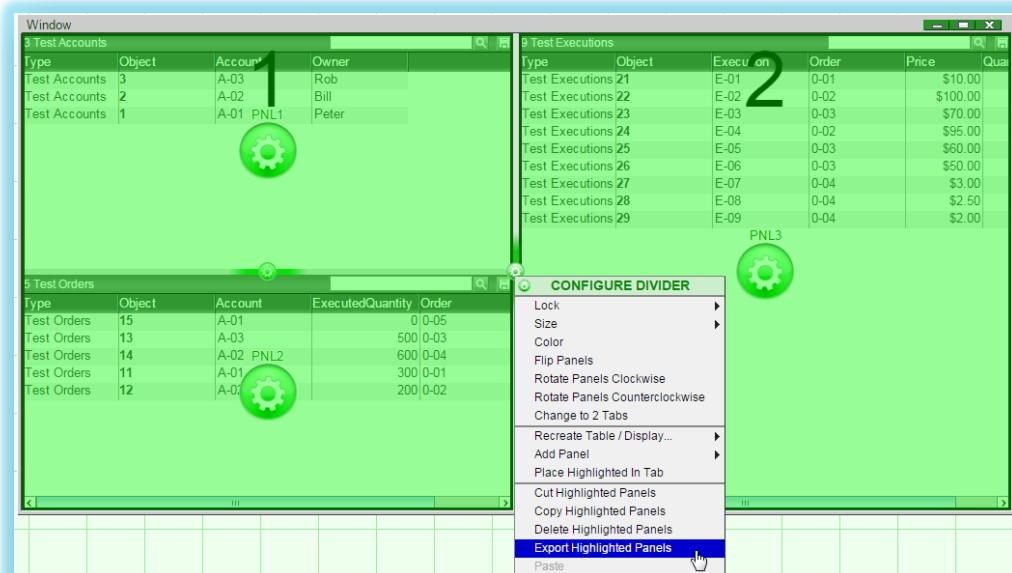


- 4) In the **Import configuration to blank Portlet** window that appears, paste the exported text and click *import*

Exporting/Importing Multiple Panels or Entire Windows

- 1) In order to export multiple panels or entire windows, click on the configuration button of the appropriate divider – this will show which panels are selected





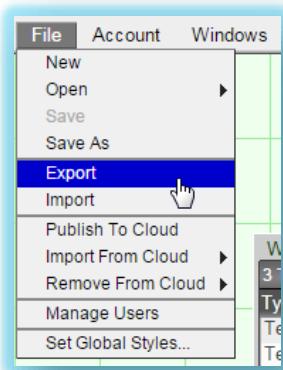
- 2) Click on **Export Highlighted Panels** and follow steps 2-4 from *Exporting/Importing Individual Panels*

Note: Panels with relationships will maintain this relationship when being exported and imported

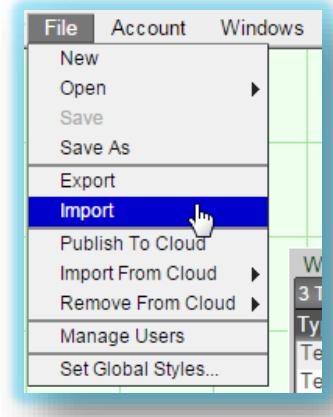
Exporting/Importing Dashboards

Dashboards which contain multiple windows can be exported and imported in their entirety.

- 1) Click File in the menu and select **Export**



- 2) This will bring up the **Export configuration** window. Copy the entire text that appears in this window
- 3) Open a new layout and select **Import** under File in the menu



- 4) In the **Import configuration** window that appears, paste the exported text and click *import*.

Note: The import option under File can also be used to import individual or multiple panels and windows into new dashboards

Data

Data Statistics

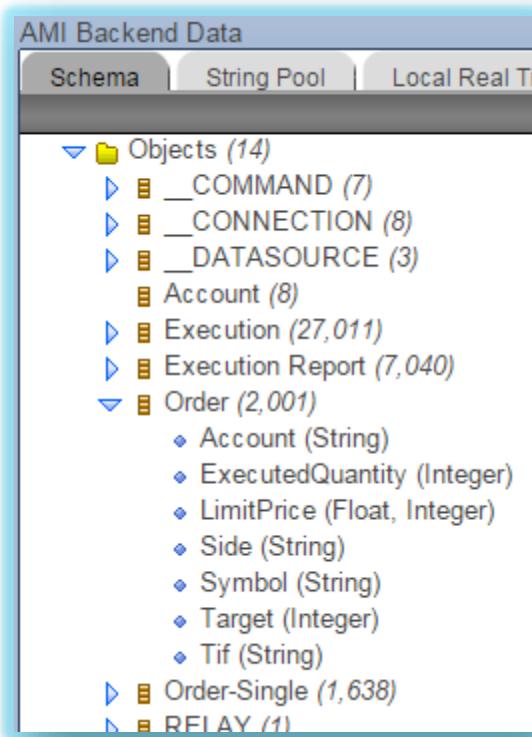
The Data Statistics window gives an overview of all the data that is stored in AMI's in-memory storage within the **AMI Center**. Data queried from databases will not show up in this window, but the connections to the databases will be displayed.

Schema

The schema tab shows an overview of the stored data's structure. Each item under **Objects** represents a table, while the sub-items represent the fields within that table. Items with "__" before the object name represent reserved system tables. All other objects are data that have been pushed into AMI.

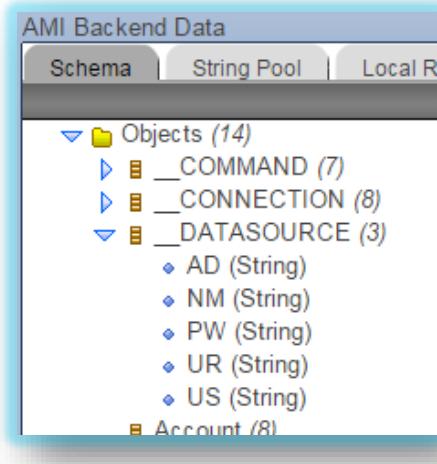
Example – schema for an object

This is the schema for a type of object called *Order*. The number in parentheses next to the title indicates the number of Order Objects that have been pushed into AMI. Expanding the section below that shows the different fields that have been provided with the Orders. One field on an Order below is the Symbol.



Example – schema of system data

The following is a Schema for the information about applications connecting to AMI. Every time a connection is established, an object is produced that describes that connection. The items below are the different parameters that describe a connection



String Pool

The string pool tab is a more advanced view of the Data Schema. To conserve space in memory, AMI will take specific string values and turn them into string pools.

This tab displays all of the strings that AMI has pooled. All names of Objects and their parameters are pooled and given an ID, and labeled as "TYPE / APP ID.". Furthermore, if single quotes are used to define the values of a parameter, then those values will also be pooled and given an ID, and labeled as "PARAM VALUE." Single quotes are very useful for values that tend to repeat themselves across rows of data, such as the day of the week. (See API for more information on string pooling.)

Local Real Time Objects

This tab is useful for determining how much the browser is being stressed. The table displays the data that is being pushed to the front end. This does not include static views and data being pulled from databases.

The "Count" column displays the number of data points for that type of object. The "Consuming Panels Count" column displays the number of panels that are actively displaying that data for the current user's dashboard. If the same real time data is being displayed in multiple tables in a dashboard, this number will tell the user how many tables it's actively updating.

Commands

Commands allow users to take action on objects in the front-end. Commands are developed and established through the back-end API. Once a command is established in the back-end, it will be available for use to the front-end users. Commands can be used by the front-end user to send information to the back-end or take action on an object (such as stopping a certain trade). Commands can be configured to be available to certain users (e.g., users with the right clearance) and/or on specific objects that meet certain parameters (such as a large quantity order). This part of the documentation will only cover what the user will see on the dashboard, please refer to the **Backend API** documentation on further details on developing and using commands.

Below is a real-time table with some sample executions (including quantity and price)

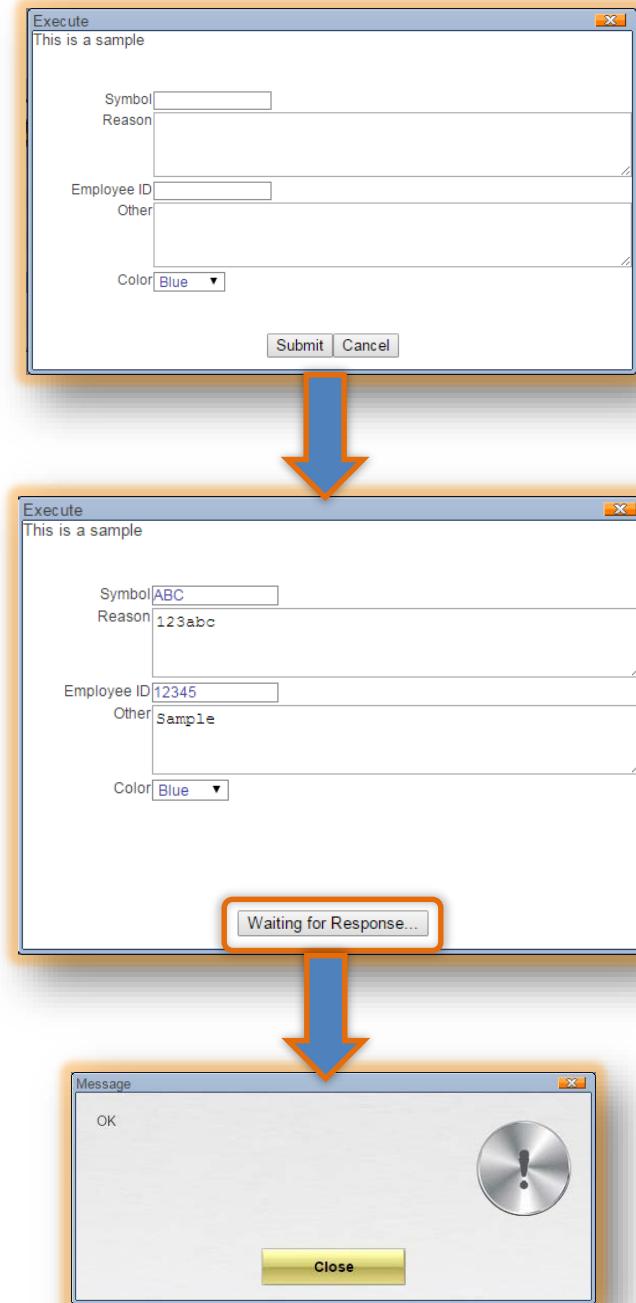
9 Test Executions					
Type	Object	Order	Execution	Price	Quantity
Test Executions	21	0-01	E-01	\$10.00	300
Test Executions	22	0-02	E-02	\$100.00	100
Test Executions	23	0-03	E-03	\$70.00	300
Test Executions	24	0-02	E-04	\$95.00	100
Test Executions	25	0-03	E-05	\$60.00	100
Test Executions	26	0-03	E-06	\$50.00	100
Test Executions	27	0-04	E-07	\$3.00	300
Test Executions	28	0-04	E-08	\$2.50	200
Test Executions	29	0-04	E-09	\$2.00	100

Commands are accessed by right-clicking on the object in the table. Notice that certain commands are greyed out and may not be accessed. This means that the object does not meet the right requirements for that command. In this example, Command 2 has been established to only work when the quantity is equal to 100. Also notice that commands can be categorized (such as Command 4)

9 Test Executions					
Type	Object	Order	Execution	Price	Quantity
Test Executions	21	0-01	E-01	\$10.00	300
Test Executions	22	0-02	E-02	\$100.00	100
Test Executions	23	Command 2	E-03	\$70.00	300
Test Executions	24	Command 1	E-04	\$95.00	100
Test Executions	25	Command 3	E-05	\$60.00	100
Test Executions	26	Category A	E-06	\$50.00	100
Test Executions	27	Command with Fields	E-07	\$3.00	300
Test Executions	28	0-04	E-08	\$2.50	200
Test Executions	29	0-04	E-09	\$2.00	100

Clicking on a command will bring up the **Execute** window where the user may input information into fields (if available) and then submit the command to the backend. Once submitted, a **Waiting for Response** button will appear at the bottom of the window and a **Message** from the back-end will soon appear

A command with fields



Commands can be directly called by clicking on an object. This is done by inputting a command in the **Action** field of a column menu. The Action must be prefixed with “ami:

To establish a direct command, add a new column or edit a column in order to view the **Custom AMI Column** menu

In the Action field of the menu, the syntax should be:

```
ami:application_ID/command_name?param1=value1&param2=value2
```

Example

Action (optional)

```
"ami:sample application login/Test?color=brown&employee ID="+empid
```

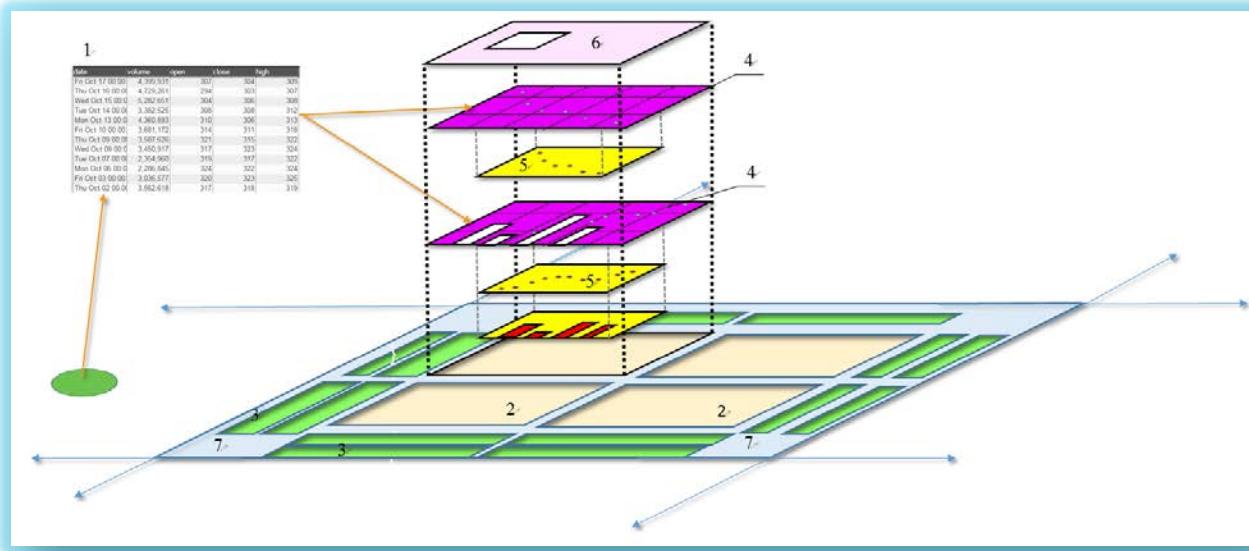


Clicking on an object in the column will now run the command **Test**

Note: Custom formatting can be used in order to indicate that commands are available for use in the column

Charts

In AMI, charts are a highly customizable way of displaying vast amounts of data. Just as tables, charts can be built on any empty panel. Below is a logical representation of the various components that make up a chart.



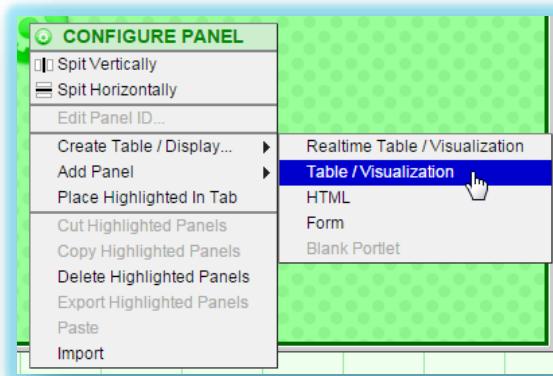
1. Data Model(s)	The data that is to be represented in the chart. The data can be from a data source or an Object stored within AMI Center. <u>Note:</u> a chart can have multiple data models.
2. Plot	Area where the chart will be created
3. Axis	A set of data
4. Rendering layer	A ‘slice’ of the chart associated with a specific data model
5. Series	Reference lines tied to a series
6. Legend	A key of the series represented in the plot
7. User Control	Controls the transparency of all of the series in the chart, available in work mode

Outline of steps in creating a chart:

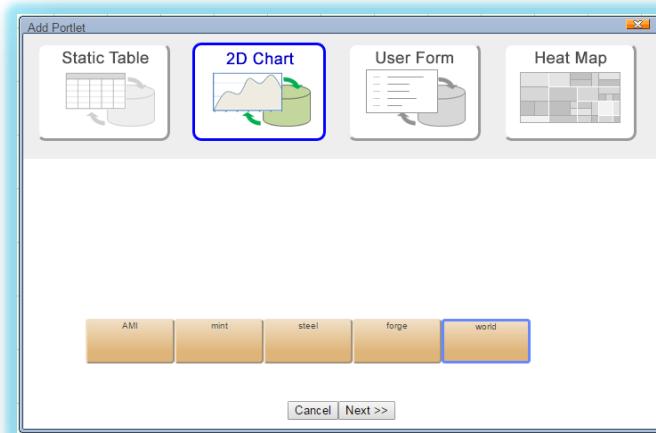
Create a new chart panel → add data model(s) → add rendering layer(s) → add series → add legend (optional)
+
Modify styles & establish relationships

Creating a new chart & adding a data model

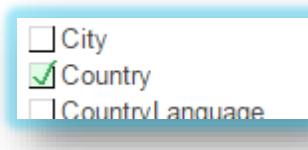
- 1) In order to create a new chart, click on the configuration button of a blank window/panel and select **Table/Visualization**



- 2) This will bring up the Add Portlet wizard (datamodel platform). Select **2D Chart** and the datasource from which the data will be retrieved.

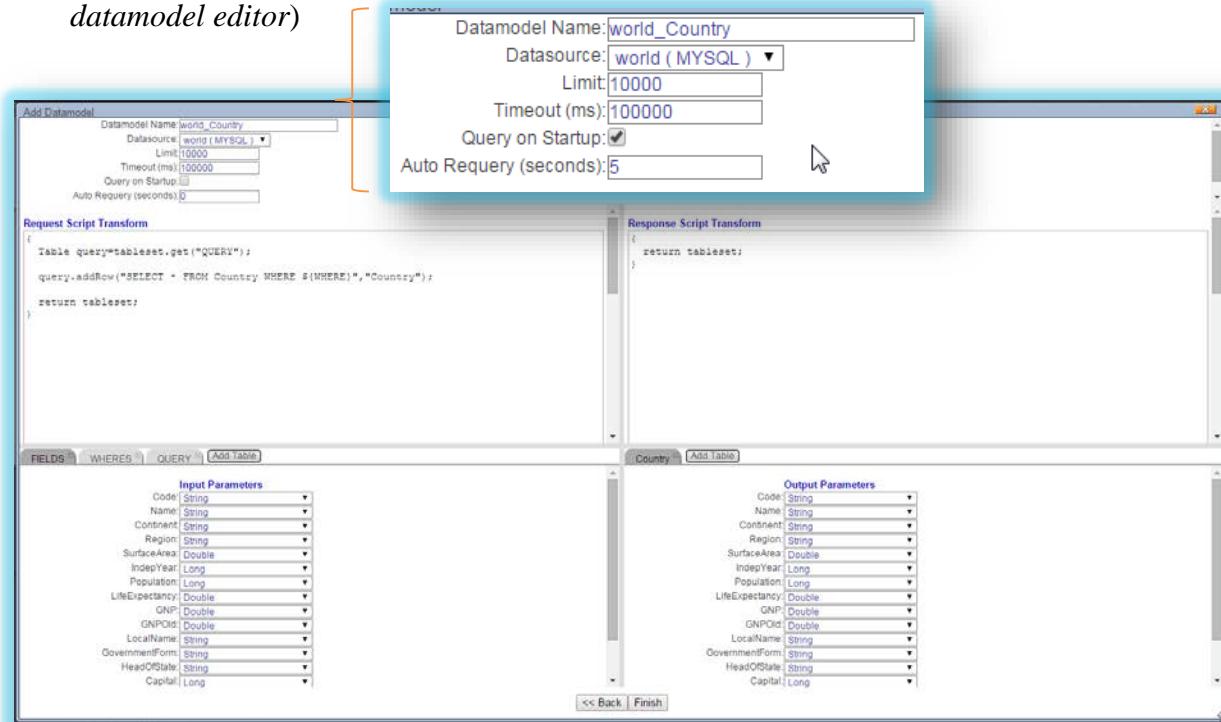


- 3) In the next window, select the data to be used and click **Next**.

