What's New

Axiom Software

Version 2018.2

KaufmanHall

axiom software

Contents

Introduction

Kaufman Hall is pleased to announce the release of **Axiom Software Version 2018.2**. This release features continued enhancements to the Web Client user experience and to web-based reporting.

Enhancements in this release include:

- Redesigned Web Client navigation and toolbars, for an updated appearance and increased ease of use
- Various enhancements to the KPI Panel component, to accommodate a wide variety of information and use cases:
    - New KPI kinds to emphasize either the numeric value or the status
    - Support for multiple KPI sizes—from full details to titles only
    - Additional supported properties such as subtitles, additional values, and sparkline charts
    - Ability to display a menu of commands on each KPI box
- Enhanced ability to display grouped data in a Data Grid component, including displaying different columns in each grouping and using different grid properties
- New slider refresh variables, to select values by dragging a slider handle
- Ability to configure the rejection behavior for plan file processes, to reject plan files back to specific steps

This *What's New* document provides information on all new features and enhancements in this release. Reviewing this document should give you a basic understanding of how these new features work, and what benefits they may provide to your organization. For full details on any new feature, please see the Axiom Software Help files or the PDF guides.

IMPORTANT: Before upgrading to version 2018.2, make sure you have reviewed the separate *Release Notes* document to understand any important technical changes and upgrade considerations in this release.

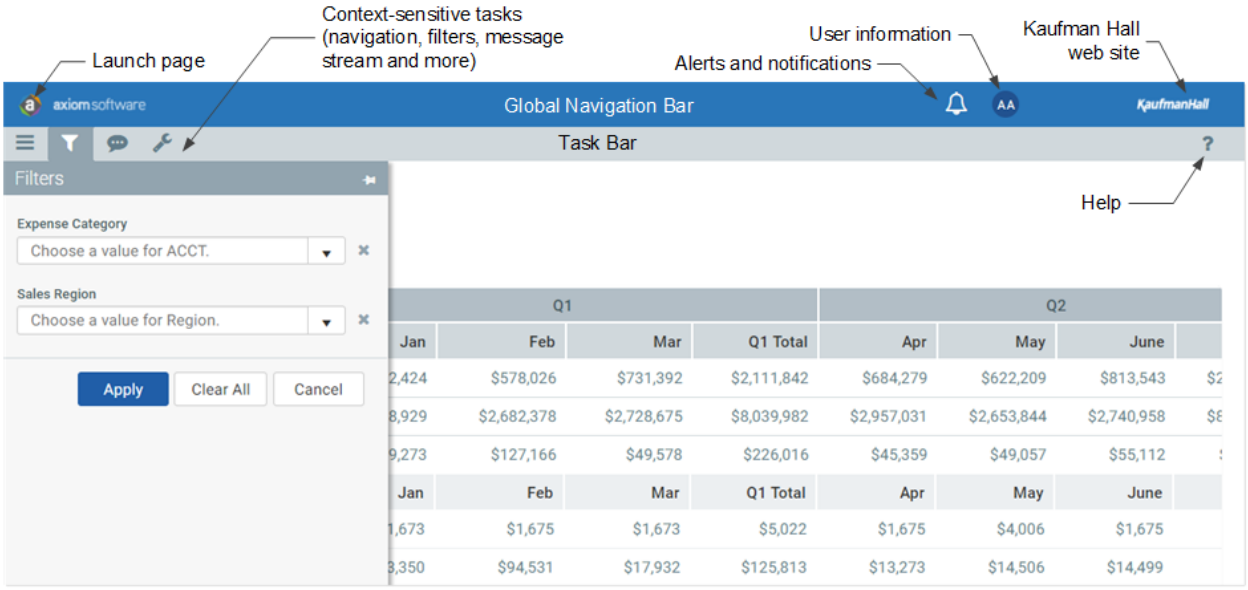
Web Client user interface

In this version, we introduce some enhancements to the Web Client user interface. These enhancements are intended to update the general appearance and improve ease-of-use, as well as provide a framework to support future standardization of the user interface for Axiom forms, web reports, and other Web Client features.

Redesigned navigation and task bars

The basic user interface for navigating the Web Client and performing tasks (also known as the "Web Client Container") has been redesigned. In previous releases, this was a single blue bar across the top of the page, providing access to various features, login information, and help. This single bar has been split into two bars:

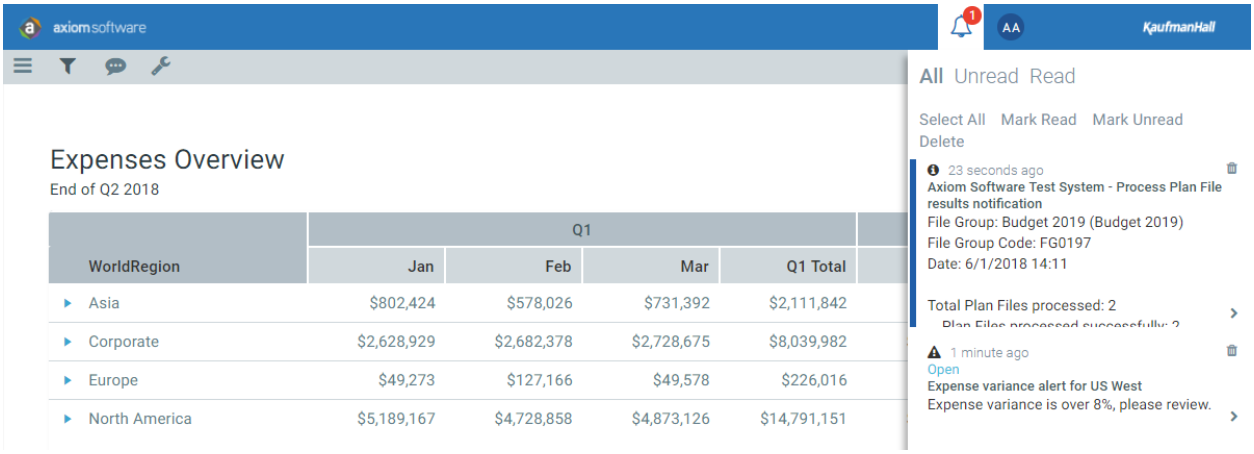
- **Global Navigation Bar**: The blue bar at the top of the page provides access to system-level areas and features.
- **Task Bar**: The gray bar underneath it provides access to various tasks and features that are specific to the current page or document.



For the most part, existing features are located in the same general area that they were previously, with minor display changes. However, in a few cases features were moved to different locations and/or experienced notable design changes, as discussed in the following sections.

Notifications

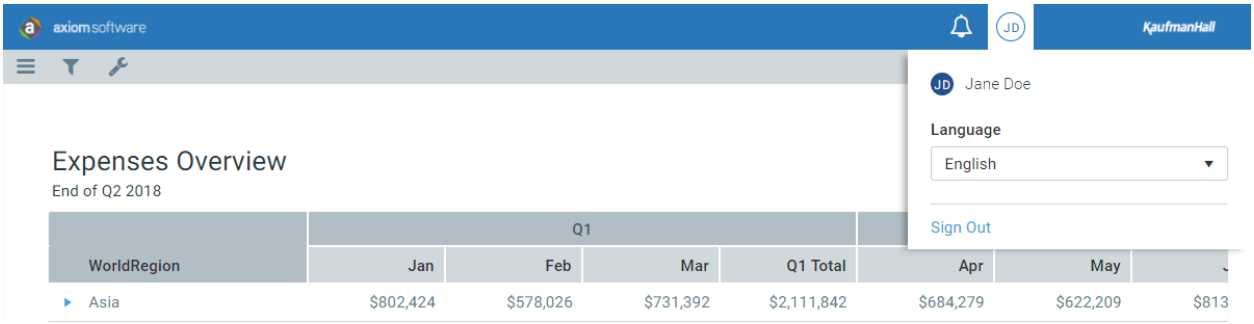
The Notifications panel is no longer located in the left-hand side of the container with the context-sensitive tasks. Because the Notifications panel displays system-wide notifications, it is now positioned in the top bar next to the user login information.



New placement of Notifications panel

User Information

The user login information is no longer displayed directly in the container. Instead, click the user avatar (initials in a circle) in the top right of the blue bar to see the user information. From here, you can view the user name and log out.



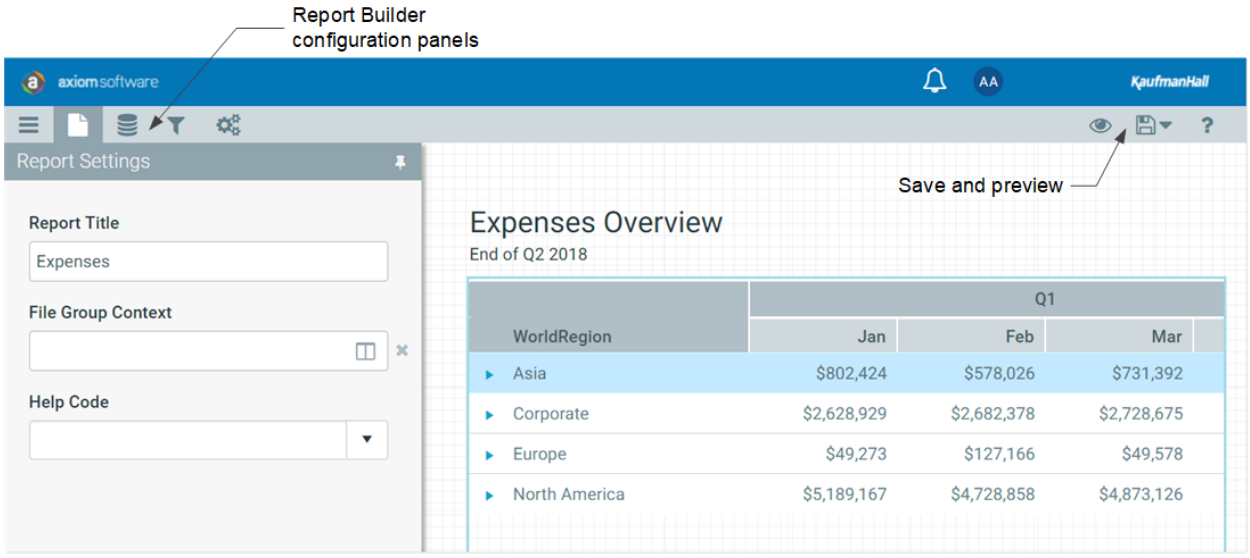
New user information panel

Additionally, you can now optionally override the default language, in order to use Axiom Software in a different language than the current locale. For example, if your current locale is French, then by default Axiom Software displays in French. However, if you want to view Axiom Software in English instead, you can select English from the **Language** list. If you launch the Desktop Client (Excel or Windows) from the Web Client launch page, it also displays in the selected language.

Report Builder

When using the Report Builder, the icons for the different configuration panels now display directly in the container task bar, instead of underneath it. Just like before, you can click the icons to configure different aspects of the report.

