



Creating DeepSee Dashboards

Version 2017.2
2020-06-25

Creating DeepSee Dashboards

Caché Version 2017.2 2020-06-25

Copyright © 2020 InterSystems Corporation

All rights reserved.

InterSystems, InterSystems IRIS, InterSystems Caché, InterSystems Ensemble, and InterSystems HealthShare are registered trademarks of InterSystems Corporation.

All other brand or product names used herein are trademarks or registered trademarks of their respective companies or organizations.

This document contains trade secret and confidential information which is the property of InterSystems Corporation, One Memorial Drive, Cambridge, MA 02142, or its affiliates, and is furnished for the sole purpose of the operation and maintenance of the products of InterSystems Corporation. No part of this publication is to be used for any other purpose, and this publication is not to be reproduced, copied, disclosed, transmitted, stored in a retrieval system or translated into any human or computer language, in any form, by any means, in whole or in part, without the express prior written consent of InterSystems Corporation.

The copying, use and disposition of this document and the software programs described herein is prohibited except to the limited extent set forth in the standard software license agreement(s) of InterSystems Corporation covering such programs and related documentation. InterSystems Corporation makes no representations and warranties concerning such software programs other than those set forth in such standard software license agreement(s). In addition, the liability of InterSystems Corporation for any losses or damages relating to or arising out of the use of such software programs is limited in the manner set forth in such standard software license agreement(s).

THE FOREGOING IS A GENERAL SUMMARY OF THE RESTRICTIONS AND LIMITATIONS IMPOSED BY INTERSYSTEMS CORPORATION ON THE USE OF, AND LIABILITY ARISING FROM, ITS COMPUTER SOFTWARE. FOR COMPLETE INFORMATION REFERENCE SHOULD BE MADE TO THE STANDARD SOFTWARE LICENSE AGREEMENT(S) OF INTERSYSTEMS CORPORATION, COPIES OF WHICH WILL BE MADE AVAILABLE UPON REQUEST.

InterSystems Corporation disclaims responsibility for errors which may appear in this document, and it reserves the right, in its sole discretion and without notice, to make substitutions and modifications in the products and practices described in this document.

For Support questions about any InterSystems products, contact:

InterSystems Worldwide Response Center (WRC)

Tel: +1-617-621-0700

Tel: +44 (0) 844 854 2917

Email: support@InterSystems.com

Table of Contents

About This Book	1
1 Introduction to Dashboards	3
1.1 Displaying DeepSee Dashboards	3
1.2 About Dashboards	5
1.3 Location of Dashboard Definitions	6
2 Creating and Modifying Dashboards	9
2.1 Introduction to the Autosave Feature	9
2.2 Creating a Dashboard	10
2.3 Modifying Dashboard Settings	10
2.4 Adding a Widget to a Dashboard	14
2.5 Overview of Widget Types	15
2.5.1 Widget Types and Subtypes	15
2.5.2 Comparison of Widget Types	16
2.6 Reconfiguring a Widget	16
2.6.1 Toolbar Settings	18
2.6.2 Sidebar Settings	18
2.7 Specifying the Data Source for a Widget	19
2.7.1 Names of Data Sources	20
2.7.2 Kinds of Data Sources	21
2.7.3 Linked Widgets	21
2.8 Saving a Widget to the Widget Catalog	22
2.9 Removing a Widget from a Dashboard	22
2.10 Clearing the Autosave State of a Dashboard	22
2.11 Saving a Dashboard	23
2.12 Copying a Dashboard	23
2.13 Deleting a Dashboard	24
2.13.1 Deleting a Dashboard in the User Portal	24
3 Specifying Common Widget Options	25
3.1 Customizing Print Settings for a Widget	25
3.1.1 Requirements for Printing	26
3.1.2 Special Options for the Title and Subtitle	26
3.1.3 Listing Settings	26
3.1.4 Filter Settings	26
3.2 Adding Widget Properties	27
3.3 Managing Widget Properties	28
3.4 Specifying Numeric Format Strings	29
3.4.1 Selecting a Format String	29
3.4.2 Typing a Numeric Format String	29
3.5 Specifying Table Text Styles	30
3.6 Specifying Chart Text Styles	31
3.7 Specifying Line Styles	32
4 Adding Pivot Table Widgets	33
4.1 Adding a Pivot Table Widget	33
4.2 Specifying Widget Settings for a Pivot Table Widget	34
4.3 Customizing the Columns (When Displaying a KPI)	35

4.4 Displaying a Listing in a Pivot Table Widget	35
4.5 Customizing the Appearance of a Pivot Table	35
4.5.1 Size and Appearance Options	36
4.5.2 Color and Style Options	38
4.6 Removing a Pivot Table Redefinition	40
5 Adding Chart Widgets	41
5.1 Adding a Chart Widget	41
5.2 Customizing the Chart Type	42
5.3 Chart Requirements	42
5.4 Available Chart Types	42
5.4.1 Bar and Column Charts	43
5.4.2 Line and Area Charts	44
5.4.3 Combo Charts	44
5.4.4 High Low Charts	47
5.4.5 Pie Charts	49
5.4.6 X/Y Charts	50
5.4.7 Bubble Charts	52
5.4.8 Time Charts	53
5.4.9 Tree Maps	54
5.4.10 Bullseye Charts	55
5.4.11 Swirl Charts	56
5.5 Displaying an Ensemble Business Metric	58
6 Customizing the Appearance of a Chart Widget	61
6.1 Size and Appearance Options	61
6.2 Title and Label Options	67
6.3 No Data Warning	72
6.4 Color and Style Options	73
6.5 x-Axis Options	78
6.6 y-Axis or Axes Options	81
6.7 Series Detail Options	81
6.8 Chart Legend Options	82
7 Adding Meter Widgets	85
7.1 Adding a Meter Widget	85
7.1.1 Data Sources, Properties, and Meters	86
7.2 Specifying Widget Settings for a Meter	87
7.3 Meter Types	87
7.4 Basic Meter Options	90
7.5 Customizing Speedometers	91
7.6 Customizing Text Meters	96
7.7 Customizing Fuel Gauges	97
7.8 Customizing Traffic Lights	98
7.9 Customizing Light Bars	99
7.10 Customizing Smiley Faces	101
8 Adding Scorecard Widgets	103
8.1 Overview of Scorecards	103
8.1.1 One-Measure Pivot Tables in Scorecards	104
8.2 Adding a Scorecard Widget	104
8.3 Specifying Widget Settings for a Scorecard	105
8.4 Adding Columns	105

8.5 Configuring a Scorecard Column	106
8.6 Specifying a Scorecard Formula	108
8.7 Configuring a Column of Row Numbers	109
8.8 Configuring a Column of Row Labels	110
8.9 Configuring a Column of Values	110
8.10 Configuring Trend Lines and Trend Bars	111
8.11 Configuring Plot Boxes	112
8.12 Configuring Arrows	114
8.13 Configuring Lamps	115
8.14 Customizing the Appearance of a Scorecard Widget	115
8.14.1 Size & Appearance Options	116
8.14.2 Title Options	116
8.14.3 Colors & Style Options	117
9 Adding Other Widgets	123
9.1 Adding a Calendar Widget	123
9.2 Adding a Map Widget	124
9.2.1 Configuring the infoWindow Popup	125
9.3 Adding a Widget That Uses a Portlet	125
9.4 Adding a Controls Widget	126
10 Defining and Using Themes	127
10.1 Overview of Themes	127
10.2 Creating a New Theme Definition	128
10.3 Adding Style Information to a Theme	128
10.4 Applying a Theme	128
10.5 Deleting a Theme	129
11 Adding Widget Controls	131
11.1 Adding a Control	132
11.2 Reconfiguring a Control	134
11.3 Adding an Explicit Filter Control	134
11.3.1 Specifying a Default Value	135
11.4 Adding an OnClick Filter Control	136
11.4.1 Example: Listing with OnClick Filter	136
11.5 Adding a Refresh Control	137
11.6 Adding a Control to Print the Dashboard	138
11.7 Adding a Control to Reload the Dashboard	138
11.8 Adding a Control to Change the Display Type	139
11.9 Adding a Control to Change the Row or Column Sort	139
11.10 Adding a Control to Change the Row or Column Count	140
11.11 Adding a Control to Change the Row or Column Specification	140
11.11.1 Variation: Providing a List of Choices	141
11.12 Adding a Control to Set a Pivot Variable	141
11.13 Adding a Control to Display a Different Pivot Table	142
11.13.1 Variation: Providing a List of Choices	142
11.14 Adding a Control to Display a Listing	143
11.15 Adding a Control to Display the Pivot Analysis Window	143
11.16 Adding a Control to Display Another Dashboard	144
11.16.1 Passing Values	144
11.17 Adding a Control to Display a Web Page	145
11.18 Adding a Control to Perform a Custom Action	145

11.18.1 Specifying the Value of a Row	146
11.19 Hiding a Control	146

About This Book

This book describes, to users and developers, how to create and modify dashboards in DeepSee. It includes the following chapters:

- [Introduction to Dashboards](#)
- [Creating and Modifying Dashboards](#)
- [Specifying Common Widget Options](#)
- [Adding Pivot Table Widgets](#)
- [Adding Chart Widgets](#)
- [Customizing the Appearance of a Chart Widget](#)
- [Adding Meter Widgets](#)
- [Adding Scorecard Widgets](#)
- [Adding Other Widgets](#)
- [Defining and Using Themes](#)
- [Adding Widget Controls](#)

For a detailed outline, see the [table of contents](#).

The other user books for DeepSee are as follows:

- [DeepSee End User Guide](#) describes how to use the DeepSee User Portal and dashboards.
- [Using the DeepSee Analyzer](#) describes how to create and modify pivot tables, as well as use the Analyzer in general.

The following books are primarily for developers who work with DeepSee:

- [Getting Started with DeepSee](#) briefly introduces DeepSee and the tools that it provides.
- [DeepSee Developer Tutorial](#) guides developers through the process of creating a sample model that consists of a cube, subject areas, pivot tables, and dashboards.
- [DeepSee Implementation Guide](#) describes how to implement DeepSee, apart from creating the model.
- [Defining DeepSee Models](#) describes how to define the basic elements used in DeepSee queries: cubes and subject areas. It also describes how to define listing groups.
- [Advanced DeepSee Modeling Guide](#) describes how to use the more advanced and less common DeepSee modeling features: computed dimensions, unstructured data in cubes, compound cubes, cube relationships, term lists, quality measures, KPIs, plugins, and other special options.
- [Using MDX with DeepSee](#) introduces MDX and describes how to write MDX queries manually for use with DeepSee cubes.
- [DeepSee MDX Reference](#) provides reference information on MDX as supported by DeepSee.
- [Tools for Creating DeepSee Web Clients](#) provides information on the DeepSee JavaScript and REST APIs, which you can use to create web clients for your DeepSee applications.

For information on creating Ensemble business metrics, which you can display in these dashboards, see [Developing Ensemble Productions](#).

Also see the article *Using PMML Models in Caché*.

For general information, see the *InterSystems Documentation Guide*.

1

Introduction to Dashboards

This chapter introduces dashboards in DeepSee. It discusses the following topics:

- [How to display DeepSee dashboards](#)
- [Introduction to dashboards](#)
- [Location of dashboard definitions](#)

Be sure to consult the online *InterSystems Supported Platforms* document for this release for information on system requirements for DeepSee.

1.1 Displaying DeepSee Dashboards

To access DeepSee dashboards from the Management Portal, click **DeepSee** and then click **User Portal**.

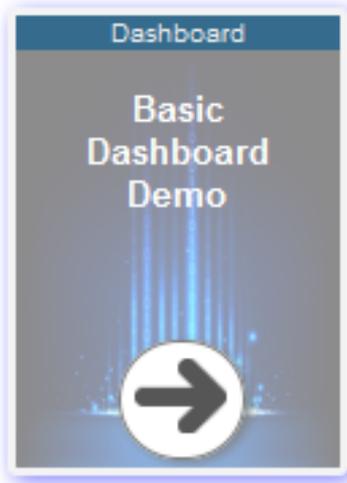
Or use a supported browser (see “Supported Browsers” in the online *InterSystems Supported Platforms* document for this release) and go to the following URL:

```
http://localhost:57772/csp/samples/_DeepSee.UserPortal.Home.zen
```

In either case, the right area then lists the available dashboards in this namespace. This area initially displays the items in Covers View, as follows:



When you hover the cursor over a cover, DeepSee displays an arrow on the cover, as follows:

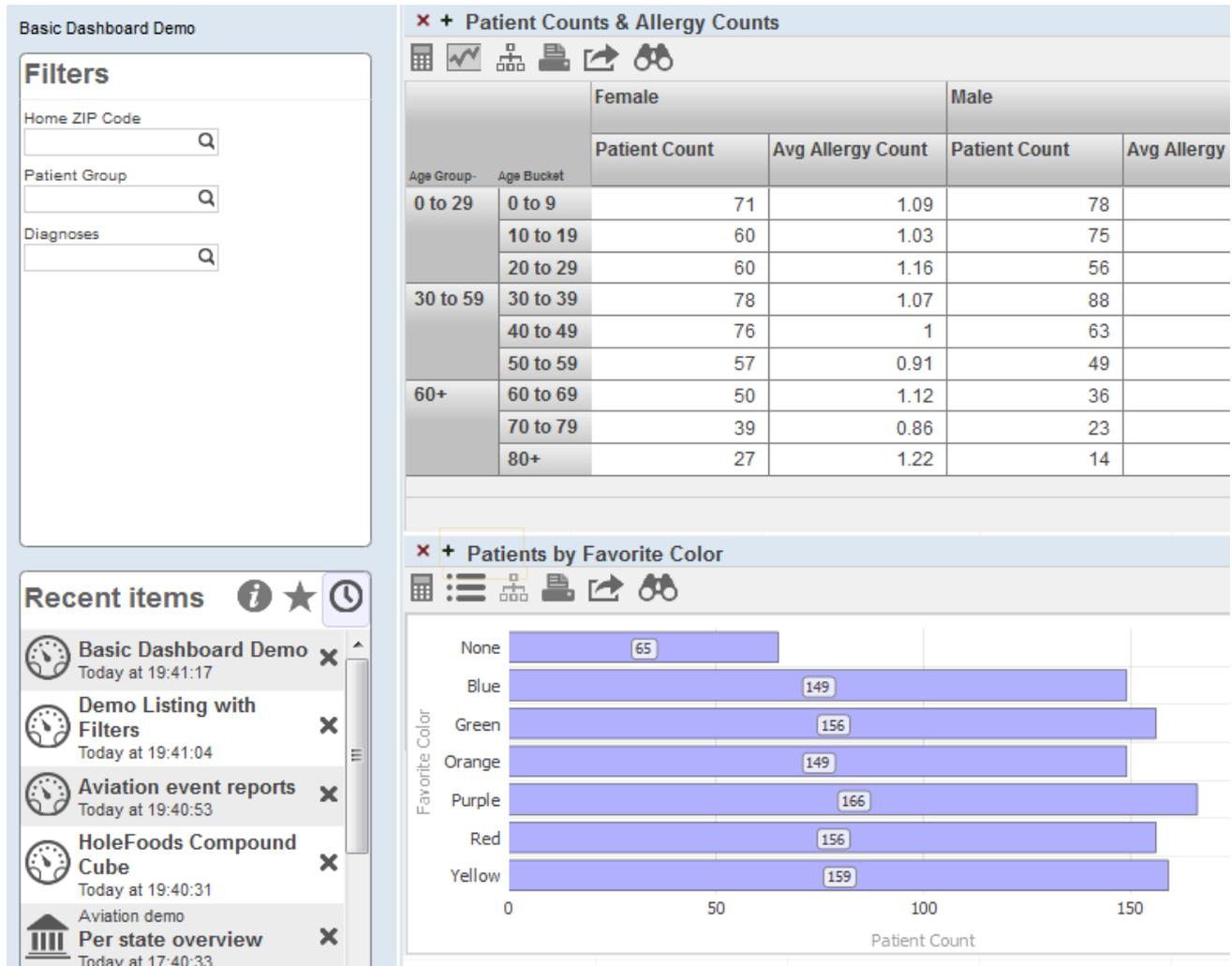


If you click the cover arrow, DeepSee displays the dashboard. If you have permission to do so, you can then edit the dashboard.

For more information on using the User Portal, see the [DeepSee End User Guide](#).

1.2 About Dashboards

A *dashboard* typically includes multiple widgets and controls, as in the following example:

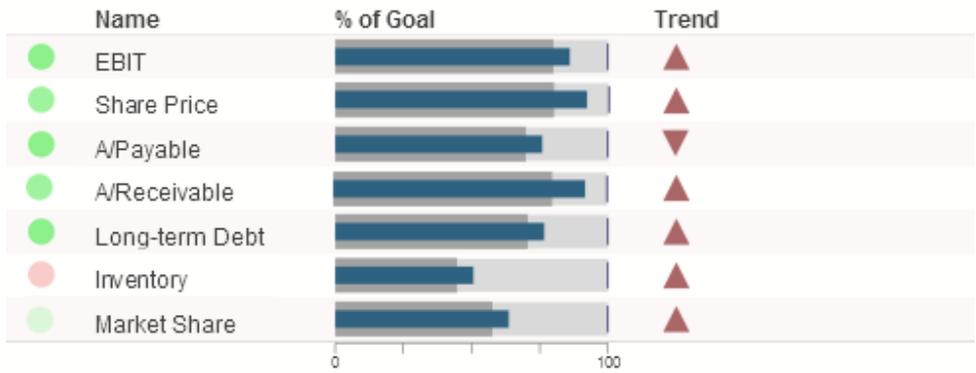


The left area of the dashboard displays two *worklists*. These are configurable areas you can use to display controls (such as the filters shown in the upper worklist) and other useful items (such as the recent items list shown in the lower worklist). This book does not discuss worklists in detail. For information on using worklists, see the [DeepSee End User Guide](#). For information on configuring them, see the [DeepSee Implementation Guide](#).

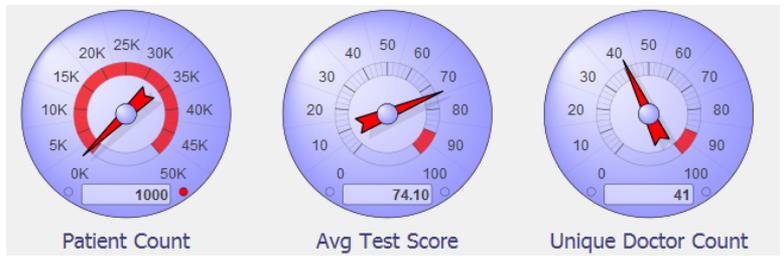
This example dashboard also includes two widgets, which are both *pivot table widgets*. Note that a pivot table widget can display any kind of data source.

Dashboards can include other kinds of widgets, including *meters* and *scorecards*. A scorecard shows multiple rows of data in a tabular format that also includes features such as value-dependent lamps and arrows. For example:

Corporate Benchmarks

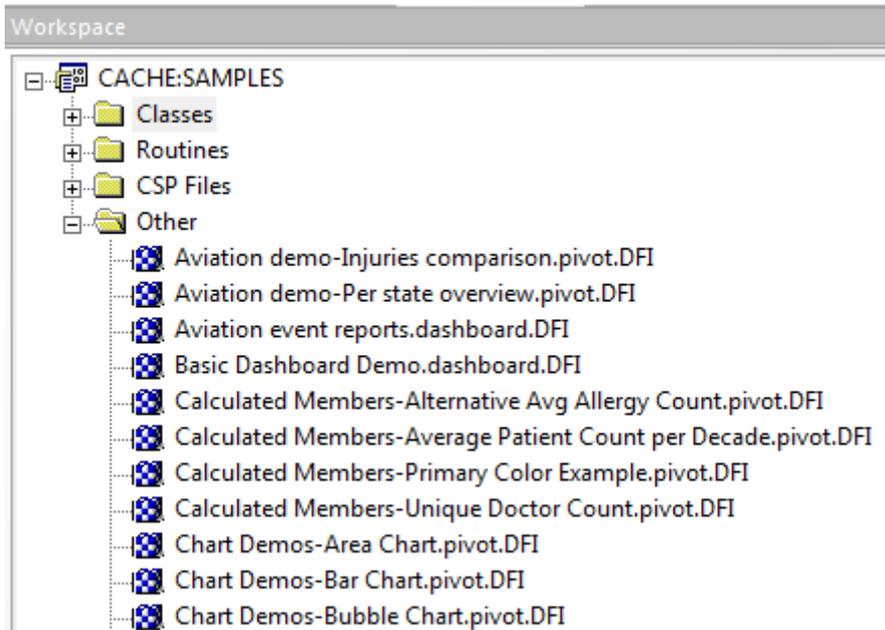


A meter shows one or more values in graphical objects like the following:



1.3 Location of Dashboard Definitions

When you create a dashboard, its definition is visible in the Studio **Workspace** window, within the **Other** folder. This folder lists the dashboard definition as an item with the extension `.dashboard.DFI`. For example:



Note that this folder also lists pivot table definitions and other DeepSee folder items.

These definitions are not class definitions but can be copied and pasted into a special container class for easier deployment. For details, see “[Packaging DeepSee Elements into Classes](#)” in the *DeepSee Implementation Guide*.

2

Creating and Modifying Dashboards

This chapter describes how to create and edit dashboards. It discusses the following topics:

- [The autosave feature](#)
- [How to create a dashboard](#)
- [How to modify dashboard settings](#)
- [How to add a widget to a dashboard](#)
- [Widget types](#)
- [How to reconfigure a widget](#)
- [How to specify the data source for a widget](#)
- [How to save widgets to the widget catalog](#)
- [How to remove widgets from a dashboard](#)
- [How to clear the autosave state of a dashboard](#)
- [How to save a dashboard](#)
- [How to copy a dashboard](#)
- [How to delete a dashboard](#)

For information on customizing covers for dashboards, see the *DeepSee End User Guide*.

2.1 Introduction to the Autosave Feature

DeepSee provides an optional autosave feature, which can automatically save the state of the User Portal for each user, for each dashboard. If the autosave feature is enabled for the User Portal:

- When you access the User Portal, DeepSee displays that user interface as you last saw it.
- When you display a dashboard, DeepSee displays that dashboard as you last saw it.

For information on this option, see “[Specifying Basic Settings](#)” in the *DeepSee Implementation Guide*. Note that there is a separate autosave feature for the Analyzer.

2.2 Creating a Dashboard

To create a dashboard, use the following overall process:

1. Log in to the User Portal in the desired namespace, as described [earlier](#).
2. Click **Menu > New Dashboard**.

Or click the Add button , which is above the list of dashboards, as follows:



Then click **Add Dashboard**.

Or click the < button at the left side of the User Portal, click **Add**, and then click **Add Dashboard**.

In any case, the system then displays a dialog box that prompts you for basic information about the new dashboard.

3. For **Folder**, optionally type a folder name or select an existing folder.

Also see “[Preparing for Folder Item Localization](#)” in the *DeepSee Implementation Guide*.

4. For **Dashboard Name**, type the name of the dashboard.

Also see “[Preparing for Folder Item Localization](#)” in the *DeepSee Implementation Guide*.

Note that this name is used (along with **Folder**) as the logical name of the dashboard; DeepSee displays this logical name when you use the **Save As** option, for example.

5. Optionally provide values for the other options as given in the [next section](#).
6. Click **OK**.

The system creates, saves, and displays the dashboard, which is initially empty.

7. To configure the dashboard, use a combination of the following tools:

- The Dashboard Editor. For an introduction, see “[Modifying Dashboard Settings](#),” later in this chapter.
- The **Menu**. For an introduction, see “[Adding a Widget to a Dashboard](#),” later in this chapter.

2.3 Modifying Dashboard Settings

To modify settings of a dashboard or of the widgets in a dashboard:

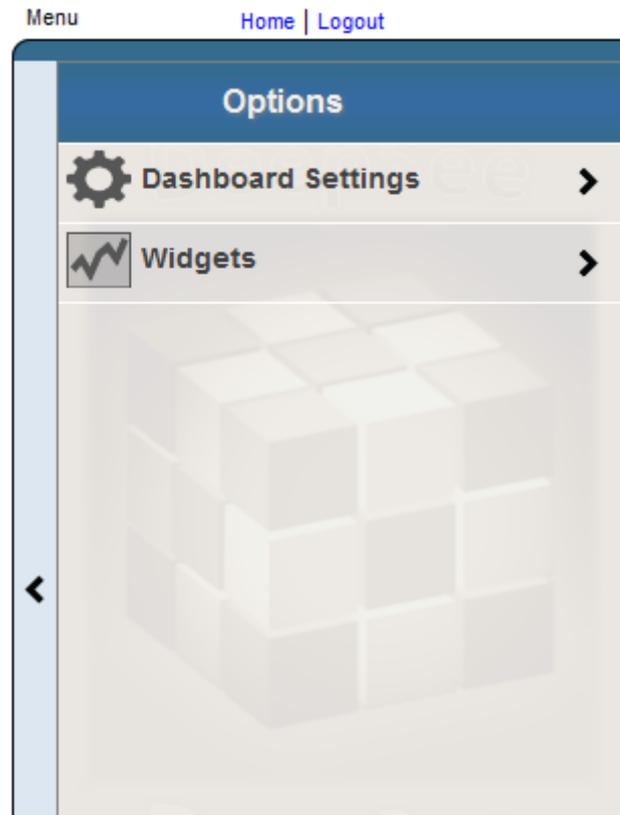
1. Display the dashboard as described in “[Accessing DeepSee Dashboards](#),” earlier in this book.
2. Click the > button to the left of the dashboard. (If this button is not visible, either the dashboard is not editable or you do not have permission to edit it.) The following shows an example:

The screenshot displays a dashboard editor interface with the following components:

- Basic Dashboard Demo**: The main dashboard title.
- Filters**: A section with three search input fields:
 - Home ZIP Code
 - Patient Group
 - Diagnoses
- Recent items**: A list of recently viewed dashboards:
 - Basic Dashboard Demo (Today at 16:30:43)
 - D2DASH/Charts
 - XYChart (Today at 16:24:05)
- Patient Counts & Allergy Counts**: A data widget showing a table for Female patients.

Age Group- Age Bucket		Patient Count	Avg Allergy
0 to 29	0 to 9	69	
	10 to 19	64	
	20 to 29	53	
30 to 59	30 to 39	80	
	40 to 49	84	
	50 to 59	54	
60+	60 to 69	51	
	70 to 79	23	
	80+	20	
- Patients by Favorite Color**: A data widget showing a bar chart with three categories:
 - None
 - Blue
 - Green

DeepSee then expands the Dashboard Editor as follows:



To hide the Dashboard Editor, click the < button.

3. Click **Dashboard Settings**.
4. Specify any of the following options as needed:
 - **Locked** — Select this option to temporarily prevent changes to this dashboard. If you select this option, you cannot edit the dashboard again unless you first clear the **Locked** option again.
 - **Public** — Select this option to control whether this dashboard is displayed in the User Portal main area.

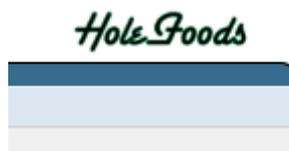
Important: This option does not control access to the dashboard. Access is controlled instead by the **Owner** and **Resource** options, discussed later in this list.

Note that even if the dashboard is not marked as public, you can access it via **Main > Open**. Also, you can find it in the User Portal by using the Find option. See the [DeepSee End User Guide](#).

- **Page Settings > Company Name** — Specify a name to display in the upper right corner. For example:



- **Page Settings > Company Logo** — Select an image to display in the upper right corner. For example:



If you specify this option, **Page Settings > Company Name** is ignored.

- **Title** — Specify the title of the dashboard.

If you specify a title, the main area of the User Portal displays the title of the dashboard. Otherwise, it displays the name of the dashboard.

Also see “[Preparing for Folder Item Localization](#)” in the *DeepSee Implementation Guide*.

- **Owner** — Specify the Caché user who owns this dashboard. If a dashboard has an owner, then only the owner can specify the **Resource** option for the dashboard; see the next item.
- **Resource** — Specify the Caché resource that is used to control access to this dashboard. See “[Visibility of Folders and Folder Items](#),” in the chapter “Using the Portal Features” in the *DeepSee End User Guide*.
- **Category** — Specify the category to which this dashboard belongs. The User Portal uses this to group items in the [Covers View](#). See the *DeepSee End User Guide*.
- **Description** — Briefly describe the dashboard.
- **Keywords** — List any keywords or phrases to help you find this dashboard later. Specify a comma-separated list of keywords or phrases.

These keywords are displayed in the User Portal.

- **Resize** — Select this option to control whether users can move and resize the widgets on the dashboard.
- **Modify** — Select this option to control whether users can modify the widgets on the dashboard.
- **Title Bars** — Use this submenu to control the display of title bars on the widgets on this dashboard. You can also customize their appearance. This submenu has the following options:
 - **Title Bars** — Select **ON** or **OFF**.
 - **Title Bar Color** — Select a color to use for the title bar, to be used by default.
 - **Title Bar Opacity** — Specify the opacity of the title bar, to be used by default. The value 1 means 100%.
 - **Text Color** — Select a color for the text of the title bar, to be used by default.
 - **Selected Color** — Select a color to use for the title bar, to be used when the widget is selected.
 - **Selected Opacity** — Specify the opacity of the title bar, to be used when the widget is selected. The value 1 means 100%.
 - **Selected Text Color** — Select a color for the text of the title bar, to be used when the widget is selected.
 - **Font** — Select the typeface to use for the text in the title bar.
- **Widget borders** — Use this option to hide or display borders on all widgets.
- **Work Lists** — Specify the number of worklists in the dashboard. Note the following defaults:
 - In a two-worklist dashboard, by default, the top worklist is the **Filters** box, and the bottom worklist is the **Favorites/Recent** items box.
 - In a one-worklist dashboard, by default, the worklist is the **Filters** box.
- **Background** — Use this submenu to specify the background of the dashboard. This submenu has the following options:
 - **Background Color** — Select the background color. Note that this color is used only if
 - **Background Image** — Select an optional background image. If you choose an image, the **Background Color** option is ignored.

- **Background Opacity** — Specify the opacity of the background color or background image (whichever is used). The value 1 means 100%.
- **Scalable Grid, Grid Rows, and Grid Columns** — Use these options to specify the position and size of all widgets relative to a resizable grid.

If you enable the **Scalable Grid** option, when you move a widget, the upper left corner of widget aligns to the nearest grid point. Similarly, when you resize a widget, the lower right corner aligns to the nearest grid point. Use the **Grid Rows** and **Grid Columns** options to specify the grid. Note that the grid itself is not displayed.

If you resize the dashboard, all the widgets are automatically resized. **Scalable Grid** is the only option that supports this resizing behavior.
- **Snap Into Place** — Use this option to control how widgets are placed on the dashboard. If you clear this option, you can position widgets freely anywhere on the dashboard, even overlapping each other. If you select this option, the system automatically aligns the widgets next to each other or below each other.

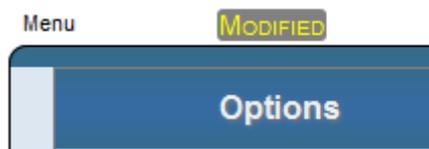
This option is available only if you clear the **Scalable Grid** option.

For details on the other parts of the Dashboard Editor, see the following chapters:

- [Adding Pivot Table Widgets](#)
- [Adding Chart Widgets](#)
- [Adding Meter Widgets](#)
- [Adding Scorecard Widgets](#)
- [Adding Other Widgets](#)
- [Adding Widget Controls](#)

In some cases, the changes are saved automatically.

In other cases, DeepSee does not save the changes automatically. In these cases, it displays an indicator like the following as a reminder:



In such cases, save the dashboard as described in “[Saving a Dashboard](#).”

2.4 Adding a Widget to a Dashboard

To add a widget to a dashboard, use the following general procedure:

1. Display the dashboard as described in “[Displaying DeepSee Dashboards](#),” earlier in this book.
2. If you make unrelated changes to the dashboard, and if you want to keep those changes, save them.

When you add a widget, the Dashboard Designer saves the dashboard with the newly added widget but discards other unsaved changes.

3. Do one of the following:

- Click **Menu > Add New Widget...** — Use this option to add a new widget that does not include any configuration details.
- Click **Menu > Add Widget from Catalog...** — Use this option to reuse a configured widget that you previously saved to the widget catalog.

You can reconfigure this widget, if needed; you can change any detail except for the widget [type](#). However, you generally use this option when you have saved widgets that are already configured in common ways for your needs.

- Display the [Dashboard Editor](#), click **Widgets**, and then click the plus sign button.

DeepSee then displays a wizard where you select a widget and specify basic information for it.

4. Navigate to a widget [type](#) and (if applicable) subtype.
5. (For most types of widgets) Optionally specify of the following items:
 - **Data Source** — Select the data source for this widget. See “[Names of Data Sources](#)” and “[Kinds of Data Sources](#),” later in this chapter.
 - **Link to** — Select another widget on this dashboard to use as the source. See “[Linked Widgets](#),” later in this chapter.

Or specify a data source later, as described in “[Specifying the Data Source for a Widget](#).”

6. Specify a name for **Widget Name**. Note that each dashboard widget must have a name.
7. Click **OK** to add the widget to your dashboard.
8. Optionally [configure](#) the widget. For information on specific types of widgets, see the following chapters:
 - [Adding Pivot Table Widgets](#)
 - [Adding Chart Widgets](#)
 - [Adding Meter Widgets](#)
 - [Adding Scorecard Widgets](#)
 - [Adding Other Widgets](#)
9. Optionally add controls to the widget. See the chapter “[Adding Widget Controls](#).”

2.5 Overview of Widget Types

This section provides an overview of widget types.

2.5.1 Widget Types and Subtypes

After you add widgets, you can reconfigure them in any way other than modifying their type. Note that you can modify the subtype, if applicable. DeepSee provides the following widget types and subtypes:

- [Pivot table](#) widgets. This type of widget includes the table subtype and many chart subtypes.
Because there are so many chart options, [charts](#) are discussed separately from [tables](#).
- [Meter](#) widgets. This type of widget includes many meter subtypes.

- [Scorecard](#) widgets. This type of widget includes two subtypes: regular and big.
- [Calendar](#) widgets. This type of widget includes one subtype.
- [Map](#) widgets. This type of widget includes one subtype.
- [Portlet](#) widgets. The subtypes include any portlets that you define.
- [Controls](#) widgets. This type of widget includes two subtypes: horizontal and vertical.