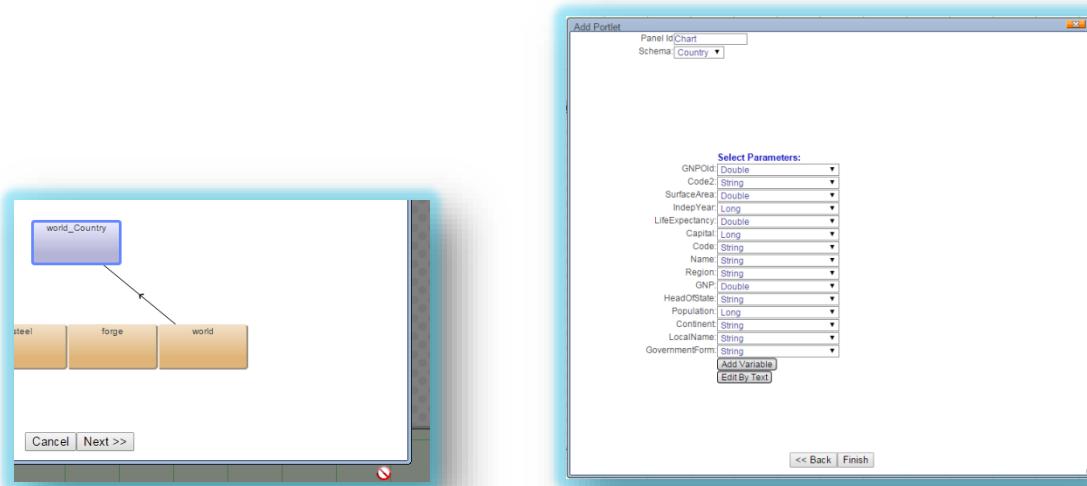


- 4) In the final part of the wizard (datamodel editor), make any necessary changes to the options (e.g., turning on **Query on Startup** or applying an **Auto Requery**) and click **Finish**.

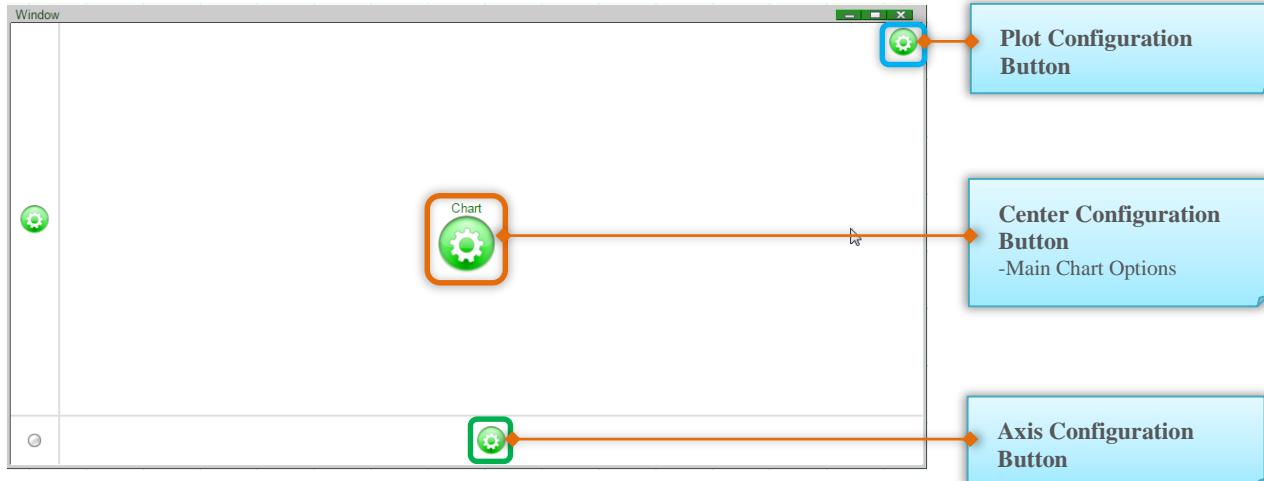
(Note: Please refer to the Data Model section for more information on the layout of the datamodel editor)



- 5) The newly created datamodel will appear in the datamodel platform. Click **Next** for the final review window. After checking to see that all of the columns are returned as the correct type, click **Finish** to create the chart.



- 6) Click on the **center configuration button** in order to view the **main chart options**.
Additional datamodels can be added and existing data models can be edited or unlinked.
(Note: Datamodels can also be edited through the datamodel platform).



- 7) In order to render (display) the data in the chart, a **rendering layer** must be added using the **plot configuration button** (located at the top right corner) – *instructions for creating a chart is continued in the Add Rendering Layer section.*
-

Main Chart Options – Center (Panel) Configuration Button

Edit Underlying Datamodel	Make changes to the data model present in the chart
View Underlying Data	View the data that is being used for the chart(s)
Add Underlying Datamodel	Add a data model to the chart
Unlink Underlying Datamodel	Remove the datamodel as the chart's datasource; datamodels with rendering layers may not be removed
Settings	
Style	Add a common style to the axes and plot(s)
Edit Panel ID	Change the panel ID. Default: Chart
Add Relationship	Add a relationship to the data model(s) being used in the chart. A relationship is required in order to display data on a chart.
Edit Relationship	Edit the relationship to the data model(s)
Remove Relationship	Remove relationship to the data model(s)
Recreate Table / Display	Select what kind of table/display the window will be used for
Add Panel	Add a blank panel to the left, right, above, or below the current chart
Place Highlighted in Tab	Places the chart panel into a tab
Cut Highlighted Panels	Cut the chart panel to paste into another panel in the same dashboard
Copy Highlighted Panels	Copy the chart panel to paste into another panel in the same dashboard
Delete Highlighted Panels	Delete the chart panel
Export Highlighted Panels	Export the highlighted panel, as text, for use in another dashboard
Paste	Paste another panel to the left, right, above, or below the chart

Note: options in bold are unique to chart panels

Edit Underlying Datamodel

- Once a data model is in use for a chart, changes can always be made to the underlying data model. Changes to the data model may be made directly through the datamodel platform [Data > Datamodels...]

View Underlying Datamodel

- Use this option to view the actual data flowing into the chart.

Add Underlying Datamodel

- Multiple data models can be added to a single chart. If multiple data models are to be displayed on a single plot, multiple rendering layers must be used. Please refer to the **Adding Rendering Layer** section for further details on using rendering layers.

Unlink Underlying Datamodel

- In order to unlink a data model that is no longer required for the chart, any rendering layers dependent on the data model must be deleted first. Unlinking a datamodel will not delete the data model from the data model platform.

Edit Global Style

- In order to change the overall style of the chart, select **Style** from the main chart menu.
 - **Background Color:** assign a uniform color to the plots and axes used in the chart
 - **Text Color & Tick Color:** assign a uniform color to be used for the texts and tick lines in the axes

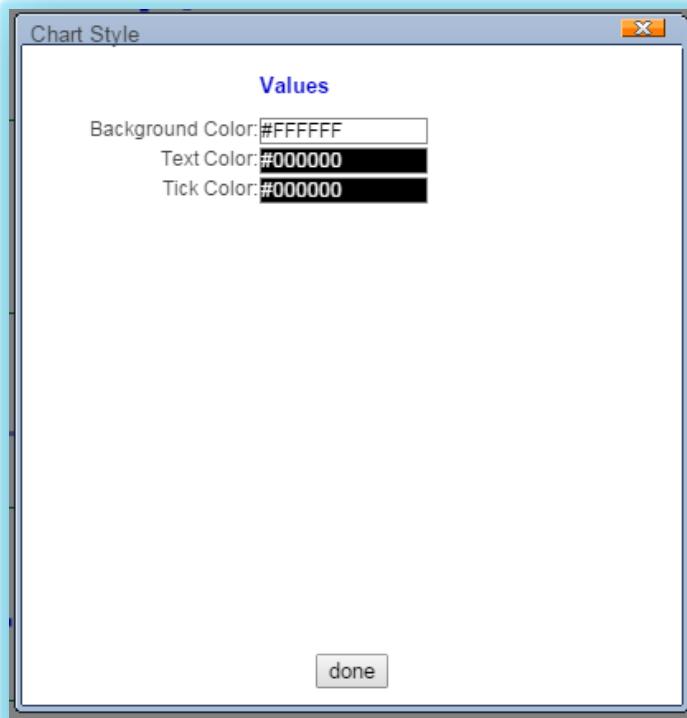
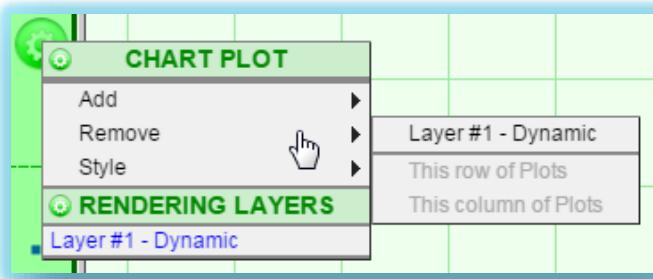
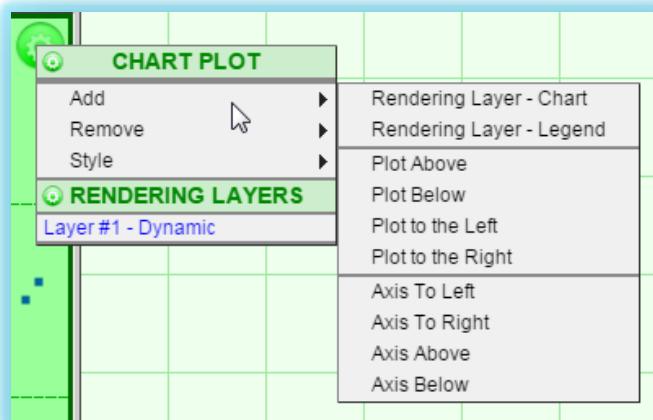


Chart Plot Options – Plot Configuration Button



Add ►

Rendering Layer – Chart	Adds a rendering layer which displays a chart using data from the selected data model
Rendering Layer – Legend	Adds a rendering layer which displays a key based on the data that is being used for the chart
Plot Above, Below, to the Left, to the Right	Add additional plots to the current chart
Axis to Left, to Right, Above, Below	Add additional axes to the current chart (the option to add additional axes is also available in the axes configuration buttons)

Remove ►

Layer	Removes the selected layer from the chart
This row of Plots	Removes the entire row of plots
This column of Plots	Removes the entire column of plots

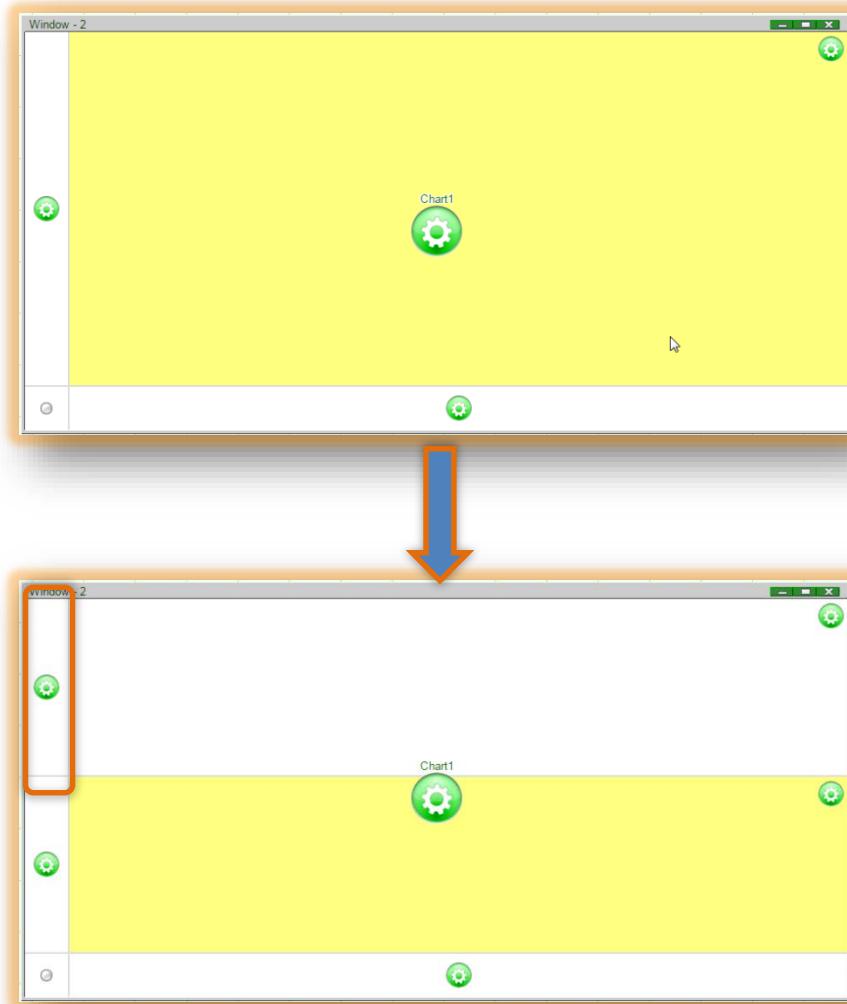
Style	Option to choose a background color for the plot
-------	--

Adding & removing plots

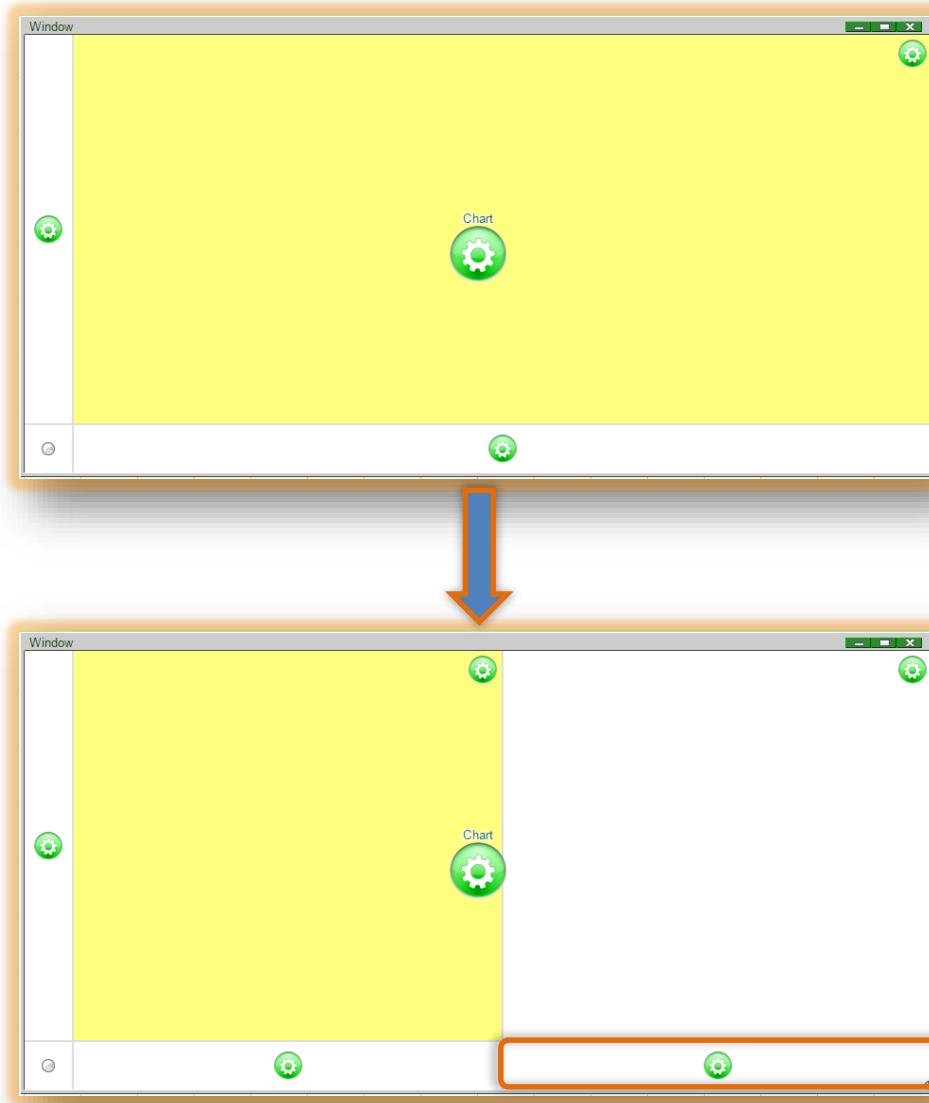
- Charts can have multiple plots and axes. Once a chart has been created, additional plots can always be added to existing plots (*axes will be discussed in the following section*)
- To add additional plots to a chart, click on the **plot configuration button** and select one of the following options under **Add ► - Plot Above, Plot Below, Plot to the Left, Plot to the Right**
 - Axes will be added automatically to the new plots

Example

Adding a plot above will automatically add an axis to the left of the new plot (the two plots will share the same bottom axis)



Adding a plot to the right will automatically add an axis to the bottom of the new plot (the two plots will share the same left axis)

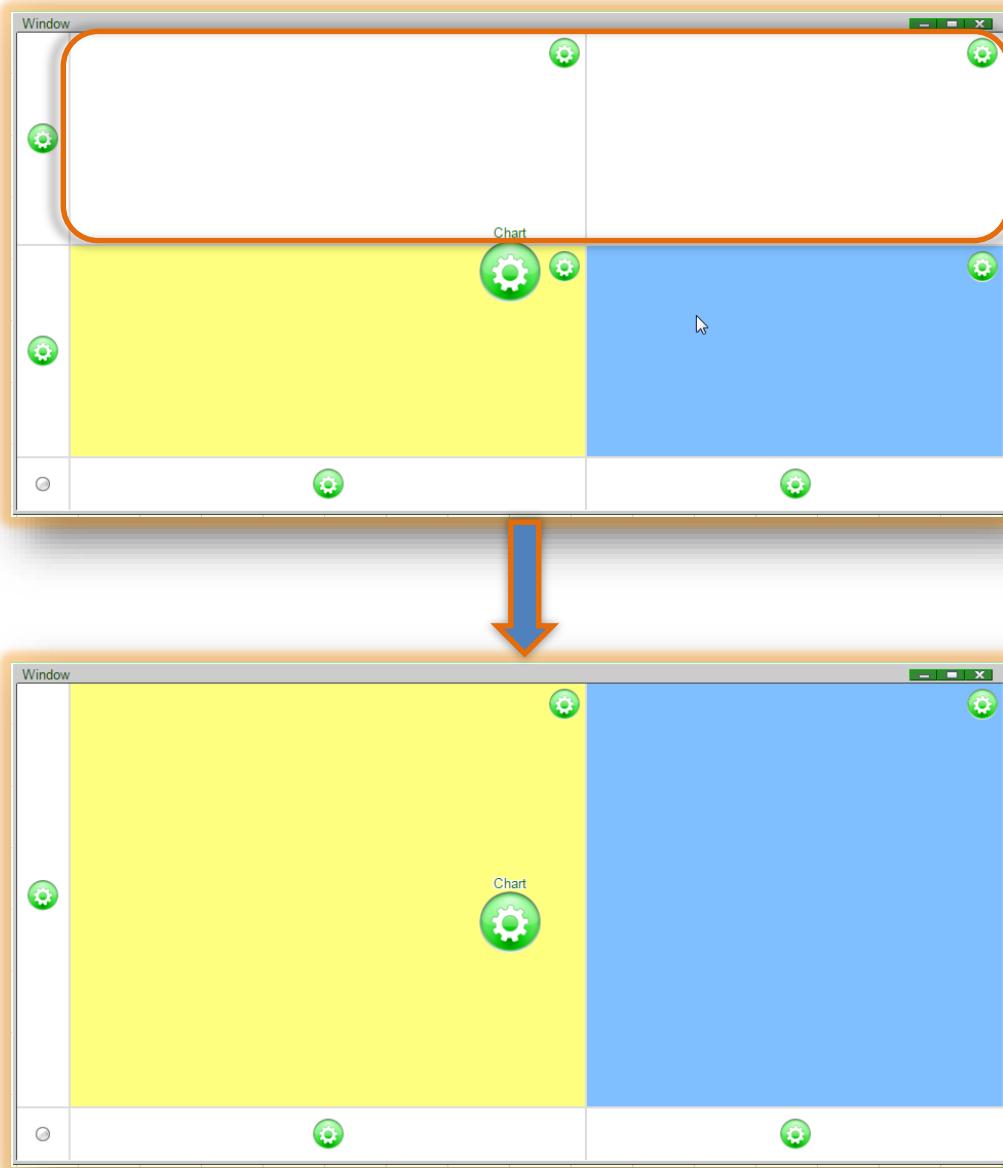


- The size of each plot can be modified by adjusting the lines dividing the plots