Additionally, the buttons to save and preview the web report are now available in the right-hand side of the gray task bar, instead of floating in the report canvas area.



Redesigned location of Report Builder icons

Note that there is no longer a separate help icon on the Report Builder panel. If you are in the Report Builder when you click the main help icon (which is now located in the right-hand side of the gray task bar), the Axiom Software Help file shows information about the currently active section of the Report Builder.

Updated help panel with support for file-level help

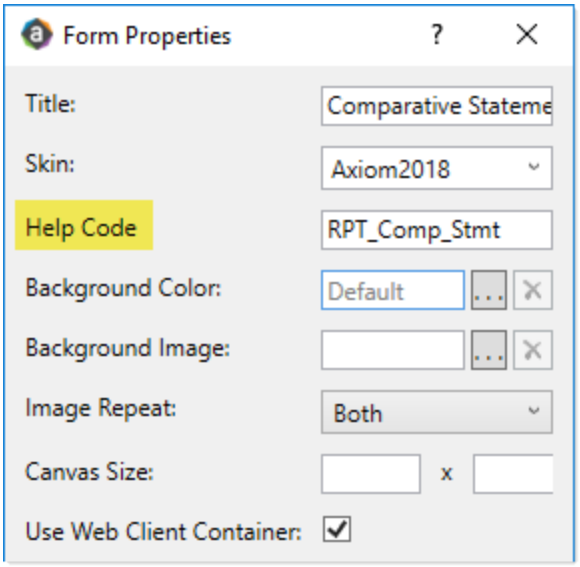
As part of the Web Client user interface changes, the Help icon has been moved to the right side of the gray task bar. This opens the Axiom Software Help to either a context-sensitive location or to the home page, depending on where you are in the Web Client. For example, if you are in the Web Client Table Manager, clicking the Help icon shows help for the current section of the Table Manager, just like in previous versions.

Additionally, you can now define custom file-level help for Axiom forms and for web reports created using the Report Builder. If file-level help is defined, the custom help text displays in a panel along the right-hand side of the page when users click the Help icon.

Example help panel displaying custom file-level help

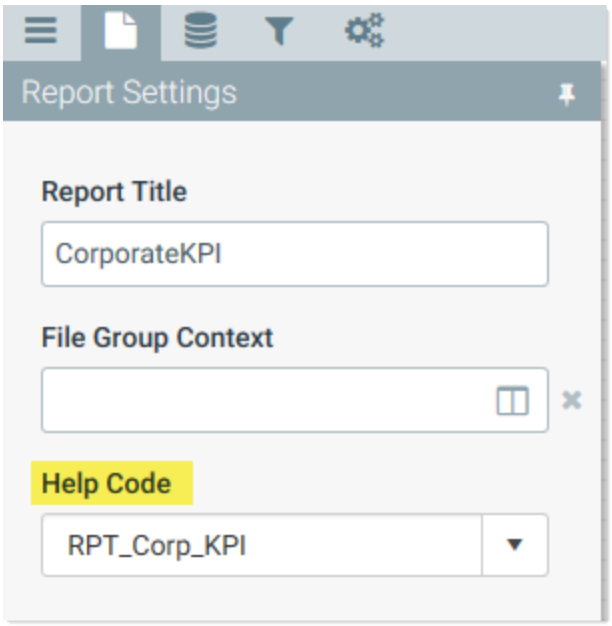
To display custom file-level help, you must first define the content using the existing **Form Help Admin** area of the Web Client. This area is only accessible to administrators. As part of this process, you create a help code that identifies the content. Once you have the help code, you can enter it into the **Help Code** property for either the Axiom form or the web report.

For Axiom forms, click **Edit Form Properties** in the Form Assistant or Form Designer to specify a help code:



Example Help Code in form properties

For web reports, you can specify a help code in the **Report Settings** panel of the Report Builder:



The help panel that displays the custom help text is the same panel used by the Form Help component in Axiom forms. This help panel has been updated to include a link to Axiom Software Help, so that users can still access the application help file as needed. Additionally, note the following changes:

- The help code no longer displays directly on the help panel, but can be seen in a tooltip that displays when hovering the cursor over the title.
- If an invalid help code is specified, the help panel now displays a message about the invalid help code (instead of being unresponsive).

KPI Panel component

This section details the new features and enhancements made to KPI Panel components.

New KPI kinds and sizes

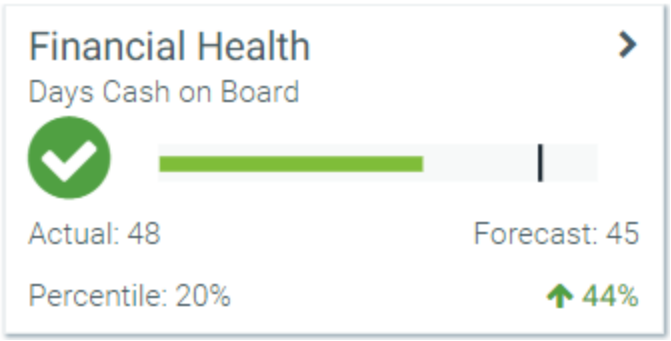
The KPI Panel component has been enhanced to support different kinds of KPIs and different sizes. These options provide greater variety to accommodate the level of detail you want to display about the KPI, as well as the information that you want to emphasize (the number itself, or its status).

These changes apply to KPI Panel components used in either Axiom forms or the Report Builder.

KPI kinds

KPI Panels now support two different KPI kinds, **Basic** and **Status**. Basic KPIs emphasize numeric data, whereas Status KPIs are intended to show whether a KPI is "good" or "bad" at a glance. Basic and Status KPIs can be mixed within the same panel.

Example Basic kind



Example Status kind

The Basic kind is very similar to the KPI version that was introduced in 2018.1 (which is now known as the Legacy kind). It uses a primary KPI numeric value (Value1), and can display supporting data and a chart.

The Status kind is intended to be used when you want to emphasize the current status of a KPI rather than its actual value. The status indicator reflects whether the KPI is up, down, or neutral, using color coding and symbols. The Status kind can also display supporting data and a chart, with the chart displayed next to the status symbol instead of at the bottom of the box.

To indicate the KPI kind, the KPISource data source and the KPI table have a new column of **Kind**. You can enter `Basic` or `Status`, depending on how you want the KPI to display. To indicate the status when using the Status kind, the KPISource data source and the KPI table have a new column of **StatusSign**. You can enter the following values for StatusSign, which display using the icons shown below:

Icons for Status KPIs

When displaying a KPI in a KPI Panel component, the behavior is as follows:

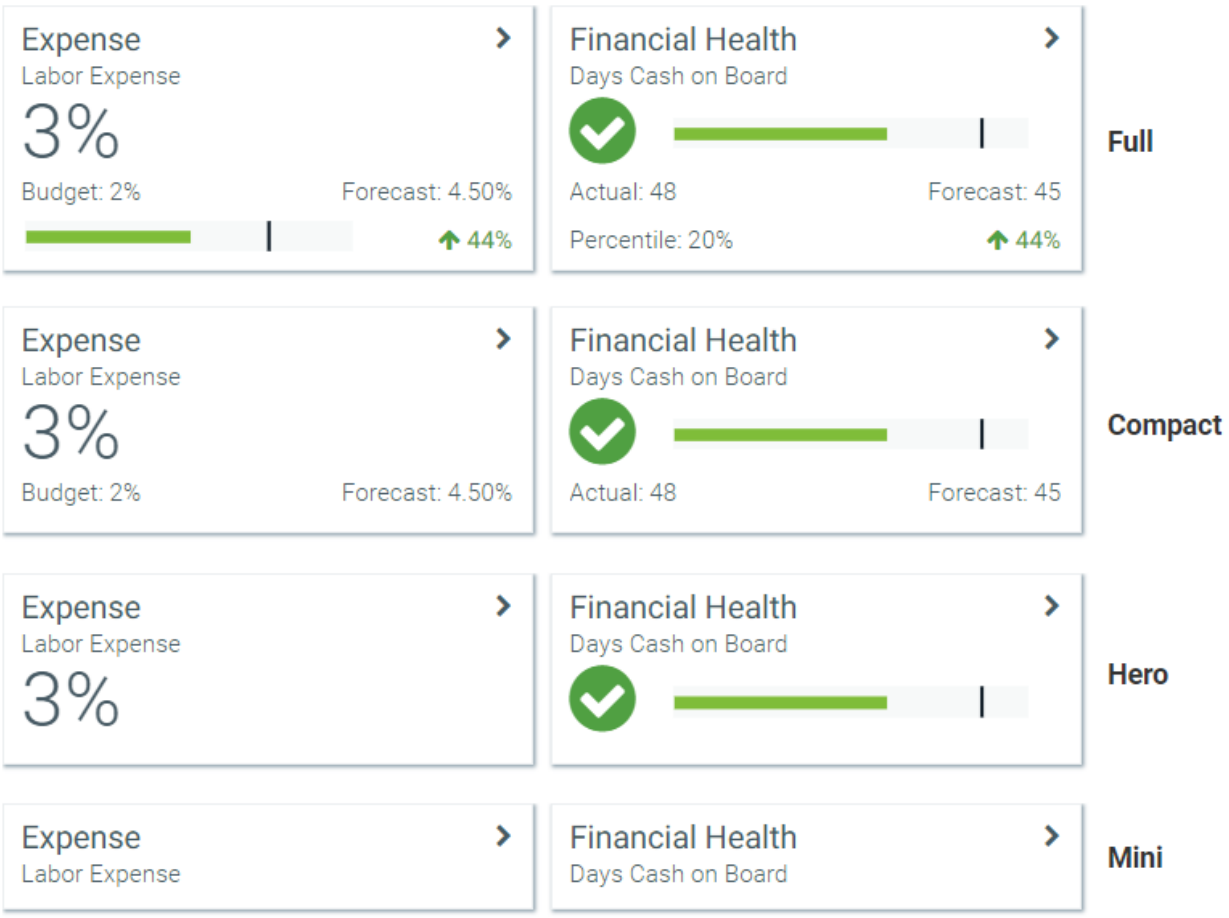
- If a KPI is Basic kind, Value1 is displayed prominently as shown in the example. StatusSign is ignored.
- If a KPI is Status kind, the StatusSign is displayed prominently as shown in the example. Value1 is ignored.
- If the Kind column is blank or omitted, the KPI displays using the legacy format introduced in 2018.1. This is intended for backward-compatibility only, so that existing KPI Panels display the same before and after the upgrade. For more information, see Upgrade behavior.