2.5.2 Comparison of Widget Types

For reference, the following table compares the key features of widgets, across the different widget types:

Widget Type	Possible Data Sources	Print Option?	Export to Excel Option?	Mini Analyzer Option?	Drilldown Option?
pivot table widget	pivot table, KPI, or Ensemble business metric	Yes	Yes	Yes *	Yes *
meter	pivot table, KPI, or Ensemble business metric	Yes	No	Yes *	No
scorecard	pivot table, KPI, or Ensemble business metric	Yes	No	Yes *	No
calendar	none	No	No	No	No
map	pivot table, KPI, Ensemble business metric, or no data source	No	No	Yes *	No
controls widgets	pivot table, KPI, Ensemble business metric, or no data source	No	No	No	No

*This option is available only if the data source is a pivot table.

For a [portlet](#), all these details depend upon the implementation.

2.6 Reconfiguring a Widget

When you reconfigure this widget, you can change any detail except for the widget [type](#). To select and reconfigure a widget on a given dashboard:

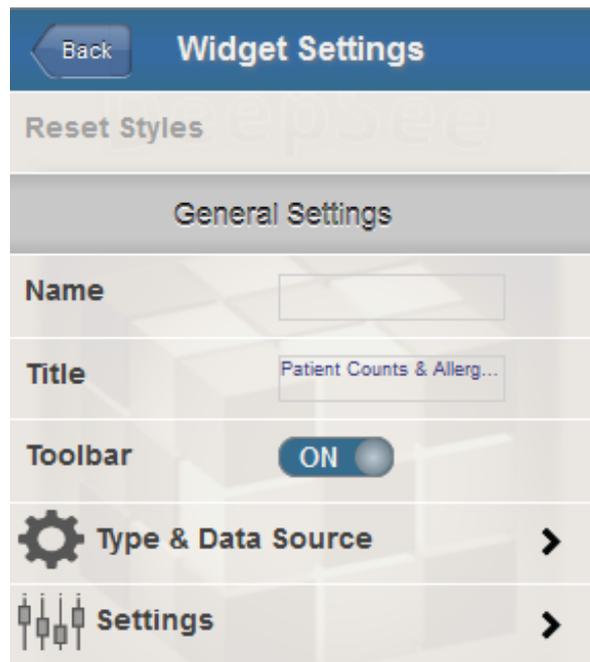
1. Display the dashboard as described in “[Displaying DeepSee Dashboards](#),” earlier in this book.
2. Display the [Dashboard Editor](#).
3. Click **Widgets**.

DeepSee then displays a list of the widgets in the dashboard.



4. Click the name of the widget in this list. This action selects the widget.

The system then displays a list of settings that you can modify. In most cases, the list includes submenus. The following shows a partial example:



5. Optionally specify the following settings, which all widgets have:
 - **Name** — Specify a name for internal use within this dashboard. One widget can refer to other widgets (for example, to filter another widget). For these names, you might want to establish a set of naming conventions, for your convenience.
 - **Title** — Specify any text to display in the title bar of the widget.
Note that the dashboard can be configured to hide all title bars; see “[Modifying Dashboard Settings.](#)”
 - **Toolbar** — Configure the toolbar for this widget. See the [first subsection](#) for details on this submenu.
 - **Sidebar** — Configure optional sidebar text for this widget. See the [second subsection](#) for details on this submenu.

The other submenus list additional items, depending on the kind of widget you select. For information, see the following chapters:

- [Adding Pivot Table Widgets](#)
- [Adding Chart Widgets](#)
- [Adding Meter Widgets](#)

- [Adding Scorecard Widgets](#)
- [Adding Other Widgets](#)
- [Adding Widget Controls](#)

2.6.1 Toolbar Settings

Use this submenu to configure the toolbar for the widget. Specify the following options:

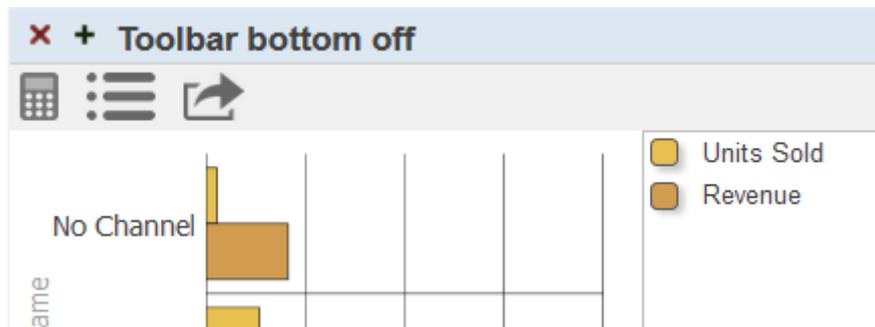
- **Toolbar** — This option enables or disables the toolbar. Select **ON** or **OFF**.

When the toolbar is enabled, DeepSee displays it if the widget is configured with any buttons or any controls that require it (but see the **Show Maximized** option). If the widget has none of these, DeepSee does not display the toolbar.

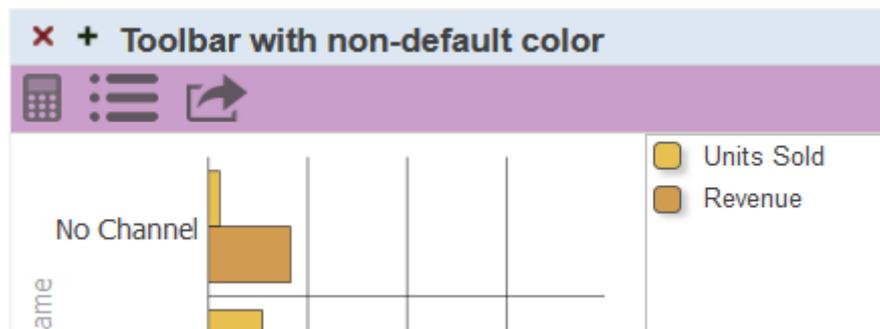
When the toolbar is disabled, DeepSee never displays the toolbar.

InterSystems recommends that you specify **Toolbar** as **ON**.

- **Bottom Border** — This option displays or hides the bottom border of the toolbar. Select **ON** or **OFF**.



- **Show Maximized** — If this option is **ON**, DeepSee displays the toolbar only when the widget is maximized.
- **Toolbar Color** — Select the color to use for the background of the toolbar. Or click X to restore the default color.



- **Toolbar Opacity** — Use this option to specify the opacity of the toolbar. The default is 1 (which is 100%).

2.6.2 Sidebar Settings

Use this submenu to configure optional sidebar text for the widget, which is displayed in a rectangle on the right side of the widget, as in the following example:

		Female		Male		This is a sidebar. It accepts limited HTML <i>formatting</i> , including breaks to control word wrap. The width of <i>this</i> sidebar is defined as 25% of the widget width.
Age Group	Age Bucket	Patient Count	Avg Allergy Count	Patient Count	Avg Allergy Count	
0 to 29	0 to 9	71	1.09	78	1.08	
	10 to 19	60	1.03	75	1.16	
	20 to 29	60	1.16	56	0.94	
30 to 59	30 to 39	78	1.07	88	1.02	
	40 to 49	76	1.00	63	1.40	
	50 to 59	57	0.91	49	0.97	
60+	60 to 69	50	1.12	36	0.83	
	70 to 79	39	0.86	23	1.13	
	80+	27	1.22	14	0.86	

Specify the following options:

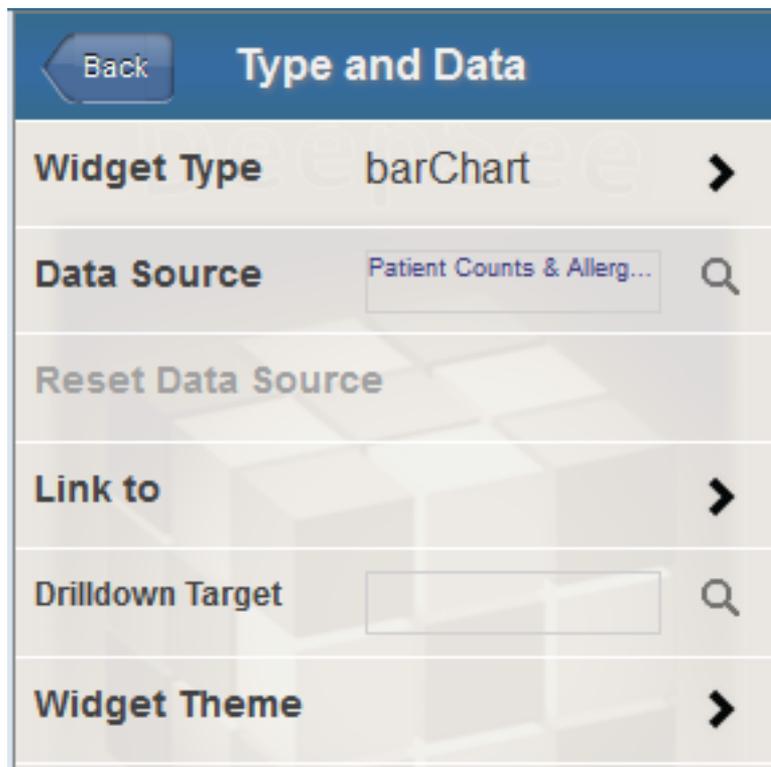
- **Sidebar** — This option enables or disables the sidebar. Select **ON** or **OFF**.
- **Sidebar Content** — Type the text to be displayed in the sidebar. You can include simple HTML markup such as the `<p>` and `</p>` tags, the `
` tag, the `<i>` and `</i>` tags, and the `` and `` tags.
If you enclose text in the `<p>` and `</p>` tags, that text is automatically wrapped as shown in the preceding example.
- **Sidebar Width** — Specify the width of the sidebar. Specify either a number (which is interpreted as the width in pixels) or a number followed by % (which is interpreted as a percentage of the width of the entire widget).

2.7 Specifying the Data Source for a Widget

To specify the data source for a widget:

1. Access the Dashboard Editor and select the widget as described in “[Reconfiguring a Widget](#).”
2. Click **Type & Data Source**.

DeepSee then displays the **Type and Data** submenu.



3. Specify one of the following options:

- **Data Source** — Select the data source for this widget. See “[Names of Data Sources](#)” and “[Kinds of Data Sources](#).”
- **Link to** — Select another widget on this dashboard to use as the source. See “[Linked Widgets](#).”
- **Drilldown Target** — This option enables the user to double-click a cell and access another dashboard. The newly opened dashboard is filtered by the context in which the user double-clicked. For example, if the user double-clicks in a cell that shows data for a specific city, the newly opened dashboard is filtered to that city.

To use this option, click the Search button , click a dashboard, and click **OK**.

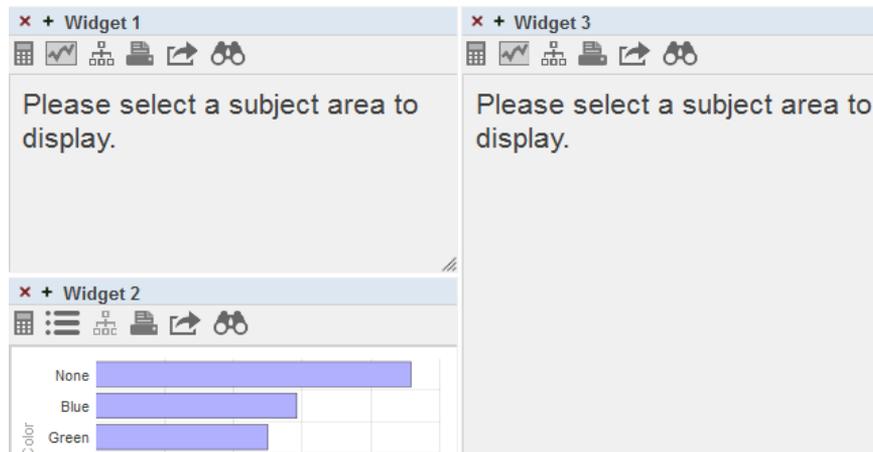
Note that the **Widget Type** option enables you to choose a different widget [subtype](#); it does not let you choose a type. See “[Widget Types and Subtypes](#),” earlier in this chapter.

The following subsections provide details.

2.7.1 Names of Data Sources

When you specify the data source for a widget, you refer to that data source by name. This logic simplifies the process of deploying dashboards and pivot tables to end users. But it also means that if you rename items, you must update any dashboards that use those elements.

When you open a dashboard, if the system cannot find a data source that it uses, the system displays as much of the dashboard as possible. For example:



2.7.2 Kinds of Data Sources

With exceptions noted below, a dashboard widget can display data from any of the following data sources:

Data source	Where created	For information, see
Pivot tables	Analyzer	Using the DeepSee Analyzer
Detail listings		
KPIs (key performance indicators)	Studio	DeepSee Implementation Guide
Ensemble business metrics (available only in Ensemble)	Studio	Developing Ensemble Productions

Exceptions:

- A detail listing can be displayed only in a pivot table widget (in table mode).

The kind of data source does not affect the appearance of the widget; for example, the user cannot tell if a given chart is displaying data from a pivot table, a KPI, or a business metric.

Note that some kinds of widgets do not use data sources.

2.7.3 Linked Widgets

When you specify the **Link to** option for a widget, you configure that widget as a *dependent* widget that is linked to another widget (an *independent* widget). The following rules govern linked widgets:

- One independent widget can be linked to any number of dependent widgets in the same dashboard.
- You cannot chain widgets; that is, **Link to** must be an independent widget rather than another dependent widget.
- When you drill down on a row in the independent widget, the system also drills down in the dependent widgets.
- If you filter the independent widget, the system also filters the dependent widgets. Any filters on the dependent widgets are ignored.
- If you use the Mini Analyzer to redefine the independent widget, the system also redefines the dependent widgets.

2.8 Saving a Widget to the Widget Catalog

You can copy a widget (including all its configuration details) to the widget catalog. To do so:

1. Display the dashboard that contains the widget. See “[Displaying DeepSee Dashboards](#),” earlier in this book.
2. Display the [Dashboard Editor](#).
3. Access the Dashboard Editor and select the widget as described in “[Reconfiguring a Widget](#).”
4. Click **Menu > Save Widget to Catalog ...**.

The system displays a dialog box in which you specify details for the widget in the catalog.

5. Specify **Folder**, **Widget Name**, and other options. For information, see “[Adding a Widget to a Dashboard](#),” earlier in this chapter.
6. Click **OK**.

If you have already saved a widget with this name, in the given folder, the system prompts you to confirm that you want to overwrite the previously saved widget.

7. Click **OK**.

No change is made to the dashboard itself.

The saved copy does not include any [local overrides](#). That is, if you have customized the pivot table by using the Mini Analyzer, those customizations are not included in the widget catalog.

2.9 Removing a Widget from a Dashboard

To remove a widget from a dashboard:

1. Display the [Dashboard Editor](#).
2. If you make unrelated changes to the dashboard, and if you want to keep those changes, save them.

When you remove a widget, the Dashboard Designer automatically saves the dashboard, now without the removed widget. In this step, the Dashboard Designer discards other changes that you made and did not save.

3. Click **Widgets**.
4. Click the X button next to the widget name. The change is saved automatically.

2.10 Clearing the Autosave State of a Dashboard

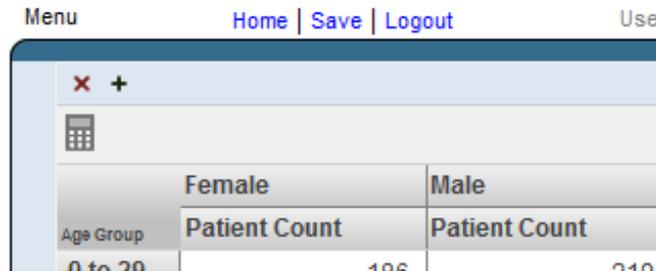
If the [autosave feature](#) for the User Portal is enabled, then when you display a dashboard, the User Portal displays that dashboard with any unsaved changes that you might have made. To remove these unsaved changes, click **Menu > Clear Autosave State**. Then the User Portal displays the dashboard as defined in its saved definition.

Or save the dashboard, thus saving the changes and changing the definition. See the [next topic](#).

2.11 Saving a Dashboard

To save the dashboard you are working on, do one of the following:

- Click **Save**.



	Female	Male
Age Group	Patient Count	Patient Count
0 to 20	105	210

The system saves the dashboard immediately.

- Click **Menu > Save**.

The system saves the dashboard immediately.

- Click **Menu > Save With Options....**

Specify some or all of the following options and then click **OK**:

- **Folder** — Specify the folder, if any, to which the dashboard belongs. Either type a folder name or select an existing folder. If desired, use the slash character (/) to specify a subfolder. For example, you can enter the following for **Folder**:

My Folder/My Subfolder

Do not start the first folder name with a dollar sign (\$); if you do, the dashboard definition is accessible only in Studio.

- **Dashboard Name** — Specify the name of the dashboard.

For information on the other options, see “[Modifying Dashboard Settings](#),” earlier in this chapter.

Or use **Menu > Save As....** See the [next section](#).

2.12 Copying a Dashboard

To copy a dashboard, do the following:

1. Display the dashboard as described in “[Displaying DeepSee Dashboards](#),” earlier in this book.
2. Click **Menu > Save As....**
3. Specify the options as described in the [previous section](#) (and in “[Modifying Dashboard Settings](#)”)”
4. Click **OK**.

2.13 Deleting a Dashboard

One way to delete a dashboard is as follows:

1. Display the dashboard as described in “[Displaying DeepSee Dashboards](#),” earlier in this book.
2. Click **Menu > Delete Dashboard**.
The system prompts you to confirm.
3. Click **OK**.

Note: When you delete a dashboard, DeepSee moves it into a folder named \$TRASH, which the User Portal does not display. If you have access to Studio, you can restore the dashboard. For details, see “[Restoring Deleted Folder Items](#)” in the chapter “Packaging DeepSee Elements into Classes” in the *DeepSee Implementation Guide*.

2.13.1 Deleting a Dashboard in the User Portal

You can also delete a dashboard in the User Portal.

- If the User Portal is in [Covers View](#), click the dashboard cover and then click the X in the dialog box. Then click **OK** to confirm.
- If the User Portal is in [List View](#), click the X button in the row that lists the dashboard. Then click **OK** to confirm.

The User Portal does not display private dashboards. To delete a private dashboard, you must change it to be public and then delete it; see “[Modifying Dashboard Settings](#).” Or delete the dashboard in the Folder Manager, which displays all items (both public and private); see “[Introduction to the DeepSee User Interfaces](#)” in *Getting Started with DeepSee*.

3

Specifying Common Widget Options

For reference, this chapter describes options that are used in multiple places when you configure widgets. It discusses the following topics:

- [How to customize print settings](#)
- [How to add properties](#)
- [How to manage properties](#)
- [How to specify numeric format strings](#)
- [How to specify table text styles](#)
- [How to specify chart text styles](#)
- [How to specify line styles](#)

3.1 Customizing Print Settings for a Widget

You can print most kinds of widgets, as noted in “[Comparison of Widget Types](#),” earlier in this book. To customize how DeepSee prints a widget, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” and click **Print Setup** and then click **Page Setup**. You can modify the following options:

- **Reset** — Click this to restore the default print settings for this widget.
- **Page Size** — Specify the size of the pages. Choose a page size.
- **Units** — (Read only) This field displays the units associated with the page size you selected.
- **Orientation** — Specify the orientation of the pages.
- **Title** — Specify an optional title to include in the PDF. See the [first subsection](#) for special options.
- **Subtitle** — Specify an optional subtitle to include in the PDF. See the [first subsection](#) for special options.
- **Show Subtitle** — Click **On** to include the subtitle in the PDF. Or click **Off** not to include the subtitle.
- **Listing Settings** — Select this to access additional options that apply when the output is a detail listing. Then specify options as described in the subsection “[Listing Settings](#).”
- **Filter Settings** — Select this to access options that control the appearance of information about any filters applied to this pivot table. Then specify options as described in the [first subsection](#).

- **Show Date** — Click **On** to include the current date in the PDF, at the top of the first page. Or click **Off** to omit this information.
- **Show User** — Click **On** to include the current user in the PDF, at the top of the first page. Or click **Off** to omit this information.
- **Page Margins** options: **Top**, **Bottom**, **Left**, and **Right** — Specify the size of the top, bottom, left, and right margins, respectively.

Note: For a pivot table widget (including any chart widget), the pivot table definition specifies the default print settings; see “[Customizing Pivot Tables](#)” in *Using the DeepSee Analyzer*.

3.1.1 Requirements for Printing

When a user invokes the Print option, Caché uses Java to call out to a third-party PDF rendering tool. This means that Java (or specifically JRE, Java Runtime Executable) is required on the user’s local machine. For information on the requirements, which also apply to Zen Reports, see “[Configuring Zen Reports for PDF Output](#)” in *Using Zen Reports*.

3.1.2 Special Options for the Title and Subtitle

Within the **Title** and **Subtitle** fields, you can include a combination of static text and the following tokens:

- `{@filter.filterName}` where *filterName* is the name of a filter defined for the given widget. Use the same name as provided in the **Filter** dropdown menu in the Dashboard Designer.
In the PDF, this token is replaced with the current value of that filter.
- `{@date.dateFormat}` where *dateFormat* is an integer date format, with the same meaning as the *dformat* argument of \$ZDATETIME.
In the PDF, this token is replaced with the date in the given format, followed by the time in 24-hour format.
- `{@variable.variableName}` where *variableName* is the name of a runtime variable.
In the PDF, this token is replaced with the current value of that variable.

3.1.3 Listing Settings

For **Listing Settings**, specify some or all of the following options, which take effect when the output is a detail listing:

- **Show Listing Filters** — Click **On** to include information on the context used for the detail listing (that is, which member or members were selected before the listing was requested, as well as any filters that were in use). Or click **Off** to omit this information.
- **Show Zebra Stripes** — Click **On** to use different colors for alternating rows. Or click **Off** to use the same color for all rows.
- **Font Size** — Specify the font size, including units, for example: 6pt

3.1.4 Filter Settings

For **Filter Settings**, specify some or all of the following options, which control the presentation of the filter information:

- **Show Filters** — Specify the presentation of the filter information. Select **By Table** (to display this information as a table on the top of the first page), **By Title** (to display this information as the subtitle or appended to the subtitle), or **Off** (to not display this information).

By default, the filter information is appended to the subtitle area to the right of any specified subtitle.

When you display filter information as a table, the table has two columns with level names (that is, the filter names) on the left and member names (that is, filter item names) on the right.

- **Table Style** — Specify options to control the appearance of the table that is used to display the filter information. These options affect both the caption column (the left column) and the item column (the right column).
- **Row Header Style** — Specify options to control the appearance of the row headers. These options override the **Table Style** options.
- **Item Style** — Specify options to control the appearance of the item column. These options override the **Table Style** options.
- **Display Format for NOW** — Specify how to display the NOW member of any time level, if that member is used in the filter and if the filter information is shown. Specify any of the *dformat* options of the \$ZDATE function. For example, specify 3 for ODBC format (for example, 2015-03-27).

If you leave this option blank, the PDF shows the NOW member as NOW.

3.2 Adding Widget Properties

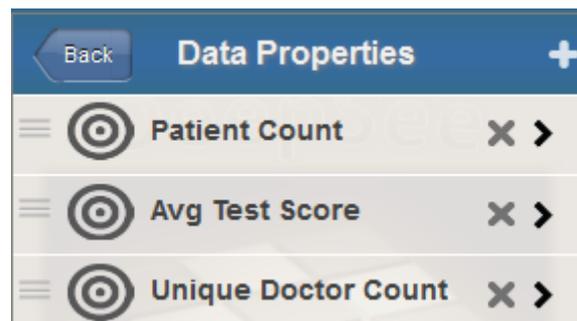
Some kinds of widgets require *properties*, each of which typically corresponds to a value in the configured data source. (For example, if the data source is a pivot table, each property of the widget can be a measure in that pivot table.)

For a meter widget, each property is displayed in a separate meter within the widget. Similarly, for pivot table widgets, each property is displayed as a separate column. Scorecards have columns, which are similar to properties and can be configured in a similar way.

To add a property to a widget:

1. Access the Dashboard Editor and select the widget as described in “[Reconfiguring a Widget](#),” earlier in this book.
2. Click **Data Properties**.

The system displays the **Data Properties** submenu, which lists any properties defined for this widget. For example:



3. Click the plus sign + button.

The system displays a dialog box where you specify the property.

4. For **Value**, select the property of the data source to display.

For a meter widget or a scorecard, you can instead type a numeric constant.

If you manually edit **Value**, note that if **Value** has a numeric value, DeepSee assumes it is a numeric constant and displays that. If **Value** has the form `= [propertyname]`, DeepSee assumes it is a property of the data source and thus displays the value of that property. When you select a property, DeepSee automatically wraps the property with `= []`.

For a scorecard, you can instead type a formula (see “[Specifying a Scorecard Formula](#)” in the chapter “[Adding Scorecard Widgets](#)”).

5. Optionally specify the following additional items:

- **Label** — Type a label or optionally click the check box to the right of **Label**. If you click the check box, DeepSee uses the localized name of this property as the label; this option applies only when **Value** is a data source property.

If you type a label, note that you can include `\n` to force a line break in the label.

- **Hidden** — Select this option to hide this value.
- **Value Column** — Select this option to configure the value in this column as the value of the row. This value is passed to any custom actions; see “[Specifying the Value of a Row](#),” later in this book.

Select this option for only one data property of the widget.

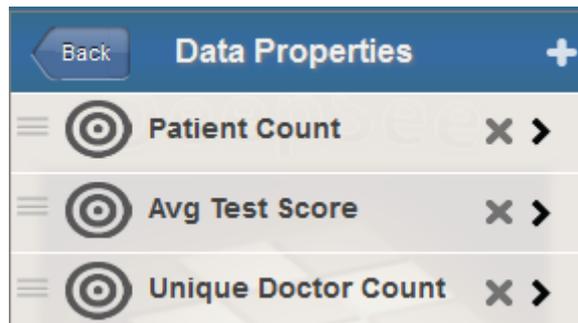
- **Format** — Type a format string. See “[Specifying a Numeric Format String](#),” later in this chapter.

Or specify these options later if you modify the property as described in the [next section](#).

6. Click **OK**.

3.3 Managing Widget Properties

For any widget that defines [properties](#), the **Data Properties** submenu lists the properties defined for this widget. For example:



In addition to the specific property options documented elsewhere, you can perform the following general activities:

- Change the order of the properties. To do so, first click the icon to the left of the property name, so that this icon is

highlighted: 

Then drag the property up or down in the list.

- Delete a property. To do so, click the X button in the row for that property.
- Reconfigure a property. To do so, click the property name. DeepSee then displays a submenu with options. For details on the options, see the [previous section](#). You can also specify **Name**. Make changes as needed.

3.4 Specifying Numeric Format Strings

In some meters and other places that display numeric values, you can use the **Format** option to control the display of numbers. In some cases, you can select an option or type a numeric format string. In other cases, you must type a numeric format string.

3.4.1 Selecting a Format String

In some cases, the Dashboard Editor provides the following set of choices for **Format**:

Option	Example
###	6609
###.#	6609.1
###.##	6609.12
###.###	6609.123
#,##	6,609
#,##.#	6,609.1
#,##.##	6,609.12
#,##.###	6,609.123
##.###%	660912.3%

3.4.2 Typing a Numeric Format String

In all applicable places, you can instead type a numeric format string, which provides more options. You can specify a string that consists of one to four pieces as follows:

`positive_piece;negative_piece;zero_piece;missing_piece`

Where *positive_piece* controls how a positive value is displayed, *negative_piece* controls how a negative value is displayed, *zero_piece* controls how zero is displayed, and *missing_piece* controls how a missing value is displayed.

Each piece is a string that includes one of the following base units:

Base Unit	Meaning	Example
#	Display the value without the thousands separator and without decimal places.	12345
#, #	Display the value with the thousands separator. Do not include any decimal places. This is the default display format for positive numbers.	12,345
#.##	Display the value without the thousands separator. Include two decimal places (or one decimal place for each pound sign after the period). Specify as many pound signs after the period as you need.	12345.67
#, #.##	Display the value with the thousands separator. Include two decimal places (or one decimal place for each pound sign after the period). Specify as many pound signs after the period as you need.	12,345.67

You can include additional characters before or after the base unit.

- If you include a percent sign (%), the system displays the value as a percentage. That is, it multiplies the value by 100 and it displays the percent sign (%) in the position you specify.
- Any other characters are displayed as given, in the position you specify.

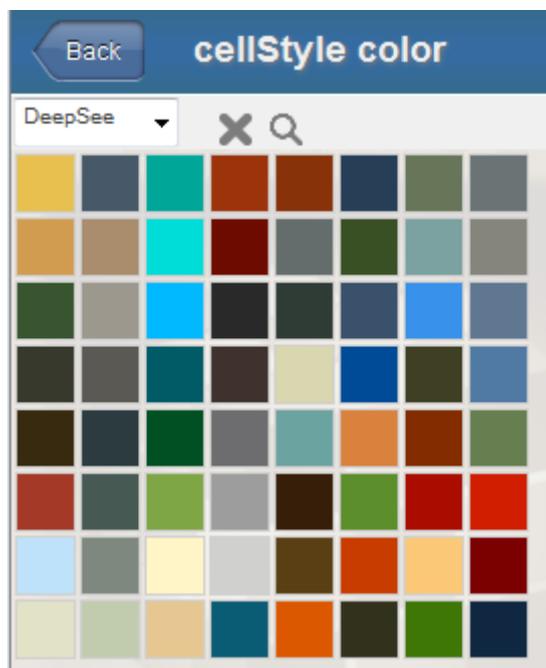
The following table shows some examples:

Example	Logical Value	Display Value
#, #; (#, #) This corresponds to the default display of numbers.	6608.9431	6,609
	-1,234	(1,234)
#, #.###	6608.9431	6,608.943
#%;	6	600%
\$#, #; (\$#, #)	2195765	\$2,195,765
	-3407228	(\$3,407,228)

3.5 Specifying Table Text Styles

In tables and chart legends, you can specify options that control the display of text. The options are as follows:

- **Color** — Specify the color of the text. The following shows an example of the user interface:



To specify a color, optionally choose the name of a color palette from the drop-down list. (For example, the list option **DeepSee** corresponds to the color palette shown in the preceding example.) Then click a color in the grid to choose that color.

Or click the magnifying glass icon, which displays a full-range color chooser. Click a color here or use the sliders to specify the red, green, and blue components of your color selection.

To clear the color selection, click the X.

Tip: An authorized user can add custom color palettes. See “[Creating Custom Color Palettes](#)” in the *DeepSee Implementation Guide*.

- **Background** — Specify the color of the background, in the same way as described in the previous item.
- **Opacity** — Use this option to specify the opacity of the background. The value 1 means 100%.
- **Font** — Specify the typeface of the text. Select a typeface. To clear the typeface selection, click **Default**, at the top of the list.
- **Font Size** — Use the slider to specify the size of the text or type a value into the input box.



To clear the size selection, click the check box.

- **Text Style** — Specify the style of the text.



Select one or more options. The first three options make the text bold, italic, and underlined, respectively. The **Cc** option changes the text to capital letters as in the following example:

The **S** option adds shadows to the text as follows:

Revenue

- **Align** — Specify the alignment of the text.



These options make the text left-aligned, centered, and right-aligned, respectively.

3.6 Specifying Chart Text Styles

In charts and other graphical elements, you can specify options that control the display of text. The options are as follows:

- **Color** — Specify the color of the text. For information on specifying this option and other color options, see the **Color** option in the [previous section](#).
- **Stroke** — Specify the color of the outline of the characters in the text. The following figure shows an example where **Color** is blue and **Stroke** is black:

My Chart Title

By default, DeepSee uses the same color for the outline as for the text itself.

- **Stroke Width** — Use this option to specify the width of the outline. The following shows an example of a medium-width outline:

My Chart Title

To clear the setting, click the check box.

- **Opacity** — Use this option to specify the opacity of the text (which is 100% by default).
- **Font, Text Size, Text Style** — See **Font, Font Size, Text Style** in the [previous section](#).

3.7 Specifying Line Styles

In numerous places, you can specify options that control the display of lines and other graphical elements. The options are as follows:

- **Fill** — Specify the color to use within the graphical element. This option applies to line markers and to parts of speedometers, for example. For information on specifying this option and other color options, see the **Color** option in “[Specifying Table Text Styles](#),” earlier in this chapter.
- **Opacity** — Use this option to specify the opacity of graphical element (which is 100% by default).
- **Line** — Specify the color of the line that outlines the graphical element.
- **Line Width** — Use this option to specify the width of the line.
- **Line Style** — Specify the style of the line. Click **Solid**, **Dashed**, or **Dotted**.

4

Adding Pivot Table Widgets

This chapter describes how to add *pivot table widgets* to a dashboard. These widgets can display any kind of [data source](#), not just pivot tables. This chapter discusses the following topics:

- [How to add a pivot table widget](#)
- [How to customize widget settings for a pivot table widget](#)
- [How to customize the columns](#)
- [How to display a listing in a pivot table widget](#)
- [How to customize the appearance of a pivot table widget](#)
- [How to remove a pivot table redefinition](#)

For information on creating pivot tables, see *Using the DeepSee Analyzer*.

Note: Chart widgets are a kind of pivot table widget. For specific information on charts, see the [next chapter](#).

4.1 Adding a Pivot Table Widget

To add a pivot table widget:

1. Click **Menu > Add New Widget...**
2. Click **Pivots and Charts**.
3. Click **Table**.
4. For **Data source**, do one of the following:
 - Click the Search button  and then select a data source.
See “[Kinds of Data Sources](#),” earlier in this book.
 - Use the **Link to** drop-down list to select another widget on the same dashboard.

You can use any type of data source, not just pivot tables.

Note: If you do use a pivot table as the data source, note that pivot tables are created in the Analyzer, discussed in [Using the DeepSee Analyzer](#). The Analyzer provides options for changing the *state* of the pivot table: filtering, drilling down, displaying listings, and displaying in chart format. When you save a pivot table, DeepSee saves the state as part of the pivot table definition. The next time the pivot table is accessed (either in the Analyzer or in a widget on a dashboard), DeepSee displays the pivot table in the state in which it was saved.

5. Optionally specify the basic settings, as described in “[Adding a Widget to a Dashboard](#),” earlier in this book.
6. Click **OK**.
7. Optionally configure the widget further as described in the rest of this chapter.
8. Optionally add controls, as described in the chapter “[Adding Widget Controls](#).”

4.2 Specifying Widget Settings for a Pivot Table Widget

To customize the available options in a widget, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” and then click **Widget Settings**. This menu primarily provides options for the user.

For a pivot table widget (including any chart widget), the **Widget Settings** menu has the following options:

- **Print** — Select this option to include the Print button  on the toolbar. This button enables users to print the widget to a PDF file. For further details and requirements, see the section “[Customizing Print Settings for a Widget](#).”
- **Chart Toggle** — Select this option to add buttons to the toolbar that can toggle the display to show either the chart or the equivalent pivot table. Select **chart** to set the initial state as a chart, select **table** to set the initial state as a pivot table, or select **none** to disable the toggle button.
- **Drill Down** — Select this option to include the Drill Down button  on the toolbar. This button enables the users to drill down.