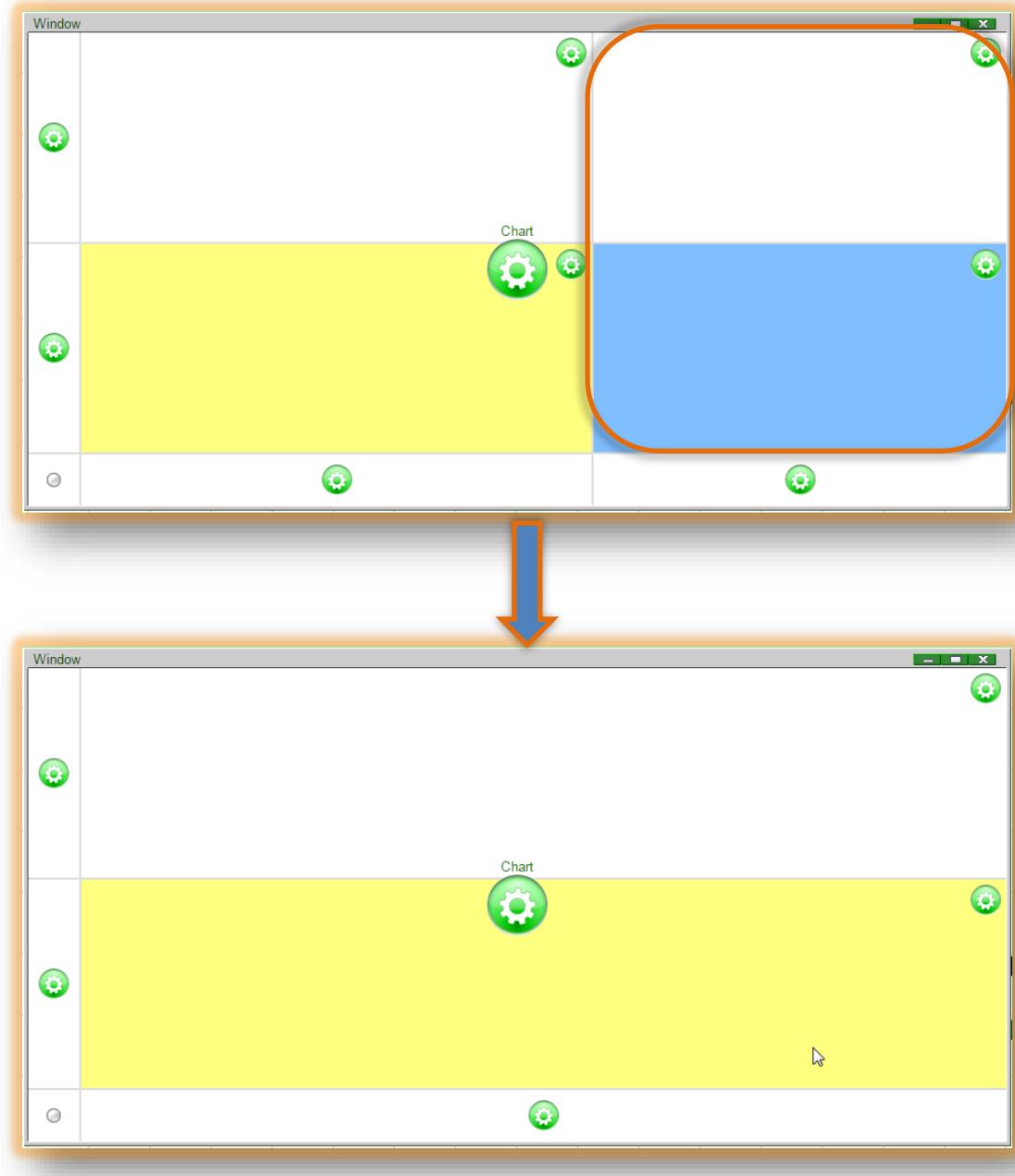
- In order to remove plots, select the following options under Remove ► - This row of Plots, This column of Plots.
 - When there are more than two plots on a chart, entire rows or columns of plots will be removed.

Example

Removing a **row** of plots:



Removing a **column** of plots:

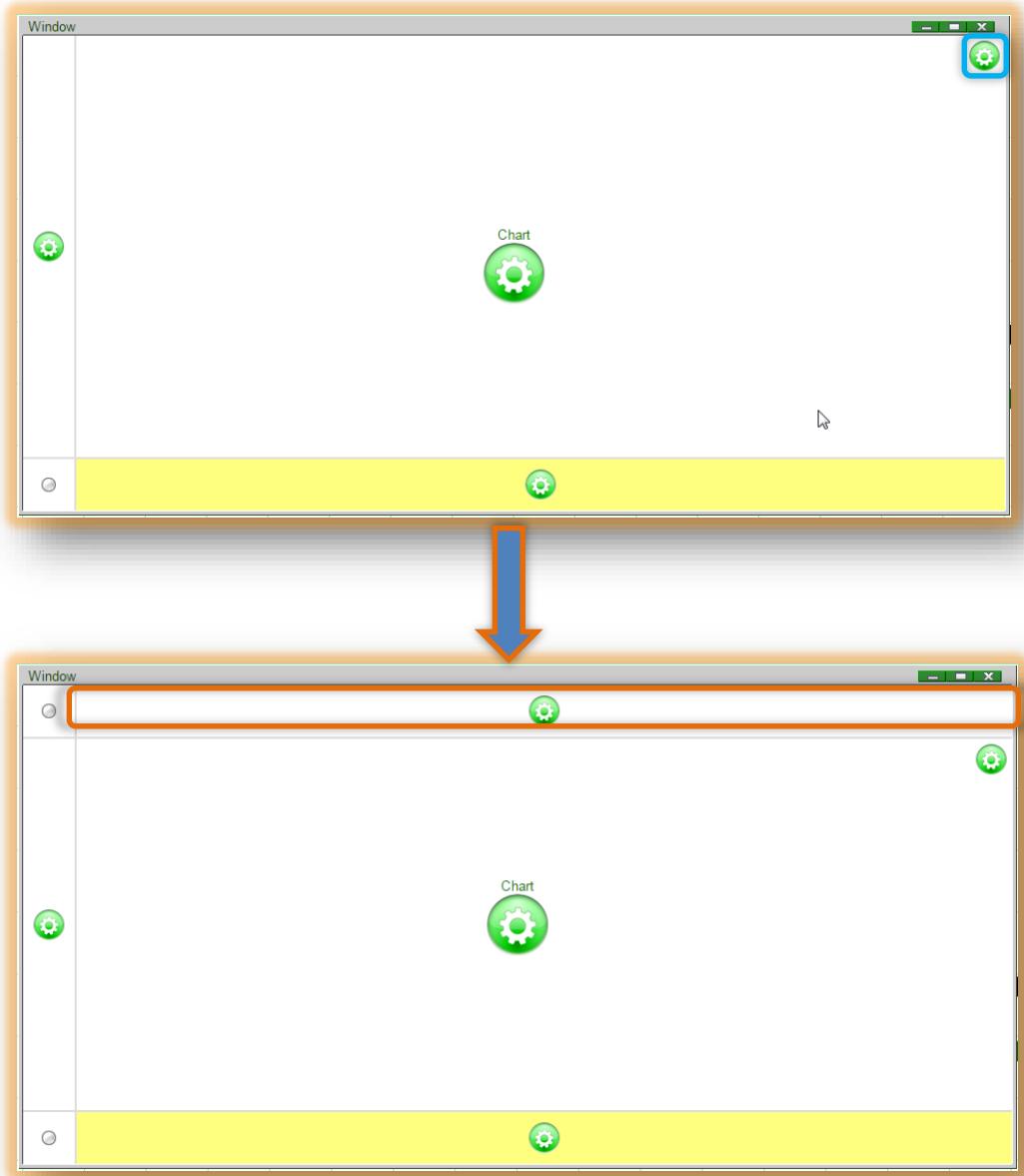


Adding axes

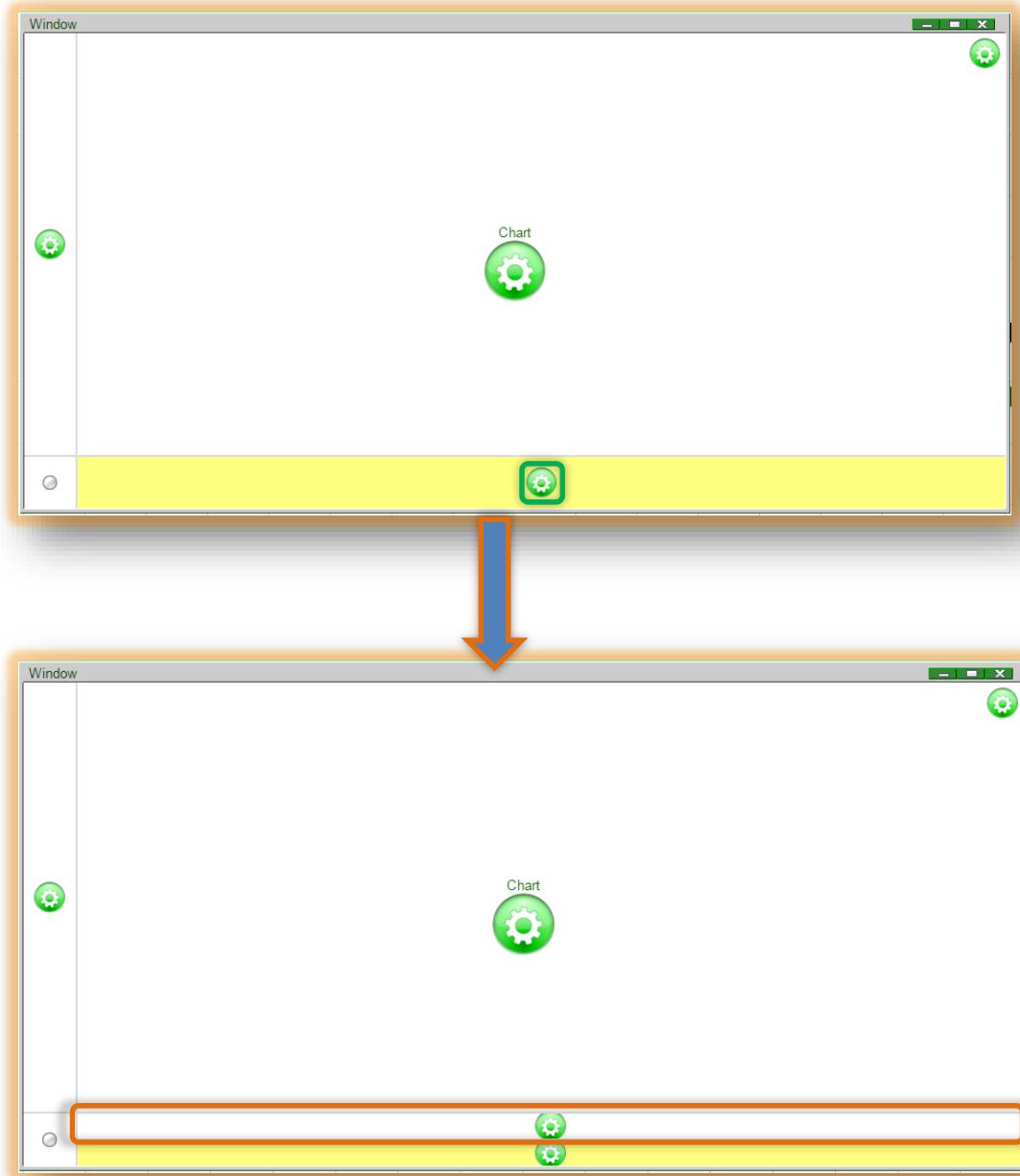
- In addition to adding axes using the **plot configuration button**, axes can be added directly to existing axes using the **axis configuration button**
 - New axes **must** be added using the **plot configuration button**

Example – adding an axis above using the **plot configuration button** vs. adding an axis above an existing axis

Adding an axis above using the **plot configuration button** will add an axis above the plot:



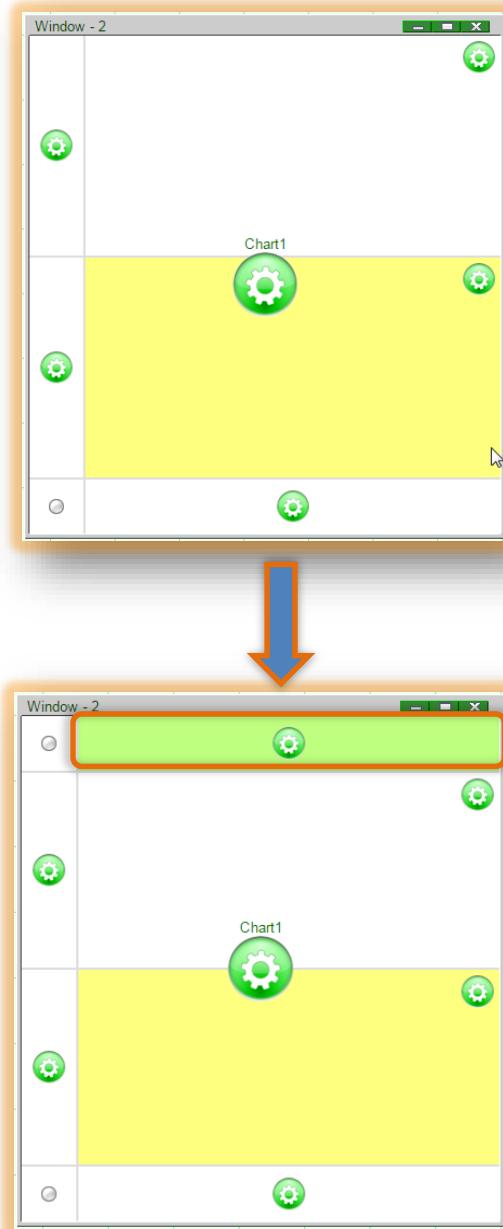
Adding an axis above using the **axis configuration button** will add an axis above the selected axis:



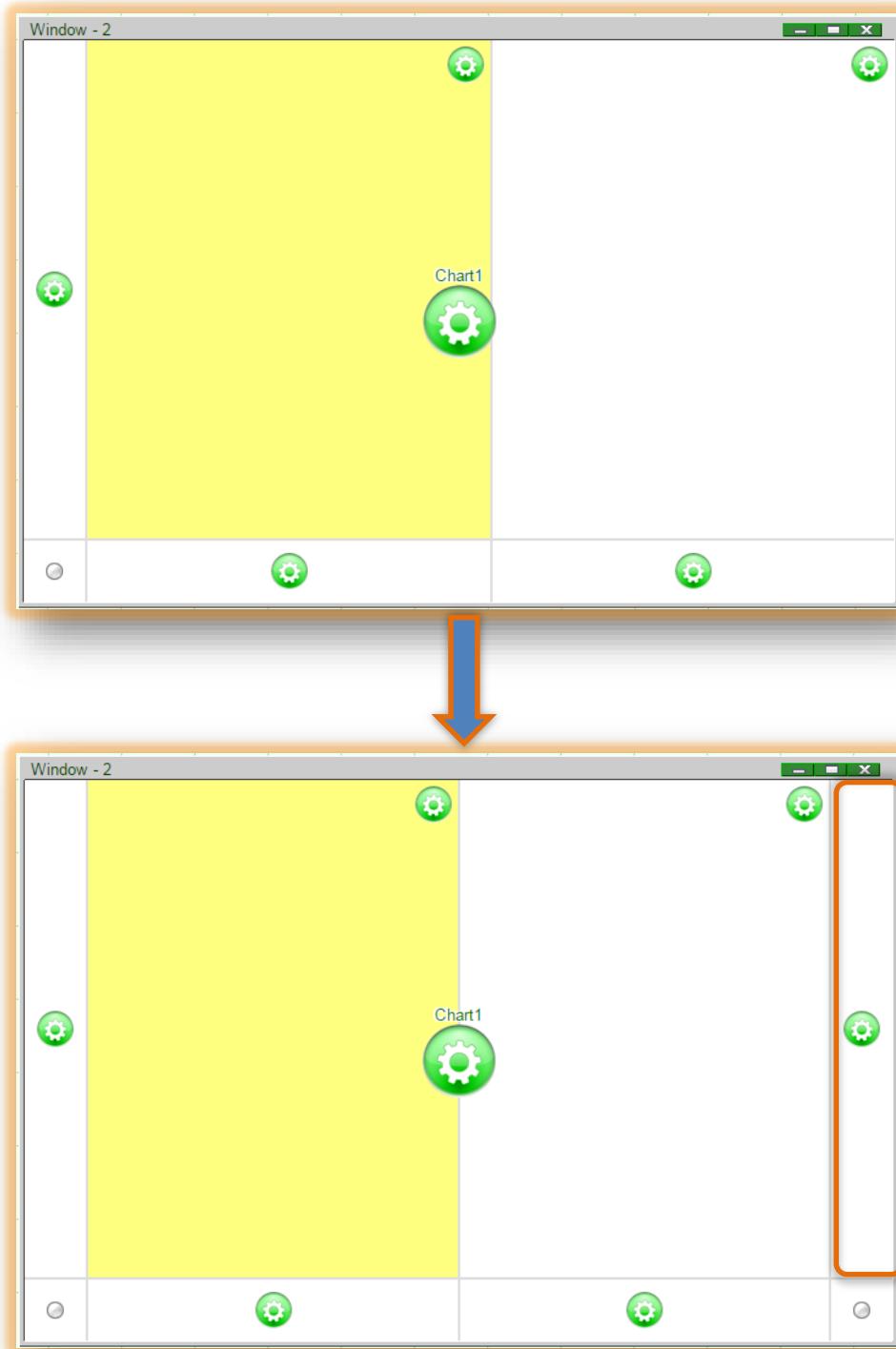
Example – adding axes to charts with multiple plots

- When there are multiple plots used in a chart, new axes are added to the outside edges of the chart

Choosing to add an axis above on any of the plots in a column will add an axis to the very top of the chart