KPI sizes

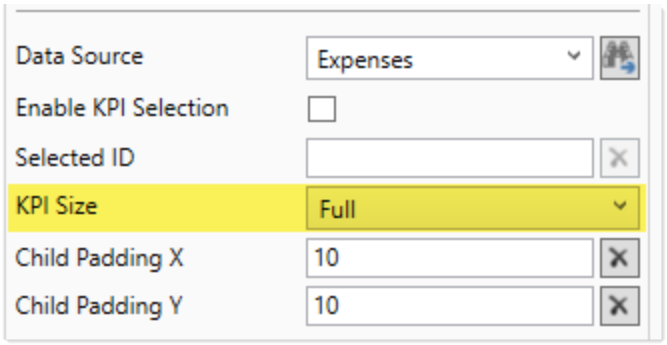
You can now specify the size of the KPI boxes for the KPI Panel component. The size determines how much detail shows for each KPI, ranging from titles-only to all available detail. KPIs support the following sizes:

- **Full** (default): All available KPI values for the specified kind are shown, using the full size of the box.
- **Compact**: The bottom row of values does not display in the KPI. This is intended for KPIs where you have some supporting values but you do not need the full level of detail.
- **Hero**: Only the most important values display in the KPI, including the title and subtitle, Value1 for Basic KPIs, and the StatusSign and chart for Status KPIs. This is intended for KPIs where you do not need to show any supporting values. You only want to communicate the primary value or status.
- **Mini**: Only the title and subtitle display in the KPI. This is intended for cases where the KPI Panel component is being used solely as a selector tool (see Select a KPI in a KPI Panel component) or just to execute actions.

The following screenshot shows how the specified size affects the presentation of Basic and Status KPIs.



The KPI size is set at the component level, so that all KPI boxes shown within the component are the same size and flow nicely. In Axiom forms, you can configure the size in the Form Designer or Form Assistant. By default, the size is set to **Full**.



New KPI Size in KPI Panel component properties

For the Report Builder and web reports, the KPI size is set at the template level and cannot be changed in the Report Builder. Because components cannot be resized or moved in the Report Builder currently, the template must be designed to accommodate a specific KPI size. The existing KPI template provided by Axiom Software uses the full KPI size. If you want to use a different size in web reports, you must create a

new template and configure it to use the desired size. For more information on creating Report Builder templates, see the following topic in Axiom Software Help: *Creating custom templates for the Report Builder* (AX1701).

Legacy KPIs do not support sizes. Legacy KPIs are always displayed at full size, regardless of the size set in the KPI Panel component properties.

Upgrade behavior

When you upgrade, any existing KPI Panel components will continue to display as is.

- Existing KPISource data sources will not have the `[Kind]` column, which means all of the KPIs in the data source will be treated as legacy.
- The Kind and StatusSign columns are automatically added to existing KPI tables, however, the Kind column will be blank so all existing KPIs in the table will be treated as legacy.
- Legacy KPIs ignore the KPI size, so the introduction of this new property has no effect on the existing KPIs.

Going forward, all KPIs should be assigned a kind. The legacy kind is intended to support backward-compatibility only, so that you can continue to use your existing KPI Panel components until you get a chance to update them for the new kinds. The legacy kind should not be mixed with other kinds within the same KPI Panel component, because the legacy kind uses slightly different formatting than the new kinds and therefore will not align with the new kinds (and will not honor size).

When you are ready, you can update your existing KPI Panel components as follows:

- If the component uses a data source, you can add the new columns to the data source manually (or use the Data Source Assistant), and then populate them as needed.
- If the component uses a KPI table, the new columns are automatically added to the table but they are not populated. You can either manually populate the new columns, or update your save-to-database report to save the appropriate values to the new columns.
- If the component is in an Axiom form and you want to use a different size, you can set the size on the component after assigning a kind to the KPIs.
- Currently, it is not possible to adjust an existing web report in the Report Builder to use a different KPI size. As previously discussed, you can create a new template that uses the desired size, and then create a new web report using that template.

New KPI display options

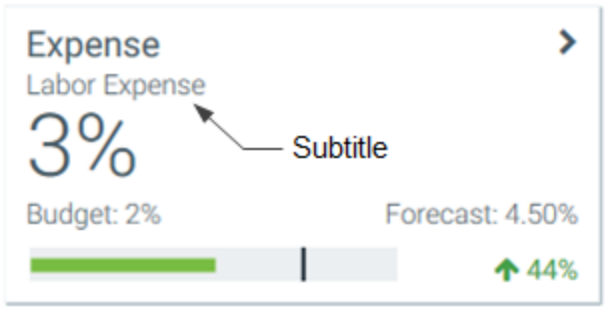
In addition to the KPI kinds and sizes, several new display options were added to KPIs. You can now optionally specify the following:

- Subtitle
- Additional supporting value (Value4)
- Sparkline chart (instead of the default bullet chart)
- Menu to display several commands (instead of the default single command)

These changes apply to KPI Panel components used in either Axiom forms or the Report Builder.

Subtitle

The KPISource data source and KPI tables now have a **Subtitle** column to display an optional subtitle for KPIs. The subtitle displays in smaller text underneath the main title, for all KPI kinds.



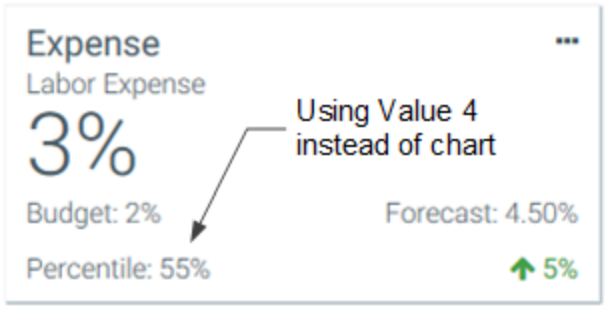
Example KPI with subtitle

Additional supporting value

The KPISource data source and KPI tables now have the following columns to display an additional supporting value for KPIs:

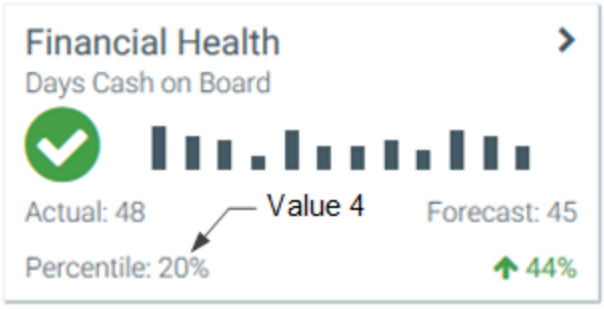
- **Value4**
- **Value4Label**

Value4 and its label display in the bottom row of the KPI, on the left-hand side. When using the Basic kind, Value4 and the chart are interchangeable, so you can display one or the other. If both are defined, the chart takes precedence.



Example Basic kind with Value4

When using the Status kind, the chart displays in a different location so you can display both if desired.



Example Status kind with Value4

Sparkline charts

You can opt to display a sparkline chart in KPIs instead of the default bullet chart. The KPISource data source has the following new columns to display a sparkline chart:

- **SparklineDataSource**: The name of an XYChart data source defined in the spreadsheet. Line charts and column charts both use XYChart data sources.
- **SparklineSeriesName**: The name of a series in the XYChart data source. Line and column series can be used.

For example, the sheet may contain an XYChart data source named Sparkline, with two series. One is a line series named LineSeries and the other is a column series named ColumnSeries.

	AE	AC	AD	AE	AF	AG	AH
5							
6		[XYChart;Sparkline]	[Kind]	[SeriesName]	[XValue]	[XValue]	[XValue]
7		[XValueName]			Jan	Feb	March
8		[Series]	Line	LineSeries	5	10	15
9		[Series]	Column	ColumnSeries	1000	800	700

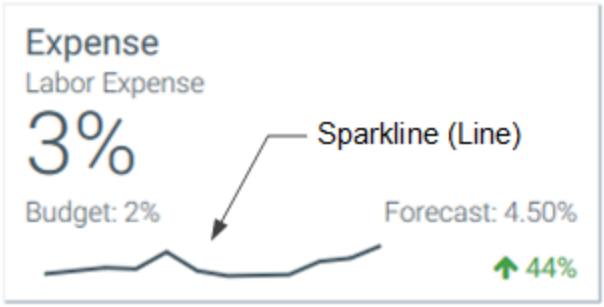
Example XYChart data source

To display these series as sparklines in KPIs, the XYChart data source name and series names are listed in the KPISource data source:

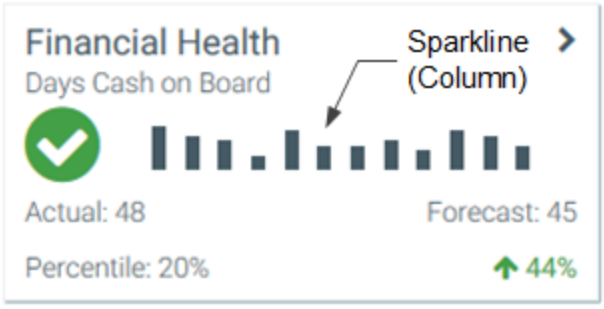
	A	B	C	D	E	F	G	H
5								
6		[KPISource;Examples]	[ID]	[Kind]	[Title]	[SubTitle]	[SparklineDataSource]	[SparklineSeriesName]
7		[KPI]	Basic1	Basic	Expense	Labor Expense	Sparkline	LineSeries
8		[KPI]	Status1	Status	Financial Health	Days Cash on Board	Sparkline	ColumnSeries

Example KPISource data source referencing sparkline series

When the KPIs are rendered, Axiom Software finds the specified series in the specified XYChart data source, and displays the data as a sparkline chart. For example:



Example Basic kind with sparkline



Example Status kind with sparkline

You can also use sparkline charts when the KPIs are sourced from a KPI table rather than a KPISource data source. The KPI table contains a new column, **SparklineData**, to store the sparkline data for KPIs. In this case, the save-to-database report that you are using to save the KPI data to the table must be set up as follows:

- The file must contain an XYChart data source that is configured to display sparkline data, such as the example XYChart data source shown previously.
- When setting up Save Type 1 in the file, the contents of the SparklineData column must contain the following special syntax to specify the XYChart data source and series: `[Datasource=DataSourceName;Series=SeriesName]`. For example:

	A	B	C	D	E
1					
2		[Save2db;DeptKPI;;;;;False]	Name	Title	SparklineData
3					
4		[save]	Basic1	Expense	[datasource=sparkline;series=lineseries]
5		[save]	Status1	Financial Health	[datasource=sparkline;series=columnseries]

When the save-to-database is executed, Axiom Software finds the designated XYChart data source, and converts the contents of the specified series into an XML string. That XML string is then saved to the SparklineData column in the KPI table. When the KPI table is used with a KPI Panel component, the XML string is used to render the sparkline chart on the KPI box.