This option is supported only when the data source is a pivot table. The option is useful only when the rows of the pivot table display members of a higher level in a hierarchy.

- **Analyzer** — Select this option to include the Mini Analyzer button  on the toolbar. This button enables the users to open the data source in the Mini Analyzer.

This option is supported only when the data source is a pivot table.

- **Excel Export** — Select this option to include the Export button  on the toolbar. This button enables the users to export the data in this widget to Microsoft Excel.
- **Row selection** — Use this option to display check boxes on the widget; the user can use these check boxes to select rows to which to apply actions. Select **single** to enable the user to select only a single row, or **multi** to enable the user to select multiple rows.

By default, these check boxes are not displayed, and the user cannot select rows in the widget. (The user can always select *cells* and *ranges of cells* in the widget.)

This option is supported only when the data source is a KPI.

- **Show Dimensions** — Select this option to include the Dimension List button  on the toolbar. This button provides an extra drilling option. Users can drag and drop items from this list on the gray part of the rows in the pivot table; when they do, DeepSee drills down as requested.

This option is supported only when the data source is a pivot table and the widget is in table format.

Note: When you enable or disable this option, the Dashboard Designer automatically saves the dashboard.

- **Dimension Source** — For advanced use, this option enables you to specify an alternative set of drill options to show in the dimension list.
- **Initial Execute** — By default, this option is selected, which means that DeepSee executes the query underlying this widget as soon as the user displays the dashboard. Clear this option if you want execute the query later. If you clear **Initial Execute**, make sure to include a [refresh control](#) on this widget. See the chapter “[Adding Widget Controls](#).”

Note: Most of these options add a button to the toolbar. If you enable any of those options, be sure the toolbar is visible (its display is controlled via the **Toolbar** option in **Settings**). See “[Reconfiguring a Widget](#),” earlier in this book.

4.3 Customizing the Columns (When Displaying a KPI)

By default, when a pivot table widget displays a KPI, it displays all the columns defined in that data source. You can customize the columns. For information, see “[Adding Widget Properties](#),” earlier in this book.

4.4 Displaying a Listing in a Pivot Table Widget

To display a listing directly in a pivot table widget:

1. Do the following in the Analyzer:
 - a. Create a pivot table with one cell. Filter this pivot table as needed for the listing.
See [Using the DeepSee Analyzer](#).
 - b.  Select the cell and click the listing button to display the listing.
 - c. Save the pivot table.
2. In the desired dashboard, add a pivot table widget that use this pivot table as its data source.
3. Optionally add filter [controls](#); these restrict the set of records that are shown.

4.5 Customizing the Appearance of a Pivot Table

DeepSee provides a rich set of options that you can use to customize the appearance of a pivot table widget. To access these options, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#)” and click **Table Settings**. This menu has two submenus:

- **Size & Appearance** — Provides options to control the size of the data cells and the form of the pivot table. See the [first subsection](#) for details.
- **Colors & Style** — Provides options to control colors, typeface, text style, and text alignment in the pivot table. See the [second subsection](#) for details.

Note: The easiest way to systematically alter the appearance of dashboards is to define and use themes, which are discussed in “[Defining and Using Themes](#)”. Themes use the options described in the following subsections.

4.5.1 Size and Appearance Options

To control the size of the data cells and the form of the pivot table, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Table Settings**, and click **Size & Appearance**. This menu has the following options:

- **Cell Width** — Specify the width, in pixels, of the data cells. For example:

Diagnoses	Patient Count
None	841
asthma	68
CHD	36
diabetes	54
osteoporosis	25

This example also uses **Cell Height**.

- **Cell Height** — Specify the height, in pixels, of the data cells. See the previous example.
- **Show Empty** — Use these options to control whether empty rows and columns are displayed. For example, if you enable the **Show Empty** option for rows, DeepSee displays empty rows as follows:

	Age Group	Patient Count
asthma	0 to 29	29
	30 to 59	32
	60+	7
CHD	0 to 29	
	30 to 59	8
	60+	28
diabetes	0 to 29	2
	30 to 59	27
	60+	25
osteoporosis	0 to 29	
	30 to 59	
	60+	25

- **Span Labels** — Use these options to control whether row and column labels are spanned.

When you display the table in a nested format, the labels are spanned by default. For example:

Age Group-	Gender	Patient Count
0 to 29	Female	186
	Male	219
30 to 59	Female	218
	Male	215
60+	Female	94
	Male	68

If you enable the **Span Labels** option for rows, DeepSee repeats the relevant labels, as follows:

Age Group-	Gender	Patient Count
0 to 29	Female	186
0 to 29	Male	219
30 to 59	Female	218
30 to 59	Male	215
60+	Female	94
60+	Male	68

- **Row Totals** — Use this option to control whether the pivot table includes a summary row at the bottom. For example:

	Female	Male	Total
Age Group	Patient Count	Patient Count	
0 to 29	186	219	405
30 to 59	218	215	433
60+	94	68	162
Total	498	502	1,000

This example also uses **Column Totals**.

- **Row Summary** — Specify the form of the summary row. For example, choose **Sum** or **Average**.
- **Sum Over**, choose either **Page** or **All Rows**. This option determines whether the summary row uses only the current page of the pivot table or all rows.
- **Column Totals** — Use this option to control whether the pivot table includes a summary column at the right. See the example for **Row Totals**.
- **Row Caption** — Use this option to display or hide the caption that appears above the row headers. In the following example, the row captions are hidden:

		Patient Count
0 to 29	Female	186
	Male	219
30 to 59	Female	218
	Male	215
60+	Female	94
	Male	68

4.5.2 Color and Style Options

To control the colors, typeface, and other such aspects of the pivot table, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Table Settings**, and click **Colors & Style**. This menu has the following options:

- **Cell Style** — Use this submenu to modify the colors, typeface, text style, and text alignment used in all data cells of the pivot table. For example:

Region	Revenue
Asia	\$3,928.35
Europe	\$2,895.98
N. America	\$3,331.64
S. America	\$1,699.09

For details on this and other submenus, see “[Specifying Table Text Styles](#),” earlier in this book.

- **Row Style** — Use this submenu to modify the colors, typeface, text style, and text alignment used in the row headers of the pivot table. For example:

Region	Revenue
Asia	\$3,928.35
Europe	\$2,895.98
N. America	\$3,331.64
S. America	\$1,699.09

- **Column Style** — Use this submenu to modify the colors, typeface, text style, and text alignment used in the column headers of the pivot table. For example:

Region	Revenue
Asia	\$3,928.35
Europe	\$2,895.98
N. America	\$3,331.64
S. America	\$1,699.09

- **Stripes** — Select this option to format the pivot table with zebra stripes. If you enable **Stripes**, the table is formatted with rows in alternating colors as follows:

Region	Revenue
Asia	\$3,928.35
Europe	\$2,895.98
N. America	\$3,331.64
S. America	\$1,699.09

- **Stripe Style** — Use this submenu to modify the colors, typeface, text style, and text alignment used in the even rows, when zebra stripes are enabled. For example:

Region	Revenue
Asia	\$3,928.35
Europe	\$2,895.98
N. America	\$3,331.64
S. America	\$1,699.09

- **Background** — Select the background color of the pivot table. For example:

× + Patients by Age & Gender			
Age Bucket	Female	Male	
0 to 9	71	78	
10 to 19	60	75	
20 to 29	60	56	
30 to 39	78	88	
40 to 49	76	63	
50 to 59	57	49	
60 to 69	50	36	
70 to 79	39	23	
80+	27	14	

- **Opacity** — Specify the opacity of the background. The value 1 means 100%.

4.6 Removing a Pivot Table Redefinition

Via the Mini Analyzer, a user can redefine a pivot table as used in a given dashboard widget. See [Using the DeepSee Analyzer](#).

To remove any redefinition performed this way:

1. Access the Dashboard Editor and select the widget as described in “[Reconfiguring a Widget](#).”
2. Click **Type and Data Source**.
3. Click **Reset Data Source**.

If the **Reset Data Source** option is grayed out, this widget uses the pivot table as it was originally defined in the Analyzer.

4. Click **OK**.

Or click the Mini Analyzer button , click **Reset**, and then click **OK**.

5

Adding Chart Widgets

This chapter describes how to add charts to a dashboard. It discusses the following topics:

- [How to add a chart widget](#)
- [How to customize the chart type](#)
- [Requirements for the data source](#)
- [A look at the available chart types](#)
- [How to display an Ensemble business metric](#)

Also see “[Specifying Widget Settings for a Pivot Table Widget](#),” in the previous chapter, as well as the next chapter (“[Customizing the Appearance of a Chart Widget](#)”).

5.1 Adding a Chart Widget

To add a chart widget:

1. Click **Menu > Add New Widget...**
2. Click **Pivots and Charts**.
3. Click a chart type.
4. For **Data source**, do one of the following:
 - Click the Search button  and then select a data source.
See “[Kinds of Data Sources](#),” earlier in this book.
 - Use the **Link to** drop-down list to select another widget on the same dashboard.

Note: If you use a pivot table as the data source, note that pivot tables are created in the Analyzer, discussed in [Using the DeepSee Analyzer](#). The Analyzer provides options for changing the *state* of the pivot table: filtering, drilling down, displaying listings, and displaying in chart format. When you save a pivot table, DeepSee saves the state as part of the pivot table definition. The next time the pivot table is accessed (either in the Analyzer or in a widget on a dashboard), DeepSee displays the pivot table in that state.

5. Optionally specify the basic settings, as described in “[Adding a Widget to a Dashboard](#),” earlier in this book.
6. Click **OK**.

7. Optionally customize the chart type, as described in the [next section](#).
8. Optionally configure the widget further as described in the rest of this chapter.
9. Optionally add controls, as described in the chapter “[Adding Widget Controls](#).”

5.2 Customizing the Chart Type

You can reconfigure a chart widget to use a different chart type. To do so:

1. Access the Dashboard Editor and select the widget as described in “[Reconfiguring a Widget](#),” earlier in this book.
2. Click **Chart Settings**.
3. Click **Chart Type**.
4. Click a type.

You can also use these steps to reconfigure a [pivot widget](#) to display a chart.

5.3 Chart Requirements

A chart is usually more useful if the data source uses members as rows and uses measures as columns.

Notes:

- If you use a pivot table that has a summary row or summary column, that row or column is ignored.
- Many of the chart types are also available in the Analyzer, where you can preview them before adding them to a dashboard.
- Except in the case of line charts and time charts, only the first 400 chart items are displayed.

5.4 Available Chart Types

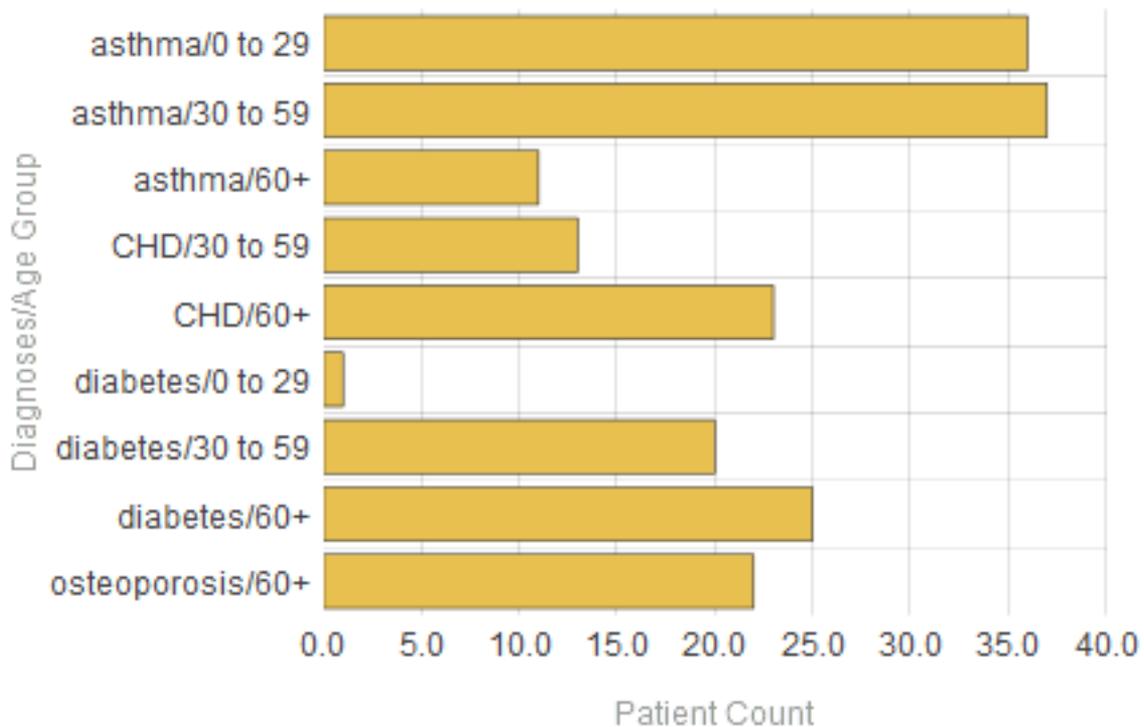
The following subsections show examples of the available chart types:

- [Bar and column charts](#)
- [Line and area charts](#)
- [Combo charts](#)
- [High low charts](#)
- [Pie charts](#)
- [X/Y charts](#)
- [Bubble charts](#)
- [Time charts](#)
- [Tree map charts](#)

- Bullseye charts
- Swirl charts

5.4.1 Bar and Column Charts

DeepSee provides several styles of bar and column charts, both horizontal and vertical. One group of bars is created for each row of the data source. In a group of bars, each bar displays a different measure. The following example shows a chart with one measure:



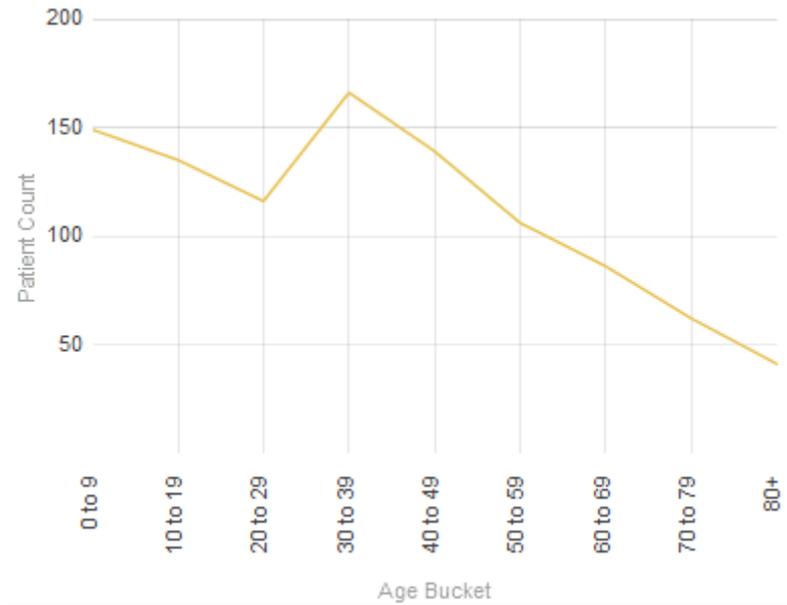
This chart is based on the following pivot table:

Diagnoses	Age Group	Patient Count
asthma	0 to 29	304
	30 to 59	278
	60+	178
CHD	0 to 29	4
	30 to 59	91
	60+	327
diabetes	0 to 29	27
	30 to 59	276
	60+	369
osteoporosis	60+	279

5.4.2 Line and Area Charts

DeepSee provides two line charts and an area chart. For each of these, the horizontal axis has one label for each row of the data source. One line is created for each column of the data source. (For the area chart, the area below the line is filled in.)

The following shows an example:



This chart is based on the following pivot table:

Age Bucket	Patient Count
0 to 9	149
10 to 19	135
20 to 29	116
30 to 39	166
40 to 49	139
50 to 59	106
60 to 69	86
70 to 79	62
80+	41

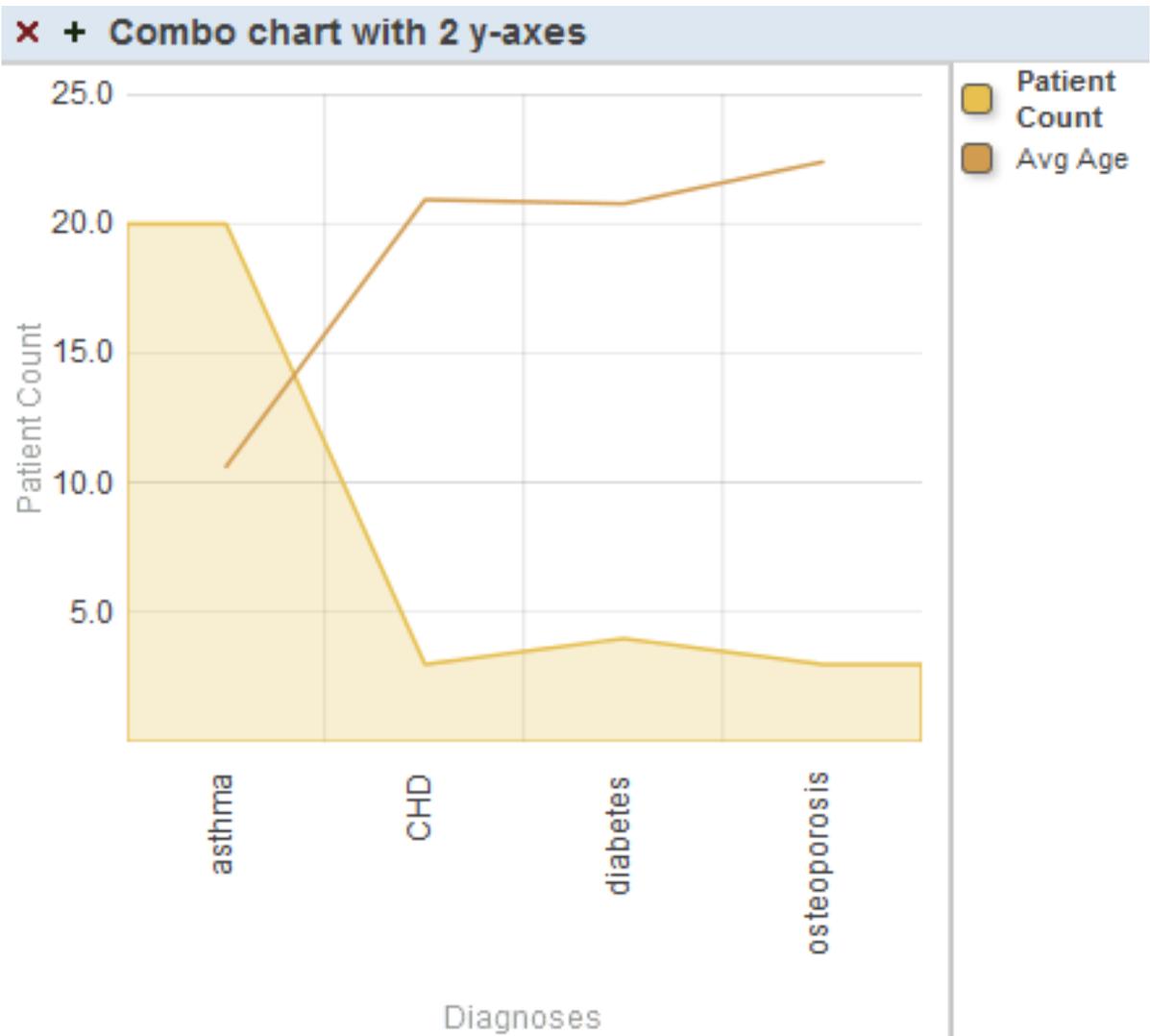
5.4.3 Combo Charts

Note: Combo charts are available only in dashboards, not in the Analyzer.

The DeepSee combo chart is similar to the preceding charts, but provides two unique features:

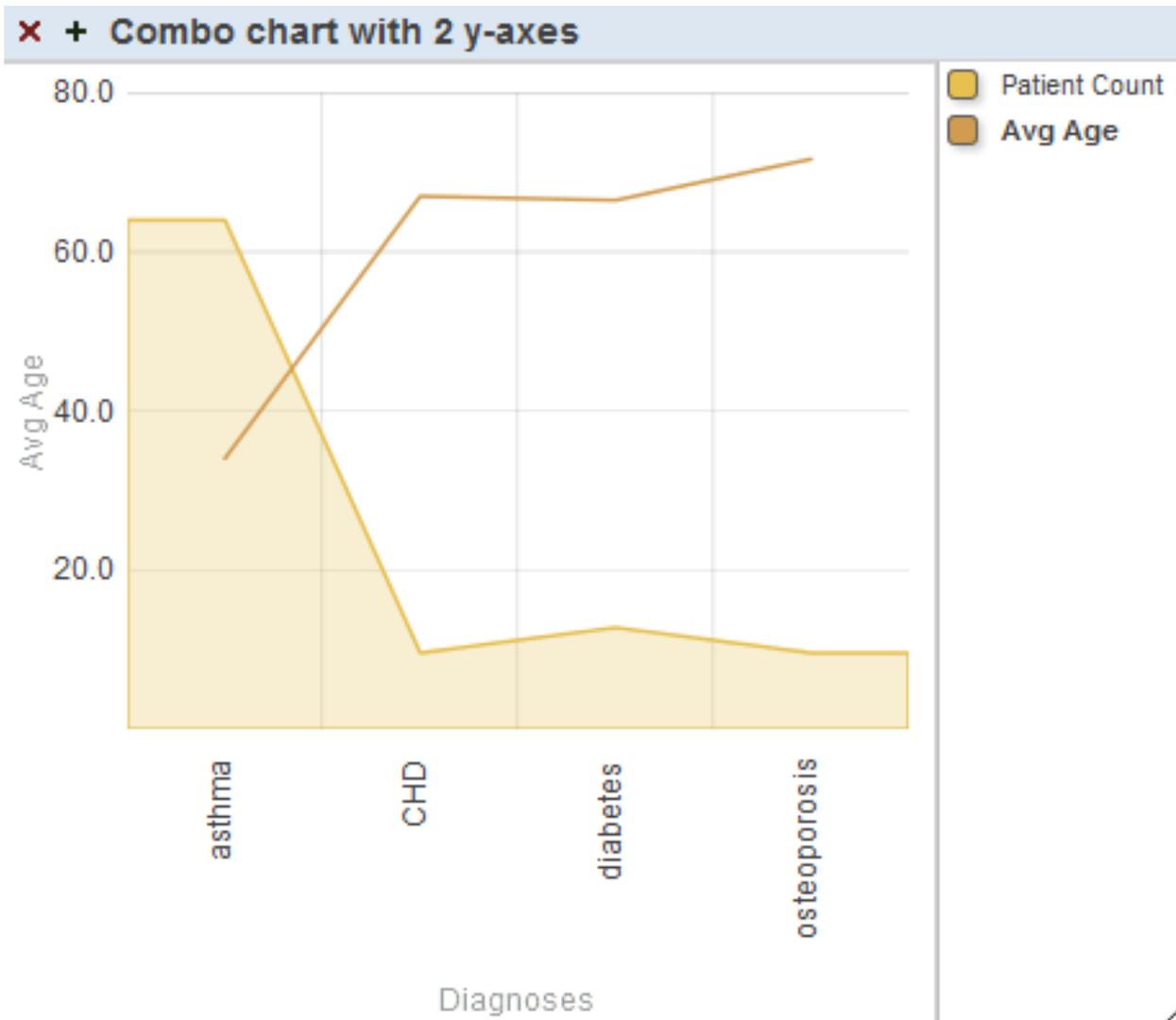
- It can have multiple y-axes.
- It can use different display types for different columns of the data source. DeepSee automatically displays lines in the foreground, bars behind the lines, and areas behind the bars.

The following chart demonstrates both features:



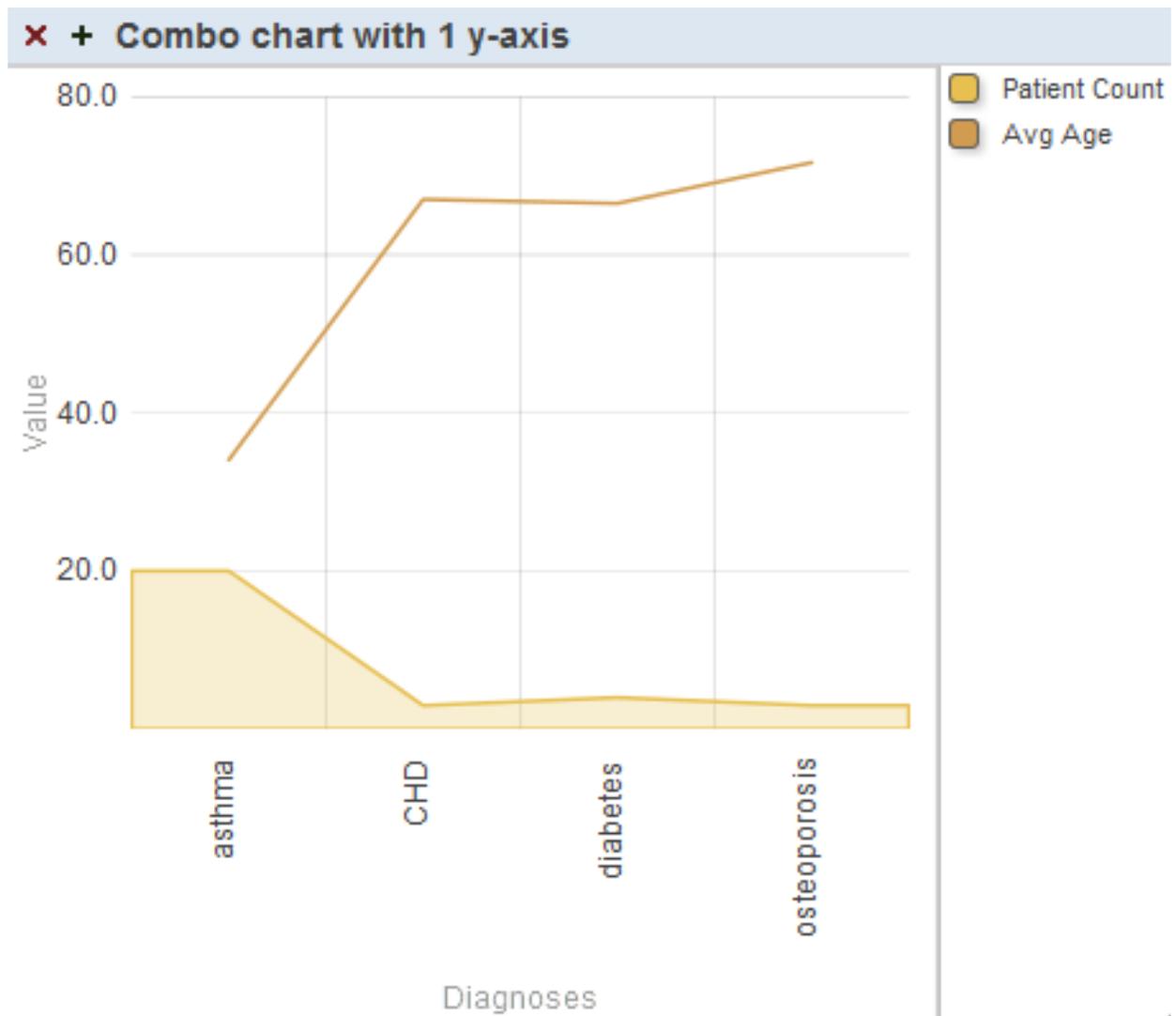
Notice that the y-axis is labeled `Patient Count`. The filled area that displays the `Patient Count` measure is scaled to match the values on this axis. In contrast, the line that displays `Avg Enc Count` does not correlate with the y-axis — this line is only scaled overall to fit within the chart.

To use a combo chart, the user clicks the box next to the name of a series, in the key. If we click the box next to `Avg Age`, the labels on the y-axis change as follows:



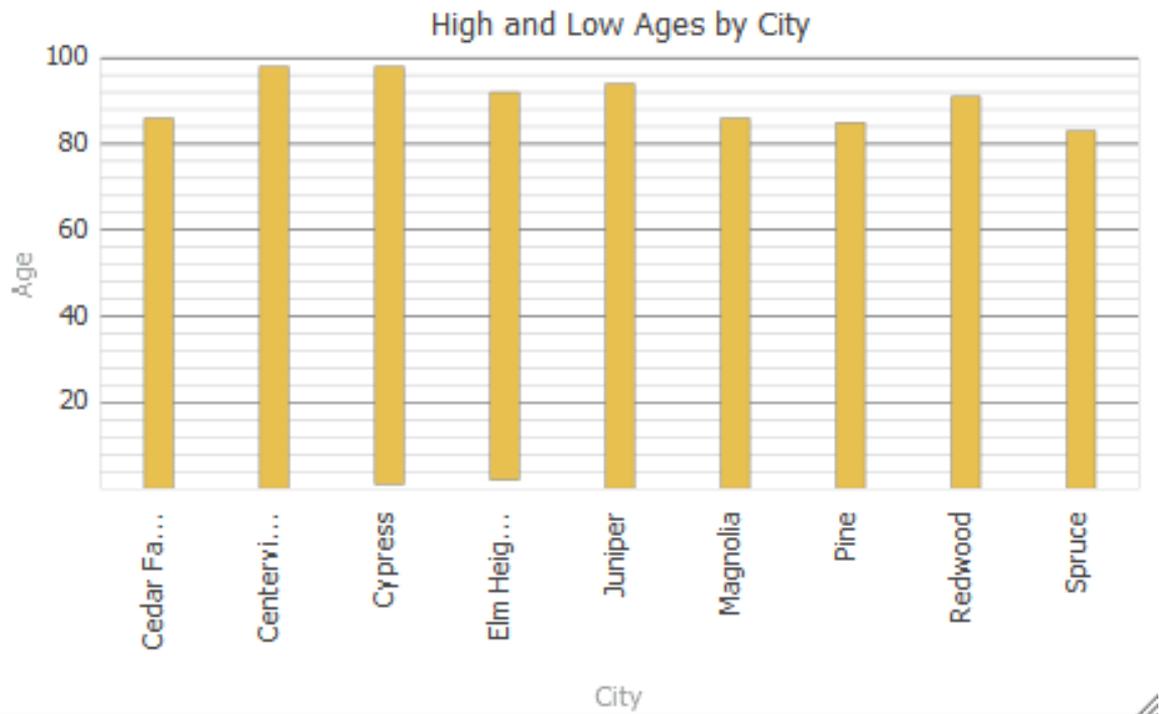
This chart is configured so that for the first series, **y-Axis** is 1, and for the second series, **y-Axis** is 2.

In contrast, if **y-Axis** is the same for both series, we would see the following (in which all the items are drawn to the same scale):



5.4.4 High Low Charts

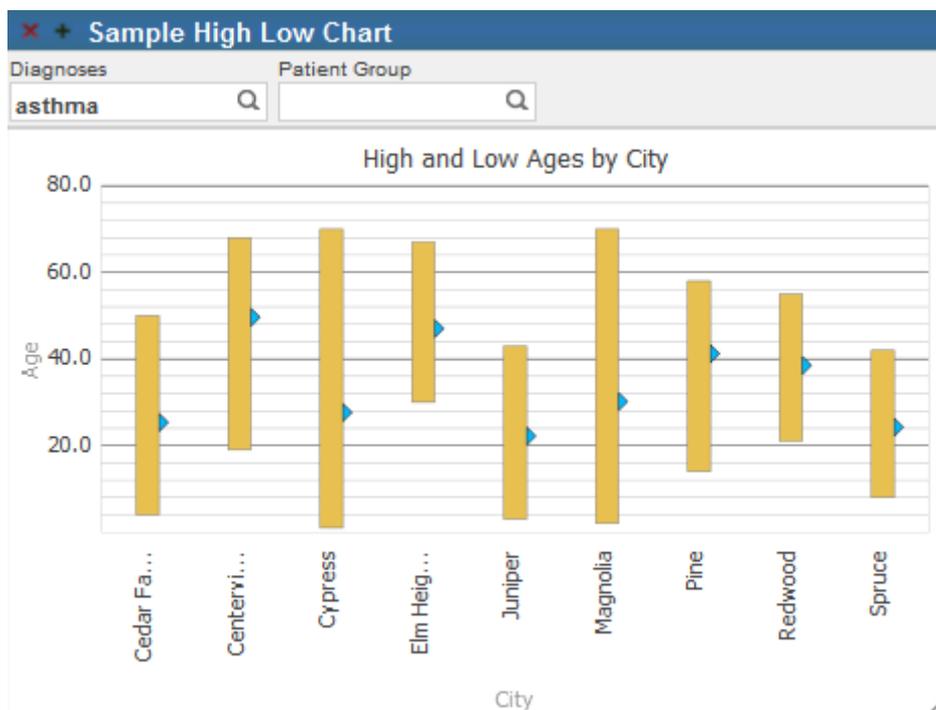
A high low chart is similar to a bar chart but instead uses each bar to display a low value *and* a high value. It requires at least two columns in the data source. It displays one bar for each row of the data source. The upper and lower values are determined by the first two columns in the data source. For example:



This chart is based on the following pivot table:

City	Minimum Age	Maximum Age
Cedar Falls	4	50
Centerville	19	68
Cypress	1	70
Elm Heights	30	67
Juniper	3	43
Magnolia	2	70
Pine	14	58
Redwood	21	55
Spruce	8	42

If there is a third column, the chart displays a triangle for that value. For example:

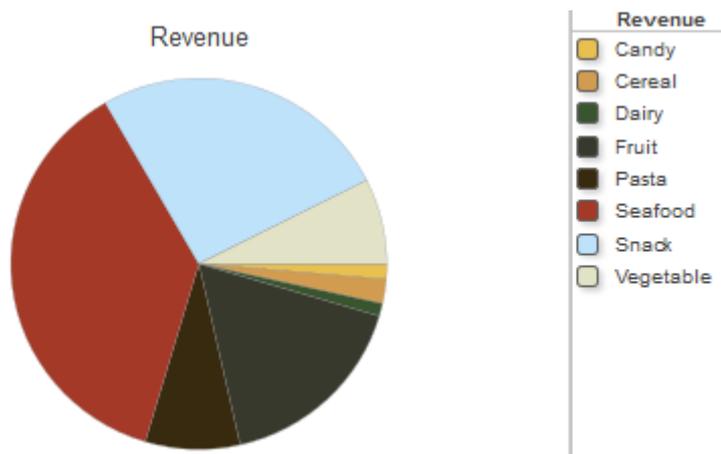


In this case, the third column displays the average values.

Other columns are ignored.