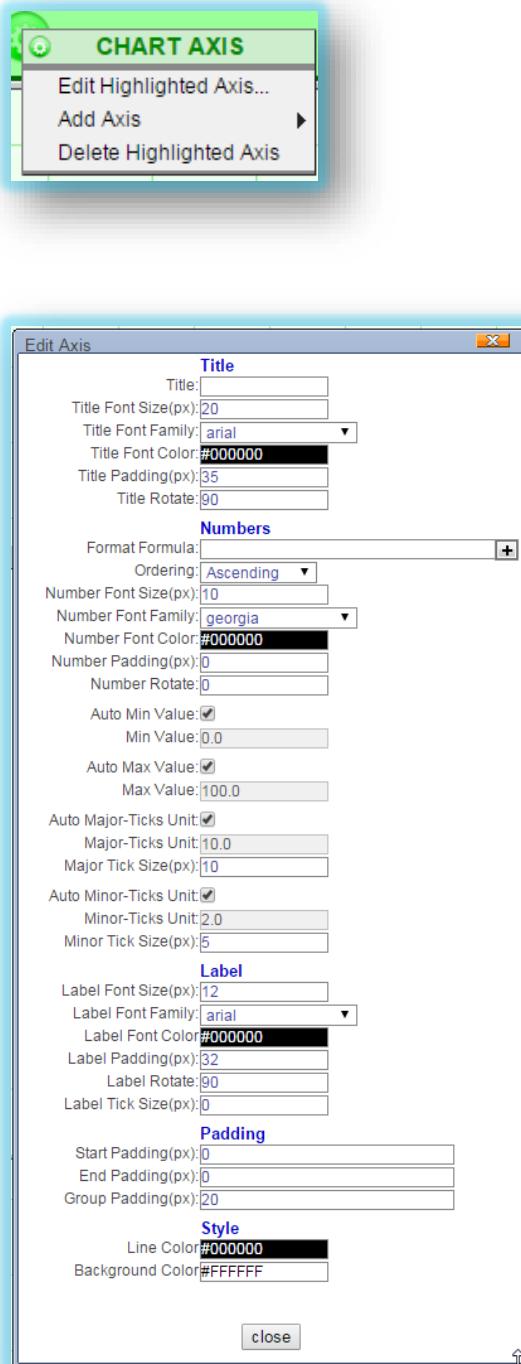


Choosing to add an axis to the right of any of the plots in a row will add an axis to the right edge of the chart



Editing & deleting axes

- Each axis in the chart is highly customizable. The option to edit or delete an axis is available through its **axis configuration button**
- The size of each axis can be modified by adjusting the line of the axis



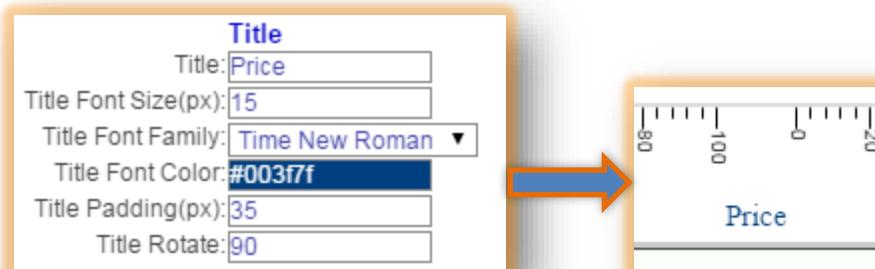
Title	
Title	Choose a title for the axis
Title Font Size (px*)	Select the size of the title
Title Font Family	Select a font for the title
Title Font Color	Choose a color for the title
Title Padding (px)	Modify the spacing between the title and the axis line
Title Rotate	Rotate the title
Numbers	
Format Formula	Modify the formatting of the values used in the axis. Operators and functions may be used to make the formatting highly specific
Ordering	Change the ordering of values – either ascending or descending
Number Font Size (px)	Adjust the font size of the axis values
Number Font Family	Assign a font to the axis values
Number Font Color	Adjust the color of the axis values
Number Padding (px)	Adjust the padding between the values of the axis and the axis line
Number Rotate	Rotate the axis values
Auto Min Value & Min Value	Minimum values to be used for the axis. Auto when checked & manual when unchecked
Auto Max value & Max Value	Maximum values to be used for the axis. Auto when checked & manual when unchecked.
Auto Major-Ticks Unit & Major-Ticks Unit	Spacing between each major tick mark. Auto when checked & manual when unchecked
Major Tick Size (px)	Assign major tick size
Auto Minor-Ticks Unit & Minor-Ticks Unit	Spacing between each minor tick mark. Auto when checked & manual when unchecked
Minor Tick Size (px)	Assign minor tick size
Label	
Label Font Size (px)	Adjust the font size of the grouping values
Label Font Family	Assign a font to the grouping values
Label Font Color	Adjust the color of the grouping values
Label Padding (px)	Adjust the padding between the grouping values and the axis line
Label Rotate	Rotate the grouping values
Label Tick Size (px)	Adjust the grouping tick size
Padding	
Starting Padding (px)	Adjust the padding at the start of the axis
End Padding (px)	Adjust the padding at the end of the axis
Text Padding (px)	Adjust the padding between the values of the groupings and the axis line
Style	
Line Color	Adjust the color of the axis lines

Background Color
 *Note: px – pixels

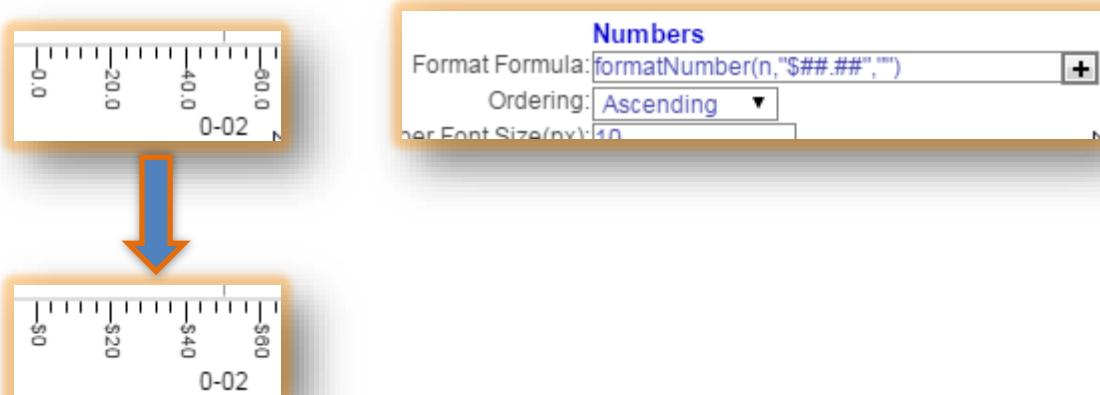
Assign a background color to the axis

Examples

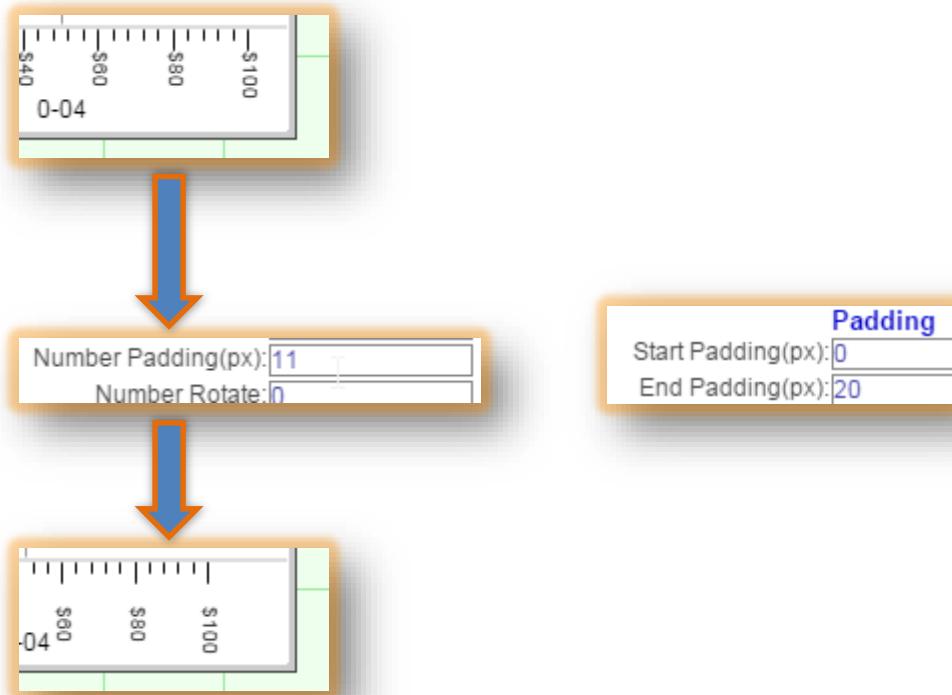
Assigning a title to the axis



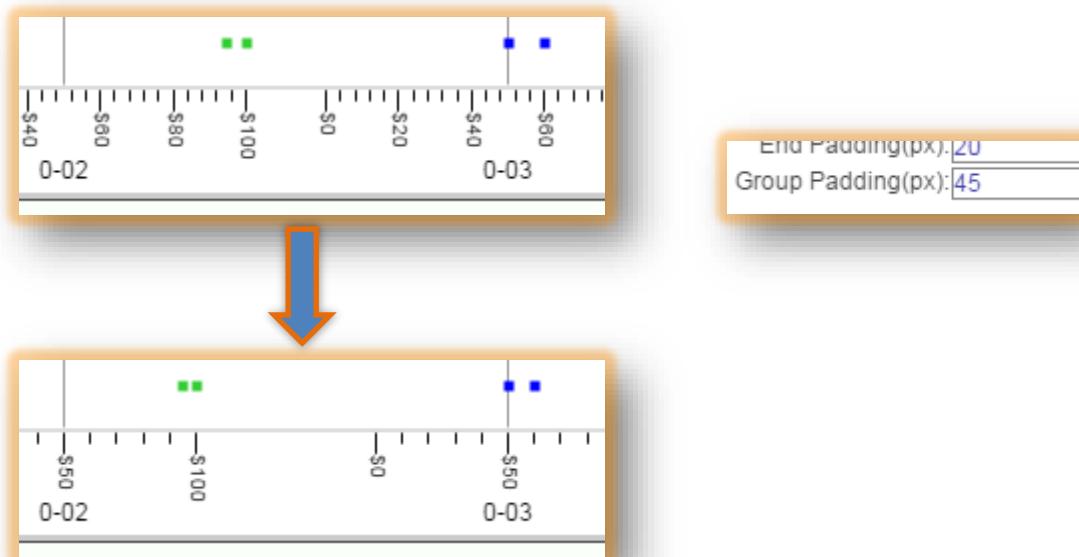
Modifying the values of the axis – *display price with a \$ sign and proper decimal places using the **formatNumber** function*



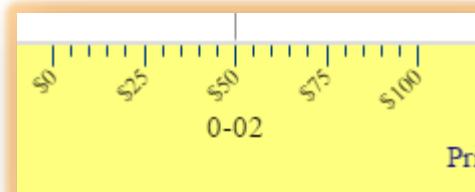
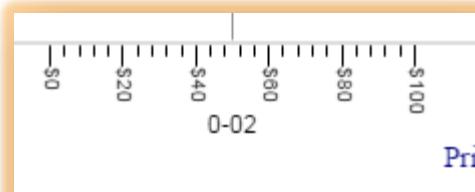
Adjusting the number padding and end padding of an axis



Adjusting the group padding – *between orders group 0-02 & 0-03*

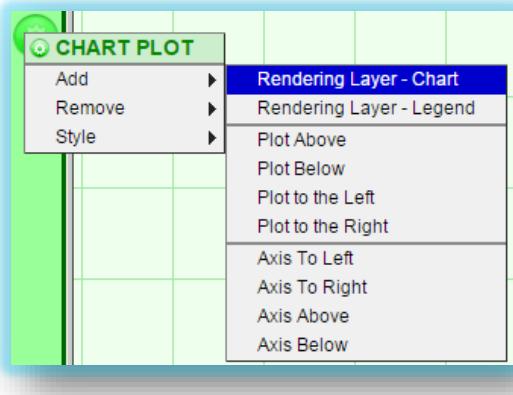


Adjusting the style of the axis

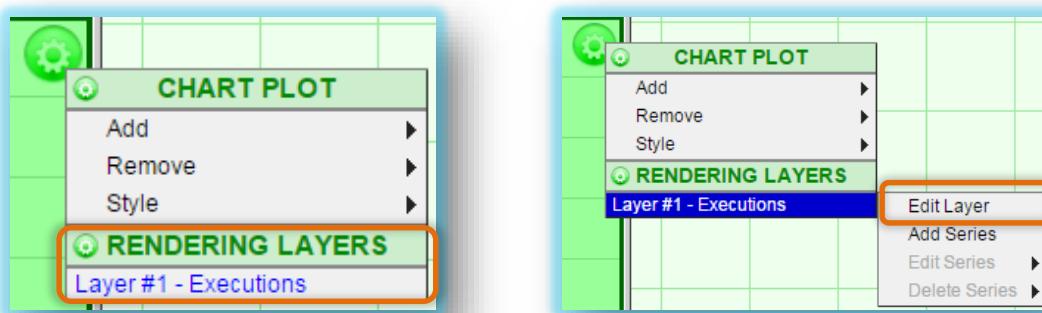


Add Rendering Layer – Chart

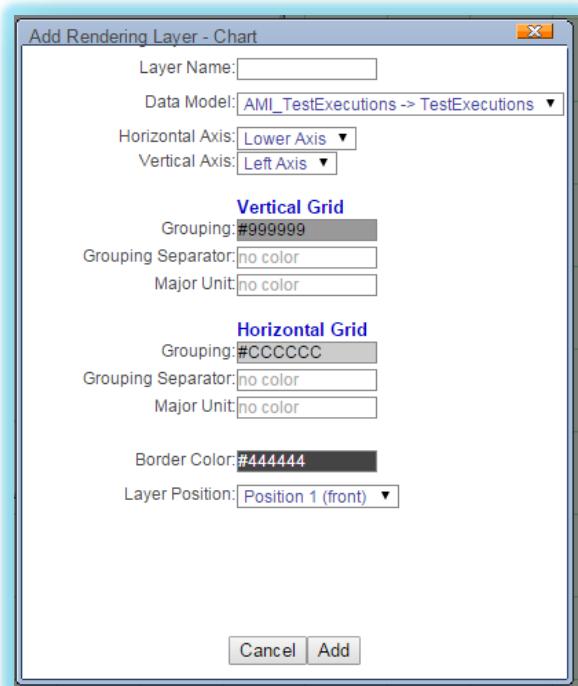
- In order to render (display) the data model in the chart, a rendering layer must be added to the plot. To add a rendering layer, click on the **plot configuration button** (located at the top right corner) and select **Rendering Layer – Chart** from the **Add** menu



- Rendering layers allow the display of multiple charts on a single plot using the same or multiple data sets:
 - If the data sets are to use the same axes, use a **single** rendering layer with **multiple** series
 - If the data sets are to use **different** axes, use **multiple** rendering layers
 - If two or more data models are used, you **must** use **multiple** rendering layers
- Once the rendering layer has been added, it will appear in the **Rendering Layers** list – found under the **Chart Plot** menu (**plot configuration button**).



- The rendering layer can always be edited by selecting **Edit Layer** from the layer menu

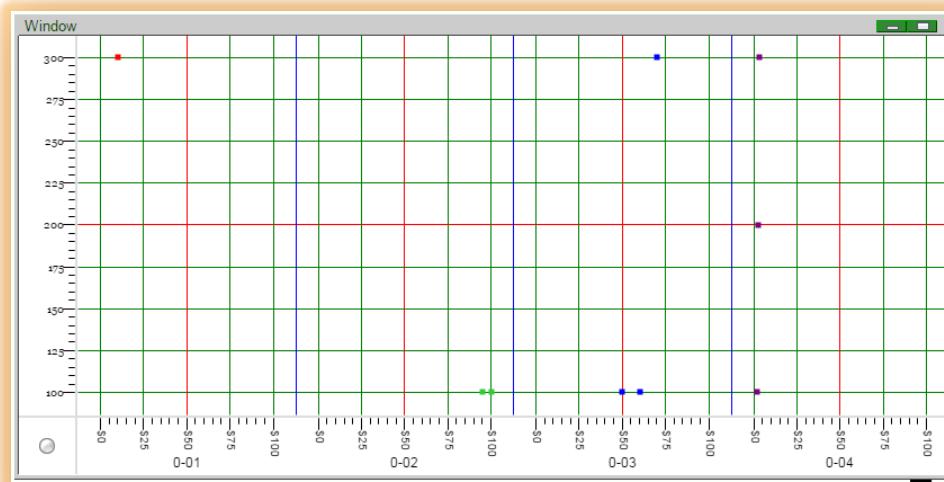
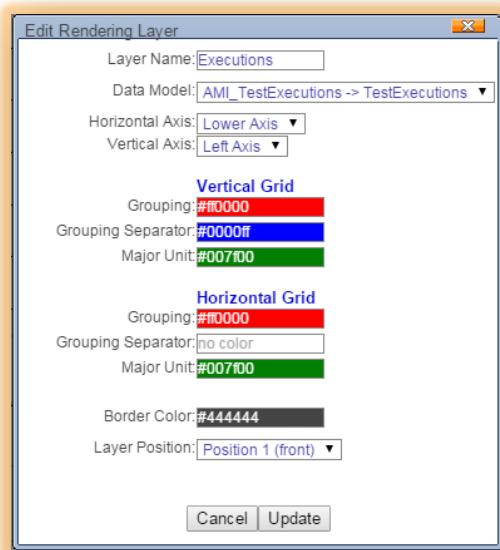


Note: The **Add** Rendering Layer menu and **Edit** Rendering Layer menu are the same

Layer Name	Assign a name to the layout
Data Model	Choose the data model that is to be displayed in the chart
Horizontal Axis	Choose the axis to be used for the horizontal (X) axis
Vertical Axis	Choose the axis to be used for the vertical (Y) axis
Vertical Grid	
Grouping	Select a color to be used for the group line. When X groupings is used, this line arises from the center of each group; otherwise, this line arises from the center of the axis
Grouping Separator	Select a color for the lines separating groups. <u>Note:</u> Only applies when X groupings is used in a series
Major Unit	Select a color to be used for the vertical grid lines. Select 'No color' to remove vertical grid lines
Horizontal Grid	
Grouping	Select a color to be used for the group line. When Y groupings is used, this line arises from the center of each group; otherwise, this line arises from the center of the axis
Grouping Separator	Select a color for the lines separating groups. <u>Note:</u> Only applies when Y groupings is used in a series

Major Unit	Select a color to be used for the horizontal gridlines. Select ‘No color’ to remove horizontal grid lines
Border Color	Select a color to be used for the outer border. Select ‘No color’ to remove border lines
Layer Position	When using more than 1 rendering layer, assign the position for each layer. Selecting Position 1 for the current layer will reassign the position for the remaining layers. Position 1 is the position in the very front.

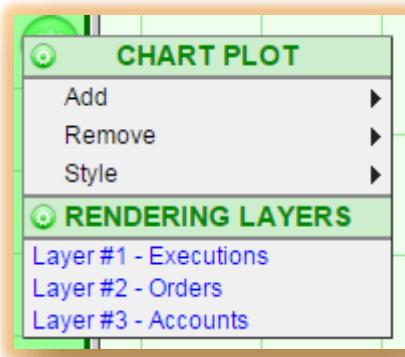
Note: Select **no color** for each field in order to remove all grid lines



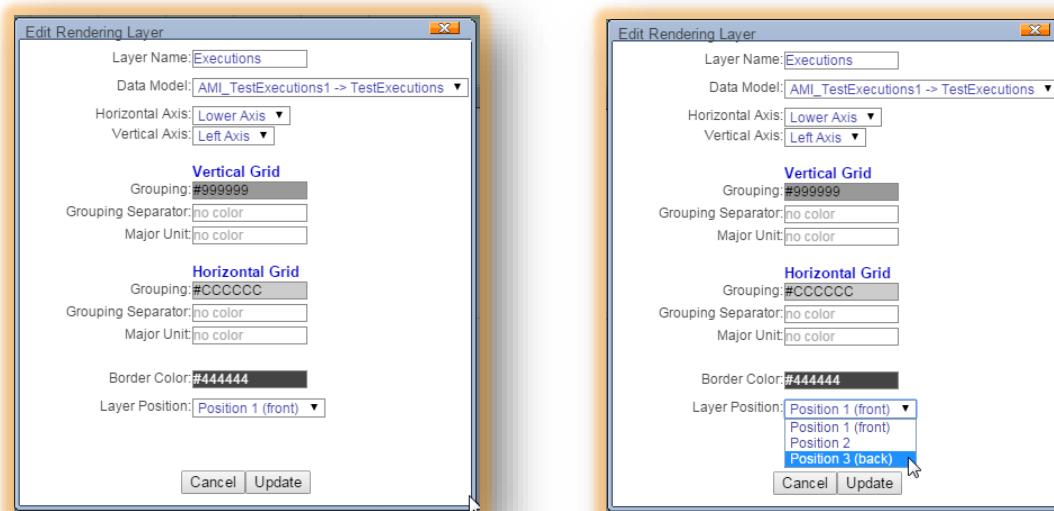
Example – changing the ordering of the layers

When working with multiple layers on a plot, it may be necessary to re-order the rendering layers in order to figure out what works best for the chart. This is because the chart representations of certain data may be easier to interpret if it is positioned in front of or behind another set of data. (Position 1 is the layer closest to the user).