Menu

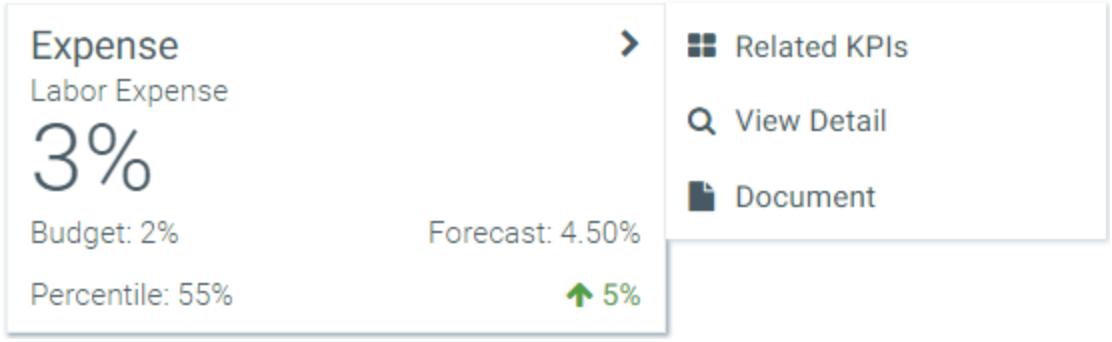
You can now configure a custom menu for each KPI box, in order to display multiple actions to execute. The menu is an alternative to using the existing Command column, which allows you to execute a single command from the top right corner of the box. The KPISource data source now has a **MenuDataSource** column, to specify the name of a KPIMenu data source.

Using the KPI Menu data source, you can define multiple menu items for display on a KPI box, including the menu item name, icon, tooltip, and command. Menu items can also be dynamically flagged as disabled or hidden. The following screenshot shows an example KPI menu data source with three items.

	B	C	D	E	F	G	H
14							
15		[KPIMenu;KPIMenuSourceMultipleItems]	[ID]	[Name]	[Icon]	[Tooltip]	[Command]
16		[MenuItem]	1	Related KPIs	fa-th-large	Related KPIs	command://ShowFormDialogPanel?DialogPanel=DialogPanelDetailKPI
17		[MenuItem]	2	View Detail	fa-search	View Detail	command://ApplySharedVariables?ShowChart=On
18		[MenuItem]	3	Document	fa-file	File	document://\Axiom\SystemFolderName_ReportsLibrary\Forms\report.xlsx?ds

Example KPIMenu data source

When the KPIMenu data source has multiple items, a carat icon displays in the top right corner of the KPI box. Users can click the icon to open the menu. If the KPIMenu data source has just one visible item, the associated icon displays in the top right corner instead of the carat icon, and users can click the icon to execute the associated command directly.



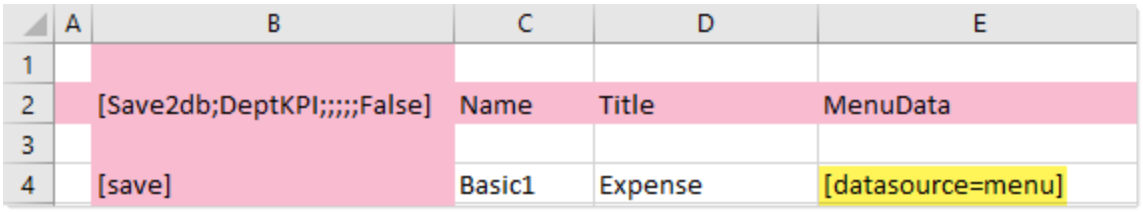
Example KPI with menu

You can create a new KPIMenu data source using the right-click menu: **Create Axiom Form Data Source > KPI Menu**. Valid commands are the same items that are supported for the Command column: web URLs, document shortcuts, and commands from the Command Library.

You can also use a KPIMenu data source when saving KPI values to a KPI table. The KPI table contains a new column, **MenuData**, to store the menu data for KPIs. To do this, the save-to-database file that you use to save KPI data to the table must be set up as follows:

- The file must contain a KPIMenu data source. This data source is set up as normal, on any sheet of the file.

- When setting up Save Type 1 in the file, the contents of the MenuData column must contain the following special syntax to specify the KPIMenu data source to save: `[Datasource=DataSourceName]`. For example:



	A	B	C	D	E
1					
2		[Save2db;DeptKPI;;;;;False]	Name	Title	MenuData
3					
4		[save]	Basic1	Expense	[datasource=menu]

When the save-to-database is executed, Axiom Software finds the designated KPIMenu data source, and converts the contents of it into an XML string. That XML string is then saved to the MenuData column in the KPI table. When the KPI table is used with a KPI Panel component, the XML string is used to render the menu on the KPI box.

NOTE: When using a KPI table with a KPI Panel component in the Report Builder, commands from the Command Library are not supported. This applies to the Command column and the MenuData column. Any commands used must be web URLs or document shortcuts.

Upgrade behavior

Existing KPI Panel components are not affected by the introduction of these new features.

- If you are using a KPISource data source, the new columns are not present in existing data sources. You can add the columns manually (or use the Data Source Assistant) and populate them as desired.
- Although the associated columns are automatically added to any existing KPI tables, these columns will be blank and therefore will not affect any existing KPIs. If you want to use the new features, you must modify your save-to-database report to populate the columns as desired as part of the save.

In both cases, if you are updating existing KPIs to use new features, you should also specify a kind for each KPI.

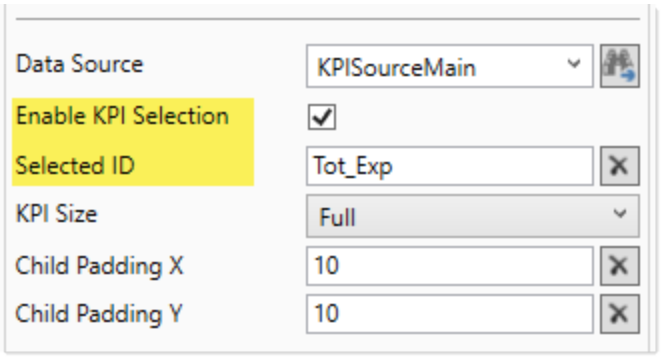
Select a KPI in a KPI Panel component

You can now select a specific KPI box in a KPI Panel component, and configure the form to change in some way based on that selection. For example, you may want to dynamically show additional KPIs related to the selected KPI, or change the data shown in a chart based on the selected KPI.

NOTE: This feature only applies when using a KPISource data source in an Axiom form. It is not possible to select KPIs when using a KPI table.

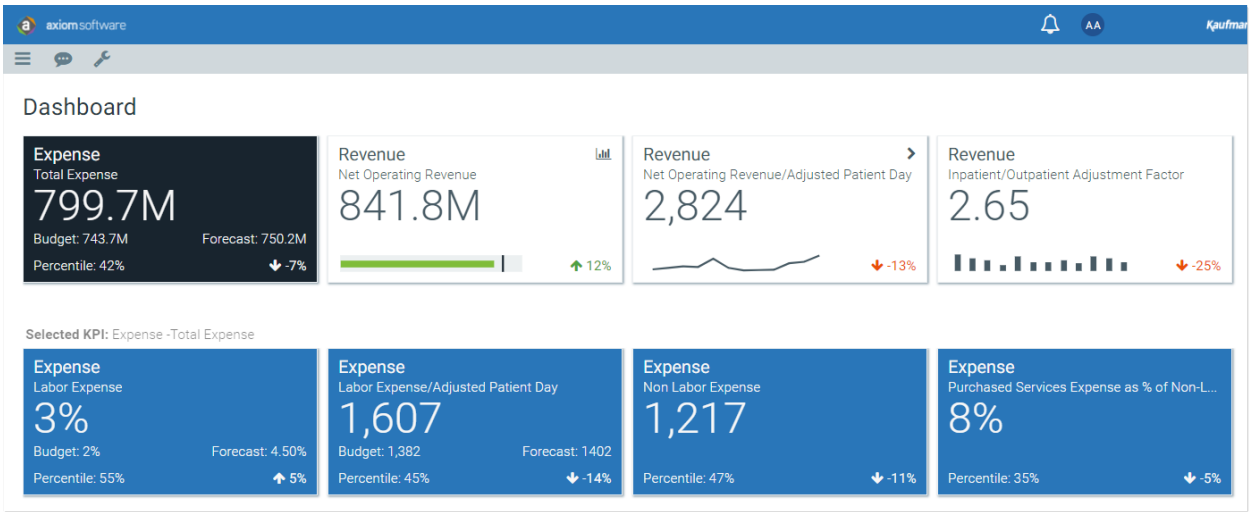
The following properties were added to KPI Panel components:

- An **ID** column was added to the KPISource data source. In order to select KPIs, this column must be populated with unique values to identity each KPI.
- The option **Enable KPI Selection** was added to the KPI Panel component properties. Select this check box if you want users to be able to select KPIs.
- The field **Selected ID** was added to the KPI Panel component properties. When a user selects a KPI, the associated ID is written to this field. You can then configure the form to change in some way based on the currently selected ID. You can also enter an ID into this field, to specify a default selected ID to be used when the form is initially opened.



New properties for KPI Panel component

To differentiate the selected KPI from the other KPIs in the panel, the selected KPI is shown with a black background. In this example, selecting a KPI in the top panel causes the bottom panel to populate with related KPIs.



Example form with selected KPI

A similar feature is available when executing commands from the KPI panel. If the ID column is present in the KPISource data source and populated with values, then when a user executes a command from a KPI box, the ID of the corresponding KPI is written to the **Triggering ID** field. You can use this value to impact the command being executed. For example, if the command opens a dialog panel, you can reference the

triggering ID to list the name of the current KPI in the dialog panel. The Triggering ID field is only present on the Form Control Sheet; it does not display in the Form Assistant or Form Designer.

Additional KPI Panel enhancements

Style options

KPI Panel components now support the following styles in Axiom forms to change the general formatting of the panel. These styles can be set in the Form Designer or Form Assistant using the Style property for the KPI Panel component:

- **no-kpi-box shadow**: Removes the drop-shadow on the KPI boxes in the panel.
- **no-kpi-border**: Removes the border on the KPI boxes in the panel.

Although both styles are add-on styles (with a plus sign), it is not necessary to specify a base style for the KPI Panel component, as the base style is assumed.

Currently, it is not possible to set component styles in the Report Builder. Instead, the styles are determined by the template used to create the report. If you want to use these styles in a web report, you must create a Report Builder template with the desired styles applied.

Change to Sign icon

In previous releases, the up/down indicator for the Sign property was a carat symbol that pointed up or down. This has been changed to use an up or down arrow, to make the up/down indication more clear. This applies to all KPIs, including legacy KPIs.

Data Grid component

This section details the new features and enhancements made to Data Grid components.

New way to group data in data grids

Data Grid components can show data in hierarchical groupings, so that you can expand groupings to see the data shown at a lower level of detail. This release introduces an alternate way to set up these groupings, to provide more flexibility. Instead of using the existing **Show Hierarchical Column Data** option in the component properties, you can use a HierarchicalGrid data source.

You should use a HierarchicalGrid data source if:

- You need different grouping levels to show different columns. When using a HierarchicalGrid data source, each grouping level has its own DataGridColumns data source to determine the columns that display for that level.
- You need different grouping levels to use different primary tables and/or filters. When using a HierarchicalGrid data source, each grouping level specifies its own primary table and filter, so each level can query a different set of data.
- You need different grouping levels to use different grid options, such as enabling drilling or the ability to select rows. When using a HierarchicalGrid data source, each grouping level has its own defined set of grid options that apply only to that level.