5.4.5 Pie Charts

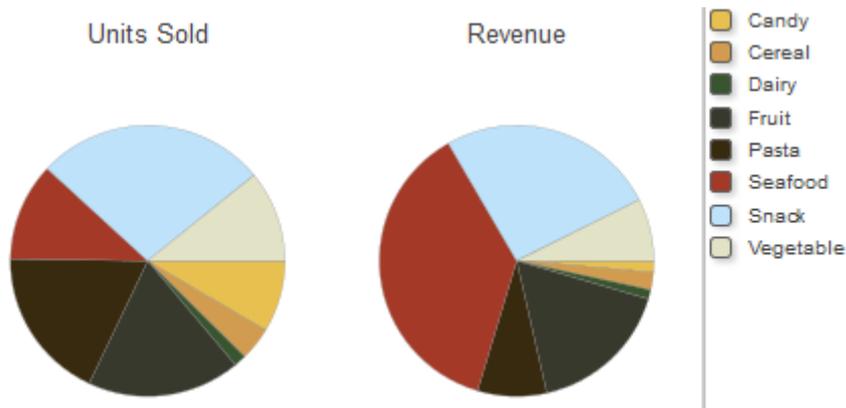
A pie chart displays one wedge for each cell of the data source. The area of each wedge is proportional to the value in the cell. If the data source displays only one column, a pie chart might look like this:



This chart is based on the following pivot table:

Product Category	Revenue
Candy	\$166.99
Cereal	\$263.12
Dairy	\$138.06
Fruit	\$2,108.77
Pasta	\$1,013.68
Seafood	\$4,626.85
Snack	\$3,223.85
Vegetable	\$925.96

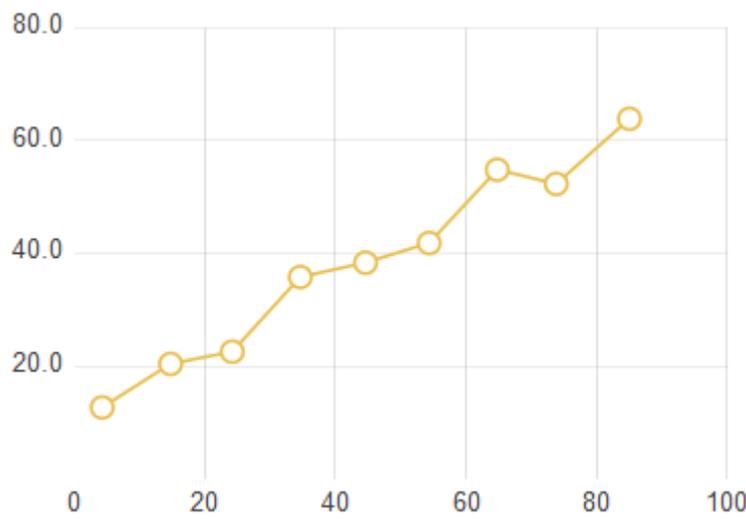
If you enable the **Show Multiples** option, the widget displays one chart for each column of the data source. The chart displays one wedge for each cell in that column. For example:



5.4.6 X/Y Charts

An X/Y chart plots one value on the x-axis and another value on the y-axis. The data source must have at least two columns. The chart displays a data point for each row of the data source, and it connects these data points with lines.

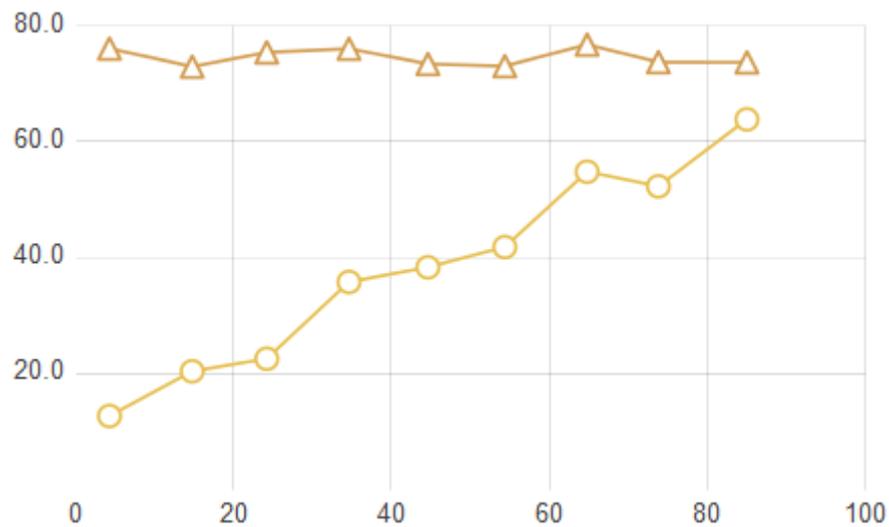
The first column determines the horizontal position of the data point, and the second column determines the vertical position. The following shows an example:



This chart is based on the following pivot table:

Age Bucket	Avg Age	Avg Enc Count
0 to 9	4.32	12.73
10 to 19	14.79	20.44
20 to 29	24.23	22.60
30 to 39	34.65	35.83
40 to 49	44.64	38.36
50 to 59	54.38	41.85
60 to 69	64.79	54.83
70 to 79	73.82	52.32
80+	85.03	63.86

If the underlying pivot table contains additional columns, they are used to create additional sets of connected data points on the chart. That is, for each additional column, the chart contains a set of data points at the same horizontal positions. For example:



This chart is based on the following pivot table:

Age Bucket	Avg Age	Avg Enc Count	Avg Test Score
0 to 9	4.32	12.73	76.07
10 to 19	14.79	20.44	72.95
20 to 29	24.23	22.60	75.41
30 to 39	34.65	35.83	76.04
40 to 49	44.64	38.36	73.41
50 to 59	54.38	41.85	73.04
60 to 69	64.79	54.83	76.75
70 to 79	73.82	52.32	73.73
80+	85.03	63.86	73.69

5.4.7 Bubble Charts

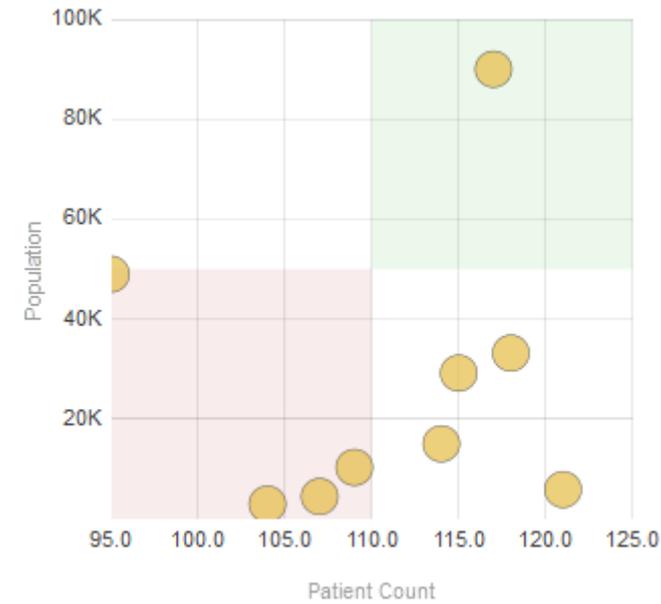
Note: Bubble charts are available only in dashboards, not in the Analyzer.

A bubble chart is similar to the X/Y chart. It plots one value on the x-axis and another value on the y-axis. The data source must have at least two columns. The chart displays an elliptical bubble for each row of the data source. The measures are used as follows:

- The first column determines the horizontal position of the bubble.
- The second column determines the vertical position.
- The third column, if included, determines the radius of the bubble.
- The fourth column, if included, controls additional colorization that is added to the bubbles. All data rows that have the same value for the fourth measure are shown in the same color.
- The fifth column, if included, determines the opacity of the bubble.

Other columns are ignored.

The following example shows a bubble chart that uses two columns:



This chart is based on the following pivot table:

City	Patient Count	Population
Cedar Falls	117	90,000
Centerville	95	49,000
Cypress	104	3,000
Elm Heights	118	33,194
Juniper	109	10,333
Magnolia	107	4,503
Pine	114	15,060
Redwood	115	29,192
Spruce	121	5,900

5.4.8 Time Charts

Note: Time charts are available only in dashboards, not in the Analyzer.

A time chart is useful for displaying values over time. For this kind of chart, the data source must have time members (such as years) for rows. The following shows an example:



This chart is based on the following pivot table:

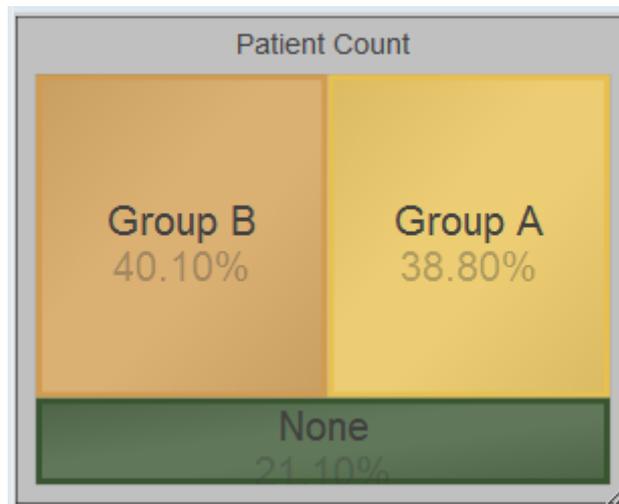
Year Sold	Units Sold	Revenue
2007	228	\$1,534.03
2008	258	\$2,200.88
2009	263	\$2,253.58
2010	319	\$2,608.65
2011	284	\$2,498.32
2012	370	\$3,073.54

Time charts are supported only for levels that are based on the following time functions: **DayMonthYear**, **MonthYear**, **WeekYear**, and **Year**. Also, time charts are not recommended for spans of time greater than approximately ten years.

5.4.9 Tree Maps

Note: Tree maps are available only in dashboards, not in the Analyzer.

A [tree map](#) has one rectangle divided into smaller rectangles, each of which corresponds to one cell in the data source. The area of each rectangle is proportional to the value in the cell. If the data source displays only one measure, a tree map might look like this:



This tree map uses the following pivot table:

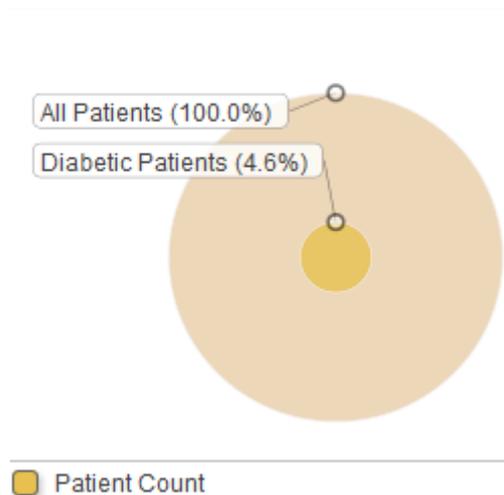
Patient Group	Patient Count
Group A	388
Group B	401
None	211

If the data source has multiple columns, be sure to enable the **Show Multiples** option. In this case, the widget displays a subdivided rectangle for each column.

5.4.10 Bullseye Charts

Note: Bullseye charts are available only in dashboards, not in the Analyzer.

A bullseye chart can be useful to demonstrate the size of a set relative to a larger set. The following shows an example:



This chart is based on the following pivot table:

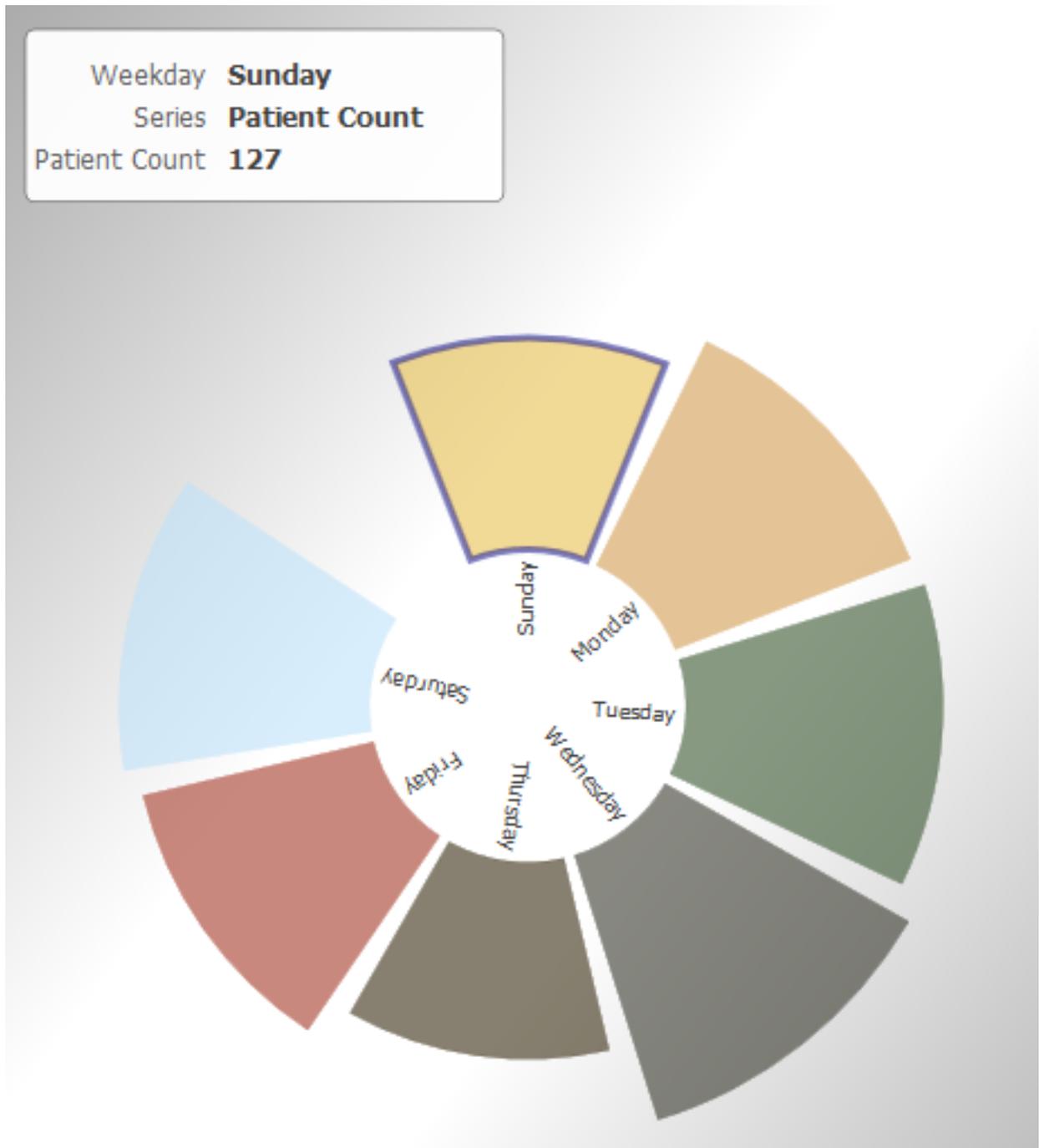
	Patient Count
Diabetic Patients	46
All Patients	1,000

If the data source has multiple columns, be sure to enable the **Show Multiples** option. In this case, the widget displays a subdivided circle for each column in the data source.

5.4.11 Swirl Charts

Note: Swirl charts are available only in dashboards, not in the Analyzer.

A swirl chart is similar to a pie chart. It displays one wedge (or group of wedges) for each row of the underlying pivot table. The following shows an example:



This chart is based on the following pivot table:

Weekday	Patient Count
Sunday	127
Monday	149
Tuesday	152
Wednesday	166
Thursday	118
Friday	140
Saturday	148

If the underlying pivot table has multiple columns, the swirl chart includes multiple wedges for any given row. For example:



This chart is based on the following pivot table:

Weekday	Patient Count	Allergy Count
Sunday	127	92
Monday	149	93
Tuesday	152	103
Wednesday	166	117
Thursday	118	75
Friday	140	79
Saturday	148	105

5.5 Displaying an Ensemble Business Metric

This section discusses suitable ways to display Ensemble business metrics in charts. There are four general possibilities:

- The metric defines only one instance (the default) and does not define *AUTOHISTORY* for any of the properties. In this case, when you display the metric as a pivot table, it appears with one row as follows:

✕ + Current Workflow Tasks, All Departments		
	Assigned	Unassigned
1	18	18

In this case, any kind of chart is suitable.

- The metric defines multiple instances and does not define *AUTOHISTORY* for any of the properties. In this case, when you display the metric as a pivot table, it appears as follows:

✕ + Current Workflow Task Count, by Department		
	Assigned	Unassigned
Department A	17	3
Department B	17	6
Department C	1	1

In this case, the most suitable chart types are [bar and column charts](#) and [combo charts](#).

- The metric defines only one instance (the default) and defines *AUTOHISTORY* for some or all of the properties. In this case, when you display the metric as a pivot table, it appears as follows:

✕ + Workflow Tasks Over Time, All Departments		
	Assigned	Unassigned
1	70	29
2	52	3
3	45	15
4	16	28
5	42	7
6	35	23

In this case, the most suitable chart types are [bar and column charts](#), [line charts](#), and [combo charts](#).

Or, if a given property does not specify *AUTOHISTORY*, the column for that property is empty except for the last row. You might prefer to display only the properties that specify *AUTOHISTORY*. To do so, you can add a [filter control](#) and filter the data to show only those properties. Note that you can hide this control from the user.

- The metric defines multiple instances *and* defines *AUTOHISTORY* for some or all of the properties. In this case, when you display the metric as a pivot table, the *AUTOHISTORY* is ignored by default, and the metric appears as follows:

	Assigned	Unassigned
Department A	10	6
Department B	16	6
Department C	7	7

In this case, the most suitable chart types are [bar and column charts](#) and [combo charts](#).

To force the metric to display the *AUTOHISTORY*, you can add a [filter control](#) and filter the data to show only one instance. Note that you can hide this control from the user. When filtered to display one instance, the metric appears as follows:

Department			
		Department C	Q
	Assigned	Unassigned	
1	13	13	
2	4	4	
3	9	9	
4	11	11	
5	11	11	
6	12	12	

In this case, the most suitable chart types are [bar and column charts](#), [line charts](#), and [combo charts](#).

6

Customizing the Appearance of a Chart Widget

DeepSee provides a rich set of options that you can use to customize the appearance of a chart widget. To access these options, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” and use the options in the **Chart and Pivot** section. The following submenus affect the appearance of a chart:

- **Chart Settings > Chart Type** — Enables you to select a chart type.
- **Chart Settings > Size & Appearance** — Provides options to control the [chart overall](#).
- **Chart Settings > Titles & Labels** — Provides options to control the text and formatting of [titles and labels](#) on the chart.
- **Chart Settings > No Data Warnings** — Provides options to control [an optional warning](#) if there is no data to display.
- **Chart Settings > Colors & Style** — Provides options to control [colors and line styles](#) in the chart.
- **Chart Settings > x Axis** — Provides options to customize the text and formatting of the [x-axis](#) of the chart.
- **Chart Settings > y Axis** — Provides options to customize the text and formatting of the [y-axis or y-axes](#) of the chart.
- **Chart Settings > Series Details** — Provides options to customize how each [series](#) (row of the data source) is displayed.
- **Chart Legend** — Provides options to disable, enable, and customize the [chart legend](#).

The following subsections provide details.

Note: The easiest way to systematically alter the appearance of dashboards is to define and use themes, which are discussed in “[Defining and Using Themes](#)”. Themes use the options described in many of the following subsections.

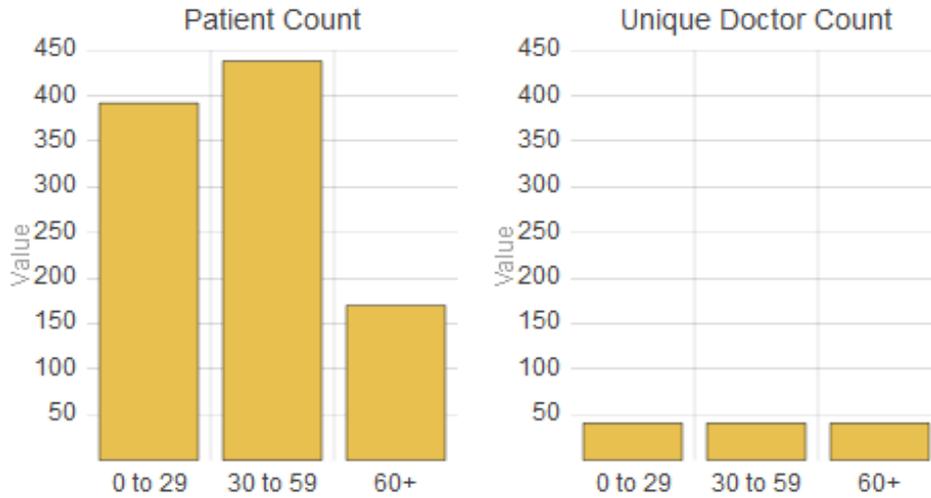
6.1 Size and Appearance Options

To customize the overall size and appearance of a chart, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Chart Settings > Size & Appearance**, and use the options listed there. The following options are available:

- **Margins** — Use this submenu to customize the margins of the chart. Use the **Top**, **Left**, **Right**, and **Bottom** sliders to separately control the top, left, right, and bottom margins. For each margin, use the **Units** setting to specify the margin either as a percentage of the widget size or in pixels.

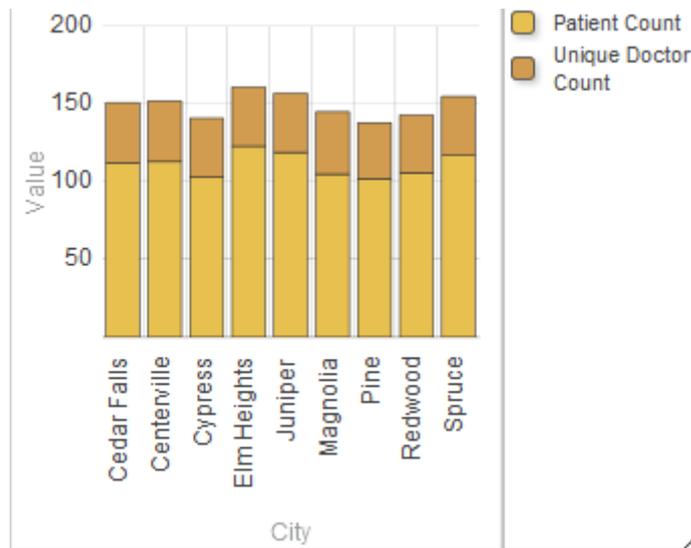
Or specify **Units** as **Auto** to use the default margins.

- Multiples** — Select this option to display one chart for each series (row of the data source). For example:



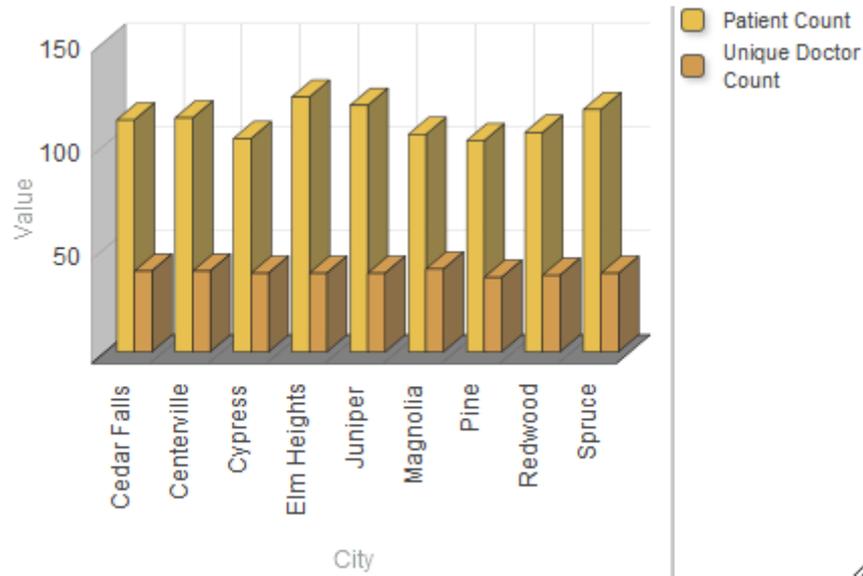
This option is recommended for [pie charts](#) and [tree maps](#).

- Pivot** (applies only to charts that have x- and y-axes) — Select this option to switch the x- and y-axes.
- Stacked** (applies only to charts that have x- and y-axes) — Select this option to display the series (rows of the data source) in stacked form. For example:



This option affects only the series that are displayed as bars.

- Filled** (applies only to [line and area charts](#)) — Select this option to control whether DeepSee fills in the area below the series lines on the chart.
- Appearance** — Use this option to control whether the chart is displayed with a three-dimensional appearance. For example:

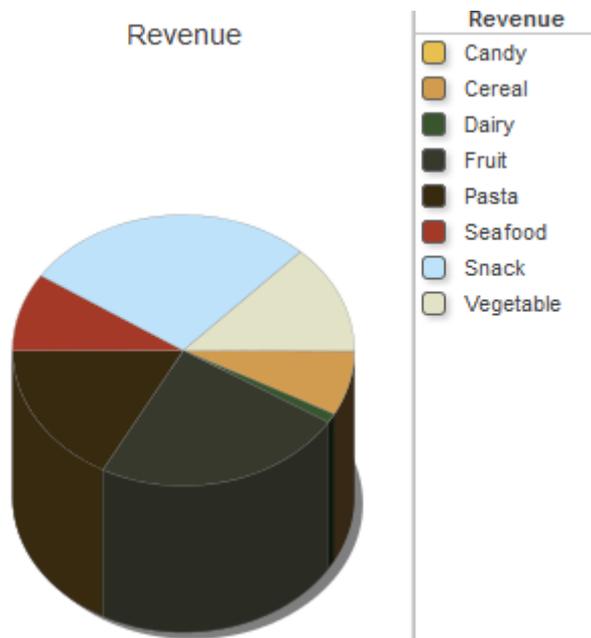


- **Zoom** — Enable this option to display zoom buttons on the lower left of the chart. For example:

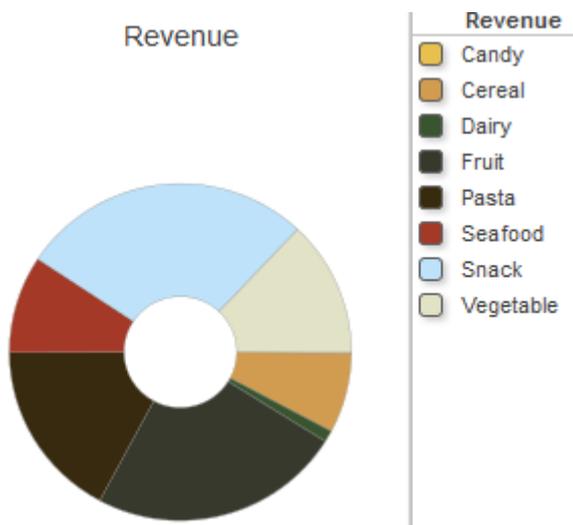


This option applies only to charts that have x- and y-axes.

- **Pie Height** (applies only to [pie charts](#)) — Use this option to control the height of the pie chart, if **Appearance** is **3D**. For example:

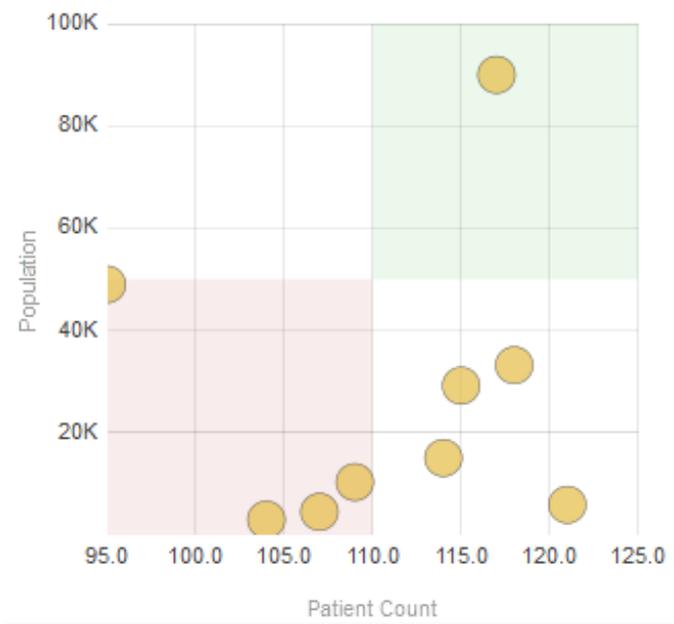


- **Pie Size** (applies only to [pie charts](#)) — Use this option to control the radius of the pie chart.
- **Pie Hole Size** (applies only to [pie charts](#)) — Use this option to control the size of the pie hole. For example:

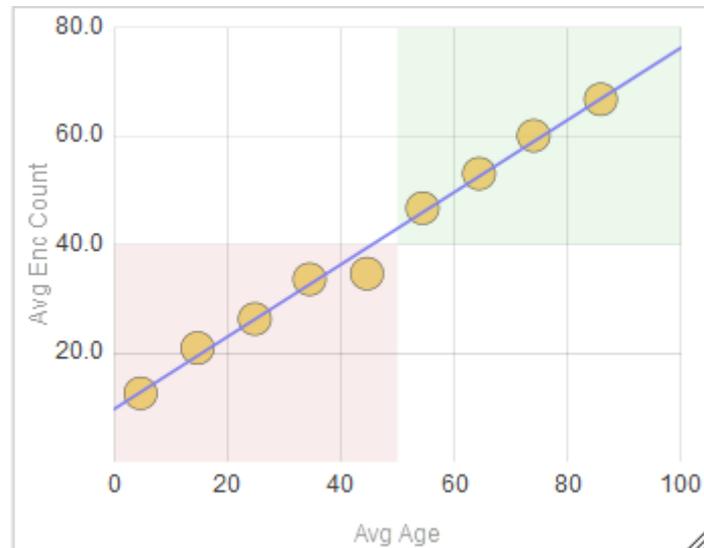


The default hole size is zero.

- **Quadrant** (applies only to [bubble charts](#)) — Select this option to divide the chart into quadrants, marked by different colors. For example:

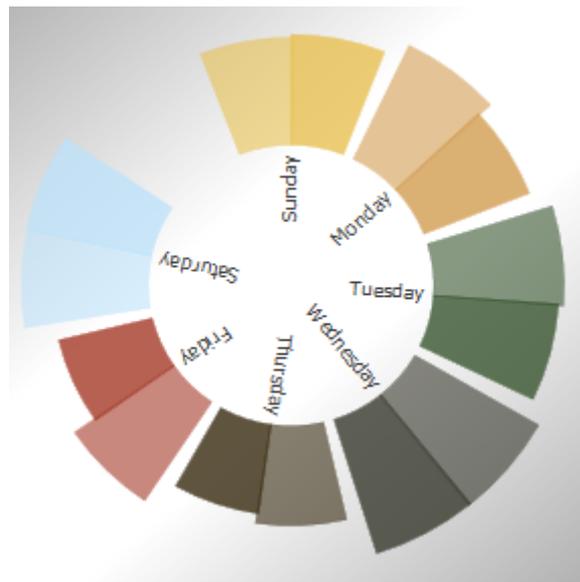


- **Radius** (applies only to [bubble charts](#)) — Use this option to control the basic size of the bubbles in a bubble chart.
- **Opacity** (applies only to [bubble charts](#)) — Use this option to control the opacity of the bubbles in a bubble chart.
- **Regression** (applies only to [bubble charts](#)) — Select this option to display a regression line on the chart. For example:

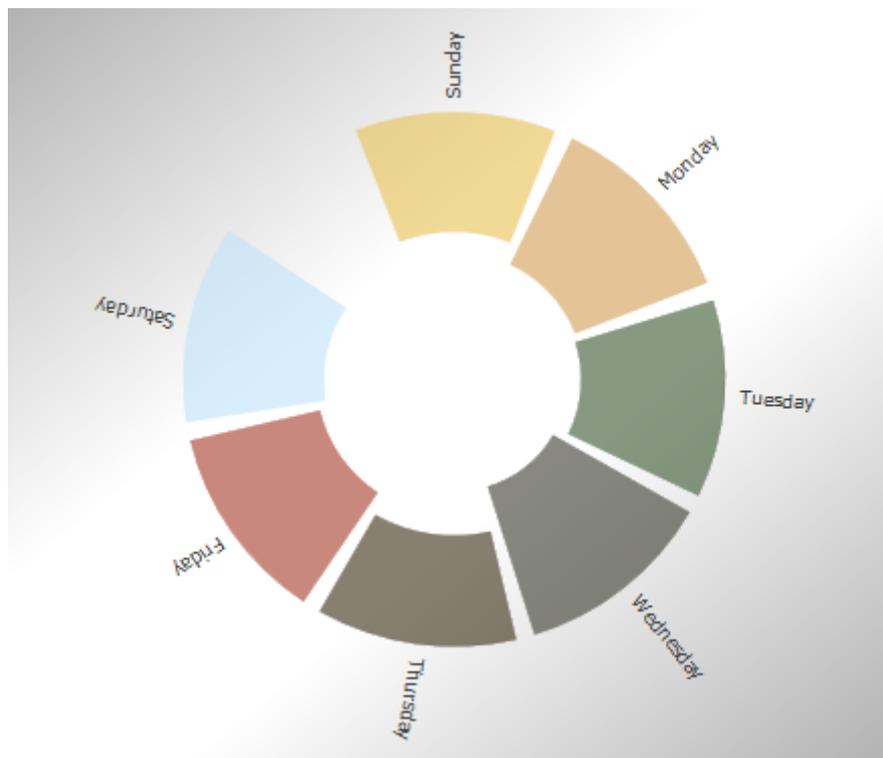


- **Separate Series Scale** (applies only to [swirl charts](#)) — Controls the scaling of the wedges, in the case when the underlying pivot table has multiple columns. By default, the chart uses a single scale to determine the size of each wedge, and the wedge for the maximum of one column is likely to be different from the wedge for the maximum of another column. (For examples, see **Plot By**.)