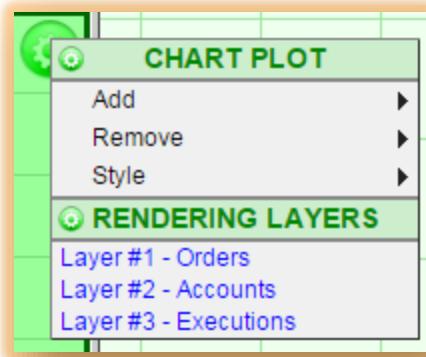
In the example below, the layers are initially in the following order (from first to last): *Executions* → *Orders* → *Accounts*. For the example, the layers will be reordered as follows: *Accounts* → *Orders* → *Executions*. This can be done by changing the position of only two (*Executions* and *Accounts*) of the three layers.



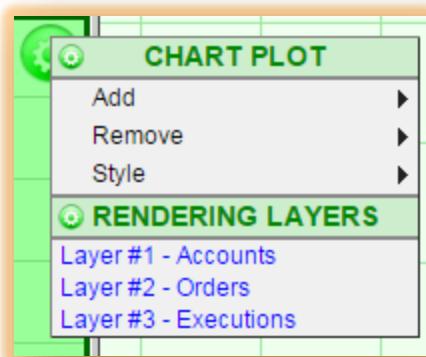
In order to change the position of *Executions* and *Accounts*, select **Edit Layer** for *Executions*. In the **Edit Rendering Layer** window, modify the **Layer Position** by selecting Position 3 (back) from the drop down menu and clicking Update.



The repositioning of the *Executions* layer reassigns positions to the other layers. The *Orders* layer has been moved up to position 1 from positon 2 and the *Accounts* layer has been moved up to positon 2 from positon 3.



Repeat the steps for the *Accounts* layer and select Positon 1 (front) from the drop down menu. All of the layers are now in the correct order:

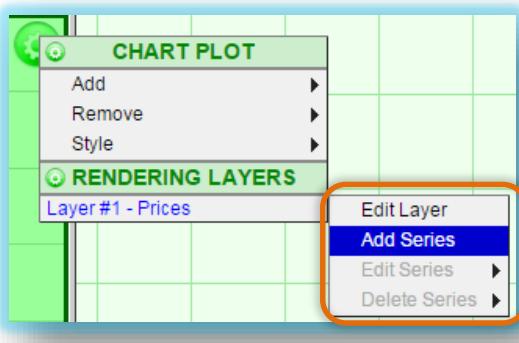


Adding Series

- Once a data model has been added and rendering layers established, the final step to creating a basic chart is to add the series.
 - Just as how multiple rendering layers can be applied to a single plot, multiple series can be applied to a single rendering layer
- 1) To add a series to a rendering layer, click on the **plot configuration button** to open the **Chart Plot** menu.



- 2) In the Rendering Layers list, locate the layer where the series will be added to and select **Add Series** from the menu. This will bring up the window to edit the series.
 - *The option to edit or delete a series will be available once a series has been added.*



Add Series To Rendering Layer

Options

Series name:
Where: [+]
Layer position: Position 1 (front) [+]

Axis

X: [+]
Y: [+]
X groupings: [+]
Y groupings: [+]

Labels

User selectable: [+]
Description: [+]
Hover over: [+]

Markers

Shape: [+]
Color: [+]
Width(px): [+]
Height(px): [+]
Border color: [+]
Border thickness: [+]

Marker Position Override

Top: [+]
Bottom: [+]
Left: [+]
Right: [+]

Lines

Line color: [+]
Line thickness: [+]

Partitioning (For Lines and Aggregates)

Partition By: [+]
Order Partition By: [+]

Area

X2: [+]
Y2: [+]
Line2 color: [+]
Line2 thickness: [+]
Connection color: [+]
Connection thickness: [+]
Fill Color: [+]

Cancel Add Preview

Name	
Series name	Assign a name to the series
Where	Use to apply a filter to the data displayed in the chart
Layer position	Select the position of the series within the layer
Axis	
X	Assign a variable to be used in the X axis
Y	Assign a variable to be used in the Y axis
X groupings	Apply a grouping of the X variables on the axis
Y groupings	Apply a grouping of the Y variables on the axis
Labels	
User selectable	Makes the markers selectable; must evaluate to Boolean value.
Description	Apply a right-click description to the markers (used in conjunction with selection)
Hover over	Apply a description box which appears with mouse hover over (used in conjunction with selection). Supports HTML
Markers	
Shape	The shape to be used for the markers – circle, square, and triangle
Color	The color of the markers
Width (px) & Height (px)	The width and height of the markers in pixels
Border color	Applies a color to the border
Border thickness	Applies a border to the markers
Marker Position Override	
Top	Applies a top limit to the marker
Bottom	Applies a bottom limit to the marker. To create bar graphs, set the marker bottom to 0
Left	The extent of the marker to the left
Right	The extent of the marker to the right
Lines	
Line color	Select a color for the line
Line thickness	Determine the size of the line connecting the markers (in pixels)
Partitioning	
Partition By	Organize the variables into divisions
Order Partition By	Apply a ordering of the variables within the divisions
Area	
X2	Assign a X axis variable for the 2 nd line ¹
Y2	Assign a Y axis variable for the 2 nd line
Line2 color	Select a color for the 2 nd line

Line2 thickness	Determine the size of the line 2 nd line
Connection color ²	Select a color for the lines connecting the areas
Connection thickness	Determine the size of the connection lines
Fill Color	Assign a color for the area
¹ : An area is created between the first and second lines.	
² : Areas are created for each space between two markers	

Notes:

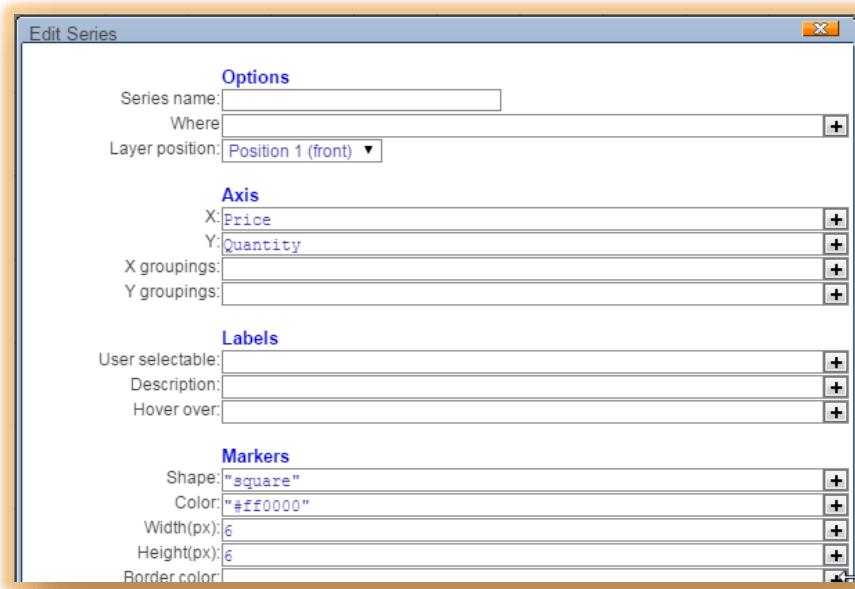
- Strings cannot be assigned to X or Y values. To use a string for the axis, use the [X/Y groupings](#) instead.
- **Marker Shape** and **Color** are the minimum requirements for displaying makers. Either **Marker Width** and **Marker Height** OR **Marker Left, Right, Top** and **Bottom** are also required (left, right, top and bottom take precedence of width & height)
- **Line color** and **Line thickness** are required to display lines
- Click on the **Preview** button in order to see how the chart will look. Click off the wizard (this icon will appear) in order to turn off the shadow, making it easier to see the true colors being used

Example – building of a basic chart including the addition of series; highlighting the use of special variables for charts and the formatNumber and cycle functions

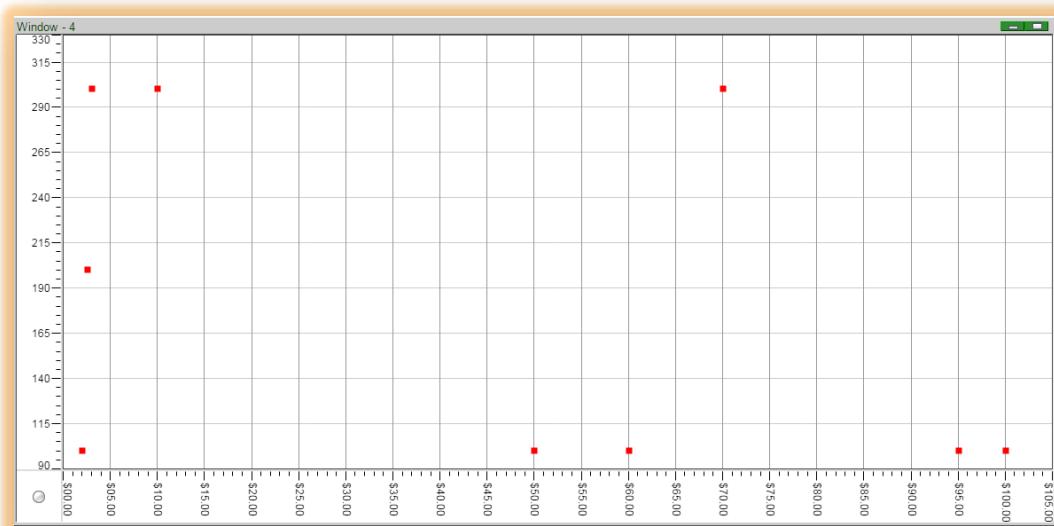
For this example, the following table has been added as the data model for the chart:

Window				
9 Rows				
Execution	Order	Price	Quantity	
E-01	0-01	10	300	
E-02	0-02	100	100	
E-03	0-03	70	300	
E-04	0-02	95	100	
E-05	0-03	60	100	
E-06	0-03	50	100	
E-07	0-04	3	300	
E-08	0-04	2	200	
E-09	0-04	2	100	

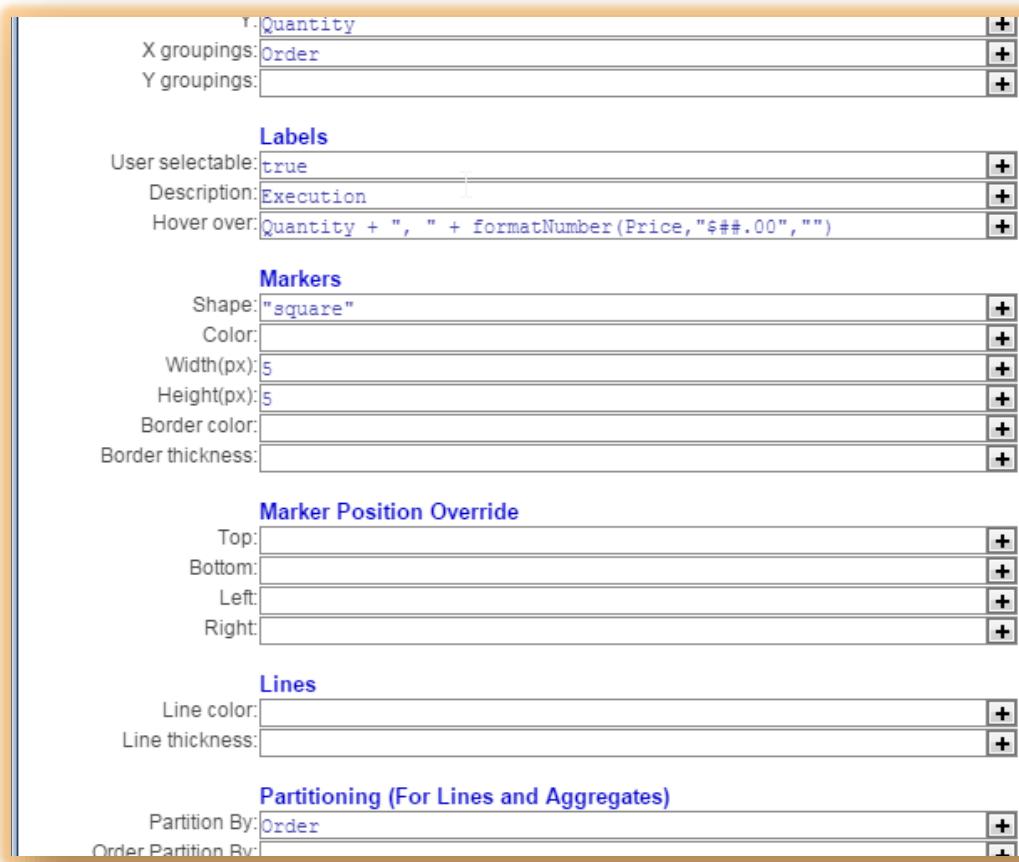
It is a basic data model of *Executions* showing their *Order* numbers, *Price*, and *Quantity*. After applying a single rendering layer, a series with the following minimum requirements is added to the rendering layer:



Applying only the minimum requirements for a series results in the following chart:



In order to further modify the chart in order to better represent the data model, it is necessary to modify the series by adding to several fields:



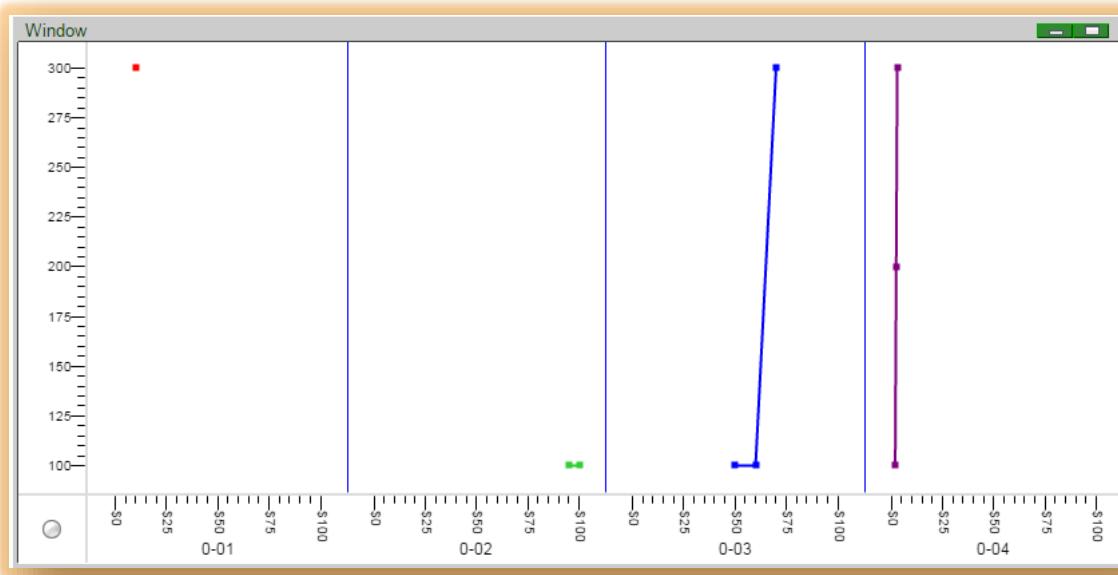
- Applying **X groupings** will divide the axis based on the variable used. In this example, the X-axis will be divided using the **Order** numbers.
- Applying a **Partition By: Order** will organize the *executions* under their respective *order* numbers. This will create **4** different series since there are **4** different orders (0-01, 0-02, 0-03, & 0-04) – *essentially creating series within a series*
 - Note: using **X groupings** will place the *executions* according to their respective *orders* but they will remain in the same series unless **Partition By** is used
- Making the markers **Selectable** will enable **Descriptions** and **Hover Over** – in this example, the description (right-click) will show the *execution* number and the hover over will show the *quantity* and *price*.
 - In order to properly list price with a \$ sign, it is necessary to use the **formatNumber** function as seen above. This function is not used for quantity because it is already in its proper form.

In order to further distinguish the *executions*, markers can be modified and lines can be added:

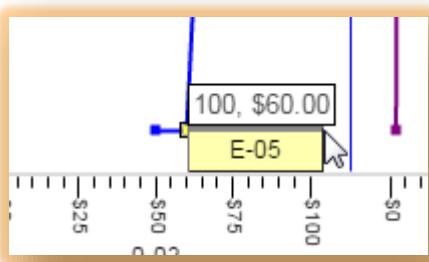


- The **cycle** function is used in Markers Color. This function, used in conjunction with the special variable **__series_num** (Series Number), allows the cycling through of the established series and applies the chosen colors to all of the available series.
 - In this example, since there are **4** different series for the **4** different orders, **4** different colors are used.
 - The other special variable available in charts is **__row_num** (Row Number). This variable used in conjunction with the cycle function in marker color will alternate the chosen colors for each row of the data table.
 - **Note:** a **Partition By** must be used in conjunction with these special variables
- A line is added and the same statement used in Markers Color is applied to **Line color** in order to keep the colors the same.

Applying all of the changes to the options results in the following chart:



Since **selection**, **description**, and **hover over** are enabled, selecting any of the markers will show the execution number along with the quantity and price.



The alternative to the method used in this example is to add **4** different series to the rendering layer and using the **where** option for each series:

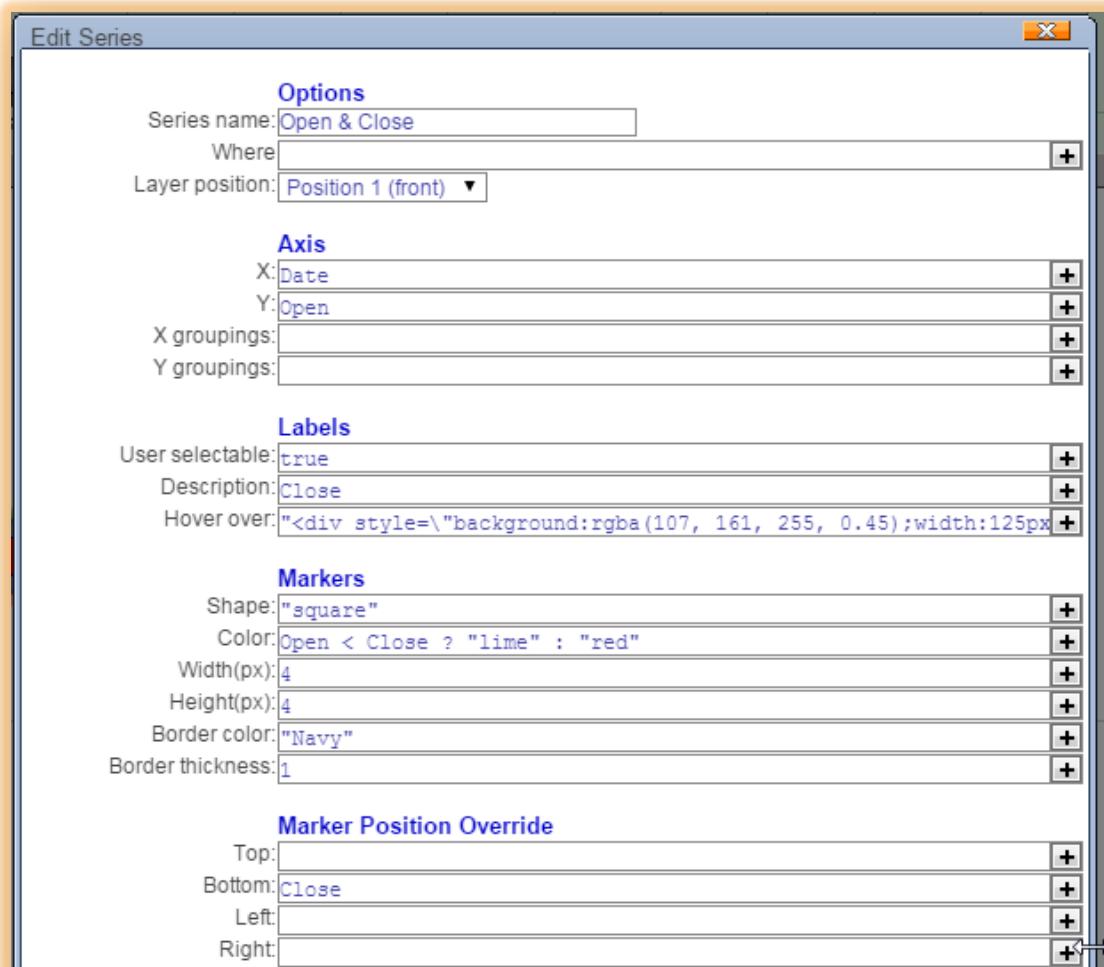
Options	
Series name:	<input type="text"/>
Where	<input 0-01""="" type="text" value="Order == "/>
Layer position:	<input type="button" value="Position 1 (front) ▾"/>

However, using the **Partition By** option as seen in this example allows for the creation of series within a series and makes it unnecessary to establish multiple series for the same data model.