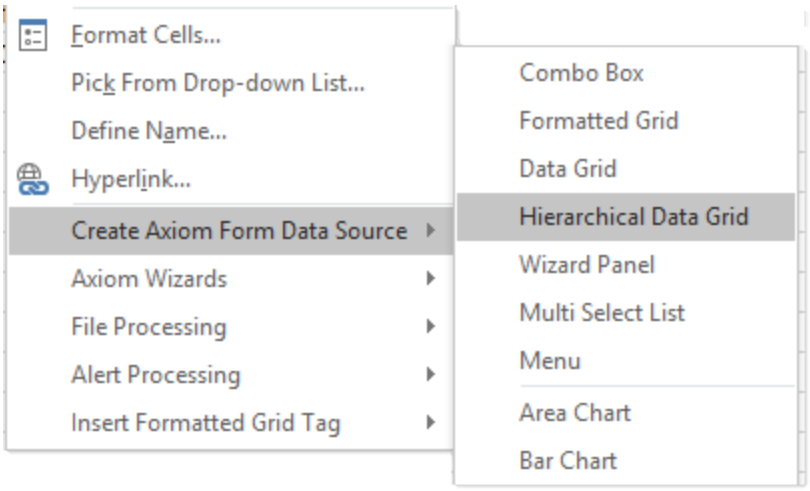
The basic setup steps to use a HierarchicalGrid data source are as follows:

- Create a DataGridColumns data source for each grouping level. This is the same data source that is normally used with Data Grid components.
- Create a HierarchicalGrid data source that lists each grouping level and its corresponding DataGridColumns data source.
- Configure the Data Grid component to use the HierarchicalGrid data source instead of a DataGridColumns data source.

NOTE: The HierarchicalGrid data source is only available for Data Grid components in Axiom forms.

Defining a HierarchicalGrid data source

The HierarchicalGrid data source defines each grouping level that you want to show in the grid. To create a HierarchicalGrid data source, use the **Create Axiom Form Data Source** wizard on the right-click menu:



New data source on the right-click menu

For each grouping level, you specify the following:

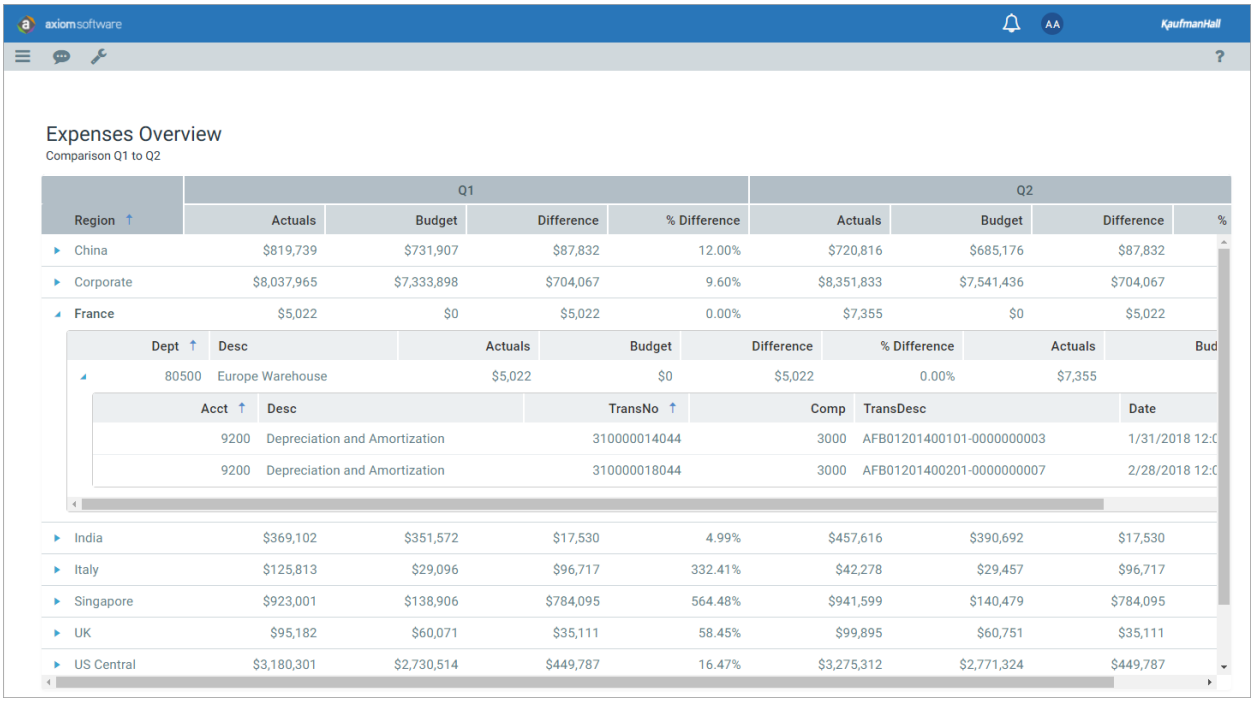
- The DataGridColumns data source that defines the columns for the grouping level
- The primary table for the grouping level and an optional filter
- The page size for the grouping level
- Whether row selection and auto-submit are enabled for the grouping level
- Whether drilling is enabled for the grouping level as well as the drill settings

The following screenshot shows an example HierarchicalGrid data source to define three grouping levels:

	A	B	C	D	E	F	G	H	I	J	K
3											
4			[HierarchicalGrid;RegionalGroup]	[GridColumnsDataSource]	[PrimaryTable]	[Filter]	[PageSize]	[EnableRowSelection]	[AutoSubmit]	[EnableDrilling]	[DrillButtonTooltip]
5			[Grid]	Region	GL2018	Acct.AcctCat='Expense'					
6			[Grid]	Dept	GL2018	Acct.AcctCat='Expense'	5	TRUE	TRUE		
7			[Grid]	GLDetail	GLDetail2018	Acct.AcctCat='Expense'	10				

Example HierarchicalGrid data source

The resulting data grid looks as follows:

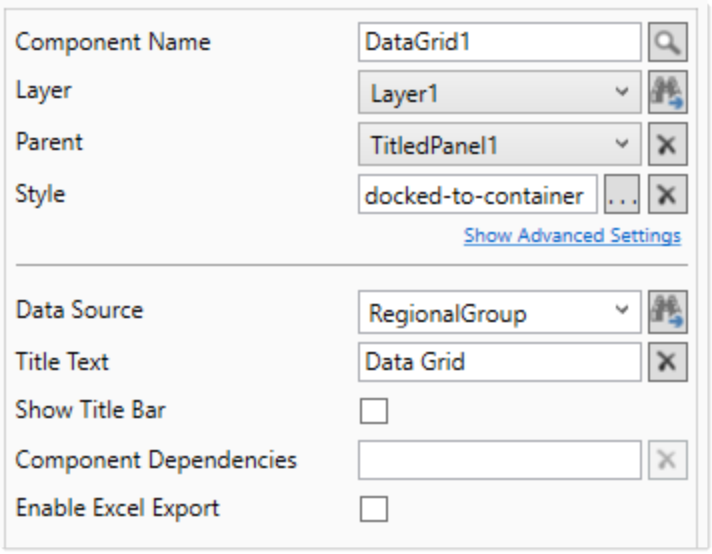


Example data grid using a HierarchicalGrid data source

In this example, the first two grouping levels are showing the same basic columns and querying the same table, just using a different sum by level. The third level queries an different table and shows a completely different set of columns.

Configuring a Data Grid component to use a HierarchicalGrid data source

When you configure the **Data Source** for a Data Grid component, you can select either a regular DataGridColumns data source or a HierarchicalGrid data source. When you select a HierarchicalGrid data source, most of the component-level properties become hidden because they are no longer set at the component level.



Example component properties when a HierarchicalGrid data source is selected

Notice that settings such as Primary Table, Data Filter, and Enable Row Selection no longer display in the component properties. Instead, these properties are set individually for each grouping level, within the HierarchicalGrid data source.

When the Data Grid component is rendered in the form, it reads the grouping levels from the HierarchicalGrid data source. Essentially, each row in the data source is treated as a separate Data Grid component, with its own component properties and DataGridColumns data source. The top-level grid is the first row of the HierarchicalData data source. If you expand an item in this top-level grid, it displays a child grid based on the second row of the HierarchicalData data source, and so on.

When you expand a grouping to see the next grouping level underneath it, the data is filtered to only show the relevant rows for the expanded grouping. For example, if you have levels of Region > Dept, then when you expand region US West to see the Dept grouping level, you only see the departments that belong to that region. This is accomplished by applying a filter to the child grid for the second grouping level, based on the sum by column of the first grouping level. This means that all of the grouping levels in your HierarchicalGrid data source must use compatible primary tables and use sum by columns that can be applied as filters to the lower grouping levels.

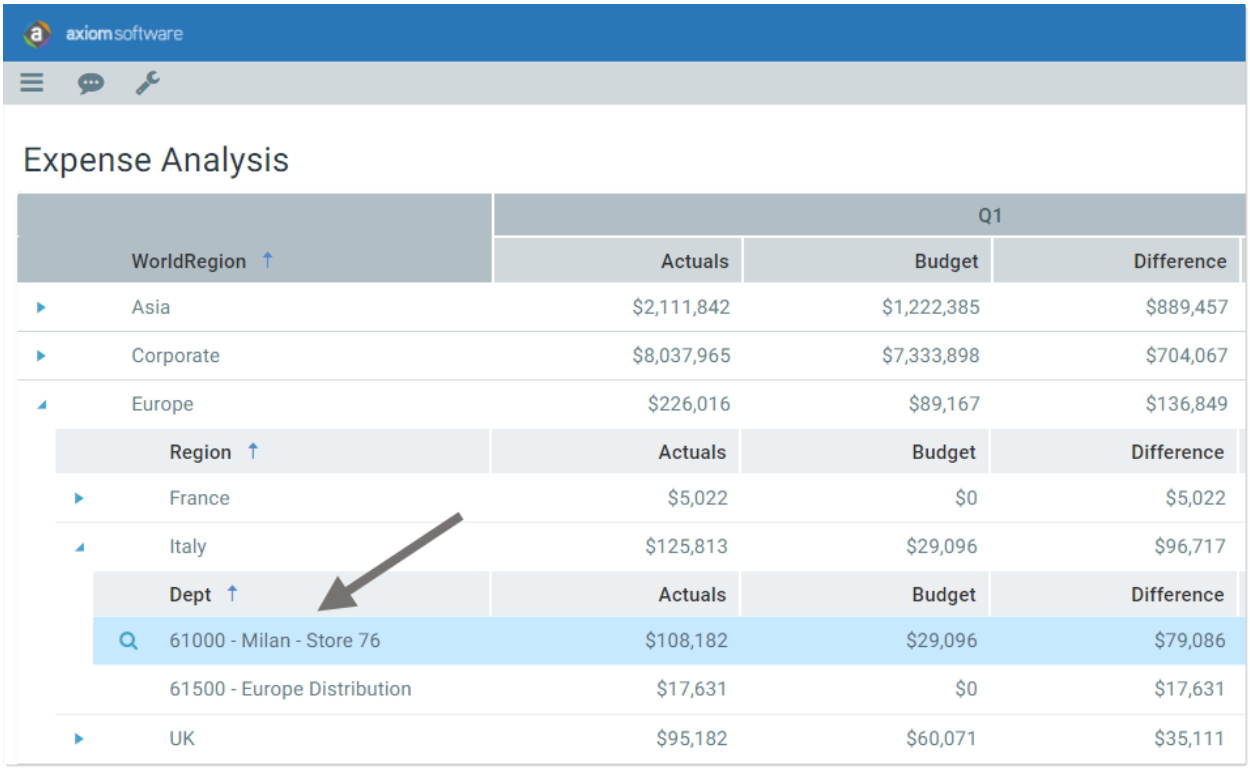
Define display formats for columns in data grids

You can now define a display format for a column in a Data Grid component, so that you can combine the contents of multiple columns together.