If **Separate Series Scale** is **On**, the chart uses a separate scale for each column, and the maximum wedge for any column is the same size, as in the following example (which uses both Patient Count and Allergy Count as columns) :



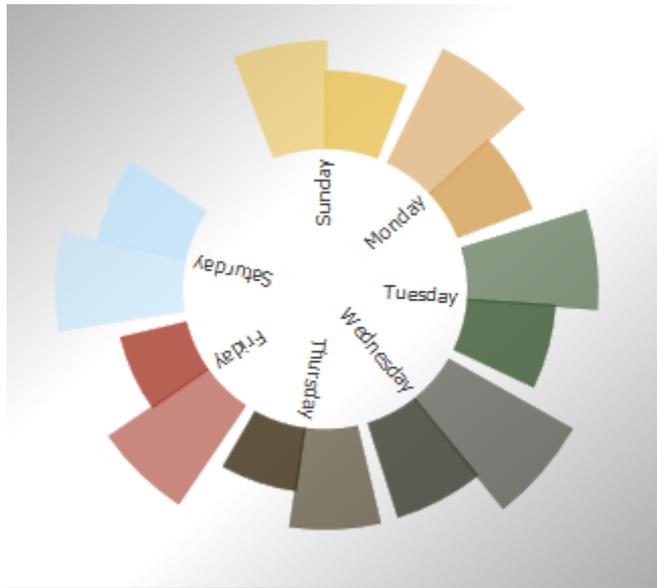
- **Invert** (applies only to [swirl charts](#)) — Click **On** to display the labels in the outer area of the chart, rather than within the center. For example:



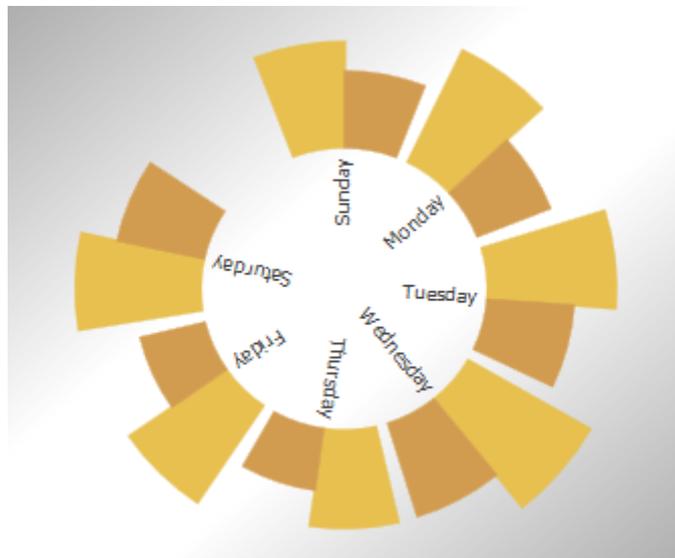
- **Gap** (applies only to [swirl charts](#)) — Use this to control the size of the gaps between the wedges. The following shows an example with a minimal gap:



- **Plot By** (applies only to [swirl charts](#)) — Controls how the wedges are colored; this is particularly useful if the underlying pivot table displays multiple columns. If you click **items**, the chart uses similar colors for the all wedges that belong to the same row in the pivot table on which this chart is based. For example, the following swirl chart is based on a pivot table that uses Patient Count and Allergy Count for columns. Notice that, for example, Sunday has two wedges of similar color; one is for Patient Count and the other is for Allergy Count.



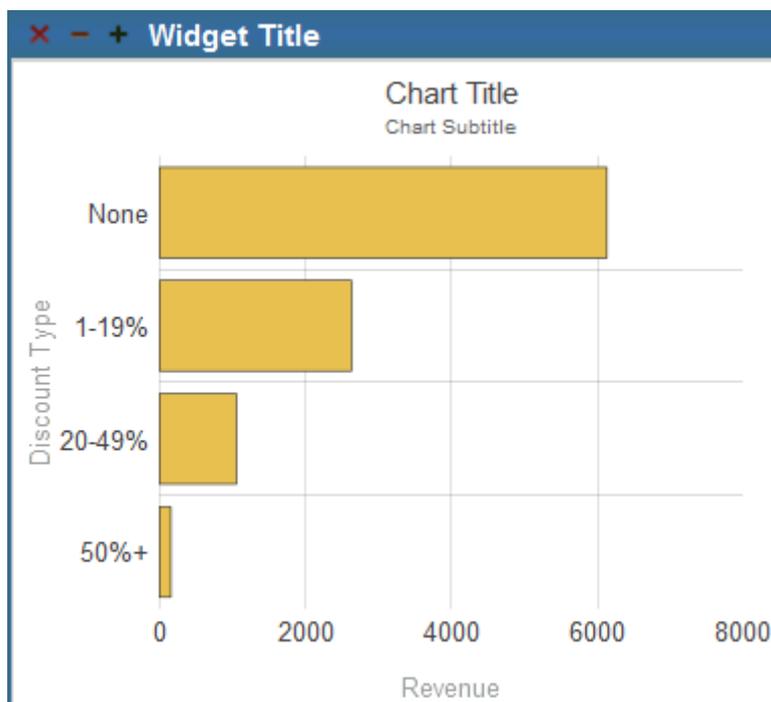
If you click **series**, the chart uses a single color for the all wedges that belong to the same column, so that the chart has a separate color for each column. For example, the following swirl chart is the same as the preceding except that **Plot By** is set to **series**:



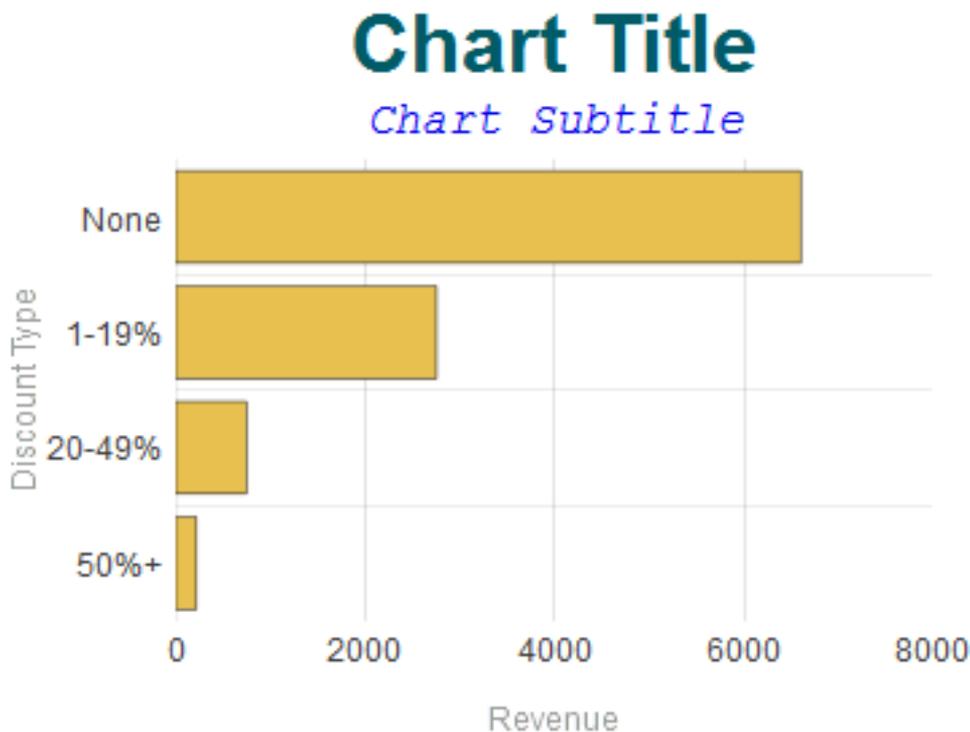
6.2 Title and Label Options

To customize the titles and labels of a chart, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Chart Settings > Titles & Labels**, and use the options listed there. The following options are available:

- **Title** — Use this option to specify an optional title within the chart. For example:



- **Title Style** — Use this submenu to control the style of the chart title. For example:

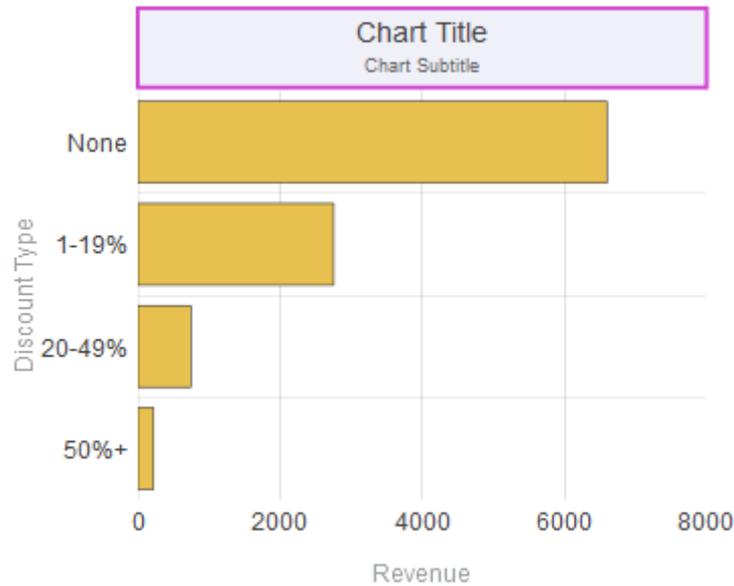


See “[Specifying Chart Text Styles](#),” earlier in this book.

- **Subtitle** — Use this option to specify an optional subtitle within the chart. See the example for **Title**.
- **Subtitle Style**— Use this submenu to control the style of the chart subtitle. See the example for **Title Style**.

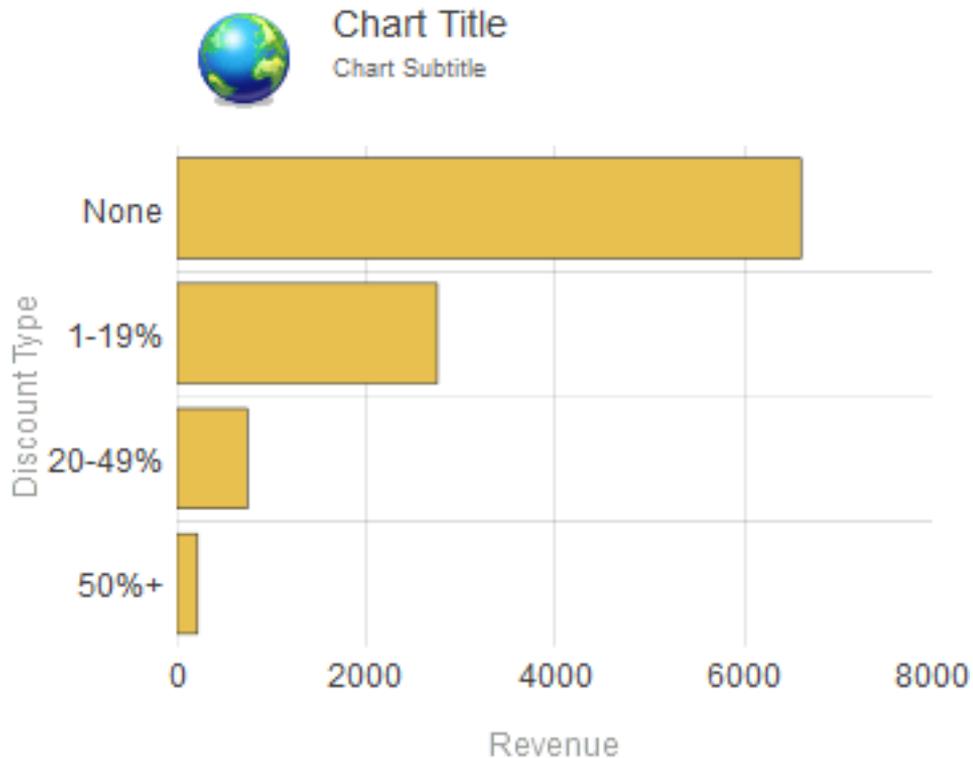
See “[Specifying Chart Text Styles](#),” earlier in this book.

- **Title Align** — Use this option to control the alignment of the chart title and subtitle. These titles can be left-aligned, centered, or right-aligned.
- **Title Box** — Use this submenu to add an optional box around the chart title and subtitle. For example:



See “[Specifying Line Styles](#),” earlier in this book.

- **Title Image** — Use this option to add an optional image to the title. For example:



In this example, the titles are also left-aligned (see the **Title Align** option).

- **Image Width, Image Height, Image Top, and Image Left** — Use these options to control the size and position of the optional title image.

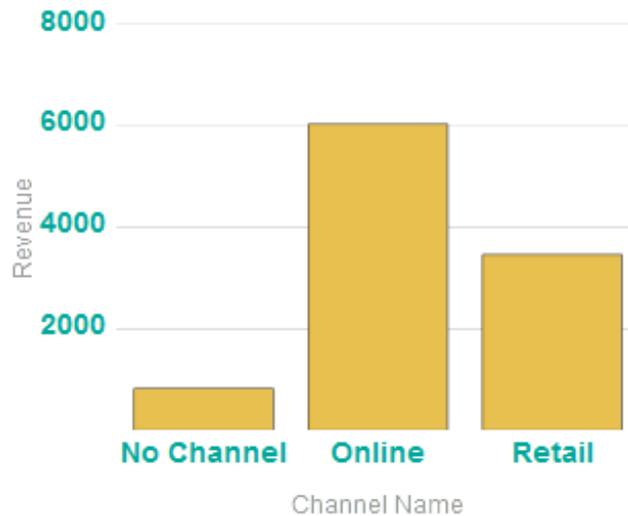
- **Labels** — Enable this option to display labels for the axes of the chart. The following shows a chart with labels displayed:



This example also uses the **Value Labels** option (which controls the labels shown at the top of each column).

To understand which labels are controlled by this option, recall that the axes of the chart are determined by the axes of the pivot table on which the chart is based. For example, the preceding chart is based on a pivot table whose x-axis (rows) displayed No Channel, Online, and Retail and whose y-axis (columns) displayed Revenue.

- **Label Style** — Use this submenu to control the style of the chart labels. For example:



See “[Specifying Chart Text Styles](#),” earlier in this book.

- **Label Length** — Use this option to specify the maximum length of the labels; any additional characters are truncated. For example:



In this example, **Label Length** is 5.

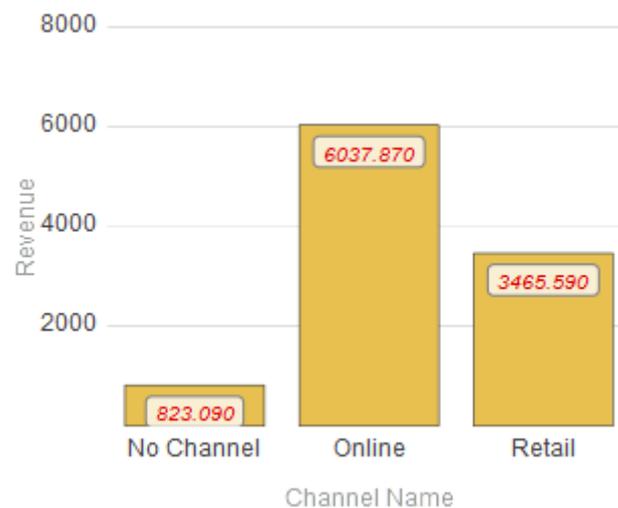
- **Axis Title**— Use this submenu to control the style of the axis titles. For example:



See “[Specifying Chart Text Styles](#),” earlier in this book.

You can also separately control the style of the *x-axis* and the *y-axis* or *y-axes*.

- **Value Labels** (applies only to [bar charts](#)) — Select this option to display a label above or next to each column, indicating the value displayed by the column. For examples, see **Labels** and **Value Style**.
- **Value Style**— Use this submenu to control the style of the value labels. For example:



This example also uses a non-default option (`#.###`) for **Value Format**. See the next item.

See “[Specifying Chart Text Styles](#),” earlier in this book.

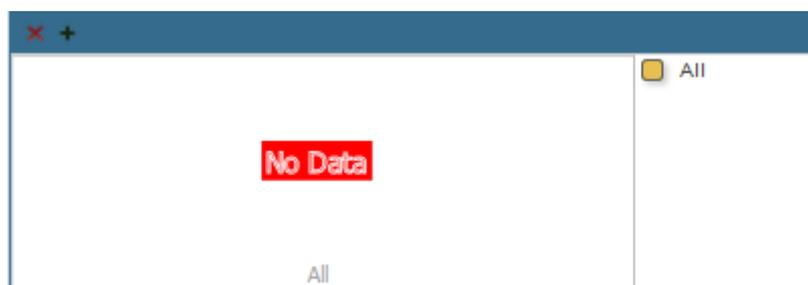
- **Value Format**— Use this option to control the numeric formatting of the value labels. See the example for **Value Style**. For details, see “[Specifying Numeric Format Strings](#),” earlier in this book.

6.3 No Data Warning

To customize the appearance of a chart when there is no data to display, select the widget as described in “[Reconfiguring a Widget](#),” click **Chart Settings > No Data Warning**, and use the options listed there. The following options are available:

- **No Data Message** — Type the message to display if there is no data to display in this chart.
- **Warn if no data?** — Select **ON** to display the given message if there is no data. The message is shown in a box in the widget as shown after this list. The default is **OFF**.
- **Background opacity** — Specify the opacity of the background of the box that contains the given message. The value 1 means 100%.
- **Background color** — Select the color of the box that contains the given message.
- **Message color** — Select the color of the message.

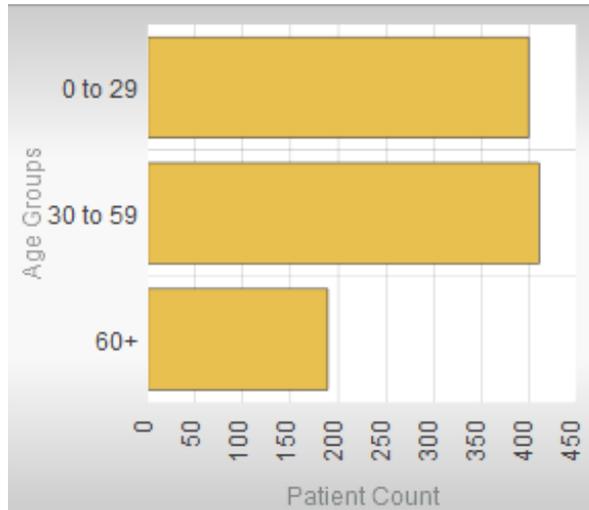
The following shows an example:



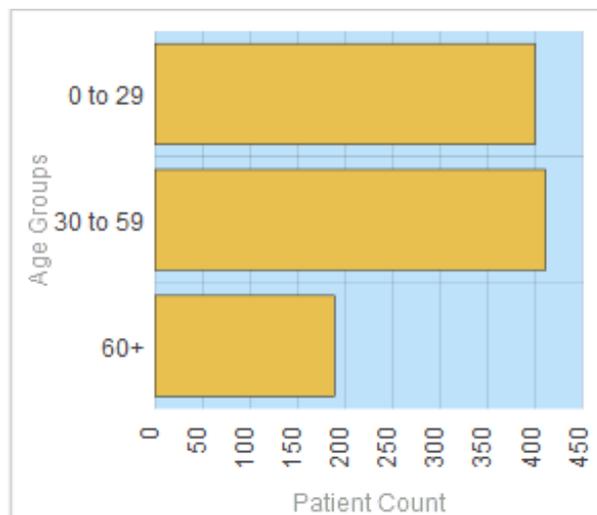
6.4 Color and Style Options

To customize the colors and line styles in a chart, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Chart Settings > Colors & Style**, and use the options listed there. The following options are available:

- **Series Scheme** — Select a scheme, which is a set of colors to use for the series shown in this chart.
- **Background** — Use this submenu to specify the style of the background that surrounds the chart. For example:

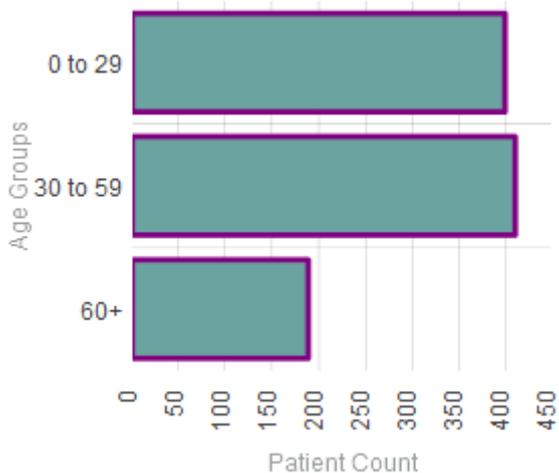


- **Plot Area** — Use this submenu to specify the style of the plot area of the chart. For example:



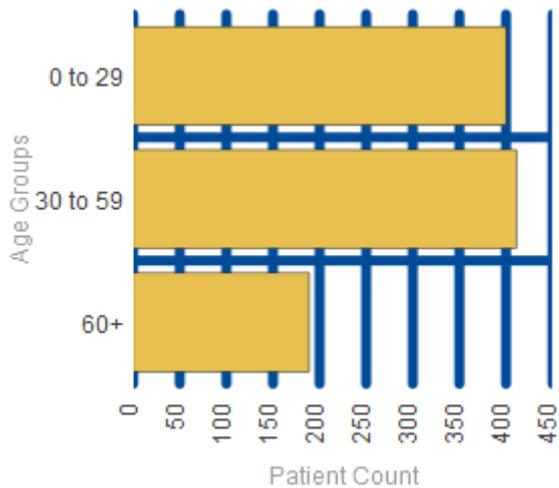
- **Plot Style** — Use this submenu to specify the style of the series that are plotted in the chart. This submenu affects only bar charts, area charts, and pie charts.

For example:

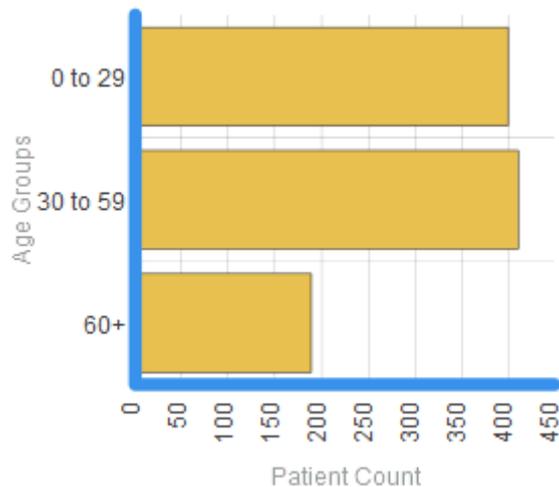


The **Plot Style** menu treats all series in the same way. Also note that this menu overrides the options in **Chart Settings > Series Details**; see “[Series Details](#),” later in this chapter.

- **Grid Style** — Use this submenu to specify the style of the grid (if any) displayed within the chart. For example:

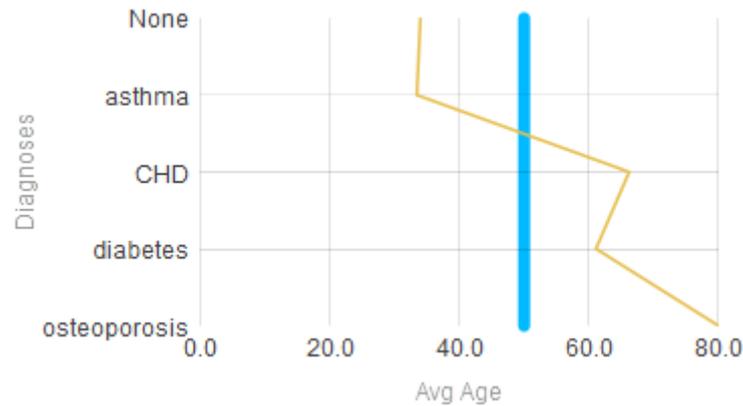


- **Axis Line Style** — Use this submenu to specify the style of the axis lines. For example:



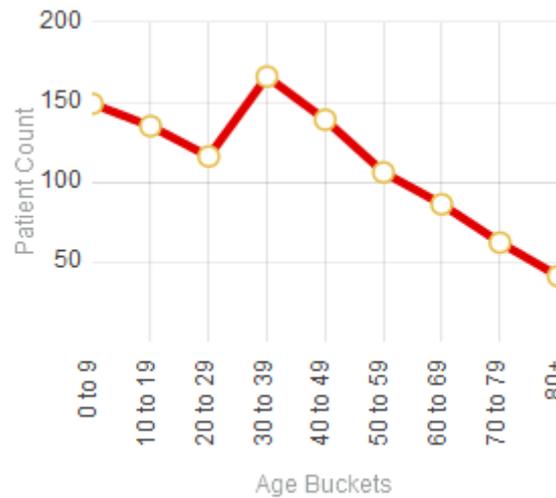
- **Base Line Style** — Use this submenu to specify the style of the base line (if any) in the chart.

For example:

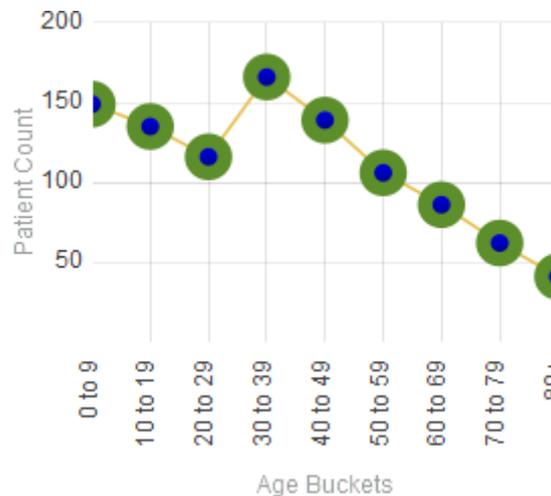


To specify a base line, see the **Base Value** option in the [next section](#).

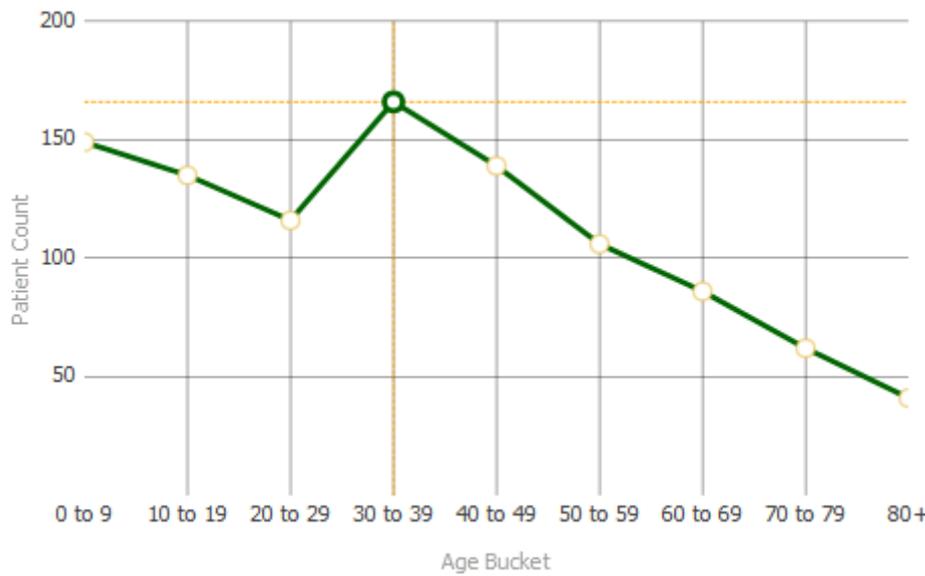
- **Line Style** — Use this submenu to specify the style of the lines (if any) displayed in the chart. For example:



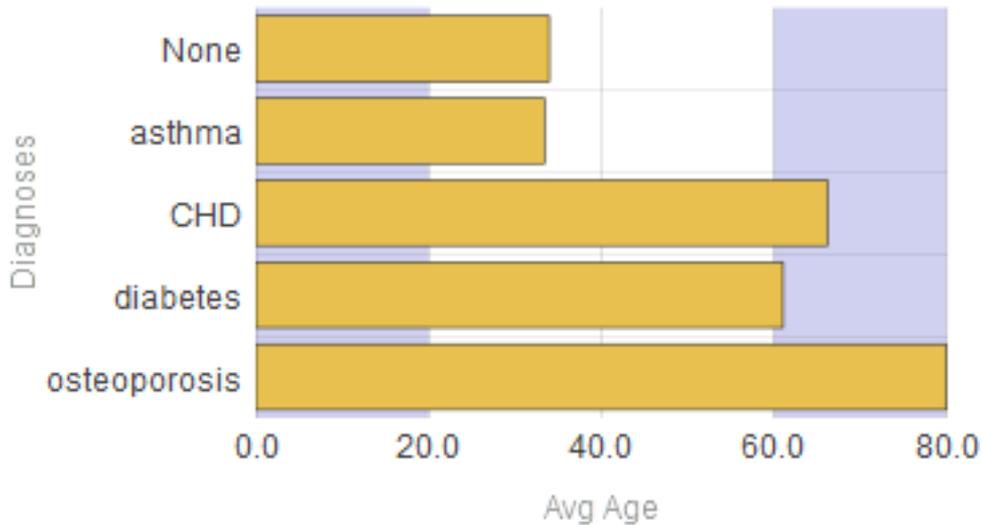
- **Markers Visible** — Enable this option to display markers in the chart.
- **Marker Style** — Use this submenu to specify the style of the markers (if any) displayed in the chart. For example:



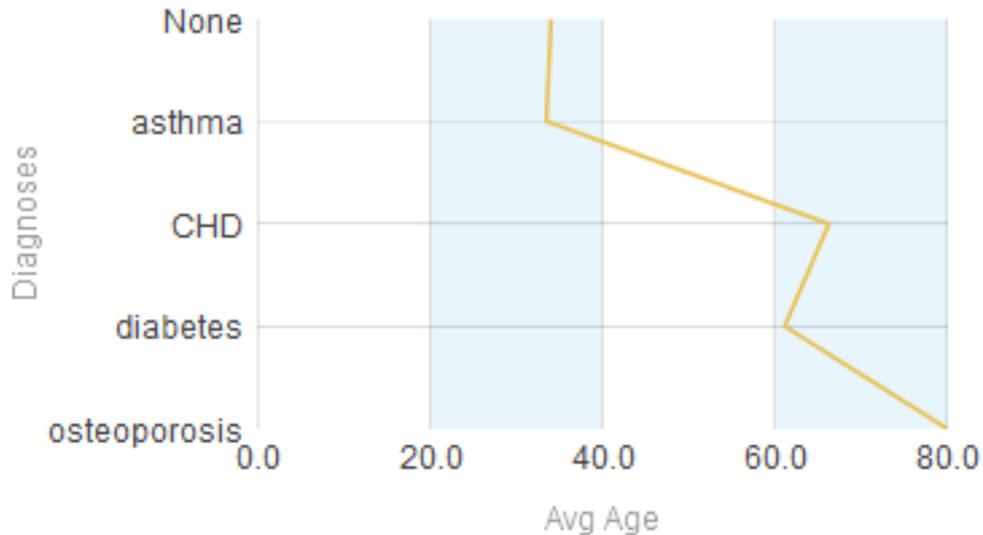
- **Marker Size** — Use this option to modify the size of the markers (if any).
- **Indicators Visible** — Enable this option to display crosshairs when a user selects an item in the chart. For example:



- **Indicator Style** — Use this submenu to specify the style of the indicator (if any) displayed in the chart. See the previous item.
- **Band Upper** — Type a value to define an upper area of this axis that is styled as a band. The following example uses both **Band Upper** (specified as 60) and **Band Lower** (specified as 20):

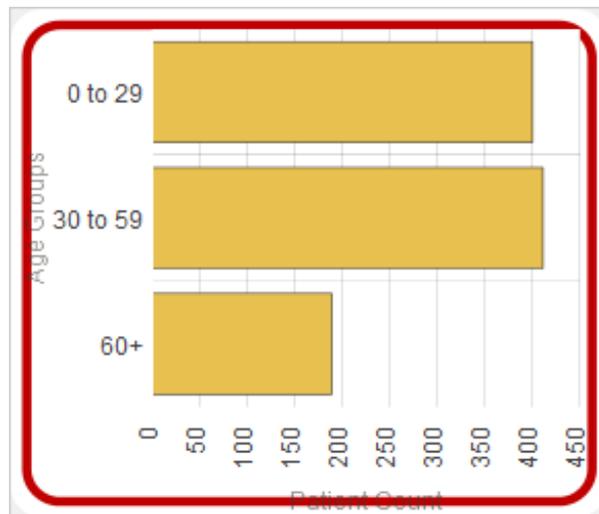


- **Upper Style** — Use this submenu to specify the style of the upper band, if any.
- **Band Lower** — Type a value to define a lower area of this axis that is styled as a band. See the example for **Band Upper**.
- **Lower Style** — Use this submenu to specify the style of the lower band, if any.
- **Stripes** — Use this option to add stripes to this axis. For example:



This example also uses **Stripe Style**.

- **Stripe Style** — Use this submenu to specify the style of the stripes, if any.
- **Border Style** — Use this submenu to specify the style of the border of the chart. The border surrounds the plot area, and is enclosed by the background. The following shows an example:



This example also uses a non-default value for **Border Radius**, discussed later in this list.

- **Border Offset** — Use this option to modify the position of the border, relative to the background.
- **Border Radius** — Use this option to modify the radius of the corners of the border. By default, these corners are right angles. For a non-default example, see **Border Style**.
- **Data-driven Colors, Termlist** option — Use this option to specify a set of CSS colors to use for specific series in this widget. To use this option, first use the Term List Manager (see *Defining DeepSee Models*) and define a term list as follows:
 - The key for each item must be the exact display name of a series in this widget.
 - The value for the item must be a CSS color value. You can use any legal CSS color name. You can find these at <https://www.w3.org/TR/css3-color/> and other locations on the Internet.

The following shows an example:

Terms		
	Key	Value
1	Avg Age	#4B0082
2	Avg Allergy Count	SeaGreen
3	Avg Test Score	Coral
4	Patient Count	rgb(255,0,0)

Then, for the **Data-driven Colors, Termlist** option, type the exact name of the term list.

If this term list does not include a term for a given series in the widget, DeepSee displays that series in gray.

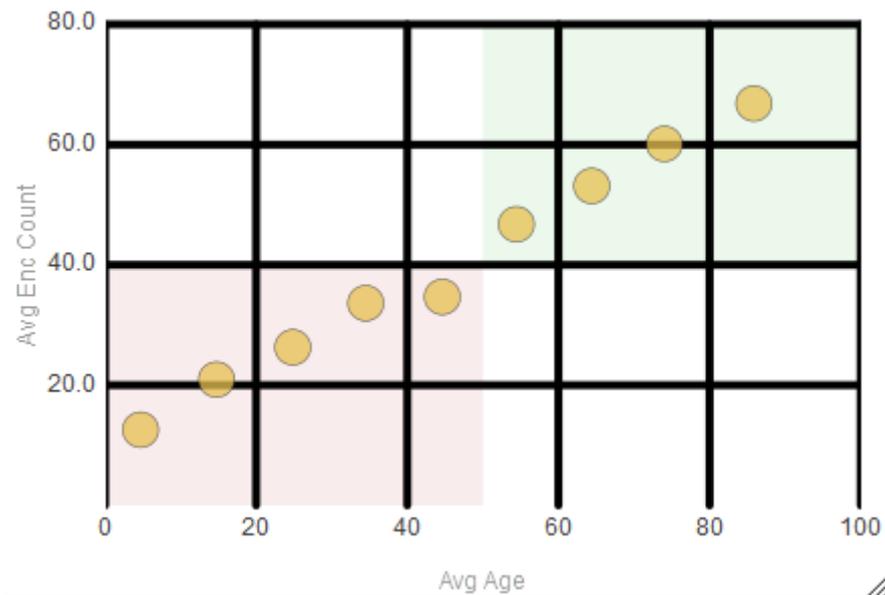
Note that this feature does not provide any support for localization.