Example – building a candle stick chart using historical prices of a company

For this example, a company's historical quote of 1 year will be used to build a candle stick chart. The historical quote is added as a data model to the chart and an initial rendering is added (named Candlestick).

Two different series will be added to the Candlestick layer – one series will contain the **Open** and **Close** prices whereas the other series will contain the **High** and **Low** prices:

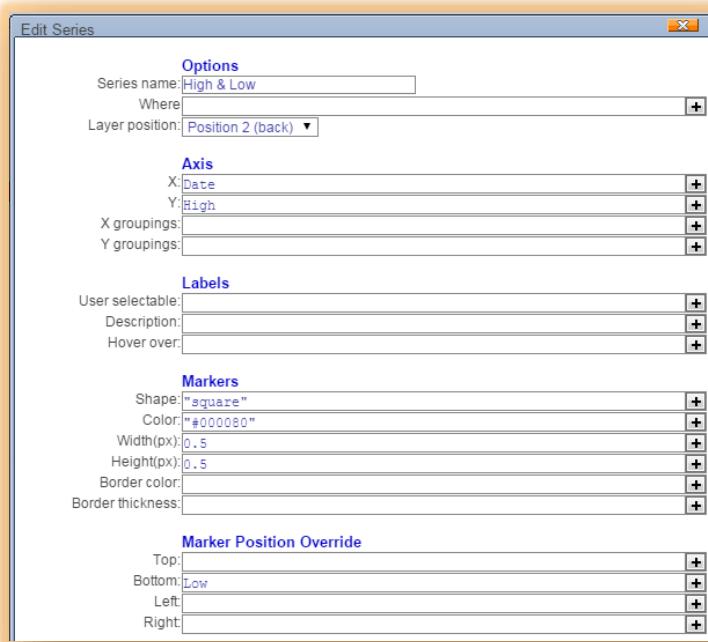


- The use of the [Open < Close ? "Lime" : "Red"] statement in **Markers Color** is a key component of the candlestick chart. This states “if opening price is lower than closing price, the marker will be lime, otherwise it will be red”
 - A lime color marker indicates a **bullish** candlestick whereas a red color maker indicates a **bearish** candlestick

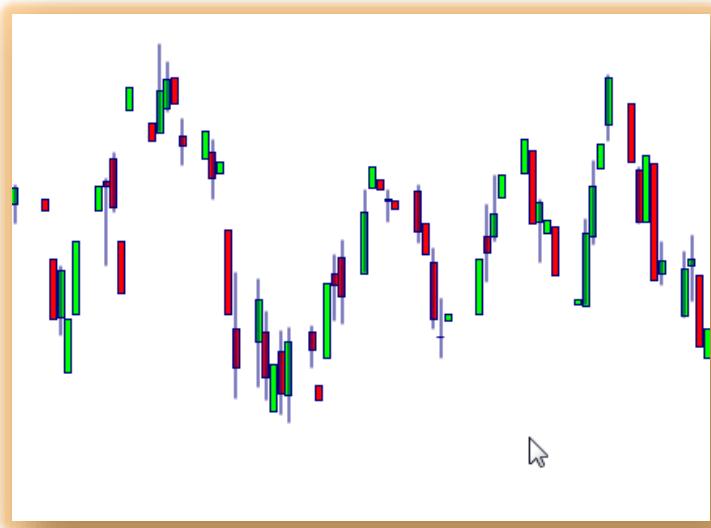
- Also note the use of *opening prices* as the **Y values** and *closing prices* as **Markers**
- Bottom**
- Applying the above series will result in the following chart:



Dates are used for the X axis and *Price* is used for the Y axis. In order to complete the candlestick chart, the high and low prices of each day should be added:



Applying both series to the rendering layer may result in the following chart with the high and low price markers placed in front of the opening and closing price markers:



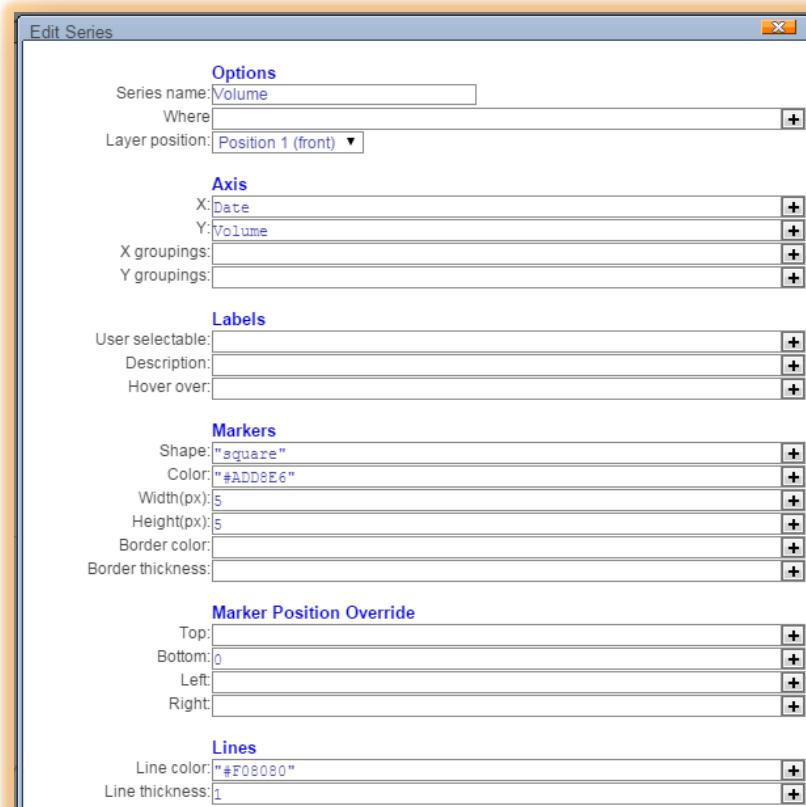
Reordering the positions of the series will return a proper candlestick chart:



With **selection** enabled and **hover over** (HTML) added:



Adding a **Volume** layer to the same plot can further enhance the information displayed in the chart. Prior to adding the volume layer, a right Y axis is added to be used for the Volume.

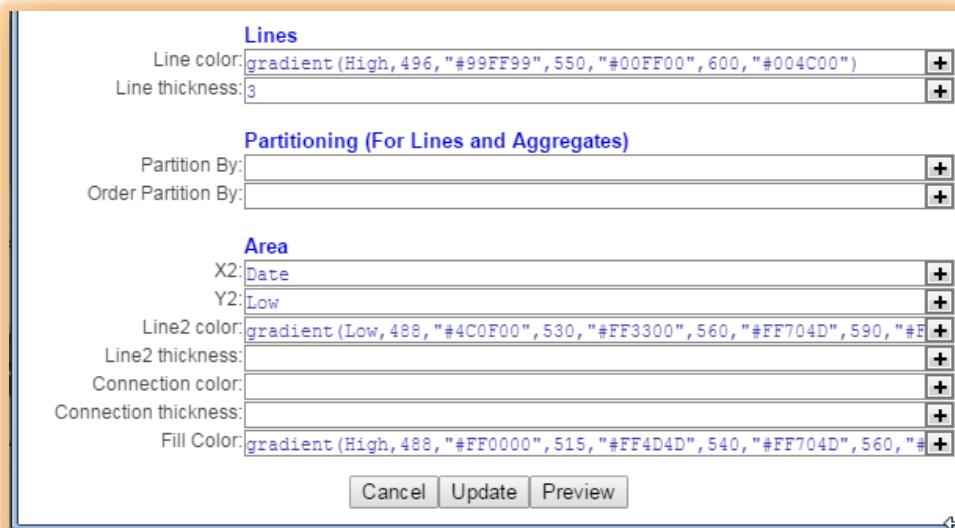


- In order to create a **bar graph**, the Marker Bottom must be set to **0** (extend the bottom of the marker to the X axis)



Example – using the gradient function in conjunction with the **Area** section:

- Areas are established on charts between the 1st and 2nd lines. In this example, the first line is drawn using the *High* markers. A second line is drawn using the *Low* markers. The **Fill Color** will be the color used for the area between the two lines. Using the gradient function will apply a different shade of the colors chosen based on the marker's position in the range specified.
- Note: Please refer to the gradient function section for further details



Transparency of Layers

In charts with multiple layers, it may be necessary at times to change the transparency of certain layers in order to view specific data.

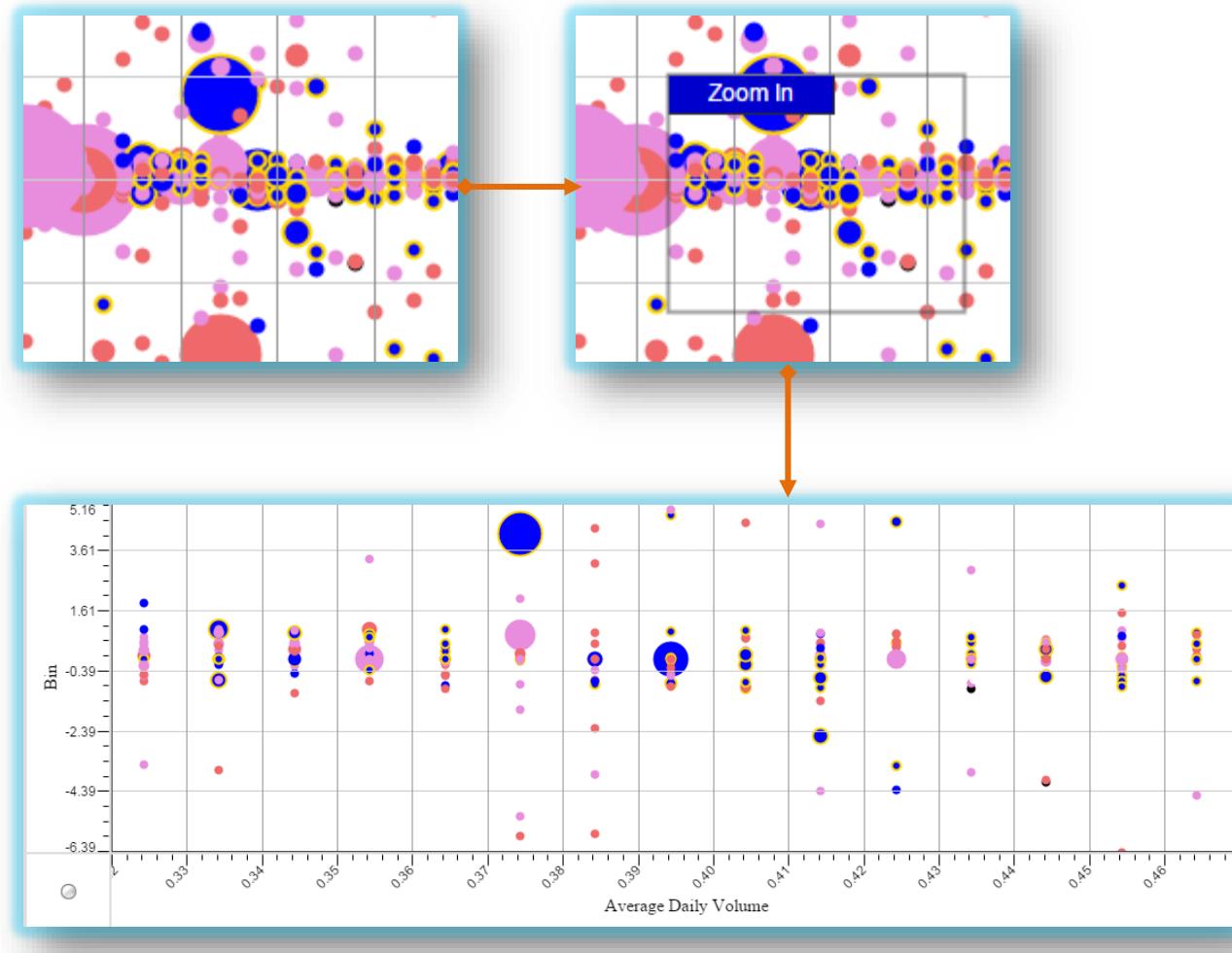
- In order to change the transparency of a layer, click on any of the corner buttons of a chart in order to bring up the **transparency menu**. Use the slider in order to adjust the transparency of the layer.



Zooming

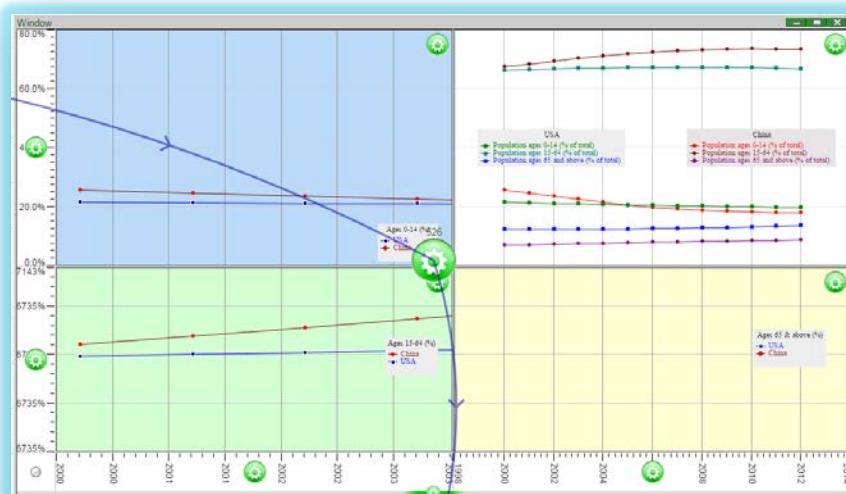
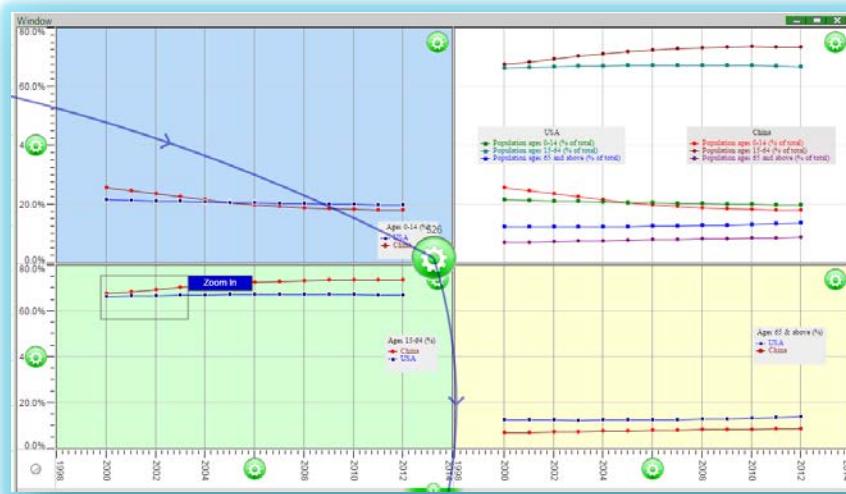
When viewing charts with large amounts of data, it may be necessary to zoom in on an area in order to view the data more clearly

- In order to zoom in, hold the right mouse button and select the area to be zoomed in. Let go of the right mouse button and the option to **zoom in** will appear. Select this option to zoom in to the area



The axes associated with the plot will adjust automatically. When zooming into plots that are part of a larger chart with multiple plots, any plots that are sharing axes with the plot that is being zoomed in will also zoom in. The zoom can be cancelled in these plots in order to return the view to normal.

In the following chart, an area in the lower left plot (green) will be zoomed in

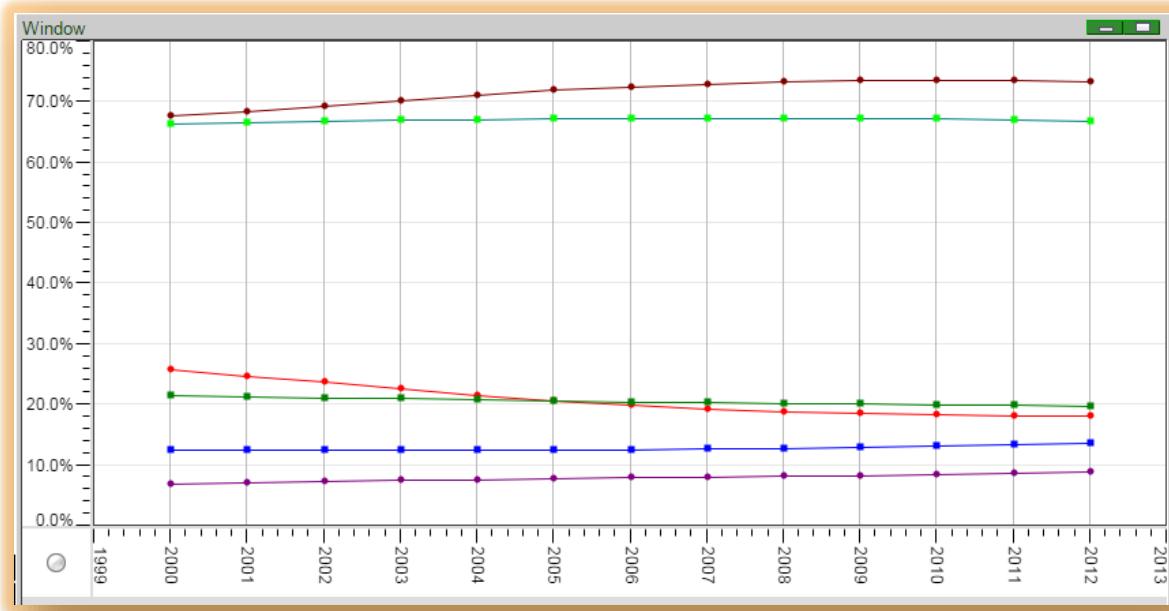


- Since the upper left plot (blue) and the lower right plot (yellow) share the same axes with the green plot, they will also be zoomed in.
- Cancelling the zoom (right click and select **Cancel Zoom**) in the original (green) plot will return all of the associated plots to the normal view. Cancelling the zoom in any of the associated plot will return the view to normal only in that plot and will adjust the view of the original zoomed in plot (since one of its axes is being returned to normal)

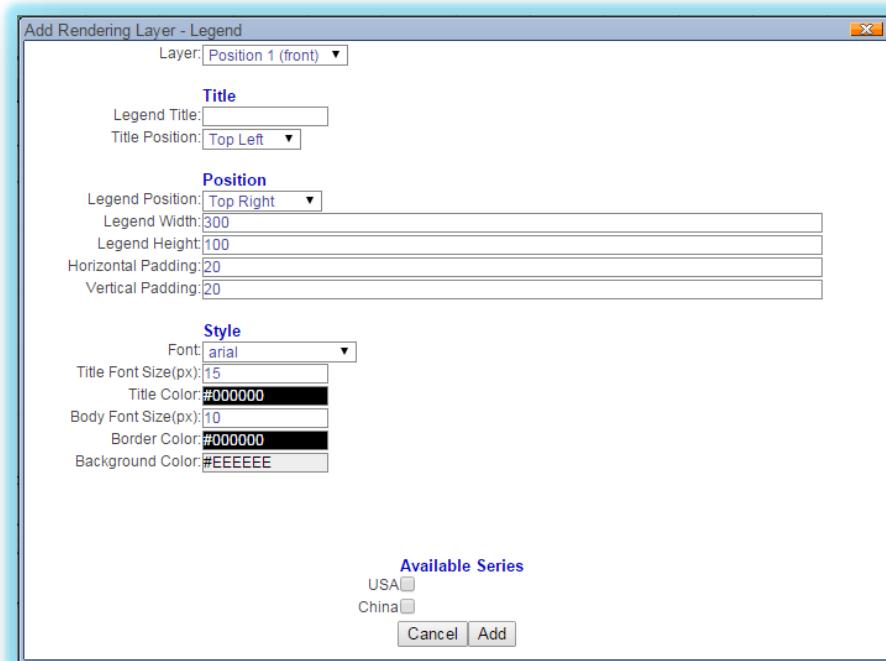
Chart Legends

When working with charts with many data points and lines, it may be necessary to add a legend in order to make it easier to interpret the data being presented.