The DataGridColumns data source now has a column **DisplayFormat** to define the optional display format. For example, if you have a column for `Dept.Dept` but you want to display the description in the same column as the department code, you can define a display format as follows:

```
{Dept.Dept} - {Dept.Description}
```

When the grid is rendered, the column with this display format displays as follows:



Example column using display format

Use fully qualified Table.Column syntax and place column references in curly brackets. The display format can include additional text and characters, such as the hyphen in the previous example. Any column listed in the display format must be valid in the context of the primary table.

The primary use case for this feature is when using the **Show Hierarchical Column Data** option for Data Grid components. When using this option, each grouping level shows the same columns. This means you cannot include description columns as individual columns in the grid, because they will show at all levels instead of just the level they apply to. Instead, you can define a display format as needed to concatenate the description to the dimension column itself.

NOTE: The ability to define a display format is only available for Data Grid components in Axiom forms.

Refresh variables

This section details the new features and enhancements made to refresh variables.

New slider refresh variables

The new slider refresh variables can be used to prompt users to select a value within a range of available values, using a slider handle. The selected value can then be used to impact the report in some way. The following variable types are available:

- **Slider**: Select a single value within a defined range of values.
- **RangeSlider**: Select a minimum and maximum value within a defined range of values.

Variable setup

The Slider and RangeSlider refresh variables are defined like other refresh variables in the RefreshVariables data source.

- The **VariableType** property must be set to `Slider` or `RangeSlider`.
- New properties of **MinValue** and **MaxValue** are available to define the minimum and maximum values for the range.
- A new property of **StepFrequency** is available to define the step values within the range. The slider handle snaps to step values, so the step frequency determines which values the user can select within the range. For example, if the minimum value is 0 and the maximum value is 100, and the step frequency is 10, then the user can select 0, 10, 20, 30, and so on.
- The existing property of **DisplayFormat** can be used to define a display format for the step values, using an Excel number format string. Only very basic numeric formats are supported, such as to specify decimal places, currency symbols, and percentage signs. For example, you can specify 0.0 to indicate a single decimal place. The number format must be preceded by an apostrophe in the cell so that it is read as a string, or the cell format must be set to Text.
- The **IsRequired** property does not apply to slider variables. Technically, there is no way to clear the slider or force a selection, because the slider handle is always on a value within the range. When a user applies the variable, the current value is written back as the selected value, even if the user has not interacted with the slider.

When inserting a new RefreshVariables data source into a file, the data source now includes the new property columns for the slider variables. If you have an existing data source, you can manually add the columns or you can use the Data Source Assistant.

For more information on defining a slider variable, see the following topics in Axiom Software Help: *Slider refresh variable* (AX1533) and *RangeSlider refresh variable* (AX1685).

The slider refresh variables can be used in the Desktop Client and the Web Client, for spreadsheet Axiom files and Axiom forms. Slider refresh variables are not currently available in web reports.

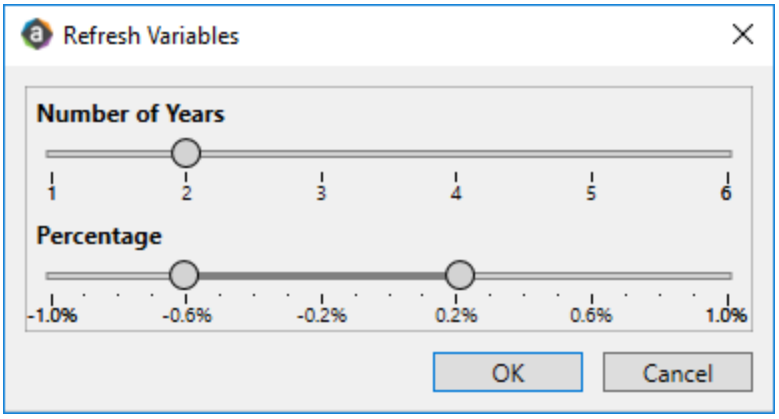
Variable behavior

The following screenshot shows example slider variables defined in the RefreshVariables data source.

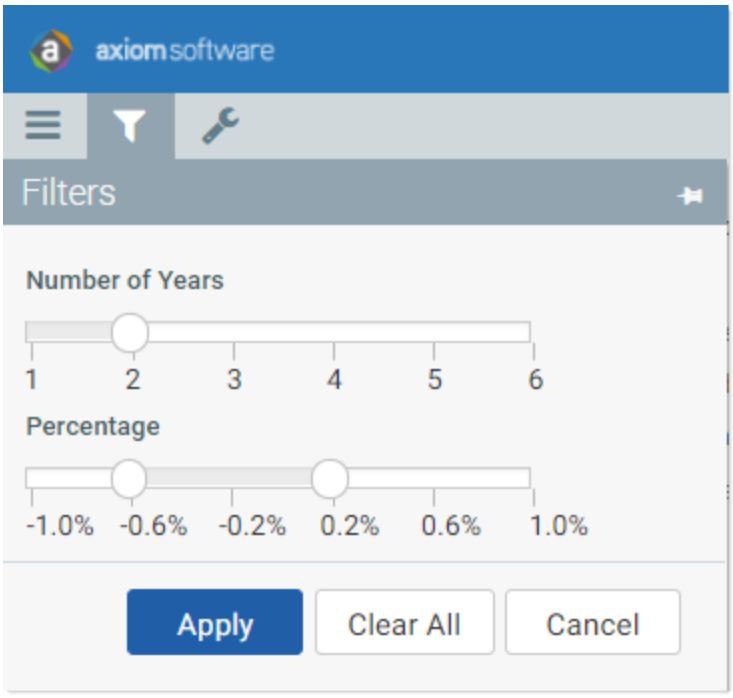
	C	D	E	F	G	H	I	J	K	L	M
20											
21		[RefreshVariables]	[Name]	[DisplayName]	[VariableType]	[IsEnabled]	[SelectedValue]	[MinValue]	[MaxValue]	[DisplayFormat]	[StepFrequency]
22		[Variable]	SliderNum	Number of Year	Slider	True		1	6	0	1
23		[Variable]	SliderPercent	Percentage	RangeSlider	True		-1.0%	1.0%	0.0%	0.10%

Example slider variables in RefreshVariables data source

When you refresh the file in the Desktop Client, or open the Filters panel in the Web Client, the variables look as follows:



Example slider variables in Desktop Client



Example slider variables in Web Client

Up to six major tick marks are labeled with step values on the slider. Other available step values display with minor tick marks in the Desktop Client, but not in the Web Client. As the user moves the slider handle, the available step values display in a tooltip so that the user can see the values between the major tick marks.

For the Slider variable, you can slide the handle to the desired single value. That single value is written back to the RefreshVariables data source as the selected value.

For the RangeSlider variable, you can slide each handle to the desired minimum and maximum value. The range of values is written back to the RefreshVariables data source as the selected value, using the following syntax: `MinValue;MaxValue`.

	C	D	E	F	G	H	I	J	K	L	M
20											
21		[RefreshVariables]	[Name]	[DisplayName]	[VariableType]	[IsEnabled]	[SelectedValue]	[MinValue]	[MaxValue]	[DisplayFormat]	[StepFrequency]
22		[Variable]	SliderNum	Number of Year	Slider	True	2	1	6	0	1
23		[Variable]	SliderPercent	Percentage	RangeSlider	True	-0.006;0.002	-1.0%	1.0%	0.0%	0.10%

Example selected values for slider variables

Behavior change when displaying lists of column values

Grid, ComboBox, and RadioButton refresh variables have been enhanced to better determine the list of values when using a multi-level lookup to specify the value column.

In previous releases, if you specified the value column using a multi-level lookup, the list of values would be sourced from the lookup table instead of the starting table. For example, if you specified

`GL2018.Dept.VP`, the list would display all VPs in the lookup Dept table, instead of just the VPs from the departments listed in the GL2018 starting table.

Going forward, the list is now filtered by the starting table, which is the expected behavior for multi-level lookups. This provides you with the ability to either display all values in the lookup table (`Dept.VP`) or to display only the values used in a particular table (`GL2018.Dept.VP`).

Additionally, these refresh variables were optimized to always include the description column by default if the value column is a key column (and no additional columns are specified). In previous releases, specifying `GL2018.Dept` would not include the description column from the Dept table, but now it will.

As part of this change, optimizations were made to the process used to read additional columns and filters for these features. In the vast majority of cases, existing entries will continue to work as expected. In rare cases, you may have an additional column or filter that uses invalid syntax and now must be corrected to avoid causing an error.

Upgrade behavior

When you upgrade, the affected refresh variables now use the new behavior. In most cases, if you notice this behavior change in an Axiom file, the effect will be positive. Lists that were previously not filtered as expected will be, and description columns will display. In rare cases you may need to correct a value column that was specified incorrectly, or correct invalid syntax in additional columns or a filter. If you know you have files that use these features heavily, you can review them after upgrade to verify the contents of the list.

Additional refresh variable enhancements

When using a multi-select Grid refresh variable in the Desktop Client, the "simple view" dialog now includes options to select all visible items or clear all selections. The simple view dialog is used when only the value column (and its description, if applicable) is displayed to users.

Plan file processes

This section details the new features and enhancements made to plan file processes.