For details on the submenus, see “[Specifying Line Styles](#),” earlier in this book.

6.5 x-Axis Options

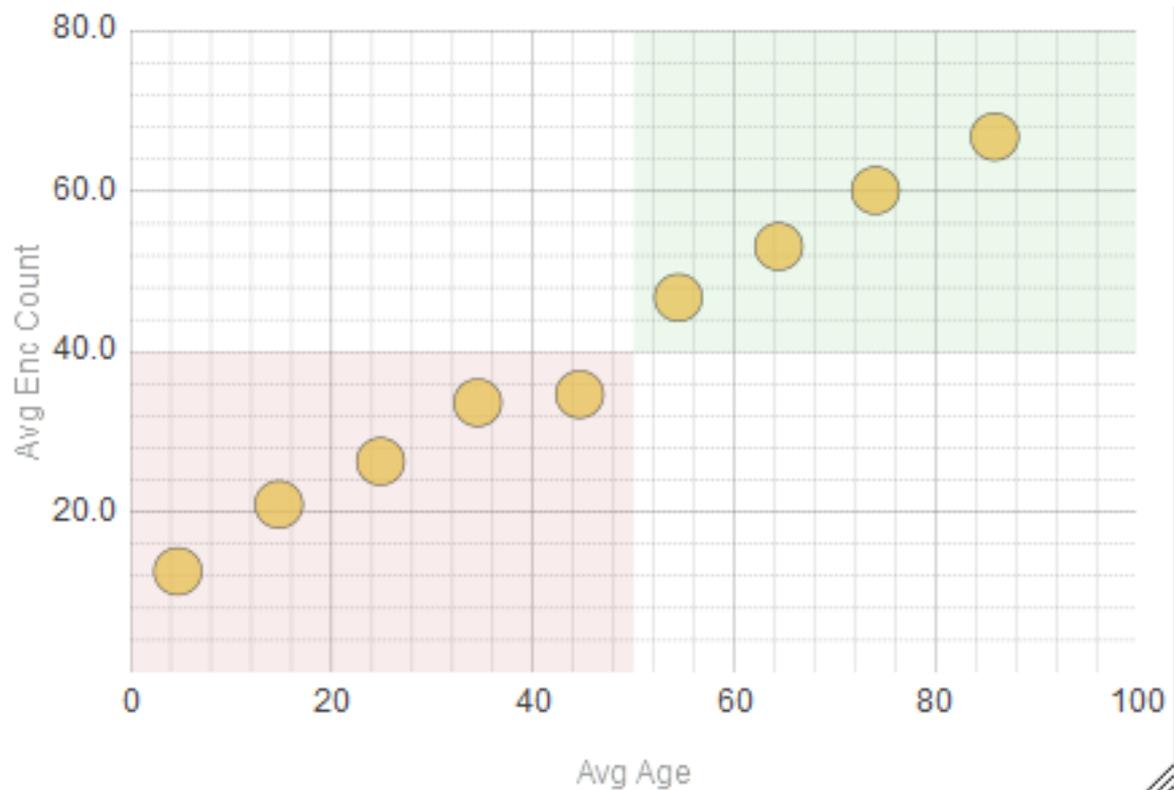
To customize the x-axis of a chart, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Chart Settings > x Axis**, and use the options listed there. The following options are available:

- **Axis Title** — Specify the title to use on the x-axis. By default, this title is taken from the data source.
- **Major Grid** — Use this option to enable or disable lines that mark major positions on the x-axis. By default, these lines are on.
- **Major Style** — Use this submenu to customize the style of the lines that mark major positions on the x-axis. In the following example, **Major Style** is specified for both the x- and y-axes:

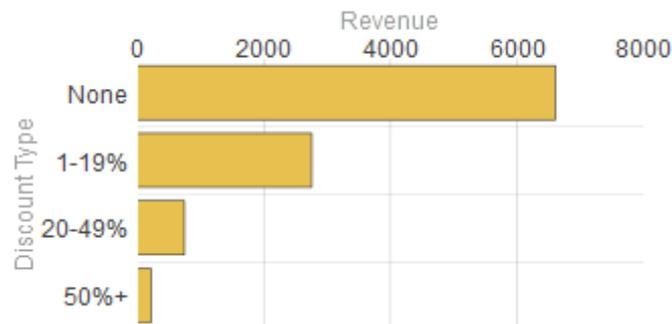


See “[Specifying Line Styles](#),” earlier in this book.

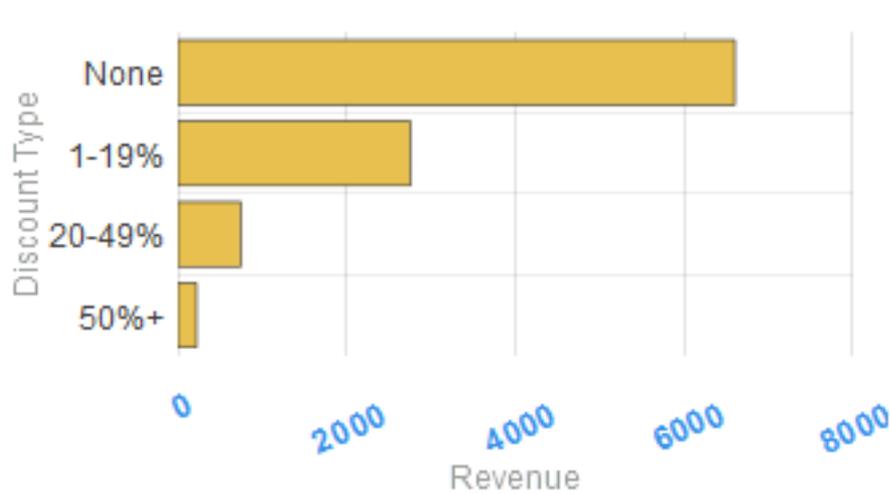
- **Minor Grid** — Use this option to enable or disable lines that mark minor positions on the x-axis. By default, these lines are off. In the following example, these lines are on for both the x- and y-axes:



- **Minor Style** — Use this submenu to customize the style of the lines that mark minor positions on the x-axis. See “[Specifying Line Styles](#),” earlier in this book.
- **Label Position** — Specifies the position (top or bottom) of the x-axis. By default, this axis is on the bottom. The following shows this axis at the top of the chart:



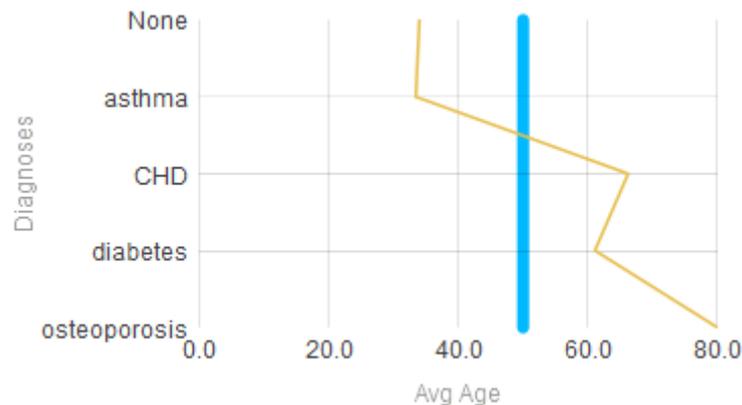
- **Label Style** — Use this submenu to specify the style of the labels of the x-axis. For example:



This example also uses the **Label Angle** option.

See “[Specifying Chart Text Styles](#),” earlier in this book.

- **Label Angle** — Use this option to specify the angle of the labels of the x-axis. See the example for **Label Style**.
- **Axis Type** — Optionally select % to express the values on the x-axis as percentages. If you select this option, DeepSee multiplies the values by 100 and displays % at the end of each value.
- **Min Value** — Use this option to specify the minimum x value shown in the chart. If you use this option, the chart is rescaled to start at this minimum x value, and lower data points are ignored.
- **Max Value** — Use this option to specify the maximum x value shown in the chart. If you use this option, the chart is rescaled to end at this maximum x value, and higher data points are ignored.
- **Base Value** — Use this option to add a reference line at the given value. For example, the following chart has a reference line at the value of 50:

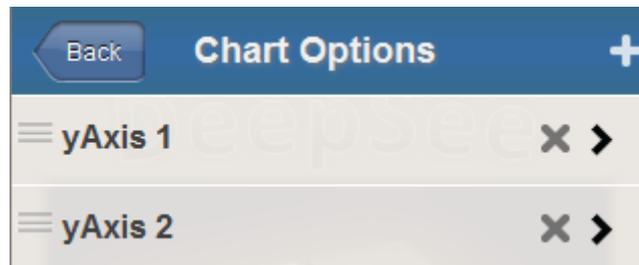


This example also uses the **Base Line Style** option in **Chart Settings > Colors & Style**; see the [previous section](#).

The same options are available when you customize a [y-axis](#).

6.6 y-Axis or Axes Options

To customize the y-axis or y-axes of a chart, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Chart Settings > y Axis**, and use the options listed there. This menu lists the y-axes currently defined in this chart. For example:



Here, you can do the following:

- Add a y-axis. To do so, click the plus sign button. DeepSee immediately adds a y-axis.
- Remove a y-axis. To do so, click the X button in the row for that axis.
- Rearrange the y-axes. To do so, drag the axis handles to the left of the axis names.
- Customize a y-axis. To do so, click its name to display a submenu of options. For details on this submenu, see the [previous section](#).

Some options, include **Major Grid**, **Major Style**, **Minor Grid**, and **Minor Style**, are supported only for the primary y-axis.

6.7 Series Detail Options

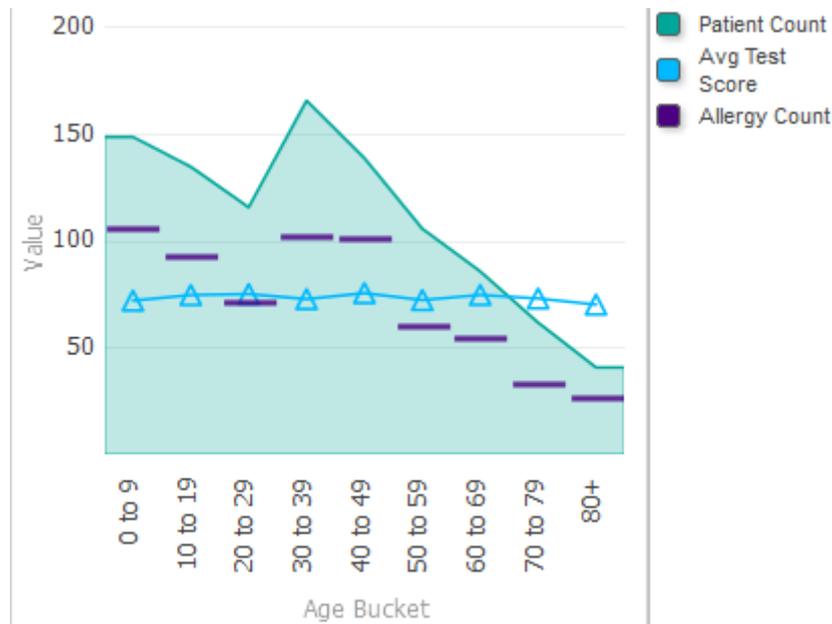
DeepSee uses a default set of colors that it uses for each series (a row in the data source). You can override these. For line charts, you can also customize the markers used for the data points.

To customize how each series is displayed in a chart, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Chart Settings > Series Details**, click a series, and then use the options listed there.

For each series, you can modify the following options (depending on the chart type):

- **Color** — Specify the color for this series.
For information on specifying colors, see “[Specifying Table Text Styles](#),” earlier in this book.
- **y Axis** — Specify the y-axis with which to associate this series (either 1 or 2).
Note that both y-axes are displayed in the same position by default. To change this position; see the **Label Position** option in **Chart Settings > Y Axis**.
- **Plot Type** (combo charts) — Specify the plot type to use for this series.
- **Marker Shape** (line, area, combo, and time charts) — Specify the marker shape to use for data points displayed in this series.

The following shows an example:



6.8 Chart Legend Options

To customize the legend of a chart, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Chart Legend**, and use the options listed there. The following options are available:

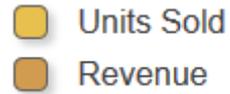
- **Position** — Select an option to specify the location (if any) of the chart legend. Select **right**, **left**, **top**, **bottom**, or **none**.
- **Title** — Specify the title to show in the chart legend, if you want to override the default title, which is based on the chart definition.
- **Show Title** — Select **ON** to display the chart legend title. Or select **OFF** to hide it.
- **Border** — Use this submenu to specify the border around the chart legend. The following shows an example:



- **Background** — Specify the color of the background of the chart legend. See the preceding example.
- **Opacity** — Specify the opacity of the chart legend. The value 1 means 100%.
- **Font** — Specify the typeface of the chart legend.
- **Font Size** — Specify the size of the text of the chart legend.
- **Text Color** — Specify the style of the text of the chart legend. The following shows an example that uses this option and other style options:

A legend bar with a teal background. It contains two items: a yellow square followed by the text "Units Sold" and an orange square followed by the text "Revenue". Both items have a subtle drop shadow.

- **Text Style** — Specify the style of the text of the chart legend.
- **Padding** — Specify the horizontal and vertical padding before and after the legend.
- **Width** — Specify the width of the legend.
- **Height** — Specify the height of the legend.
- **Show Box Shadow** — Control whether the legend displays a faint shadow for each box in the legend. The shadow is displayed by default. The following magnified view shows an example:



If **Show Box Shadow** is **OFF**, the same legend appears as follows:



7

Adding Meter Widgets

This chapter describes how to add meter widgets to a dashboard. It discusses the following topics:

- [How to add a meter widget](#)
- [How to customize common widget settings for a meter widget](#)
- [Meter types](#)
- [Basic meter options](#)
- [How to configure speedometers](#)
- [How to configure text meters](#)
- [How to configure fuel gauges](#)
- [How to configure traffic lights](#)
- [How to configure light bars](#)
- [How to configure smileys](#)

7.1 Adding a Meter Widget

To add a meter widget:

1. Click **Menu > Add New Widget....**
2. Click **Meter**.
3. Click a meter type.
4. For **Data source**, do one of the following:
 - Click the Search button  and then select a data source.
See “[Kinds of Data Sources](#),” earlier in this book.
 - Use the **Link to** drop-down list to select another widget on the same dashboard.

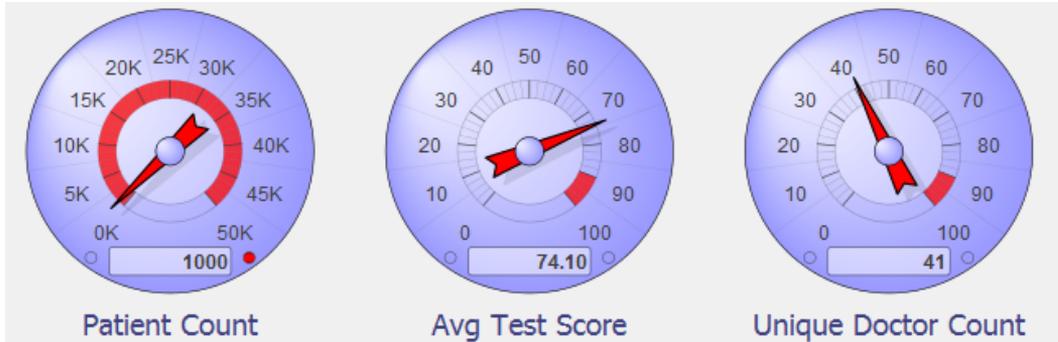
Most DeepSee data sources can have multiple rows and columns. When you display any of these in a meter widget, the widget uses only the first row of data. Any additional rows are ignored.

5. Optionally specify the basic settings, as described in “[Adding a Widget to a Dashboard](#),” earlier in this book.

Also see “[Basic Meter Options](#),” in this chapter.

6. Click **OK**.
7. Add at least one property, as described in “[Adding Properties](#),” earlier in this book.

This property becomes a meter within the meter widget. For example, the following meter widget has three properties:



By default, the type of each meter is the same as the type of the parent meter widget, but you can modify the type for individual meters.

For additional details, see the [following subsection](#).

8. Optionally configure the widget further as described in the rest of this chapter, which discusses the following kinds of meters:
 - [Speedometers](#)
 - [Text meters](#)
 - [Fuel gauges](#)
 - [Traffic lights](#)
 - [Light bars](#)
 - [Smileys](#)

For examples and a comparison, see the [next section](#).

9. Optionally add controls, as described in the chapter “[Adding Widget Controls](#).”

7.1.1 Data Sources, Properties, and Meters

A *meter widget* has multiple *properties*, each of which is displayed as a separate meter within the widget. Consider the preceding example, which shows a widget with three properties, each shown in a separate speedometer.

Most DeepSee data sources can have multiple rows and columns. When you display any of these in a meter widget, the widget uses only the first row of data. Any additional rows are ignored.

For the first row of data, each column of the data source can be displayed as a property of the meter widget, that is, as a separate meter within the widget. In the examples shown here, all the meters in a meter widget are of the same type, but you can configure the widget to use different types within a single widget.

7.2 Specifying Widget Settings for a Meter

For a meter, the **Widget Settings** menu has the following options:

- **Print** — Select this option to include the Print button  on the toolbar. This button enables users to print the widget to a PDF file. For further details and requirements, see the section “[Customizing Print Settings for a Widget.](#)”
- **Analyzer** — Select this option to include the Mini Analyzer button  on the toolbar. This button enables the users to open the data source in the Mini Analyzer.

This option is supported only when the data source is a pivot table.

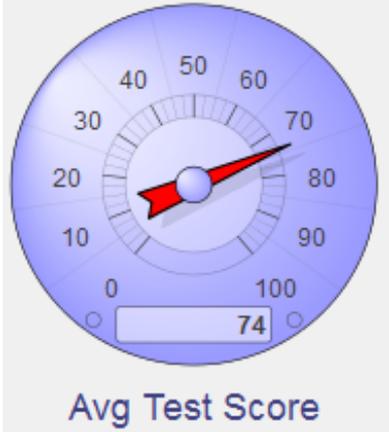
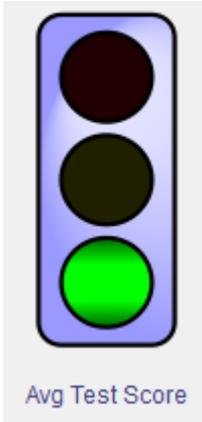
This menu includes additional options that control the look of the meter; the options depend on the kind of meter. The rest of the chapter provides the details.

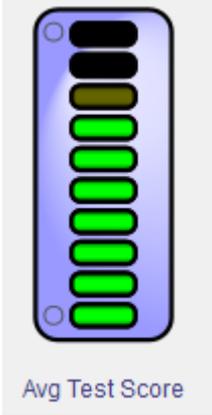
Note: These options add a button to the toolbar. If you use these options, be sure the toolbar is visible (its display is controlled via the **Toolbar** option in **Settings**). See “[Reconfiguring a Widget,](#)” earlier in this book.

7.3 Meter Types

DeepSee provides the following types of meter:

Meter Type	Example	Basics	Notes
------------	---------	--------	-------

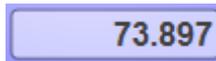
Meter Type	Example	Basics	Notes
Speedometer		Range options control the possible positions of the needle	<ul style="list-style-type: none"> The actual value is always shown in the box. You can add thresholds within the scale, shown as different color bands in the scale. You can add a target value shown as a line on the scale.
Text meter		Meter displays the actual value and a label	
Fuel gauge		Range options control the possible positions of the needle	<ul style="list-style-type: none"> The actual value is always shown in the box. You can add thresholds within the scale, shown as warning lights at the top and bottom.
Traffic light		<i>Threshold</i> options control the use of the three lamps	<ul style="list-style-type: none"> By default, a high value is considered “bad” and is shown as a red lamp. You can reverse the sense of the meter by reversing the high and low range values.

Meter Type	Example	Basics	Notes
Light bar		Range options control the bars and their colors	<ul style="list-style-type: none"> By default, a high value is considered “bad” and is shown with all bars lit, including a red bar at the top. You can reverse the sense of the meter by reversing the high and low range values. You can add thresholds within the scale, shown as warning lights at the top and bottom.
Smiley face		Range options control the possible shapes of the smile or frown	<ul style="list-style-type: none"> By default, a high value is considered “good” and is shown as a smile. You can reverse the sense of the meter by reversing the high and low range values.

7.4 Basic Meter Options

For any meter widget, you can customize a set of basic options. To customize the widget, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Meters**, and use the options listed there. The following submenus are available:

- Settings** — Use this submenu to customize the entire meter widget. This submenu provides the following options:
 - Format** — Select an option to control the numeric formatting of all meters in this meter widget. For example:



See “[Specifying Numeric Format Strings](#),” earlier in this book.

–

Analyzer — Use this option to control whether this widget displays the Mini Analyzer  button.

This option is supported only when the data source is a pivot table.

- **High value color** — Select the color to use for the section of the meter that displays values higher than the **Upper Threshold** option, which you specify in the *meter name* submenu.

This option applies only to speedometers.

- **Low value color** — Select the color to use for the section of the meter that displays values lower than the **Lower Threshold** option, which you specify in the *meter name* submenu.

This option applies only to speedometers.

- **Background color** — Select the color to use for the background around the meter widget.
- **Background opacity** — Use this option to specify the opacity of the background. The value 1 means 100%.
- *meter name* — Use this submenu to customize an individual meter ([property](#)) within the meter widget. This submenu provides the following options:
 - **Type** — Select the meter type. The default type is the meter type that you specify in the parent widget, as described in the [previous section](#).
 - **Data Property** — Select the property to display in this meter.
 - **Label** — Specify the label for this meter.
 - **Label Style** — Use this submenu to specify the color, font size, typeface, and so on for the label of this meter. For example:

Avg Test Score

- **Apply Style to All Meters** — Use this option to copy the styles from this meter to all other meters within this meter widget.

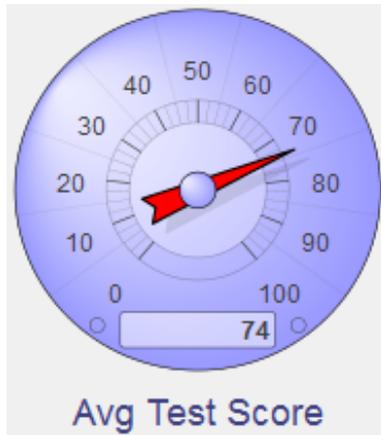
Other options depend on the kind of meter; see the following sections:

- [Speedometers](#)
- [Text meters](#)
- [Fuel gauges](#)
- [Traffic lights](#)
- [Light bars](#)
- [Smileys](#)

Note: The easiest way to systematically alter the appearance of dashboards is to define and use themes, which are discussed in “[Defining and Using Themes](#)”. Themes use many of the options described in this chapter.

7.5 Customizing Speedometers

The speedometer is a round gauge with a needle that moves from lower left, all the way around to lower right, to indicate a value within a specific range. The gauge has tick marks with numeric labels. In the following example, the gauge extends from 0 to 100:



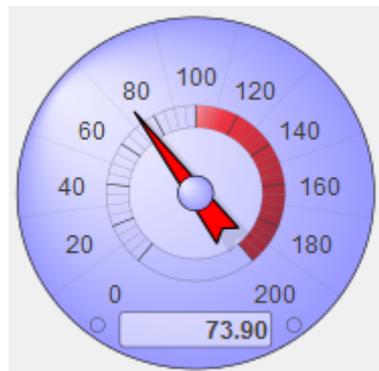
The speedometer includes an odometer box at the bottom; this box always displays the actual value from the data source, even if that value does not fall within the range of the gauge.

To customize a speedometer, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Meters**, click a meter name, and use the options listed there. The following options are available:

- *Common options* — See “[Customizing Meters](#),” earlier in this chapter.
- **Upper Range** and **Lower Range** — Use these options to specify the lowest and highest numbers displayed on the speedometer gauge. In the following example, these options are 20 and –20, respectively.

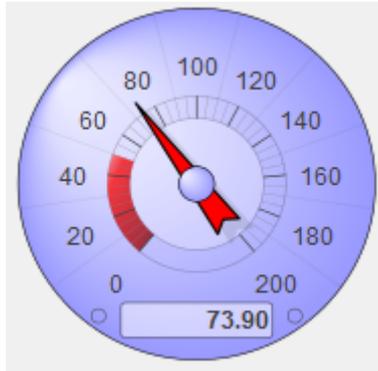


- **Upper Threshold** — Use this option to specify the upper threshold value for the speedometer gauge; values above this are displayed in a contrasting color. In the following example, this option is 100:



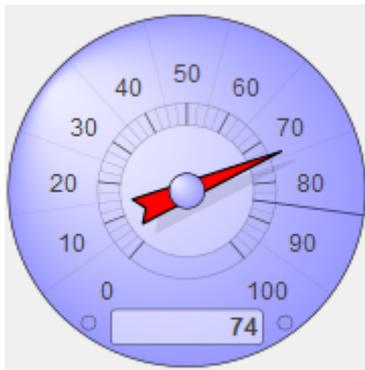
Use the **Widget Settings** menu to control the color used in the threshold area. See “[Customizing Meters](#),” earlier in this chapter.

- **Lower Threshold** — Use this option to specify the lower threshold value for the speedometer gauge; values below this are displayed in a contrasting color. In the following example, this option is 50:

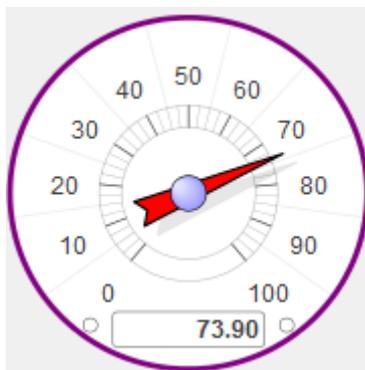


Use the **Widget Settings** menu to control the color used in the threshold area. See “[Customizing Meters](#),” earlier in this chapter.

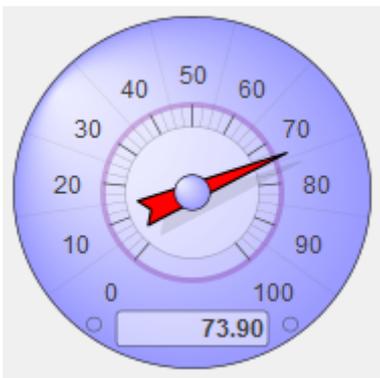
- **Target Value** — Use this option to add a target line to the speedometer gauge. For example:



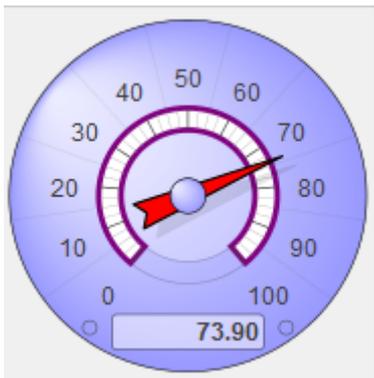
- **Outer Circle** — Use this submenu to specify the color and style of the outer circle of the speedometer. For example:



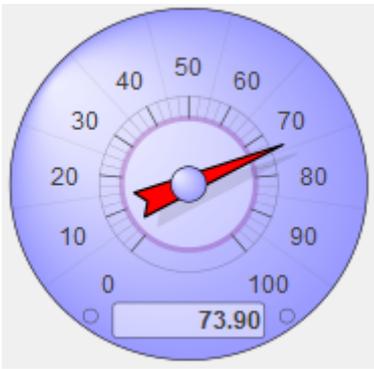
- **Mid Circle** — Use this submenu to specify the color and style of the middle circle of the speedometer. For example:



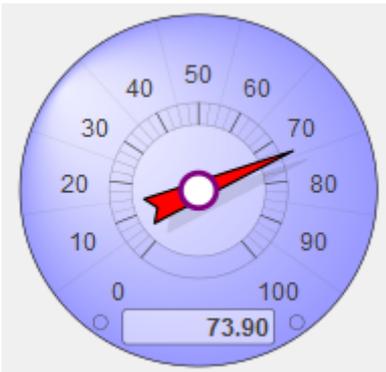
- **Ring** — Use this submenu to specify the color and style of the ring just within the middle circle. For example:



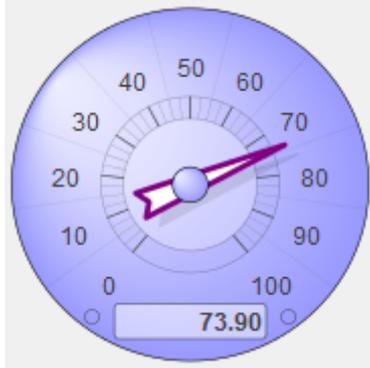
- **Inner Circle** — Use this submenu to specify the color and style of the inner circle. For example:



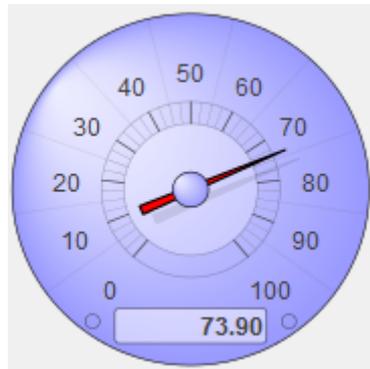
- **Nub** — Use this submenu to specify the color and style of the nub at the center of the speedometer. For example:



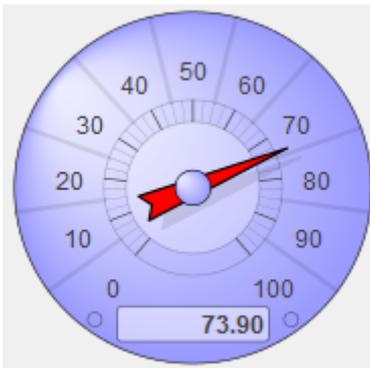
- **Needle** — Use this submenu to specify the color and style of the needle. For example:



- **Thin Needle** — Use this option to display a thin needle. For example:



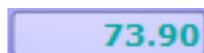
- **Separator** — Use this submenu to specify the color and style of the separators that radially divide the speedometer. For example:



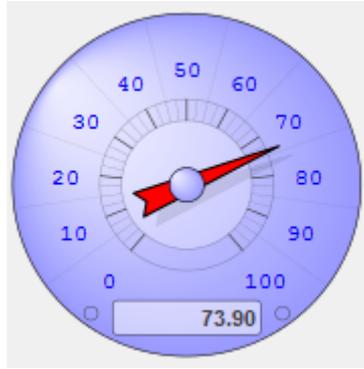
- **Odometer Box** — Use this submenu to specify the color and style of the odometer box. For example:



- **Odometer** — Use this submenu to specify the color and style of the value shown in the odometer box. For example:



- **Value Style** — Use this submenu to specify the color and style of the values marked on the speedometer gauge. For example:



Note: The easiest way to systematically alter the appearance of dashboards is to define and use themes, which are discussed in “[Defining and Using Themes](#)”. Themes use many of the options described in this chapter.

7.6 Customizing Text Meters

The text meter displays the given value as text:



To customize a text meter, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Meters**, click a meter name, and use the options listed there. The following options are available:

- *Common options* — See “[Customizing Meters](#),” earlier in this chapter.
- **Background** — Use this submenu to customize the outline and fill of the meter. For example:



- **Fill** — Use this submenu to customize the appearance of the value shown in the meter. For example:



- **Low Style** — Use this submenu to customize the appearance of the value, when the value is below **Lower Threshold**.
- **High Style** — Use this submenu to customize the appearance of the value, when the value is above **Upper Threshold**.

Note: The easiest way to systematically alter the appearance of dashboards is to define and use themes, which are discussed in “[Defining and Using Themes](#)”. Themes use many of the options described in this chapter.

7.7 Customizing Fuel Gauges

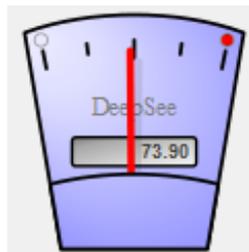
The fuel gauge is a narrow, vertical gauge with a needle that moves from left to right to indicate a value within a specific range (specified by **Lower Range** and **Upper Range**). The gauge distributes marks or “ticks” proportionally across its range, with **Lower Range** at left.

The fuel gauge displays its current value in a text box at the center of the gauge.

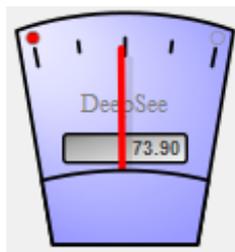


To customize a fuel gauge, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Meters**, click a meter name, and use the options listed there. The following options are available:

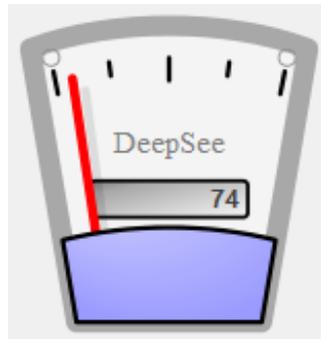
- *Common options* — See “[Customizing Meters](#),” earlier in this chapter.
- **Upper Range** and **Lower Range** — Use these options to specify the range of values represented by tick marks on the fuel gauge.
- **Upper Threshold** — Use this option to add a warning light to the right end of the meter. This warning light changes color when the meter value is equal to or greater than **Upper Threshold**. For example:



- **Lower Threshold** — Use this option to add a warning light to the left side of the meter. This warning light changes color when the meter value is equal to or lower than **Lower Threshold**. For example:



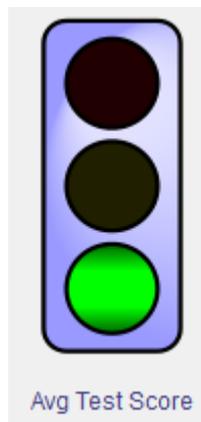
- **Body Style** — Use this submenu to control the outline and fill of the meter. For example:



Note: The easiest way to systematically alter the appearance of dashboards is to define and use themes, which are discussed in “[Defining and Using Themes](#)”. Themes use many of the options described in this chapter.