The following chart contains two series – the percentage of the total population for three age groups (**0-14, 15-64, 65 and above**) for the USA and China. Without a legend, it is difficult to determine which lines correspond to which series.



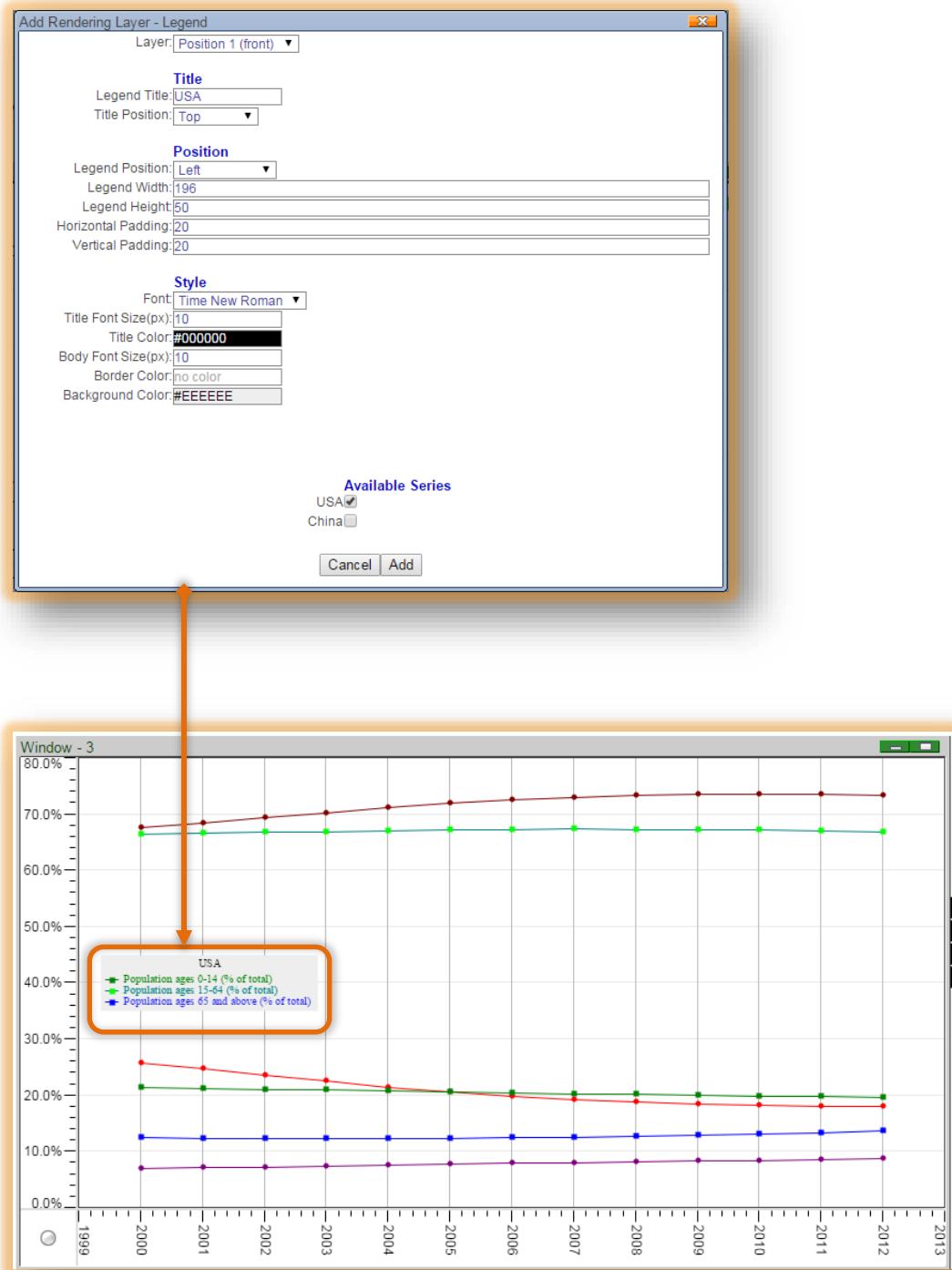
- Legends are added as rendering layers. To add a legend, open the **Chart Plot** menu (**plot configuration button**) and select **Rendering Layer – Legend** from the **Add** option. This will open up the Add Rendering Layer – Legend menu.

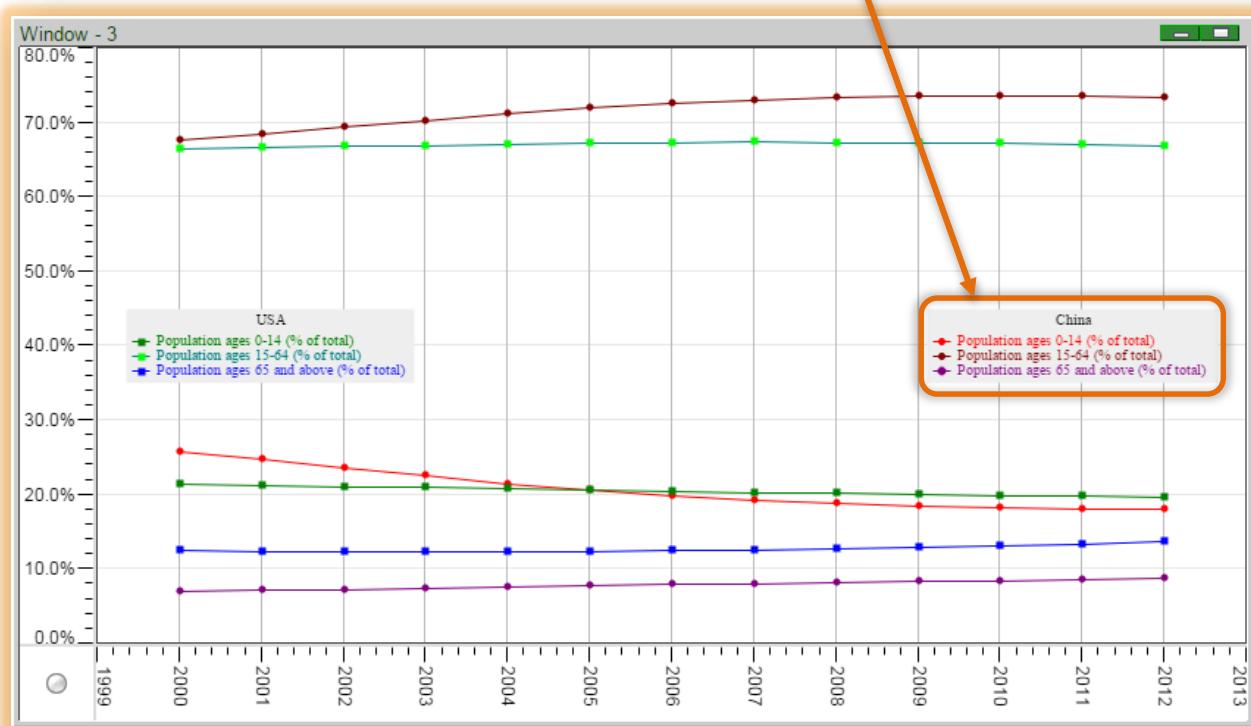
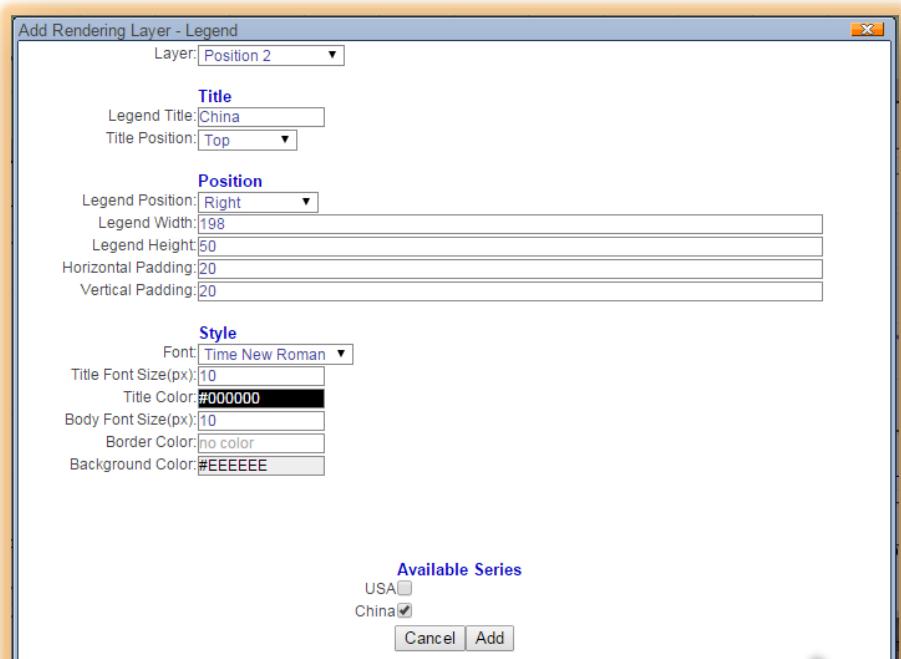


Layer	Choose a layer positon for the legend layer
Title	
Legend Title	Assign a title for the legend
Title Position	Assign a position for the title in the legend – Top Left, To Right, Top, & Hidden
Position	
Legend Position	Drop down menu for available positions on the plot for the legend
Legend Width	Set a width for the legend
Legend Height	Set a height for the legend
Horizontal Padding	Adjusts the position of the legend horizontally
Vertical Padding	Adjusts the positon of the legend vertically
Style	
Font	Assign a font for the legend
Title Font Size (px)	Adjust the size of the legend title
Title Color	Assign a color for the legend title
Body Font Size (px)	Adjust the font size of the legend body
Border Color	Applies a color for the legend border
Background Color	Applies a color for the legend background
Available Series	Select from a list of available series in the plot

Example

Using the chart seen in the beginning of this section, two separate legends will be added for the two series – percentage of the three age groups in two countries (the USA and China)





Setting up Relationships with Charts

The **Descriptions** option in series enables the user to quickly view specific information about a marker on the chart (such as seeing the *Open* and *Close* prices on a marker in the candle stick chart). However, it may sometimes be necessary to analyze other related data to the marker. Relationships can be established to and from charts – establishing a link to a static table from the chart enables the user to further examine the data related to a marker on the chart. Linking to a chart from a data table will display selected data from the table.

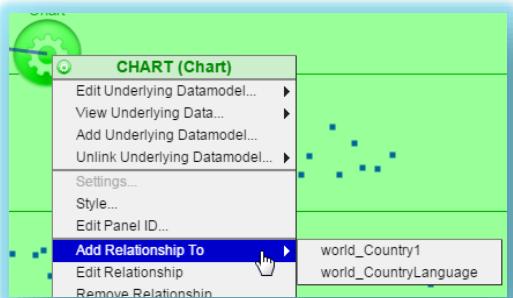
Linking to a data table is different from the **View Datasource's Data** option in that it will only display user-specified data and not the entire data from the data model. With relationships, additional filters can be used to further specify the data that is to be displayed in the tables.

Relationships are established to and from the data models within the chart. The tables do not have to be of the same data models as a relationship can be set up with related variables.

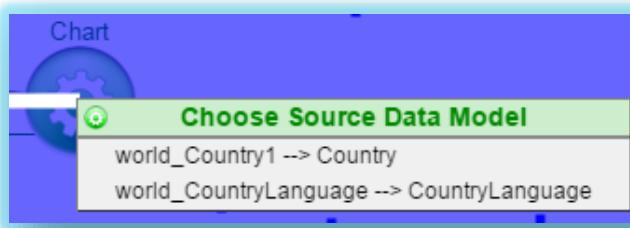
Please refer to the Relationships section for further details on relationships, the remainder of this section will be dedicated to highlighting the use of relationships to/from charts with examples

When establishing a relationship with a chart with multiple data models, the option to select a specific data model will appear:

- Establishing a relationship **TO** a chart (as the **TARGET**) with multiple data models:



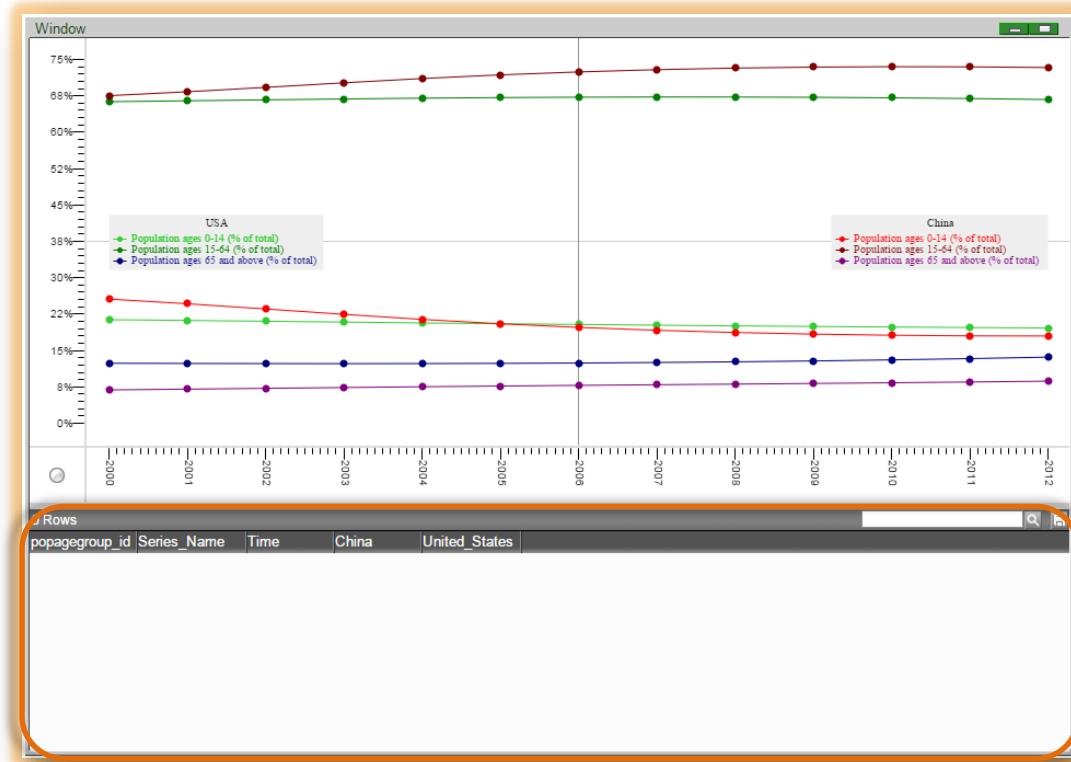
- Establishing a relationship **FROM** a chart (as the **SOURCE**) with multiple data models:



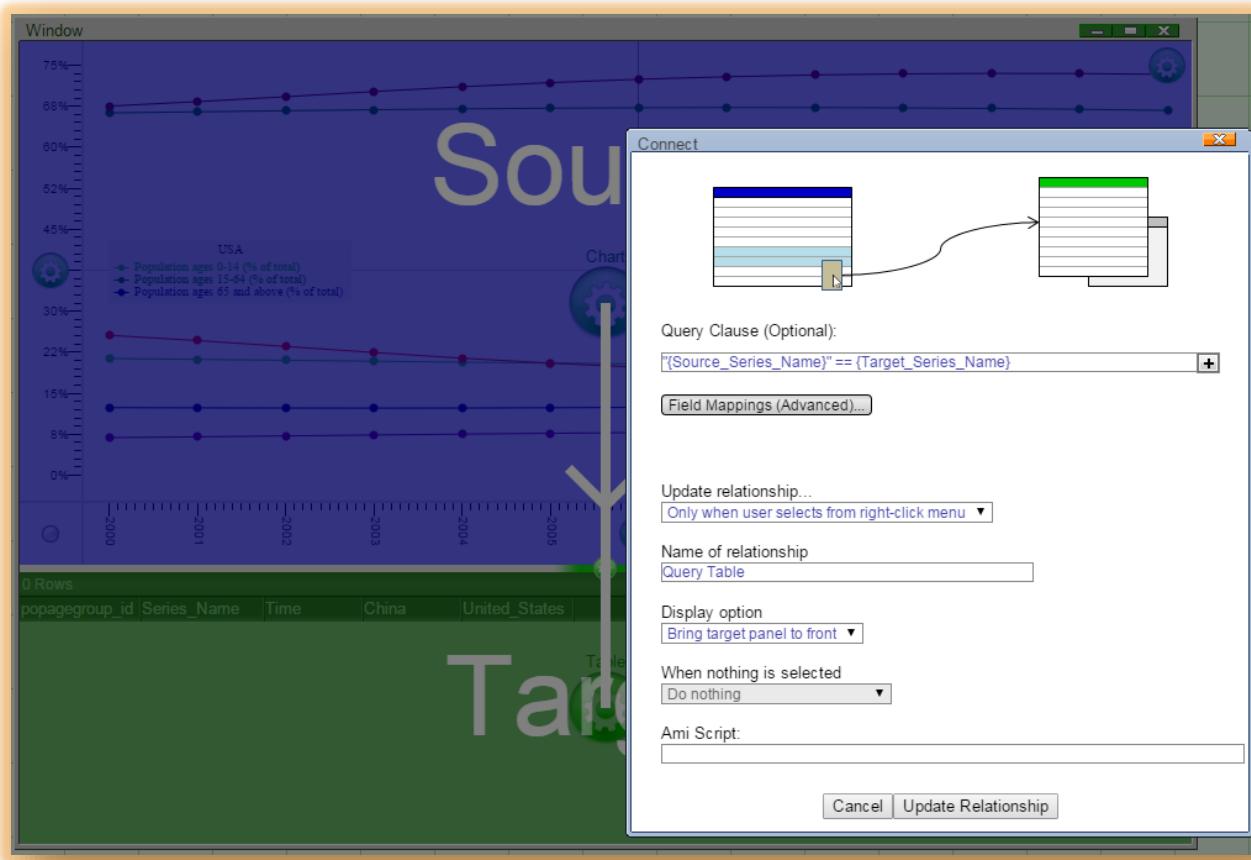
Example

Continuing with the example used in **Chart Legends**, a static table can be linked to the chart showing the percentage of the different age groups in the USA and China

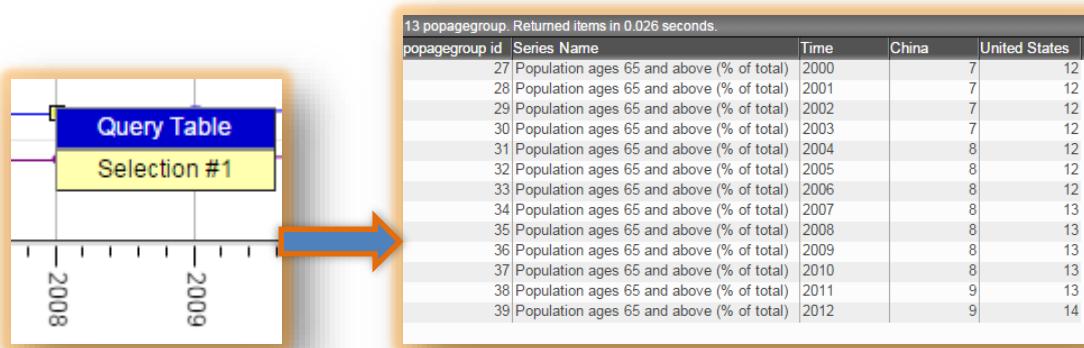
- First, a static table of the data being used for the chart is added below the chart.