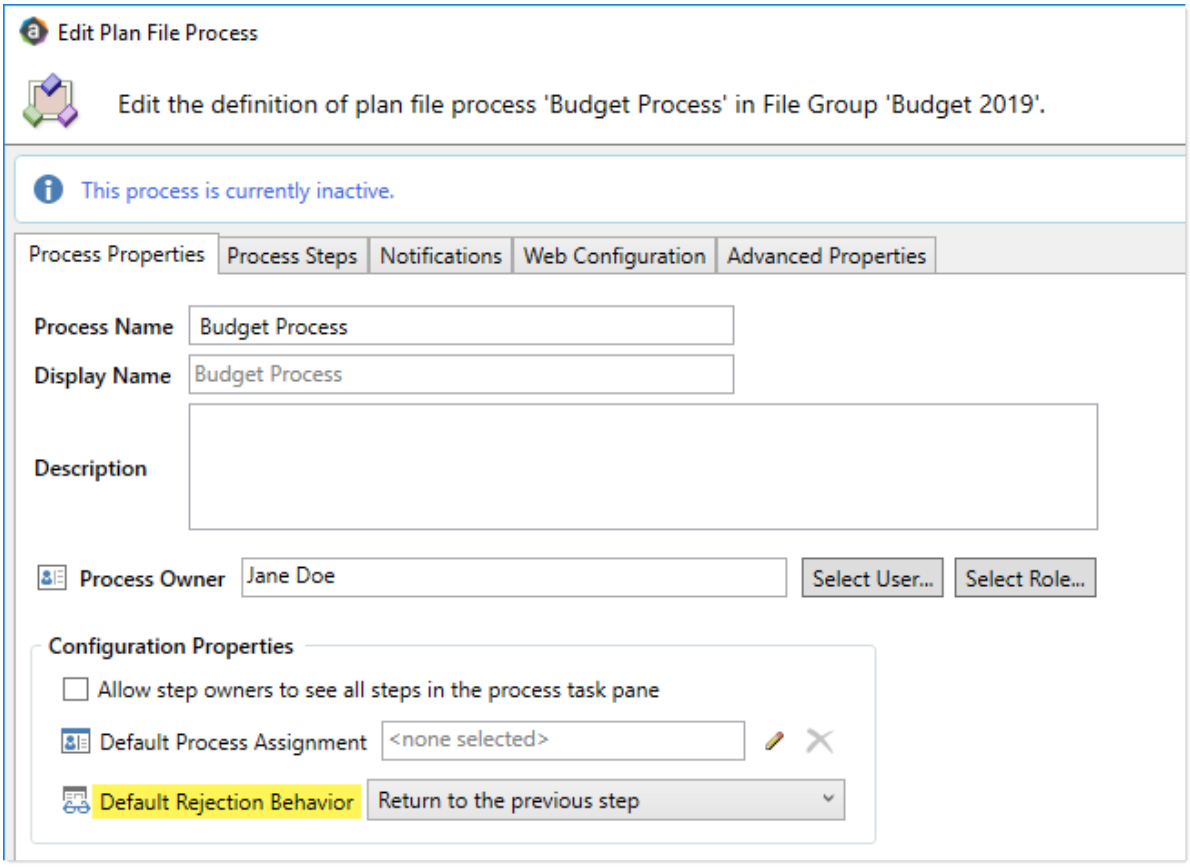
Configurable rejection behavior

You can now configure the rejection behavior for approval steps in plan file processes, to determine which previous step the plan files are returned to if the step is rejected. The rejection behavior options are:

- Return to previous step
- Return to previous edit step (bypassing any approval steps in between)
- Return to a specific step

The rejection behavior can be configured at the process level and at the step level. The process-level behavior applies to all approval steps unless it is overridden at the step level.

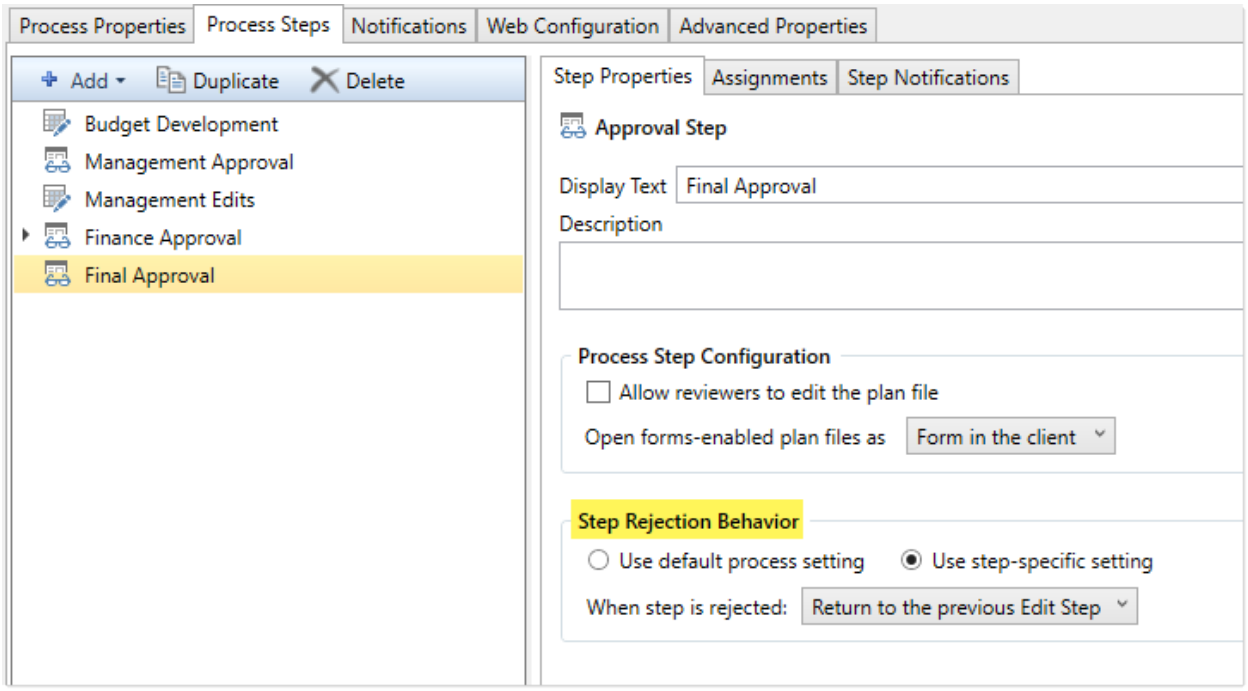
The default rejection behavior for the process is specified on the **Process Properties** tab of the plan file process definition, in the **Configuration Properties** section.



Default rejection behavior for the process

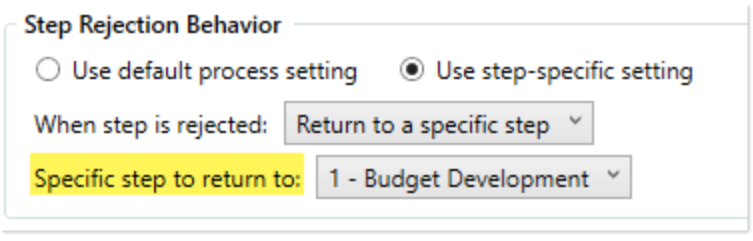
At the process level, you can choose to return to the previous step or to return to the previous edit step. Returning to a specific step is not available at the process level.

If you need to override the default rejection behavior for a particular step, you can do this in the **Step Properties**. This applies to Approval Process Steps and to Multiple Approvals Steps. If an Approval Process Step is a sub-step of a Multiple Approvals Step, then it uses the behavior configured on the parent step.



Step-specific rejection behavior

At the step level, you can choose to return to the previous step, or the previous edit step, or a specific step. For example, you can be on the final step of the process and choose to reject the plan file all the way back to step 1 if desired.



Example step configured to reject back to a specific step

Upgrade behavior

When you upgrade, all of your existing plan file processes are configured to use the **Return to the previous step** behavior, which is the behavior used in previous versions. The rejection behavior will be

the same before and after the upgrade. If you want to use a different rejection behavior, you must edit the process definition as needed.

Additional plan file process enhancements

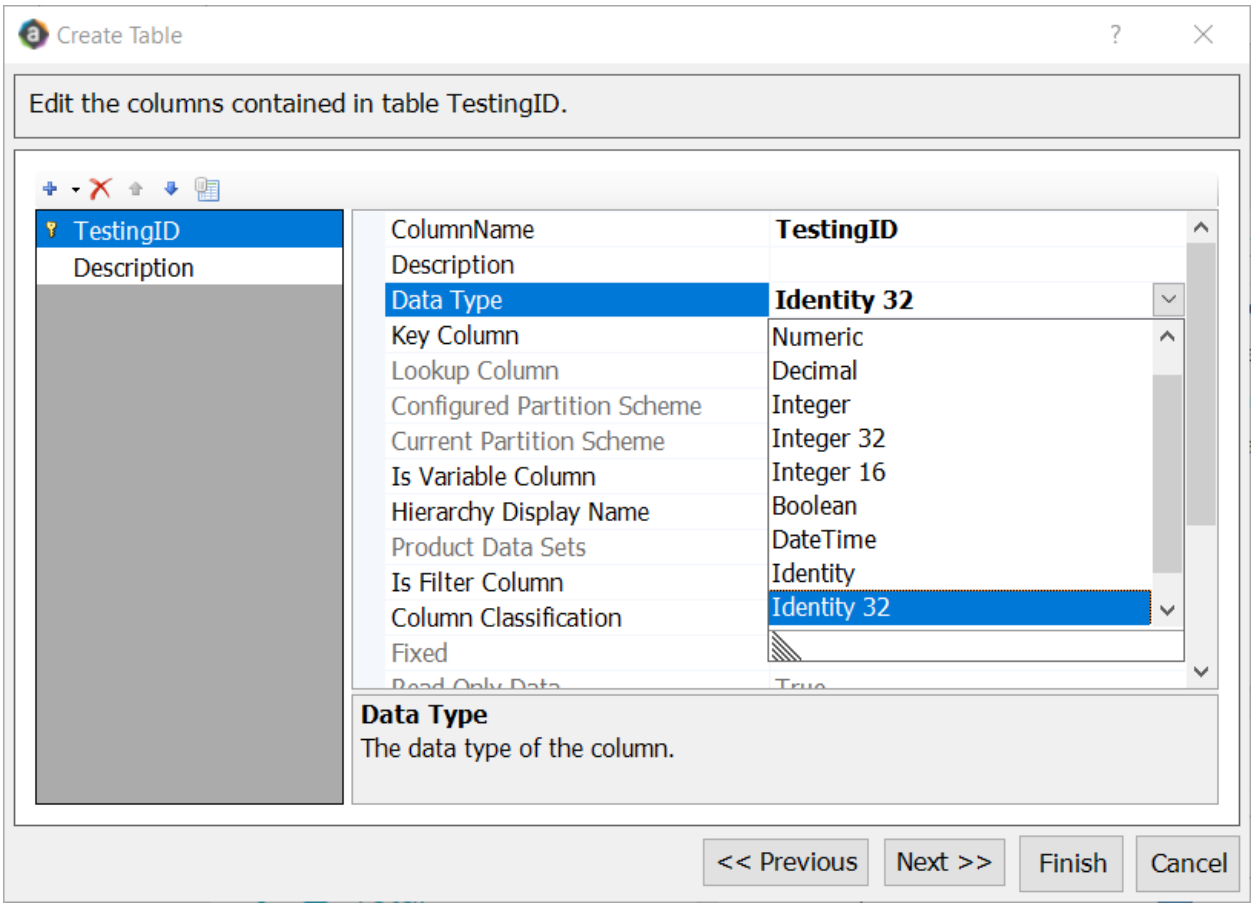
A stronger warning is now displayed when stopping a plan file process, to make it clear that all plan files will be stopped with no way to restore the current status of the process.