7.8 Customizing Traffic Lights

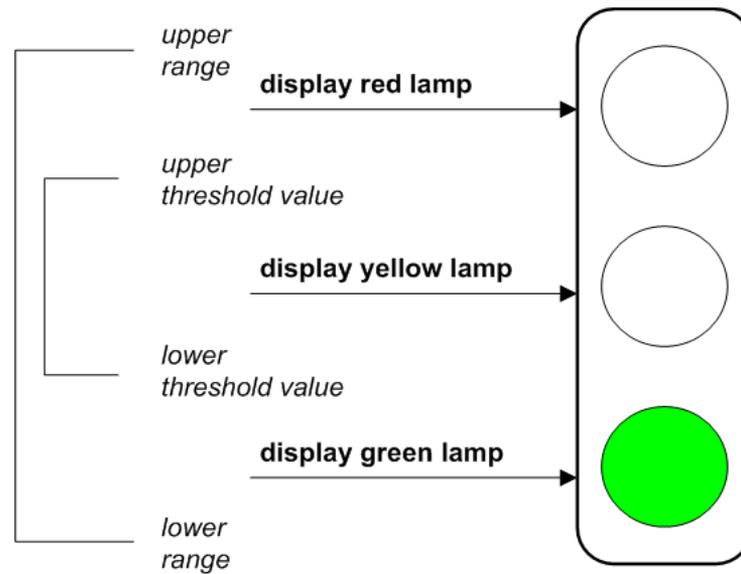
The traffic light consists of three round lamps in a column. From top to bottom, the lamps are red, yellow, and green.



To customize a traffic light, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Meters**, click a meter name, and use the options listed there. The following options are available:

- *Common options* — See “[Customizing Meters](#),” earlier in this chapter.
- **Upper Range, Lower Range, Lower Threshold** and **Upper Threshold**, — Use these options to control the three lamps in the traffic light. If **Upper Range** is greater than **Lower Range** (the default), this meter is lit as follows:
 - When the meter value is at or below **Lower Threshold**, the bottom lamp shows green.
 - When the meter value is between **Lower Threshold** and **Upper Threshold**, the center lamp shows yellow.
 - When the meter value is at or above **Upper Threshold**, the top lamp shows red.

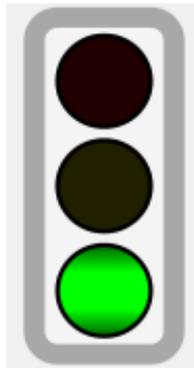
The following figure demonstrates this logic:



This system is useful when a large number indicates a condition that requires attention.

Sometimes a *low* number indicates a condition that requires attention. If so, you can reverse the sense of the meter; to do so, set **Lower Range** to a larger number than **Upper Range**. Then the red lamp indicates low values, and the green lamp indicates high values. When you reverse the range values, do not also reverse the threshold values.

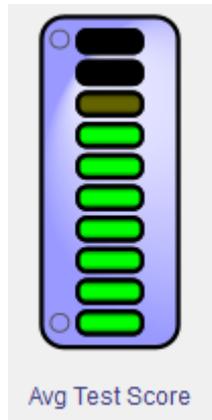
- **Body Style** — Use this submenu to control the outline and fill of the meter. For example:



Note: The easiest way to systematically alter the appearance of dashboards is to define and use themes, which are discussed in “[Defining and Using Themes](#)”. Themes use many of the options described in this chapter.

7.9 Customizing Light Bars

The light bar provides a stack of lamps arranged in a vertical bar. The light bar is similar to the traffic light, but its larger number of lamps provide a sense of movement from one end of the scale to the other.



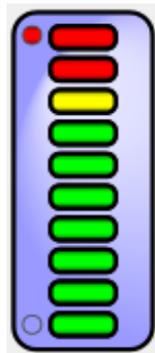
To customize a light bar, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Meters**, click a meter name, and use the options listed there. The following options are available:

- *Common options* — See “[Customizing Meters](#),” earlier in this chapter.
- **Upper Range** and **Lower Range** — Use these options to specify the range of values represented by the light bar.

The light bar is intended to appear “off” when the value is low, and fully lit when the value is high. The color of the bar shades from green (at the bottom of the scale) to yellow and red (at the top of the scale), implying that you should stop and address a problem when the value is high. This system is useful when a large number indicates a condition that requires attention.

Sometimes a *low* number indicates a condition that requires attention. If so, you can reverse the sense of the meter; to do so, set **Lower Range** to a larger number than **Upper Range**. Then the fully lit lamp indicates low values, and the “off” lamp indicates high values. When you reverse the range values, do not also reverse the threshold values.

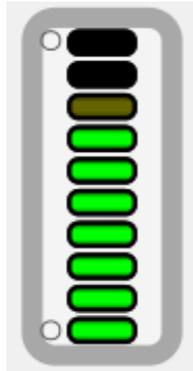
- **Upper Threshold** — Use this option to add a warning light to the upper end of the meter. This warning light changes color when the meter value is equal to or greater than **Upper Threshold**. For example:



- **Lower Threshold** — Use this option to add a warning light to the lower end of the meter. This warning light changes color when the meter value is equal to or lower than **Lower Threshold**. For example:



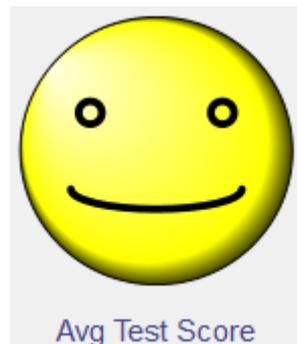
- **Body Style** — Use this submenu to control the outline and fill of the meter. For example:



Note: The easiest way to systematically alter the appearance of dashboards is to define and use themes, which are discussed in “[Defining and Using Themes](#)”. Themes use many of the options described in this chapter.

7.10 Customizing Smiley Faces

The smiley face is the familiar yellow circle with two eyes and a smile. The mouth line changes depending on the value.



To customize a smiley face, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Meters**, click a meter name, and use the options listed there. The following options are available:

- *Common options* — See “[Customizing Meters](#),” earlier in this chapter.
- **Upper Range** and **Lower Range** — Use these options to specify the range of values represented by the face, as follows:
 - The meter displays a smile when the meter value is near **Upper Range**.

- The mouth is a horizontal line at the midpoint within the given range.
- The meter displays a frown when the meter value is near **Lower Range**.

This system is useful when a large number indicates a positive condition, and a small number indicates a negative condition.

Sometimes a low number indicates a condition that requires attention. If so, you can reverse the sense of the meter. To do so, set **Lower Range** to a larger number than **Upper Range**. Then the smile occurs at low values, and the frown at high values.

- **Persona** — Use this option to control the type of face shown in this meter. For example:



- Note:** The easiest way to systematically alter the appearance of dashboards is to define and use themes, which are discussed in “[Defining and Using Themes](#)”. Themes use many of the options described in this chapter.

8

Adding Scorecard Widgets

This chapter how to add scorecard widgets to a dashboard. This chapter discusses the following topics:

- [Overview of scorecards](#)
- [How to add a scorecard widget](#)
- [How to customize widget settings for a scorecard widget](#)
- [How to add a column to a scorecard](#)
- [How to configure a column in general](#)
- [How to specify a scorecard formula](#)
- [How to configure item numbers](#)
- [How to configure row names](#)
- [How to configure values](#)
- [How to configure trend lines](#)
- [How to configure plot boxes](#)
- [How to add arrows](#)
- [How to add lamps](#)
- [How to further customize the appearance of a scorecard](#)

8.1 Overview of Scorecards

A *scorecard widget* displays a set of rows like the following:

	Region	Current	Target	% of Target			% of Target
1.	ASIA	7	10	70.00%	▲	●	
2.	EUROPE	6	8	7.50%		●	
3.	N. AMERICA	11	16	68.75%	▲	●	
4.	S. AMERICA	3	6	50.00%	▼	●	

You configure the columns of the scorecard, which can be the following:

- For a KPI or Ensemble business metric, you can use each property of that item as a column of the scorecard.
- For a pivot table, you can use each column of that item as a column of the scorecard.

In any case, you select the items to use as columns of the scorecard, and you specify the order of the columns.

For information on creating pivot tables, see [Using the DeepSee Analyzer](#). For information on creating KPIs, see the [DeepSee Implementation Guide](#). For information on creating Ensemble business metrics, see [Developing Ensemble Productions](#).

8.1.1 One-Measure Pivot Tables in Scorecards

Note that if you create a pivot table that displays only one measure, you must create that pivot table in a specific way if you want to display that pivot table in a scorecard. Specifically, drag and drop the measure to the **Columns** box in the Analyzer, rather than the **Measures** box.

Rows	Columns	Measures	Comments
<i>Anything</i>	One measure	<i>Nothing</i>	This pivot table can be displayed in a scorecard.
<i>Anything</i>	<i>Nothing</i>	One measure	This pivot table cannot be displayed in a scorecard. In this case, the measure is used as an MDX filter and cannot be configured as a property of the scorecard.

8.2 Adding a Scorecard Widget

To add a scorecard widget:

1. Click **Menu > Add New Widget...**
2. Click **ScoreCard**.
3. Click either **Regular** or **Big**.
4. For **Data source**, do one of the following:
 - Click the Search button  and then select a data source. See “[Kinds of Data Sources](#),” earlier in this book.
 - Use the **Link to** drop-down list to select another widget on the same dashboard.
5. Optionally specify the basic settings, as described in “[Adding a Widget to a Dashboard](#),” earlier in this book.
6. Click **OK**.
7. Optionally enable or disable the Mini Analyzer. For this, access to the **Widget Settings** menu and use the **Analyzer** option. This option controls whether this widget displays the Mini Analyzer  button. This option is supported only when the data source is a pivot table.
8. Add at least one scorecard column, as described in the [next section](#).
9. Optionally configure the widget further as described in the rest of this chapter.

10. Optionally add controls, as described in the chapter “[Adding Widget Controls.](#)”

8.3 Specifying Widget Settings for a Scorecard

For a scorecard, the **Widget Settings** menu has the following options:

- **Print** — Select this option to include the Print button  on the toolbar. This button enables users to print the widget to a PDF file. For further details and requirements, see the section “[Customizing Print Settings for a Widget.](#)”
- **Analyzer** — Select this option to include the Mini Analyzer button  on the toolbar. This button enables the users to open the data source in the Mini Analyzer.

This option is supported only when the data source is a pivot table.

Note: These options add a button to the toolbar. If you use these options, be sure the toolbar is visible (its display is controlled via the **Toolbar** option in **Settings**). See “[Reconfiguring a Widget,](#)” earlier in this book.

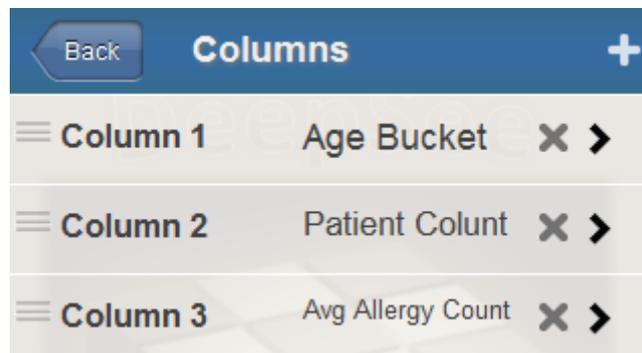
8.4 Adding Columns

A scorecard does not display any data until you add columns. A scorecard column is typically based on a property of the data source. In some cases, though, a scorecard column simply displays the item name or number and does not use a property of the data source.

To add a column to a scorecard:

1. Access the Dashboard Editor and select the widget as described in “[Reconfiguring a Widget.](#)”
2. Click **Scorecard**.
3. Click **Columns**.

The system displays a list of any columns defined for this scorecard. For example:



4. Click the plus sign + button.
The system displays a dialog box where you specify the column.
5. For **Value**, select a property of the data source, type a numeric constant, or type a formula.

For information on formulas, see “[Specifying a Scorecard Formula](#),” later in this chapter.

6. Optionally specify the following additional items:

- **Label** — Type a label or optionally click the check box to the right of **Label**. If you click the check box, DeepSee uses the localized name of this property as the label; this option applies only when **Value** is a data source property.

If you type a label, note that you can include `\n` to force a line break in the label.

- **Hidden** — Select this option to hide this value.
- **Value Column** — Select this option to configure the value in this column as the value of the row. This value is passed to any custom actions; see “[Specifying the Value of a Row](#),” later in this book.

Select this option for only one data property of the widget.

- **Format** — Type a format string. See “[Specifying a Numeric Format String](#),” later in this chapter.

7. Click **OK**.

8. Configure the column as described in the rest of this chapter.

For information on deleting and rearranging columns, see “[Managing Widget Properties](#),” earlier in this book.

8.5 Configuring a Scorecard Column

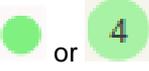
To configure a scorecard column:

1. Access the Dashboard Editor and select the widget as described in “[Reconfiguring a Widget](#).”
2. Click **Scorecard**.
3. Click **Columns**.

The system displays a list of any columns defined for this scorecard.

4. Select the column to configure.
5. Specify options as described after this list.
6. Save the dashboard.

The **Display** option controls most of the other choices for a property. For **Display**, you can choose one of the following:

Display Option	Sample	See
Item Number	1.	“Adding a Column of Row Numbers”
Label	Cedar Falls	“Adding a Column of Row Labels”
Value	1126	“Adding a Column of Values”
Arrow		“Adding Arrows”
Lamp	 or 	“Adding Lamps”
Trend Line		“Adding Trend Lines and Trend Bars”
Trend Bar		
Plot Box		“Adding Plot Boxes”
Hidden	<i>Not applicable; this property is not visible.</i>	

The other options are as follows:

- **Value** — Specifies the value, if any, to display in this column. Select a property of the data source, type a numeric constant, or type a formula.

For information on formulas, see [“Specifying a Scorecard Formula,”](#) later in this chapter.

You can omit this option if **Display** is **Item Number** or **Label**.

- **Label** — Specifies a header for this column. You can include \n to force a line break in this label.
- **Cell Caption** — Specifies an optional caption to place within each cell in this column.

For example:

Product	Revenue
Candy	Cell Caption \$128.56
Cereal	Cell Caption \$47.41
Dairy	Cell Caption \$90.46

- **Header Align** — Specifies how to align the column header for this column.
- **Align** — Specifies how to align the values in this column.
- **Width** — Specifies the width of this column.

You can also modify the widths of the columns in the **Size & Appearance menu** of the Dashboard Editor.

- **Show As** — Controls whether you display the value itself or the value as a percentage of another value, and so on. Select one of the following:
 - **Value** — Displays **Value**. This is the default setting.

- **Conditional** — Displays **Value** if the given property is defined. If the property is not defined, the column is not displayed.
- **Sum** — Displays the sum of the values given by **Value**.
- **Target** — Displays **Target Value**.
- **% of Sum** — Displays **Value** as a percentage of the total values given by **Value**.
- **% of Target** — Displays **Value** as a percentage of the **Target Value** for the same row.
- **Summary** — Controls what to display, if anything, in the footer, for this column. Select one of the following:
 - **Sum** — The summary value is the sum of the values in the column.
 - **Avg** — The summary value is the average of the values in the column.
 - **Max** — The summary value is highest value in the column.
 - **Min** — The summary value is the lowest value in the column.
 - **Value** — The summary value is specified in terms of the summary values for other columns. If you select this option, the system displays the **Summary Value** field. In this field, specify a formula that refers to other columns in the scorecard. For example:

`=Sales-Cost`

To refer to a column, use the name that you specified for the **Label** of the column. For information on creating formulas, see “[Specifying Formulas](#),” earlier in this book.
- **Format** — Specifies how to display numeric values in this column. See “[Specifying a Numeric Format String](#),” earlier in this book.
- **Style** — Controls the color and line style of the graphical objects, if any, in this column. See “[Specifying Line Styles](#),” earlier in this book.
- **Min Value** and **Max Value** — Specify the minimum and maximum values to display if the column displays plot boxes, described [later](#) in this chapter.

Min Value also affects [arrows](#). **Max Value** also affects [lamps](#).
- **Target Value** — Specifies the value to compare with the data values. Select a property of the data source, type a numeric constant, or type a formula (see “[Specifying a Formula](#),” later in this chapter).

You can omit this if **Display** is **Item Number** or **Label**.
- **Lower** and **Upper** — Control the appearance of [plot boxes](#), described later in this chapter.
- **Value Column** — Select this if this column contains the value to pass to any custom actions.

8.6 Specifying a Scorecard Formula

You can use formulas in the **Value** and **Target Value** options for a scorecard column. The rules are as follows:

- The first character must be an equals sign (=)
- The formula can include the following pieces:
 - Names of data source properties (as given in the **Value** drop-down list)

You must enclose the name in square brackets if it includes a space or any character that would be interpreted as an operator.

For example:

```
=[Total Sales]/100
```

- Numeric constants such as 100
- String constants such as "my string"
- Standard mathematical operators: + (addition), - (subtraction), / (division), and * (multiplication)
- Logical comparison operators: > (greater than), >= (greater than or equal to), = (equal to), < (less than), and <= (less than or equal to)
- Parentheses

The values 0 and "0" are treated as false, and all other numeric and string values are treated as true.

- The formula can also include the following functions:
 - IF(*test-value,value-if-test-true,value-if-test-false*) evaluates a value (*test-value*) and returns *value-if-test-true* if the logical value is true or returns *value-if-test-false* if the logical value is false.

For example, the following formula returns 25:

```
=IF(1,25,99)
```

The following formula returns either "abc" or "def" depending on the value of the PatCount property:

```
=IF(PatCount>1130,"abc","def")
```

- AND(*value1,value2,value3, ...*) returns true if all the given values are true.
- OR(*value1,value2,value3, ...*) returns true if at least one of the given values is true.
- CONCAT(*value1,value2,value3, ...*) and returns the concatenation of the given values.

You can use formulas in the following locations as well:

- For a plot box, in the options **Min Value**, **Max Value**, **Lower**, **Upper**, and **Base Value**. See “[Adding Plot Boxes.](#)”
- For a column summary, in the **Summary Value** option. This option expresses the summary of a column in terms of the summaries for other columns.

In this case, the formula cannot refer to scorecard properties; instead it can refer to scorecard columns, using their names given by the **Label** option.

8.7 Configuring a Column of Row Numbers

A scorecard can include a column of row numbers, typically as the first column. The following shows an example:

	City
1.	Cedar Falls
2.	Centerville
3.	Cypress

To create such a column, [add](#) a column and [configure](#) it with the following options:

- **Display** — Select **Item Number**.
The column will contain item numbers.
- **Label** — Optionally specify the text to use as the column header. You can include `\n` to force a line break in this label.

8.8 Configuring a Column of Row Labels

A scorecard can include a column of row labels, typically as the first or second column. The following shows an example:

	City
1.	Cedar Falls
2.	Centerville
3.	Cypress

To create such a column, [add](#) a column and [configure](#) it with the following options:

- **Display** — Select **Label**.
For a KPI, the column will contain the names of the rows, as shown on the KPI test page.
- **Label** — Optionally type the text to use as the title. You can include `\n` to force a line break in this label.

8.9 Configuring a Column of Values

A typical scorecard includes columns that display numeric values, as in the following example:

Product	Revenue
Candy	\$128.56
Cereal	\$47.41
Dairy	\$90.46
Fruit	\$2170.99
Pasta	\$1361.59
Seafood	\$3128.21
Snack	\$4787.35
Vegetable	\$964.17
	\$12678.74

To create such a column, [add](#) a column and [configure](#) it with the following options:

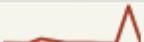
- **Display** — Click **Value**. This is the default setting.
- **Value** — Specify the value to display. Select a data source property, type a value, or type a formula. For information on formulas, see “[Specifying a Scorecard Formula](#),” earlier in this chapter.

- **Label** — Specify the text to use as the column header. You can include `\n` to force a line break in this label.
- **Target Value** — Optionally specify a target value for use as comparison with the given data value. Select a data source property, type a value, or type a formula.
- **Show As** — Use any option as described in “[Configuring a Scorecard Column.](#)”
- **Format** — Optionally specify a format string. See “[Specifying a Numeric Format String,](#)” earlier in this book.

You use **Value Column** only when you configure custom actions that require row values. See “[Specifying the Value of a Row,](#)” later in this book.

8.10 Configuring Trend Lines and Trend Bars

A scorecard column can display trend lines or trend bars. The following shows an example of trend lines:

Product	Revenue	History	Target
Bagels (dozen)	166		148
Bundt Cake	2,101		1,367
Calamari (frozen)	3,128		3,041
Cheerios (box)	47		39
Donuts (dozen)	208		273
Free-range Donuts (dozen)	1,610		1,350

The following shows an example of trend bars:

City	Patient Count	Birth Trend by Decade
Cedar Falls	117	
Centerville	95	
Cypress	104	
Elm Heights	118	
Juniper	109	

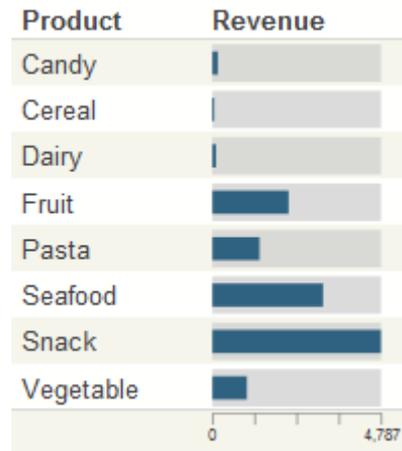
To create such a column, [add](#) a column and [configure](#) it with the following options:

- **Display** — Click **Trend Line** or **Trend Bar**.
- **Value** — Specify the values to show in the trend line or trend bar. Select a data source property that contains a comma-separated list of values.
Or type a comma-separated list of values.
- **Label** — Specify the text to use as the column header. You can include `\n` to force a line break in this label.
- **Show As** — Click **Value**.

Tip: You can display trend lines and trend bars only for KPIs, because only KPIs can contain a column of comma-separated values. In an MDX-based KPI, you can use the `%LIST` function to build a comma-separated list.

8.11 Configuring Plot Boxes

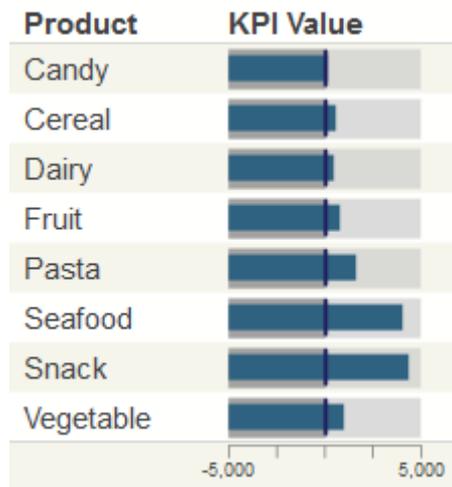
A scorecard column can display plot boxes, as in the following example:



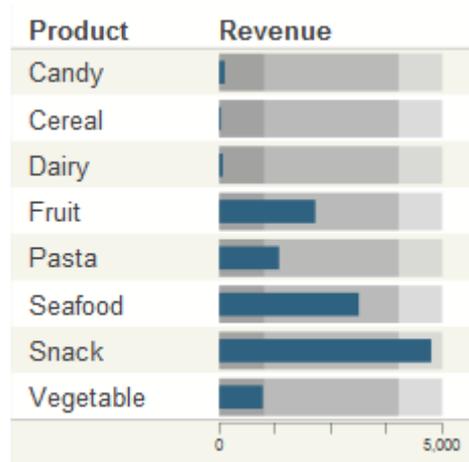
To create such a column, [add](#) a column and [configure](#) it with the following options:

- **Display** — Click **Plot Box**.
- **Value** — Specify the value to show in the plot box. Select a data source property, type a value, or type a formula. For information on formulas, see “[Specifying Formulas](#),” earlier in this book.
- **Label** — Specify the text to use as the column header. You can include `\n` to force a line break in this label.
- **Min Value** — Optionally specify the minimum value to display in the plot boxes. The default is zero.

In the following example, **Min Value** is `-5000`, and **Max Value** is `5000`:



- **Max Value** — Optionally specify the maximum value to display in the plot boxes. The default is the highest value displayed in the column. See the previous example.
- **Lower** — Optionally specify a value to divide the box into lower and middle bands. In the following example, **Lower** is `1000` and **Upper** is `4000`. These options define three bands in the plot boxes: a lower band (below 1000), a middle band (between 1000 and 4000), and an upper band (above 4000).



- **Upper** — Optionally specify a value to divide the box into middle and upper bands. See the previous example.
- **Target Value** — Optionally specify a target value for use as comparison with the given data value; the default is 0. Select a data source property, type a value, or type a formula.

The plot box includes a vertical line to mark the target value, as follows:

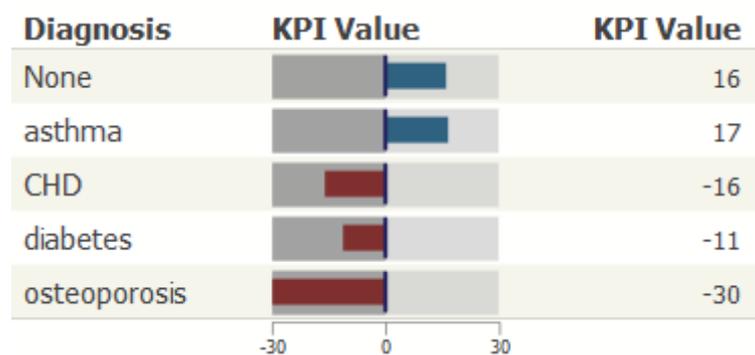


To remove these vertical lines, set **Target Value** equal to a value that is out of range for the plot boxes.

Also see the **Base Value** option for an alternative way to use a comparison value.

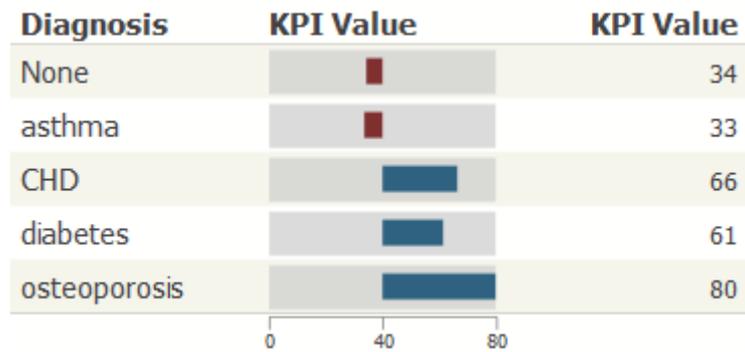
- **Base Value** — Optionally specify a value to create a visual boundary in the plot boxes. The base value is used as follows:
 - If the actual column value is less than the base value, the bar is displayed in a contrasting color (red by default).
 - In all cases, the bar is drawn between the actual column value and the base value.

The following shows an example in which the base value is 0. This example includes an additional column that shows the actual values used in the plot boxes:



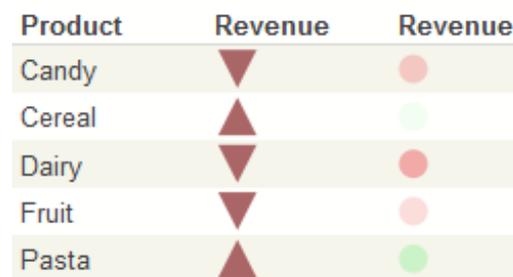
This plot box includes vertical lines at the value 0. DeepSee draws these lines because **Target Value** is 0, the default.

The following shows another example in which the base value is 40:



8.12 Configuring Arrows

A scorecard column can display triangular arrows, shown in the following example:



Arrows provide a quick way to see where the given value falls, in a desired range. They work as follows:

- They use a data source property (or possibly a constant) as a source value.
- If the **Min Value** setting is less than zero, then:
 - If the source value is negative, an arrow is displayed as a downward pointing triangle.
 - If the source value equals zero, the arrow is not displayed.
 - If the source value is positive, an arrow is displayed as an upward pointing triangle.
- If the **Min Value** setting is greater than or equal to zero, then:
 - If the source value is less than **Min Value**, an arrow is displayed as a downward pointing triangle.
 - If the source value equals **Min Value**, the arrow is not displayed.
 - If the source value is greater than **Min Value**, an arrow is displayed as an upward pointing triangle.

To create such a column, [add](#) a column and [configure](#) it with the following options:

- **Display** — Select **Arrow**.
- **Value** — Specify the value to show in the arrow. Select a data source property, type a value, or type a formula. For information on formulas, see “[Specifying Formulas](#),” earlier in this book.
- **Label** — Specify the text to use as the column header. You can include `\n` to force a line break in this label.
- **Show As** — Use any value as described in “[Configuring a Scorecard Column](#).”

- **Min Value** — Specify the value to control the direction of the arrow as described above.

8.13 Configuring Lamps

A scorecard column can display round lamps, shown in the following example:

Product	Revenue	Revenue
Candy	▼	●
Cereal	▲	●
Dairy	▼	●
Fruit	▼	●
Pasta	▲	●

Lamps provide a quick way to see where the given value falls, in a desired range. They work as follows:

- They use a data source property (or possibly a constant) as a source value.
- If the source value is negative, the lamp is displayed as a red circle.
- If the source value equals zero, the lamp is not displayed.
- If the source value is positive, the lamp is displayed as a green circle.
- The opacity of the circle is determined by the absolute value of the source value, divided by the absolute value of **Max Value**. If **Max Value** is not specified, the lamp uses the largest value in the column.

To create such a column, [add](#) a column and [configure](#) it with the following options:

- **Display** — Select **Lamp** or **Lamp with Value**.
- **Value** — Specify the value to show in the lamp. Select a data source property, type a value, or type a formula. For information on formulas, see “[Specifying Formulas](#),” earlier in this book.
- **Label** — Specify the text to use as the column header. You can include `\n` to force a line break in this label.
- **Show As** — Use any value as described in “[Configuring a Scorecard Column](#).”
- **Max Value** — Specify the maximum value to control the opacity of the lamp, as described previously.

8.14 Customizing the Appearance of a Scorecard Widget

DeepSee provides a rich set of options that you can use to customize the appearance of a scorecard widget. To access these options, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#)” and then see the options in the **Scorecard** section.

- **Scorecard > Size & Appearance** — Provides options to control the [scorecard overall](#).
- **Scorecard > Titles** — Provides options to control the text and formatting of [titles](#) in the scorecard.
- **Scorecard > Colors & Style** — Provides options to control [colors and line styles](#) in the scorecard.

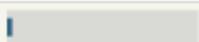
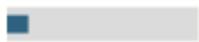
The following subsections provide details.

Note: The easiest way to systematically alter the appearance of dashboards is to define and use themes, which are discussed in “[Defining and Using Themes](#)”. Themes use many of the options described in the following subsections.

8.14.1 Size & Appearance Options

To customize the overall size and appearance of a scorecard, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Scorecard > Size & Appearance**, and use the options listed there. The following options are available:

- **Size** — Select the size of the scorecard, either **Regular** or **Big**.
- **Show Title** — Use this option to show or hide the scorecard title.
- **Show Headers** — Use this option to show or hide the scorecard headers.
- **Show Footers** — Use this option to show or hide the scorecard footers.
- **Row Height** — Use this option to control the height of the scorecard rows.
- **Arrow Size** — Use this option to control the height of any [arrows](#) in the scorecard. You can override this setting and the following settings for any individual column in the scorecard.
- **Lamp Size** — Use this option to control the height of any [lamps](#) in the scorecard.
- **TrendLine Hgt** — Use this option to control the height of any [trend lines](#) in the scorecard.
- **TrendBars Hgt** — Use this option to control the height of any [trend bars](#) in the scorecard.
- **Plotbox Height** — Use this option to control the height of any [plot boxes](#) in the scorecard. In the following example, the plot boxes are slightly taller than the default:

Product	KPI Value
Candy	
Cereal	
Dairy	

This example also uses **Plotbox Value**, which controls the height of the value boxes (shown slightly shorter than the default).

- **Plotbox Value** — Use this option to control the height of the value boxes in any [plot boxes](#) in the scorecard. See the previous example.
- **Column 1, Column 2, Column 3,** and so on — Use these options to control the widths of the respective columns. You can also modify the column widths in the **Scorecard > Columns** menu of the Dashboard Editor.

8.14.2 Title Options

To customize the titles in a scorecard, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Scorecard > Titles**, and use the options listed there. The following options are available:

- **Title** — Use this option to specify an optional title within the scorecard.
- **Title Style** — Use this submenu to control the style of the scorecard title.
See “[Specifying Chart Text Styles](#),” earlier in this book.
- **Subtitle** — Use this option to specify an optional subtitle within the scorecard.
- **Subtitle Style** — Use this submenu to control the style of the scorecard subtitle.

See “[Specifying Chart Text Styles](#),” earlier in this book.

- **Title Align** — Use this option to control the alignment of the title and subtitle. these titles can be left-aligned, centered, or right-aligned.
- **Title Box** — Use this submenu to add an optional box around the title and subtitle.

See “[Specifying Line Styles](#),” earlier in this book.

- **Title Image** — Use this option to add an optional image to the title.
- **Image Width, Image Height, Image Top, and Image Left** — Use these options to control the size and position of the optional title image.

Charts have a similar set of options; for examples, see “[Titles and Labels](#),” earlier in this book.

8.14.3 Colors & Style Options

To customize the colors and line styles in a scorecard, access the Dashboard Editor, select the widget as described in “[Reconfiguring a Widget](#),” click **Scorecard > Colors & Style**, and use the options listed there. The following options are available:

- **Background**— Use this submenu to control the style of the background around the scorecard. For example:

Product	Revenue
Candy	\$158.94
Cereal	\$592.59
Dairy	\$457.02
Fruit	\$783.48
Pasta	\$1646.22
Seafood	\$4048.52
Snack	\$4335.66
Vegetable	\$962.93
\$12985.36	

For details on specifying this style, see “[Specifying Line Styles](#),” earlier in this book.

- **Stripes**— Use this submenu to control the style of the stripes in the scorecard. For example:

Product	Revenue
Candy	\$158.94
Cereal	\$592.59
Dairy	\$457.02
Fruit	\$783.48
Pasta	\$1646.22
Seafood	\$4048.52
Snack	\$4335.66
Vegetable	\$962.93
\$12985.36	

For details on specifying this style, see “[Specifying Line Styles](#),” earlier in this book.

- **Separators**— Use this submenu to control the style of the separators between the scorecard and the header and footer. For example:

Product	Revenue
Candy	\$158.94
Cereal	\$592.59
Dairy	\$457.02
Fruit	\$783.48
Pasta	\$1646.22
Seafood	\$4048.52
Snack	\$4335.66
Vegetable	\$962.93
\$12985.36	

For details on specifying this style, see “[Specifying Line Styles](#),” earlier in this book.

- **Label Style**— Use this submenu to control the style of any [labels](#) in the scorecard. For example:

Product	Revenue
Candy	\$158.94
Cereal	\$592.59
Dairy	\$457.02
Fruit	\$783.48
Pasta	\$1646.22
Seafood	\$4048.52
Snack	\$4335.66
Vegetable	\$962.93
	\$12985.36

For details on specifying this style, see “[Specifying Chart Text Styles](#),” earlier in this book.