- A relationship is then established from the **table** to the **chart**; linking **Series Name** to **Series Name**; the relationship is given the name **Query Table**



- With the relationship established, right-clicking on a marker on the chart will now show the option – **Query Table**. Selecting this will show all of the related data on the static table:

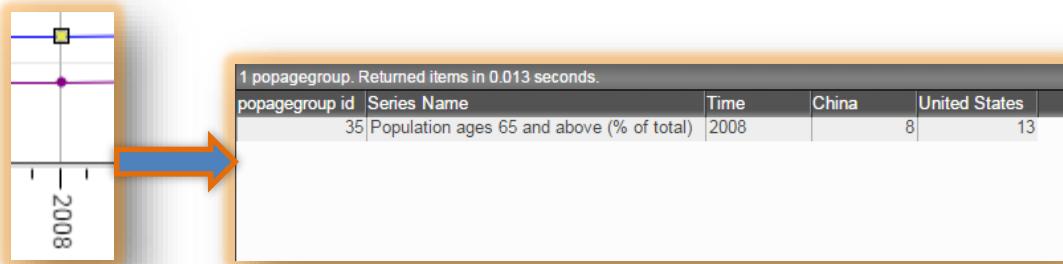


As with any relationships in AMI, the **where** clause can be modified to further specify the data displayed on the table

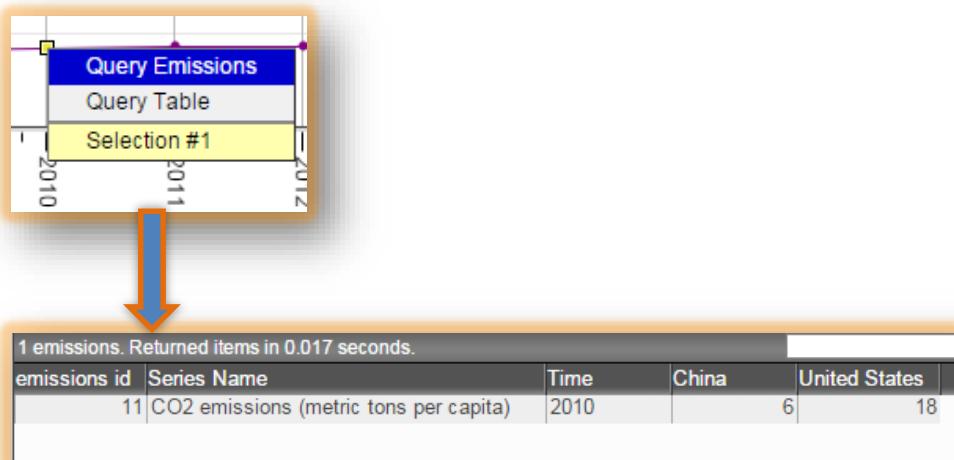
- Instead of seeing the percentages for all of the available years in the data, an additional clause can be added in order to show only the data from the same year of the marker selected:

```
Name} && ("{Source_Time}" == {Target_Time})
```

- Querying the same year (2008) as before will now show only the data from 2008:

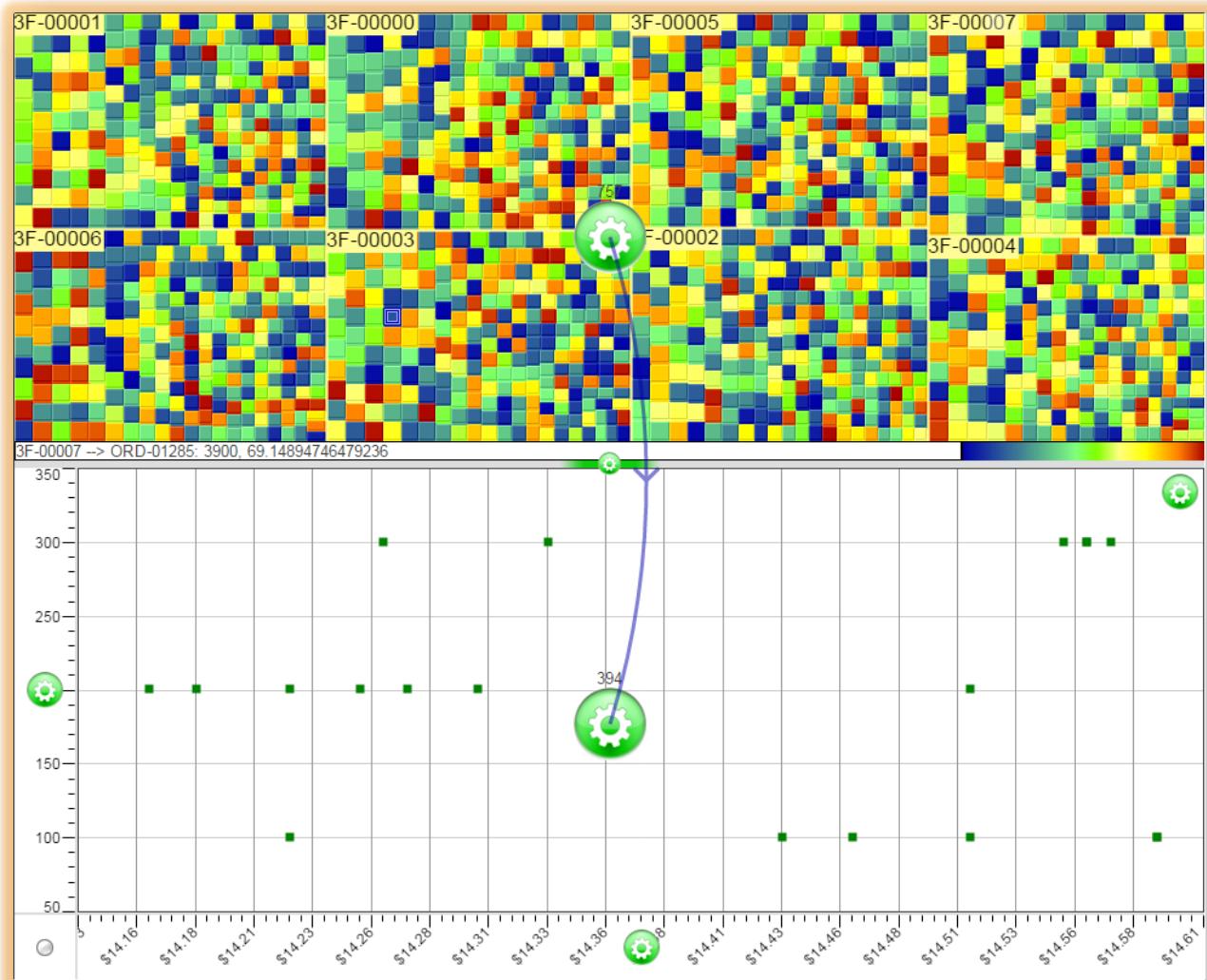


The tables do not have to be of the same data models as a relationship can be set up with related variables. Using **Time** (years) as a common variable, the chart can be linked with a static table of the CO₂ emissions by the two countries:



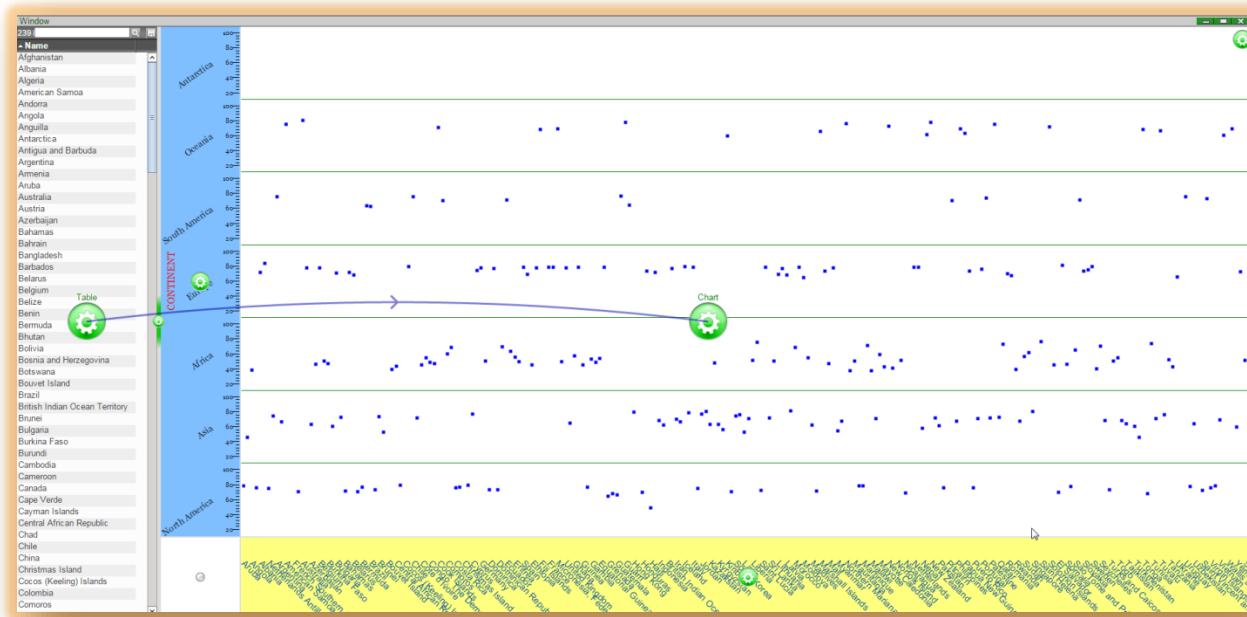
Real-time tables, static tables, and other charts can be linked to charts in order to display selected data from the tables. Layouts can even be linked to other layouts within a chart as they can be backed by different data models.

An example of a **heat map** linked to a chart:



Example – creating dynamic charts & the use of partition by and order partition by

The availability of **X/Y groupings** in series, and the ability to update relationships to static tables/charts in **real-time** enables the creation of **dynamic charts** – charts which will adjust automatically based on the selection of data being queried to the chart.



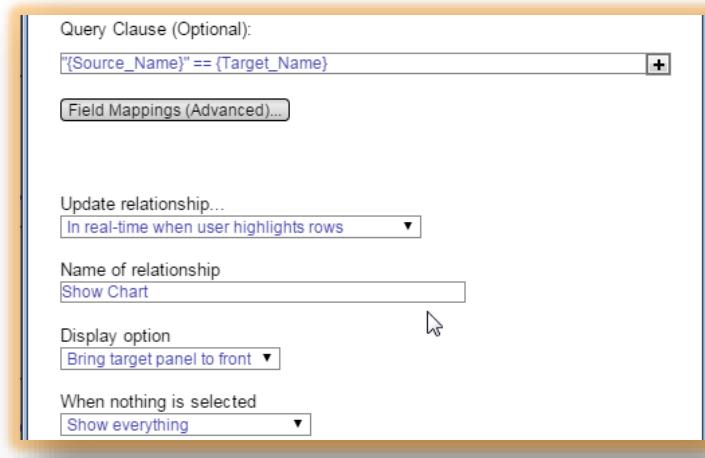
The table on the left lists all of the countries in the world and the chart on the right is based on the same data set (different data model). The chart has one rendering layer and one series – with the following set up:

The screenshot shows a configuration dialog for a chart. The dialog is divided into several sections:

- Axis**: X: (empty), Y: LifeExpectancy, X groupings: Name, Y groupings: Continent.
- Labels**: User selectable: true, Description: Name, Hover over: "<div style='background:rgba(107, 161, 255, 0.45);width:50px;height:25px;float:left; margin-right:10px;'>" + [+] button.
- Markers**: Shape: "square", Color: "#0000FF", Width(px): 5, Height(px): 5, Border color: (empty).

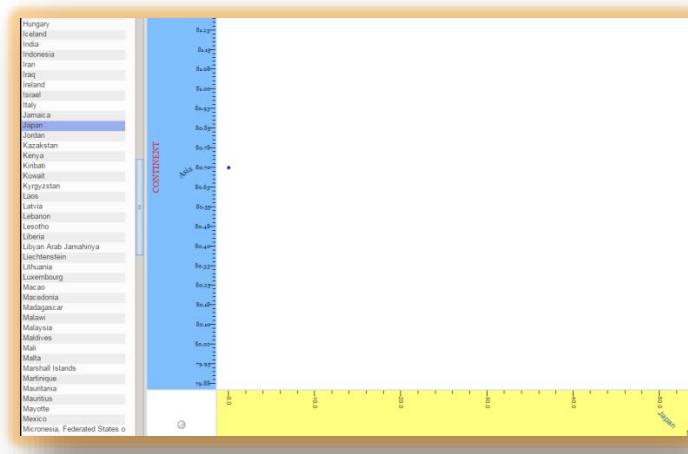
The *name of countries* is placed in the **X groupings** field because strings cannot be assigned to the **X** or **Y** fields. Assigning *continent* to the **Y groupings** field will group the **Y** variables (i.e., *life expectancy*) into separate continents.

The following relationship is established between the table and the chart:

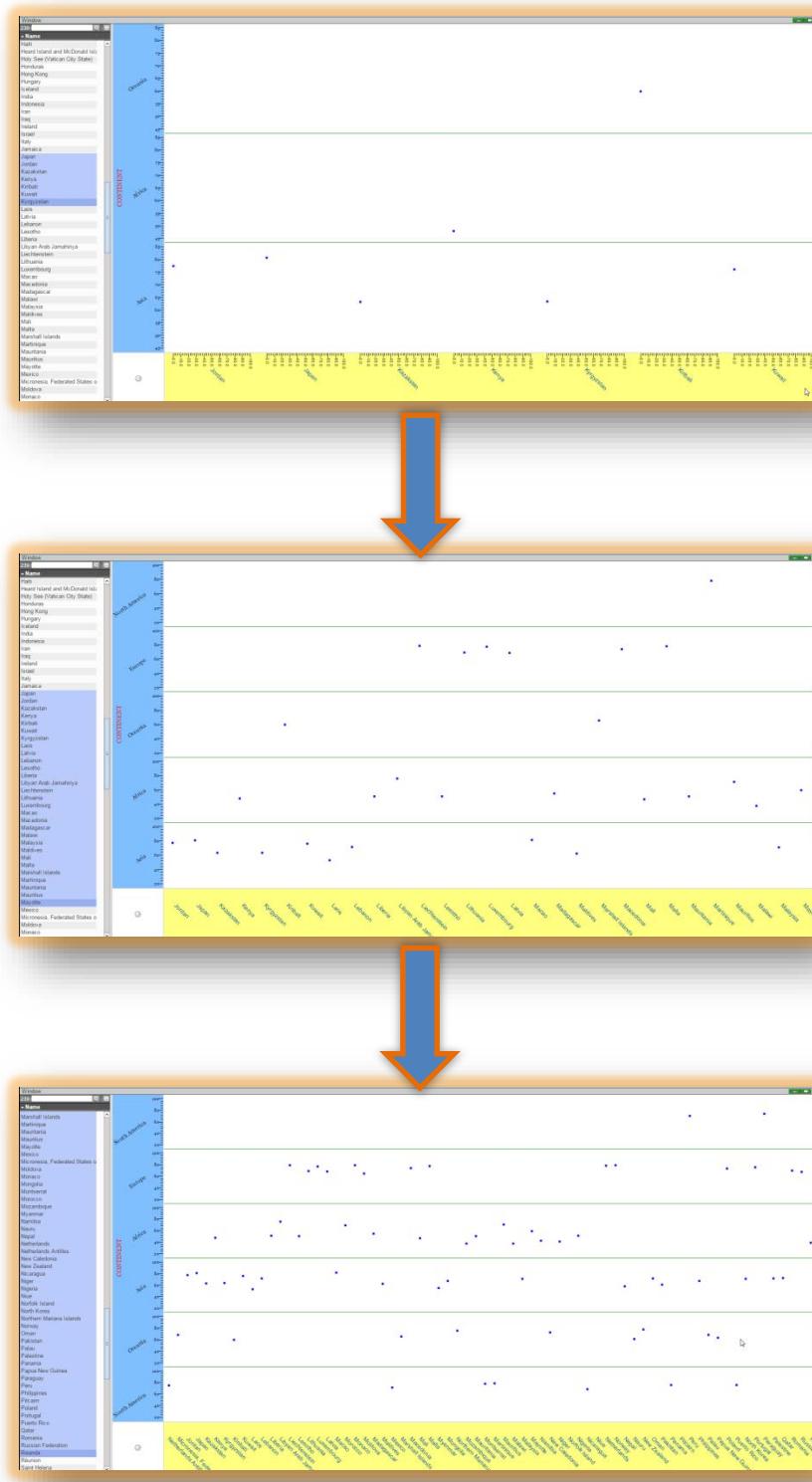


- Note:** all queries made to static tables/charts must have the source variable enclosed in quotations (e.g., "{Source_Name}" == {Target_Name}). Relationships to static tables/charts may be updated in real-time, enabling the automatic update of charts as the selection changes in the source table. Also notice that having nothing selected in the source table will display all of the points on the chart because the **show everything** option is selected for **when nothing is selected**.

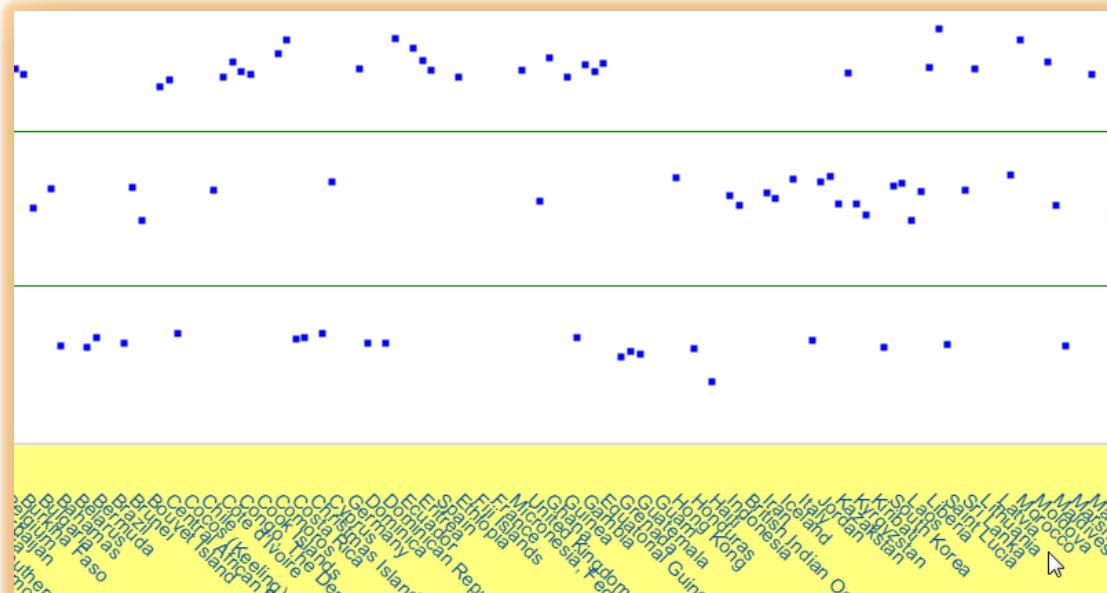
Selecting one row:



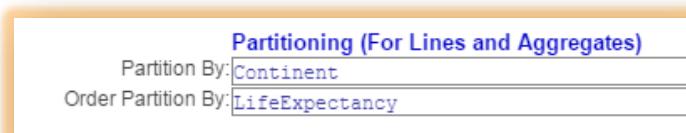
As more rows are selected on the source table, the Y axis on the chart will continue to adjust automatically to place the markers in their correct **Y groupings** (i.e., *continents*):



Without the use of `partition by` and `order partition by`, the markers will simply be displayed in alphabetical order:



Assigning a partition along with an order will provide more organization to the chart:



Using these values, the markers will be organized by continent and the countries within continent will be ordered by their life expectancy (`Y`) values