Tables

A few minor enhancements were made to table management features.

New Identity 32 data type for table columns

A new data type, **Identity 32**, is now available to create identity columns in tables. This data type takes up less storage space in the database than the current Identity data type, and is recommended to be used as the default identity data type going forward. Additionally, the Identity 32 data type can be used as a lookup column for Integer 32 columns.



Delete tables via Save Type 4

Save Type 4 to Axiom.Tables now supports using the `[Delete]` tag to delete a table as part of the process. This feature should be used with caution because deleting a table cannot be undone. Only administrators and users with the **Administer Tables** permission can execute a Save Type 4 to Axiom.Tables.

When executing the save interactively in a spreadsheet file, a warning message prompts you to confirm that you want to delete the table(s). If the save is executed in an Axiom form or via Scheduler, the tagged tables are simply deleted.

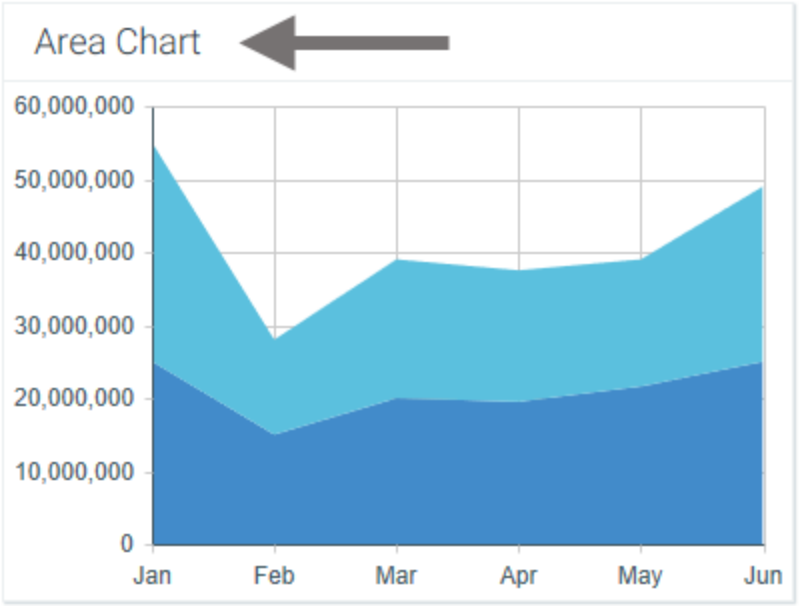
Additional enhancements

Updated title text color for Axiom2018 skin

The following items in the Axiom2018 skin have been updated to use a dark gray color (style: S1) instead of blue:

- Title bar text (for any component that uses a title bar)
- Title styles for Label components (**page-title** and **page-subtitle**)

This change applies automatically to any Axiom form that uses the Axiom2018 skin. Legacy skins are unaffected and continue to use their respective title colors.



Title bar text with updated color

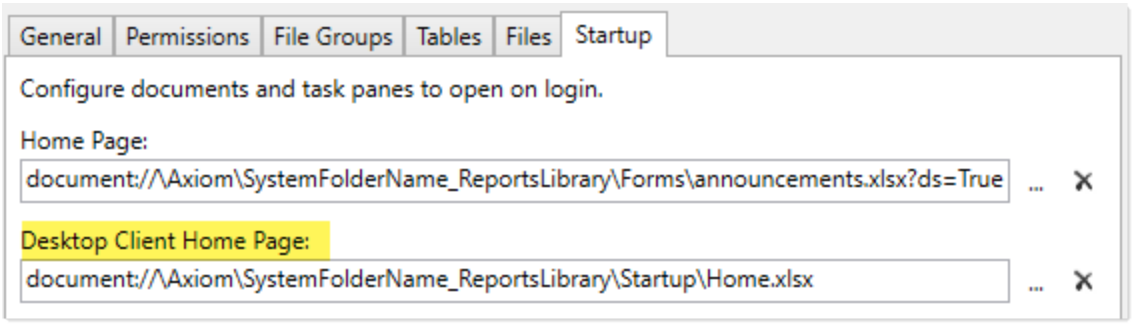
For title bars, this is now the standard title text color in the Axiom2018 skin and it cannot be changed. For Label components, if you do not want the new color you can opt to remove the title styles and instead manually apply color and size styles.

Assign separate home pages for different Axiom Software clients

You can now optionally assign different home pages for the Web Client versus the Desktop Client (Excel or Windows). On the **Startup** tab of security, users and roles now have two available home page assignments:

- **Home Page**: This existing assignment remains and is used by all clients unless it is overridden for the Desktop Client.
- **Desktop Client Home Page**: This new assignment can be used to specify a home page for the Desktop Client only.

For example, you can now specify a spreadsheet report as the Desktop Client Home Page, and a web-enabled report as the Home Page (to be used by the Web Client).



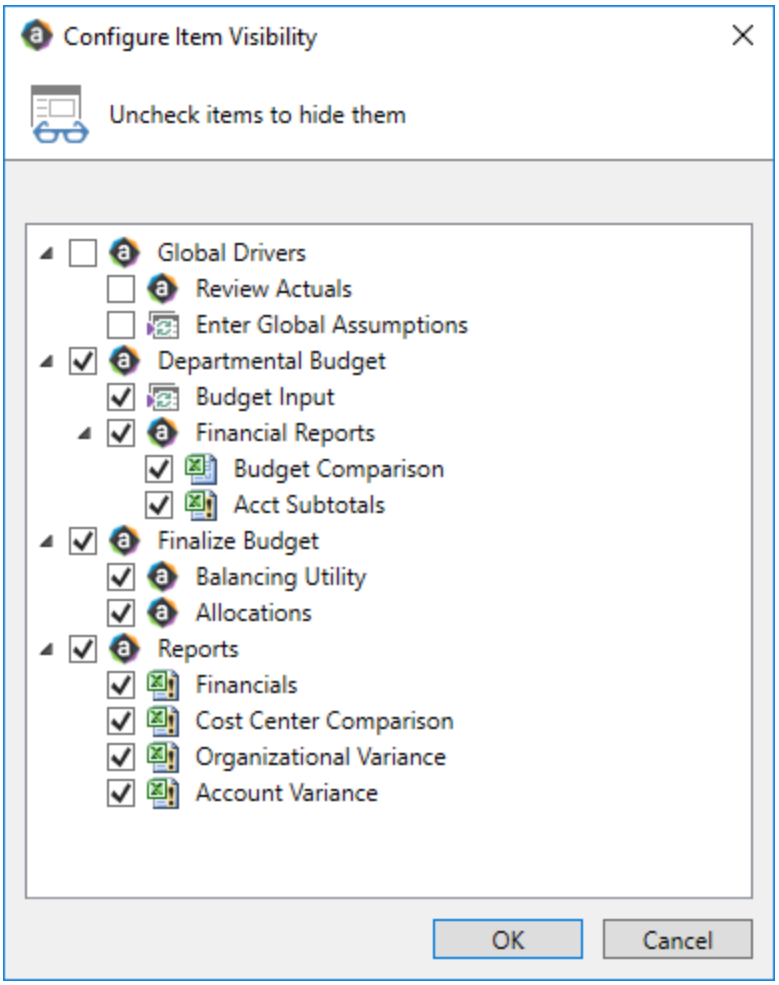
New setting for Desktop Client Home Page

Because the existing Home Page assignment remains and is used by default, there are no upgrade considerations.

Control visibility of items in a custom task pane

You can now configure certain task pane items as not visible, in order to temporarily hide items. For example, you may have an item in a task pane that is only relevant during a certain point of the planning cycle. You can toggle the item hidden when it is not needed, then toggle it visible again when it needs to be used.

To do this, you can right-click the task pane file in the Task Panes Library and select **Configure Item Visibility**. In the **Configure Visibility** dialog, clear the check box for any item that you want to hide.



To make the item visible again, repeat the same process and select the check box.

Read-write access to the task pane file is required to configure visibility. You must restart the client in order to see the effect of all changes.

Behavior change for GetDataElement function and form combo boxes

The behavior changes described for refresh variables also apply to:

- GetDataElement function
- Combo boxes in Axiom forms (ComboBox component and Select tag)

These changes were made to better support multi-level lookups for the value column, and to expand the cases where description columns are automatically displayed.

Updated System Index Maintenance job and other database tasks

The System Index Maintenance job in Scheduler has been updated. This job runs regularly to update table indexes and perform other necessary database maintenance tasks.

The existing job in your system will be replaced with the new job during the upgrade. This means that any existing customizations that you have made to the scheduling rules or the notifications will be lost. Before upgrading, you can go to Scheduler and review your existing System.IndexMaintenance job for customizations.

- **Notifications**: By default, the job is set to send a notification only on error, to `{CurrentUser.EmailAddress}`.
- **Scheduling Rules**: By default, the job is set to run nightly at 5:15.

If the settings in your job are different than this, make a note of the changes so that you can remake the customizations in the new job after upgrading.

Additionally, some database maintenance tasks performed by this job are now available as independent Scheduler tasks, so that they can be added to custom jobs as needed. The following new tasks are available:

- **Rebuild Database Indexes** task
- **Update Database Indexes** task

These tasks are preconfigured versions of the Execute SQL Command task. They can be configured to execute against either the system database or the audit database. The exact SQL code executed by the tasks is automatically generated at runtime, so that the tasks use the most up-to-date SQL code as defined by Axiom Software.

Miscellaneous enhancements

- **Limit query statements:** When using the Filter Wizard to create a limit query statement, you can now select the primary table as the limit table.
- **Filter wizard:** When using the Filter Wizard to build a standard filter criteria statement, validated column selections are no longer automatically elevated to lookup table selections. For example, selecting `GL2018.Dept` is no longer automatically elevated to `Dept.Dept`. If you want to use `Dept.Dept`, you should select it directly. This change applies to the standard Desktop Client version of the dialog. (The Web Client version always worked this way.)
- **Auditing**: Enhancements were made to the session activity table, to improve performance when purging old records.
- **Report Builder templates**: The name of the Axiom System folder for Report Builder templates was changed from Web Reports to Report Builder. The folder path is now: `\Axiom\Axiom System\Document Templates\Report Builder`. Your existing folder will be renamed on upgrade.

Deprecation of File Watcher feature

The File Watcher feature for scheduling imports has been deprecated. The feature will no longer display in new systems, or in existing systems where it is not already configured. The File Watcher feature was intended to "watch" a designated network folder for updated import files, and then automatically trigger the import. However, issues with the Windows file system functionality made it difficult to configure and maintain successfully.

If your system currently has a defined File Watcher folder in Scheduler, then the File Watcher feature will continue to be available in your system after upgrade and will work as before. However, we urge any customers who are still using this feature to migrate to a different solution.