- **Value Style**— Use this submenu to control the style of the *values* in the scorecard. For example:

Product	Revenue
Candy	<i>\$158.94</i>
Cereal	<i>\$592.59</i>
Dairy	<i>\$457.02</i>
Fruit	<i>\$783.48</i>
Pasta	<i>\$1646.22</i>
Seafood	<i>\$4048.52</i>
Snack	<i>\$4335.66</i>
Vegetable	<i>\$962.93</i>
	\$12985.36

For details on specifying this style, see “[Specifying Chart Text Styles](#),” earlier in this book.

- **Cell Caption Style**— Use this submenu to control the style of any cell captions in the scorecard. For example:

Product	Revenue
Candy	<i>\$158.94</i> Cell Caption
Cereal	<i>\$592.59</i> Cell Caption
Dairy	<i>\$457.02</i> Cell Caption

For details on specifying this style, see “[Specifying Chart Text Styles](#),” earlier in this book.

- **Header**— Use this submenu to control the style of any column headers in the scorecard. For example:

Product	Revenue
Candy	\$158.94
Cereal	\$592.59
Dairy	\$457.02

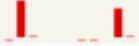
For details on specifying this style, see “[Specifying Line Styles](#),” earlier in this book.

- **TrendLine**— Use this submenu to control the style of any [trend lines](#) in the scorecard. For example:

Product	History
Candy	
Cereal	
Dairy	

For details on specifying this style, see “[Specifying Line Styles](#),” earlier in this book.

- **TrendBars Style**— Use this submenu to control the style of any [trend bars](#) in the scorecard. For example:

Product	History
Candy	
Cereal	
Dairy	

For details on specifying this style, see “[Specifying Line Styles](#),” earlier in this book.

- **Arrow Style**— Use this submenu to control the style of any [arrows](#) in the scorecard. For example:

Product	Revenue
Candy	
Cereal	
Dairy	

For details on specifying this style, see “[Specifying Line Styles](#),” earlier in this book.

- **Lamp Color** — Use this option to specify the color of any [lamps](#) in the scorecard, for use with positive values. For example:

Product	Revenue
Candy	
Cereal	
Dairy	
Fruit	
Pasta	

- **Lamp Negative** — Use this option to specify the color of any [lamps](#) in the scorecard, for use with negative values. See the previous example.

- **Lamp Value Style**— Use this submenu to control the style of the *values* for any [lamps](#) in the scorecard. For example:

Product	KPI Value
Candy	4
Cereal	5
Dairy	
Fruit	-1
Pasta	-1

For details on specifying this style, see “[Specifying Chart Text Styles](#),” earlier in this book.

- **Plotbox > Value Style** — Use this submenu to control the style of the values within any [plot boxes](#) in the scorecard. In the following example, the values are displayed as blue boxes:

Product	Revenue
Candy	
Cereal	
Dairy	

For details on specifying this style, see “[Specifying Line Styles](#),” earlier in this book.

- **Plotbox > Below Base Style** — Use this submenu to control the style of values below the configured **Base Value** in any [plot boxes](#) in the scorecard. In the following example, **Base Value** is 0 and **Below Base Style** is bright red:

Diagnosis	KPI Value	KPI Value
None		16
asthma		17
CHD		-16
diabetes		-11
osteoporosis		-30

-30 0 30

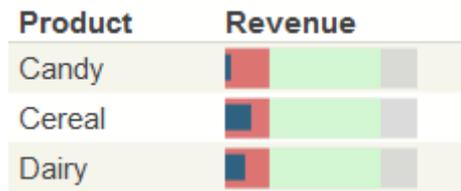
For details on specifying this style, see “[Specifying Line Styles](#),” earlier in this book.

- **Plotbox > Box Style** — Use this submenu to control the style of the enclosing boxes for any [plot boxes](#) in the scorecard. In the following example, the boxes are purple rather than the default gray:

Product	Revenue
Candy	
Cereal	
Dairy	

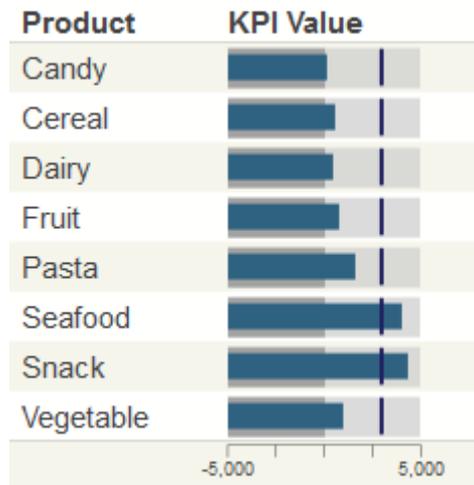
For details on specifying this style, see “[Specifying Line Styles](#),” earlier in this book.

- **Plotbox > Lower Style** — Use this submenu to control the style of the optional lower band in any [plot boxes](#) in the scorecard. The lower band indicates the values below the given **Lower Threshold**. In the following example, the lower band is red and the middle band is green:



For details on specifying this style, see “[Specifying Line Styles](#),” earlier in this book.

- **Plotbox > Mid Style** — Use this submenu to control the style of the optional middle band of any [plot boxes](#) in the scorecard. The middle band indicates the values above the given **Lower Threshold** but below the given **Upper Threshold**. See the previous example.
- **Plotbox > Target Style** — Use this submenu to control the style of the target line, if any, in any [plot boxes](#) in the scorecard. For example:



The position of the target line is controlled by the **Target Value** option.

For details on specifying this style, see “[Specifying Line Styles](#),” earlier in this book.

9

Adding Other Widgets

This chapter describes how to add other kinds of widgets to a dashboard:

- [Calendar widgets](#)
- [Map widgets](#)
- [Portlet widgets](#)
- [Controls widgets](#)

The Dashboard Editor also provides the **Worksheet** option, which is no longer documented. Note that worksheets are now deprecated.

9.1 Adding a Calendar Widget

A dashboard can include a calendar widget like the following:

«	Day	Week	Month	»		
May 2011						
Sun	Mon	Tue	Wed	Thu	Fri	Sat
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

To add a calendar widget:

1. Click **Menu > Add New Widget...**
2. In the left area, click **Calendar** and then click **Calendar**.
3. Optionally specify the basic settings, as described in “[Adding a Widget to a Dashboard](#),” earlier in this book.
4. Click **OK**.

When you open a dashboard, this widget displays the current month, for informational purposes.

9.2 Adding a Map Widget

If your cube includes latitude and longitude data, you can include a map widget like the following on your dashboards:



Important: This widget uses the Google Maps API. Be sure that your usage of this API is consistent with the Terms of Use, which you can access via a link in the widget, shown in the previous picture.

To add a map widget:

1. Obtain and enter a key for the Google Maps API. See [Specifying Basic Settings](#) for more information.
2. Optionally, in the Analyzer, define a pivot table that includes the columns `Latitude` and `Longitude` (not case-sensitive).

These columns should contain latitude and longitude values in decimal format (rather than degree/minute/second format).

For information on creating pivot tables, see [Using the DeepSee Analyzer](#).

You do not have to provide a data source for a map widget. If you provide no data source, the widget appears without any pins.

3. Display the dashboard.
4. Click **Menu > Add New Widget....**
5. In the left area, click **Map** and then click **Map**.
6. For **Data source**, select the previously defined pivot table.
7. Optionally specify the basic settings, as described in “[Adding a Widget to a Dashboard](#),” earlier in this book.
8. Click **OK**.
9. Optionally configure the widget further as follows:
 - a. Select the widget, as described in “[Reconfiguring a Widget](#),” earlier in this book.
 - b. Click **Widget Settings**.
 - c. Specify the following additional options:

- **Zoom** — Initial zoom setting to use. A larger number results in a closer view.
- **Starting latitude** — Initial latitude to display, in degrees.
- **Starting longitude** — Initial longitude to display, in degrees.
- **Draggable markers** — Controls whether the user can drag the map markers.

9.2.1 Configuring the infoWindow Popup

If the underlying pivot table contains data in addition to latitude and longitude, you can configure the map widget to display that data. If the user clicks a pin in the map widget, the widget displays a popup window that displays the additional information.

1. Access the Dashboard Editor and select the widget as described in “[Reconfiguring a Widget](#),” earlier in this book.
2. Click **Data Properties**.
The system displays a list of any properties defined for this map widget.
3. Click the plus sign + button.
The system displays a dialog box where you specify the property.
4. Specify the following options:
 - **Label** — Optionally specify the text to use as the column header. (Note that you *cannot* force a line break in this label.)
 - **Data Value** — Specify the value to display. Select a data source property. You cannot use a static value or a formula in this scenario.
 - **Format** — Optionally specify a format string. See “[Specifying a Numeric Format String](#),” earlier in this book.
5. Click **OK**.

9.3 Adding a Widget That Uses a Portlet

To add a widget that uses a custom portlet:

1. Click **Menu > Add New Widget...**
2. In the left area, click **Portlet**. The list expands to show a list of choices. This list shows the short class names for all compiled portlets in this namespace.
3. Click the name of the portlet to add.
4. Optionally specify the basic settings, as described in “[Adding a Widget to a Dashboard](#),” earlier in this book.
5. Click **OK**.

For information on creating portlets, see the [DeepSee Implementation Guide](#).

9.4 Adding a Controls Widget

To add a controls widget:

1. Click **Menu > Add New Widget...**
2. In the left area, click **Controls Widget**. The list expands to show a list of choices. Click the type of Controls Widget to add.
3. For the **Data Source**, do one of the following:
 - Click the Search button  and then select a data source.
See “[Kinds of Data Sources](#),” earlier in this book.
 - Use the **Link to** drop-down list to select another widget on the same dashboard.
4. Optionally specify the basic settings, as described in “[Adding a Widget to a Dashboard](#),” earlier in this book.
5. Click **OK**.
6. Optionally configure the widget further as described in the rest of this chapter.
7. Optionally add controls, as described in the chapter “[Adding Widget Controls](#).”

Note that it is not possible to hide the toolbar for a controls widget.

10

Defining and Using Themes

The easiest way to systematically alter the appearance of dashboards is to define and use themes. This chapter describes how to do this. It discusses the following topics:

- [Overview](#)
- [How to start a new theme definition](#)
- [How to add style information to a theme](#)
- [How to apply a theme to other widgets](#)
- [How to delete a theme](#)

10.1 Overview of Themes

To customize the appearance of your dashboards systematically, you can either specify a set of options for each widget in each dashboard or you can simply specify the theme that each widget uses. In the latter case, it is necessary only to specify the style options once, when you define the theme.

A single theme can specify the size of pivot table cells, the color and size of plot lines, background color and border, and more. The following table summarizes how themes work:

Widget type	Options determined by the applied theme, if any
Pivot table widget in table format	All options in Table Settings ; see “Customizing the Appearance of a Pivot Table.”
Pivot table widget in chart format	Most options* in Chart Settings > Titles & Labels , Chart Settings > Colors & Style , and Chart Settings > Series Details ; see “Customizing the Appearance of a Chart Widget.”
Speedometer	Most options* applicable to speedometers; see “Customizing Speedometers.”
Text meter	Most options* applicable to text meters; see “Customizing Text Meters.”
Fuel gauge	Most options* applicable to fuel gauges; see “Customizing Fuel Gauges.”
Traffic light	Most options* applicable to traffic lights; see “Customizing Traffic Lights.”
Light bar	Most options* applicable to light bars; see “Customizing Light Bars.”

Widget type	Options determined by the applied theme, if any
Smiley face	Most options* applicable to smiley faces; see “Customizing Smileys.”
Scorecard	Most options* in the Scorecard section; see “Customizing the Appearance of a Scorecard Widget.”
Other types	<i>Theme has no effect</i>

*In general, the exceptions are options that specify numeric values (such as the **Upper Range** and **Lower Range** options for a meter) and titles.

10.2 Creating a New Theme Definition

To create a new theme definition, possibly empty:

1. Access the Dashboard Editor and select a widget as described in [“Reconfiguring a Widget.”](#)

If the dashboard contains a widget with styles that you would like to use in the theme, select that widget. Or if no widgets have any styles (yet) that you would like to use, select any widget.

2. Select **Menu > Save Widget to Theme**
3. For **New Theme**, type the name of the theme.
4. Click **OK**.

DeepSee creates and saves the theme. The theme contains style settings, if any, copied from the widget. It also updates the widget to remove the local style settings and to use the theme instead, and it saves the dashboard. DeepSee then redisplay the dashboard.

10.3 Adding Style Information to a Theme

To add style information to a theme:

1. Access the Dashboard Editor and select the widget that includes the styles that you want to use.
2. Select **Menu > Save Widget to Theme**
3. For **Existing Theme**, select the theme.
4. Click **OK**.

DeepSee copies style settings, if any, from the widget to the theme and saves the theme. It also updates the widget to remove the local style settings and to use the theme instead, and it saves the dashboard. DeepSee then redisplay the dashboard.

10.4 Applying a Theme

To apply a theme to a different widget:

1. Access the Dashboard Editor and select the widget as described in [“Reconfiguring a Widget.”](#)

2. Click **Type and Data Source**.
3. Click **Widget Theme**.
4. Click the name of the theme.

DeepSee immediately updates the widget, saves the dashboard, and redisplay the dashboard.

10.5 Deleting a Theme

To delete a theme:

1. Access the Dashboard Editor and select any widget as described in [“Reconfiguring a Widget.”](#)
2. Click **Type and Data Source**.
3. Click **Widget Theme**.
4. Click the X next to the name of the theme.
5. Click **OK** to confirm this change.

DeepSee deletes the theme but does not change any dashboards. Any dashboards that use this theme now revert to their default appearance.

11

Adding Widget Controls

This chapter describes how to add controls to widgets in a dashboard. It discusses the following topics:

- [How to add a control](#)
- [How to reconfigure a control](#)
- [Explicit filter control](#)
- [Specify default values](#)
- [Onclick filter control](#)
- [Refresh control](#)
- [Print control](#)
- [Control to reload a dashboard](#)
- [Control to change the display type](#)
- [Control to change the row or column sort](#)
- [Control to change the row or column count](#)
- [Control to change the row or column specification](#)
- [Control to set a pivot variable](#)
- [Control to display a different pivot table](#)
- [Control to display a listing](#)
- [Control to display the Pivot Analysis window](#)
- [Control to display another dashboard](#)
- [Control to display a web page](#)
- [Control to execute a custom action](#)
- [How to hide a control](#)

For examples of controls and information on where they can be used, see the chapter “[Orientation to Dashboards](#)” in the *DeepSee End User Guide*.

Note: The following buttons are not controls and are not discussed in this chapter:

Button	Discussion
	Opens the pivot table in the Mini Analyzer.
	Displays the widget in table format.
	Displays the widget in chart format.
	Exports the currently viewed data to Microsoft Excel.
	Displays the dimension list.
	Drills down to a lower level in a hierarchy.

These buttons are specified as settings of the widget, as discussed in “[Specifying Widget Settings](#),” earlier in this book.

The print button  can be a control or can be specified as a widget setting.

11.1 Adding a Control

To add a control to a widget:

1. Access the Dashboard Editor and select the widget as described in “[Reconfiguring a Widget](#),” earlier in this book.
2. Click **Controls**.
3. Click the plus sign + button.
The system displays a dialog box where you specify the control.
4. Specify the options as appropriate for the type of control you need, as described in the rest of this chapter.
5. Click **OK** to add the control.
6. Optionally save the dashboard.

For reference, this section lists the control options and describes their purposes.

- **Location** — Specifies where the control is shown:
 - **Widget** (the default) displays the control on the widget toolbar.

Note: If you add a control to the widget, be sure the toolbar is visible (its display is controlled via the **Toolbar** option). See “[Reconfiguring a Widget](#),” earlier in this book.

- **Dashboard** displays the control in the **Filters** worklist.

Do not use **Dashboard** if you have configured this as a zero-worklist dashboard, because the control would not be visible to users.

- **OnClick Event** configures the control as an onclick control. No visible indication of the control is given.
- **Target** — Specifies which widgets this control affects. Specify one of the following:
 - If this option is blank, the control affects only the widget that owns it.
 - If this option is *, the control affects all widgets.
 - If this option is the name of the widget, the control affects that widget. In this case, use the value given in the **Widget Name** option for that widget.
- **Action** — Specifies the action. Custom actions are shown at the end of the list.
- **Filter** — (if **Action** is **Filter**) Specifies the filter for this control to use.
- **Type** — Specifies the visual form of the control:
 - **auto** — The system chooses the suitable form based on your other selections.
 - **dropdown** — The control is shown as a drop-down list.
 - **searchBox** — The control is shown as a searchable drop-down list. The user can type a value and then search for values that match the input.
 - **radio** — The control is shown as a set of radio buttons.
Note that the **radio** option is available only when you [reconfigure](#) a control.
 - **button** — The control is shown as a button.
 - **hidden** — The control is not shown.
 - **custom** — The control is based on a custom portlet.
- **Custom control** — (if **Type** is **custom**) — Specifies the portlet to use.
- **Control Label or Icon** — Specifies either the text to display above the control or an icon to display next to the control. For information on adding icons that can be used here, see “[Creating Icons](#)” in the *DeepSee Implementation Guide*.
- **Active When** — Specifies when this control is active. Choose one of the following values:
 - **Always** — This control is always active.
 - **Item Selected** — This control is active when the user selects one or more cells in a pivot table. The control is inactive otherwise.
 - **1 Listing Item Selected** — This control is active when the user selects a single row of a listing. The control is inactive otherwise.
 - **Listing Item Selected** — This control is active when the user selects one or more rows of a listing. The control is inactive otherwise.
- **Control Tooltip** — Specifies a tooltip to display when the user hovers the cursor over the control.
- **Control Size** — Specifies the width of the control.
- **Read Only** — If selected, this option prevents a user from changing the value in this control.
In this case, the purpose of the control is to indicate the state of the control. Either provide a default value (if applicable) or pass a value via the dashboard URL. See “[Accessing Dashboards from Your Application](#)” in the *DeepSee Implementation Guide*.
- **Default Value** (applies only to filter controls) — Specifies the default filter value. See “[Adding an Explicit Filter Control](#),” later in this chapter.

- **Required** (applies only to filter controls) — If this option is enabled, the user must specify a value for the filter. See “[Adding an Explicit Filter Control](#),” later in this chapter.

Note that you must also specify **Default Value** so that the dashboard always has a value for this filter.

Not all combinations of these options are suitable. Later sections of this chapter provide recommended configurations for specific kinds of controls.

11.2 Reconfiguring a Control

To reconfigure a control on a widget:

1. Access the Dashboard Editor and select the widget as described in “[Reconfiguring a Widget](#),” earlier in this book.
2. Click **Controls**.
3. Click the control name.
4. Make changes as needed. See the [previous section](#) for details.
5. Optionally save the dashboard.

11.3 Adding an Explicit Filter Control

To add a control to filter one or more widgets, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — If this control is going to automatically refresh the display, do one of the following:
 - Leave this option blank, if you want to refresh only this widget. This is the default.
 - Specify * to refresh all widgets.
 - Specify the name of the widget to refresh (use the value given in the **Widget Name** option for that widget).
- **Action** — Select one of the following:
 - **Apply Filter** sets the filter and automatically refreshes the display.
 - **Set Filter** sets the filter but does not refresh the display.
- **Filter** — Select the filter for this control to use:
 - For a pivot table created in the Analyzer, the system lists all levels available in the same subject area.
If there are any named filters in the subject area, the list include the item **Named Filters**. If you select this option, the user will be able to select any one of those named filters.
 - For a KPI, the system lists all filters defined in the KPI.
 - For an Ensemble business metric, the list includes **Instance** (the instance to display), **MaxHistory** (the number of historical points to display for the properties that specify *AUTOHISTORY*), and **Properties** (the metric property to display).
- **Type** — Select **auto**, **dropdown**, or **searchBox**.

- **Control Label or Icon** — Optionally type the text that you want to display above the control. Or select an icon to display next to the control.
- **Read Only** — Optionally select this to prevent the users from changing the filter. In this case, the purpose of the filter is to indicate how the data is filtered. Either provide a default value or pass a filter value via the dashboard URL. See “[Accessing Dashboards from Your Application](#)” in the *DeepSee Implementation Guide*.
- **Value Required** — If the user must specify a value for this filter, enable the **Required** option. Also specify a value for **Default Value** so that the dashboard always has a value for this filter.
- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

11.3.1 Specifying a Default Value

You may optionally type a value into **Default Value**.

Alternatively, click the magnifying glass. The system displays the following dialog box:

Here do one of the following:

- Click the magnifying glass under **Select a filter value** and select a value in the resulting dialog.
- Type a value in the **Enter an MDX Key** field. See the table after this list.
- Select a value from the **Use a run-time variable** drop-down list, which lists all available runtime variables. (For details on creating these, see the *DeepSee Implementation Guide*.)

Then click **OK**.

If you select a runtime variable, it is shown in the **Default Value** field with an at sign (@) at the beginning. For example:
@DefaultZipCode

Note: DeepSee assumes that the filter value is a member key rather than a member name. In most cases, the member key and member name are the same. For information on seeing the member key, see “[Finding the Key for a Member](#)” in the chapter “[Filtering Pivot Tables](#)” in *Using the DeepSee Analyzer*.

Also, if a user accesses the dashboard via a URL, any filter values in the URL take precedence over the default value specified in this control.

If you type a value into the **Enter an MDX Key** field, use one of the following forms:

Scenario	Expression That Returns This Value
A single member	"&[keyval]" where <i>keyval</i> is the key for the member. See “Key Values” in the <i>DeepSee MDX Reference</i> .
A range of members	"&[keyval1]:&[keyval2]"
A set of members	"{&[keyval1],&[keyval2],&[keyval3]}"
All members of the level except for a specified single member	"%NOT &[keyval]"
All members of the level except for a specified subset	"%NOT{&[keyval1],&[keyval2],&[keyval3]}"

11.4 Adding an OnClick Filter Control

Note: An onclick filter control is supported only in a widget that uses a pivot table as its data source.

To add an onclick filter control, add a control as described [earlier in this chapter](#) and configure it as follows:

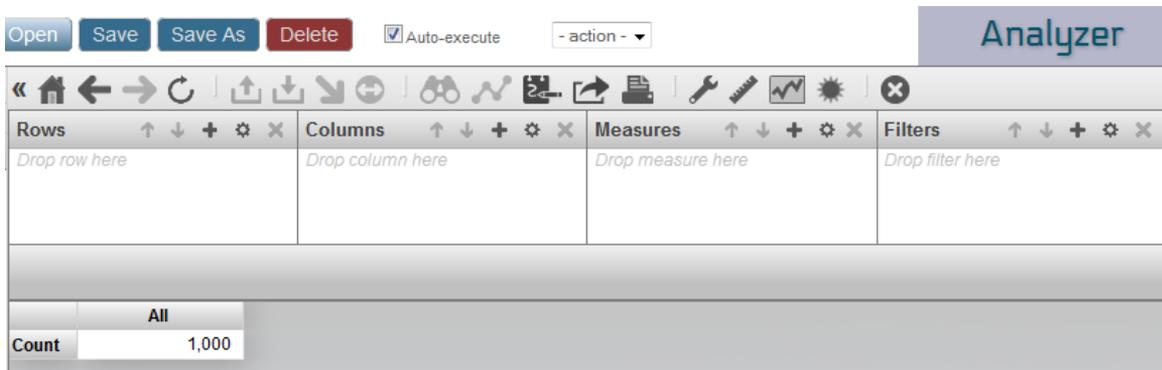
- **Location** — Click **OnClick Event**.
- **Target** — Do one of the following:
 - Specify * to refresh all widgets.
 - Specify the name of the widget to refresh (use the value given in the **Widget Name** option for that widget).
- **Action** — Select **Apply Filter**.

Note: You do not need to select the filter to apply. DeepSee automatically filters based on the complete context in which the user selects; that is, DeepSee considers the row and column context as well as any currently active filters applied to the widget. Furthermore, if you do specify a filter within the widget configuration, *that* filter is ignored.

11.4.1 Example: Listing with OnClick Filter

As an example, see the dashboard Listing with OnClick Filter, in SAMPLES. This dashboard is defined as follows:

- In the dashboard definition, the upper widget has an onclick filter whose target is the lower widget.
- The pivot table shown in the top widget — Use in Dashboards/Product Info — displays product categories as rows.
- The pivot table shown in the bottom widget — Use in Dashboards/HoleFoods Sales Listing — is defined as an unfiltered detail listing that does not auto-execute. To define a pivot table in this way, do the following in the Analyzer:
 1. Create a pivot table that uses the desired cube and that has no column or row definition and that also has no filter, like this:



2. Clear the **Auto-execute** button.

3. Click the listing button .

You can perform steps 2 and 3 in either order.

4. Save this pivot table.

11.5 Adding a Refresh Control

To add a control to refresh the dashboard (which is necessary if you have controls that set [filters](#) but that do not apply them automatically), add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — Do one of the following:
 - Leave this option blank, if you want to refresh only this widget. This is the default.
 - Specify * to refresh all widgets.
 - Specify the name of the widget to refresh (use the value given in the **Widget Name** option for that widget).
- **Action** — Select **Refresh**.
- **Type** — Select **auto** or **timer**.
 - If you select **auto**, the control appears as follows: 
 - If you select **timer**, type a value into **Time (in seconds)**. In this case, the refresh control will automatically run at the specified interval.
- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the button.

Tip: To use a custom button instead of the standard icon, specify the **Type** as **button** and specify a value for **Control Label** or **Icon**.

11.6 Adding a Control to Print the Dashboard

You can add controls with which users can print parts or all of the dashboard. The output is a PDF file (opened in a browser window) with one page for each widget.

To add such a control, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — Specify a list of the names of the widgets to print. The first widget in the list determines the print settings.

Or, if you want to print all the widgets and do not need to control the print settings, type `*`.

If you leave **Target**, only the current widget is printed. If you need only to print a single widget, however, it is simpler to just enable the print setting for that widget; see “[Customizing Print Settings for a Widget](#),” earlier in this book.

- **Action** — Select **Print Widget**.
- **Type** — Select **auto**, **dropdown**, or **button**, as applicable for the chosen action.

The default type is a small button with a picture on it.

- **Control Label or Icon** — Optionally type the text that you want to display on the control. Or select an icon to display next to the control.
- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

Then save the dashboard.

For further details and requirements, see the section “[Customizing Print Settings for a Widget](#).”

11.7 Adding a Control to Reload the Dashboard

To add a control to reload the dashboard, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — Ignore this option. All widgets are refreshed.
- **Action** — Select **Reload Dashboard**.
- **Type** — Select **auto** or **timer**.

–

If you select **auto**, the control appears as follows:



– If you select **timer**, type a value into **Time (in seconds)**. In this case, the reload control will automatically run at the specified interval.

- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the button.

Tip: To use a custom button instead of the standard icon, specify the **Type** as **button** and specify a value for **Control Label or Icon**.

11.8 Adding a Control to Change the Display Type

You can add controls with which users can change the display type from one chart to another. You can provide two different kinds of controls:

- A drop-down list from which the user can choose a type.
- A button that immediately switches to a different type. In this case, you typically provide at least two buttons, so that the user can easily return to the original type.

To add such a control, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — Leave this blank. Or type * if this control should affect all widgets in this dashboard.
- **Action** — Select **Set Chart Type** or **Choose Chart Type**. The first option adds a button (which you configure later) to set the display type to a specific type. The second option adds a drop-down list (which you configure later) of display types.
- **Type** — Select **auto**, **dropdown**, or **button**, as applicable for the chosen action.

For **Set Chart Type**, the default type is **dropdown**, and for **Choose Chart Type**, the default type is a small button with a picture on it.

- **Control Label or Icon** — Optionally type the text that you want to display on the control. Or select an icon to display next to the control.
- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

Then select the control again and display the **Control** menu. Select **Chart Type**. This displays a list of chart types, which includes the pivot table type. Then:

- If **Action** is **Set Chart Type**, click a single type.
- If **Action** is **Choose Chart Type**, click multiple types and then click **Apply Selection**.

To unselect a given type, click it a second time. To unselect all types, click **Clear Selection**.

Then save the dashboard.

11.9 Adding a Control to Change the Row or Column Sort

In a pivot table widget, if a user double-clicks a column header in a pivot table, the system automatically sorts the pivot table by the data in that column. No configuration is needed for this.

You can also add a control that affects the row or column sort. To add such a control, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — Leave this blank. Or type * if this control should affect all widgets in this dashboard.
- **Action** — Select **Row Sort** or **Column Sort**.
- **Type** — Select **auto** (which becomes a drop-down list) or **dropdown**.

- **Control Label or Icon** — Optionally type the text that you want to display on the control. Or select an icon to display next to the control.
- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

11.10 Adding a Control to Change the Row or Column Count

For pivot table widgets, to add a control that changes the row or column count, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — Leave this blank. Or type * if this control should affect all widgets in this dashboard.
- **Action** — Select **Row Count** or **Column Count**.
- **Type** — Select **auto** (which becomes a type-in box) or **dropdown**.
- **Control Label or Icon** — Optionally type the text that you want to display on the control. Or select an icon to display next to the control.
- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

11.11 Adding a Control to Change the Row or Column Specification

For widgets that display pivot tables (rather than other data sources), you can add a control that enables the user to change the specification for the rows or columns of the pivot table. To add such a control, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — Leave this blank. Or type * if this control should affect all widgets in this dashboard.
- **Action** — Select **Set Row Spec** or **Set Column Spec**.
- **Type** — Select **auto** (which becomes a button) or select **button**.
- **Row Spec** or **Column Spec** (as appropriate; this option is available only when you [reconfigure](#) a control) — Specify an MDX set expression. For example:

```
[gend].[h1].[gender].MEMBERS
```

For another example:

```
{MEASURES.[avg test score],MEASURES.[avg allergy count]}
```

Do not include the keyword `NON EMPTY`. Also note that the set expression does not include `ON ROWS` or `ON COLUMNS`.

For details, see the reference section “[Set Expressions](#)” in the *DeepSee MDX Reference*.

If this setting is an empty string, DeepSee ignores the specification and uses the specification defined in the original data source.

- **Control Label or Icon** — Optionally type the text that you want to display on the control. Or select an icon to display next to the control.
- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

Note: Such controls are supported only for widgets that use pivot tables as the data source. Also note that it is not practical to use **Set Column Spec** or **Choose Column Spec** (see the [subsection](#)) with meters and scorecards because these options change the property names of the data source.

11.11.1 Variation: Providing a List of Choices

To provide a similar control that provides a list of choices for the row or column specification:

1. Create a term list to hold the choices.

For each term list item, specify **Key** as the name of the specification and specify **Value** as an MDX set expression. For example:

Terms		
	Key	Value
1	Allergies	[allerd].[h1].[allergies].MEMBERS
2	Cities	[homed].[h1].[city].MEMBERS
3	Colors	[colord].[h1].[favorite color].MEMBERS

2. Use the preceding steps, with the following changes:
 - **Action** — Select **Choose Row Spec** or **Choose Column Spec**.
 - **Type** — Select **auto** (which becomes a drop-down list), **dropdown**, or **radio** (if there is a small number of choices). The **radio** option is available only when you [reconfigure](#) a control.
 - **Row Spec List** or **Choose Column Spec** (as appropriate; this option is available only when you [reconfigure](#) a control) — Select the term list that you created for this purpose.

11.12 Adding a Control to Set a Pivot Variable

Pivot variables provide another way for your end users to interact with your dashboards. A pivot table can be defined so that its underlying query uses pivot variables in selected parts of the query. (Similarly, an MDX-based KPI can use pivot variables in the same way.) When a dashboard includes this pivot table, that dashboard can include a *pivot variable control*, with which the user can change the value of the corresponding pivot variable. DeepSee simply substitutes the given value into the query, executes the query, and then redisplay the pivot table.

To add a control to set a pivot variable, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Action** — Select **Apply Pivot Variable**.
- **Pivot Variable** — Select the pivot variable.
- **Control Label or Icon** — Optionally type the text that you want to display on the control. Or select an icon to display next to the control.

- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

The control is useful only if the data source for the widget uses the given pivot variable.

11.13 Adding a Control to Display a Different Pivot Table

For a pivot table widget, you can provide a control that displays a different pivot table as the data source. The original data source does not have to be a pivot table. To add a control to display a specific pivot table in a widget, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — Leave this blank.
- **Action** — Select **Set Data Source**.
- **DataSource** (this option is available only when you [reconfigure](#) a control) — Select the pivot table.
- **Type** — Select **auto** or **button**.
- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

Note: Such controls are supported only for pivot table widgets.

11.13.1 Variation: Providing a List of Choices

To provide a similar control that provides a list of choices:

1. Create a term list to hold the choices.

For each term list item, specify **Key** as the name of the specification and specify **Value** as the name of a pivot table, ending with `.pivot`

For example:

Terms		
	Key	Value
1	Pivot 1	Pivot Features/Members Sorted by Measure Value.pivot
2	Pivot 2	Use in Dashboards/For Cities Scorecard.pivot
3	Pivot 3	D2IMPActions/alternateDataSource.pivot

2. Use the preceding steps, with the following changes:
 - **Action** — Select **Choose Data Source**.
 - **Type** — Select **auto** (which becomes a drop-down list), **dropdown**, or **radio** (if there is a small number of choices).
 - **DataSource List** (this option is available only when you [reconfigure](#) a control) — Select the term list that you created for this purpose.

11.14 Adding a Control to Display a Listing

You can display listings from any pivot table widget whose data source defines a listing.

Important: A map listing uses the Google Maps API. Be sure that your usage of this API is consistent with the Terms of Use, which you can access via a link displayed in this listing.

Note that in order to use the Google Maps API, you must obtain an API key. See [Specifying Basic Settings](#) in the *DeepSee Implementation Guide* for more information.

To add a control that displays a listing, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — Leave this blank.
- **Action** — Select **Show Listing** or **Show Geo Listing** (which shows a map listing).

Then choose the listing from the drop-down list to the right.

- **Type** — Select **auto**.

For a regular listing, this control appears as follows: 

For a map listing, this control appears as follows: 

- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

Tip: To use a custom button instead of the standard icon, specify the **Type** as **button** and specify a value for **Control Label** or **Icon**.

11.15 Adding a Control to Display the Pivot Analysis Window

For a widget that displays a pivot table, you can add a control that displays the Pivot Analysis window. To add such a control, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — Leave this blank.
- **Action** — Select **Pivot Analysis**.
- **Analysis** — Select the analysis option to display, from the following:
 - **Cluster** — Performs a [cluster analysis](#).
 - **Distribution** — Performs a [distribution analysis](#).
 - **Regression** — Performs a [regression analysis](#).
 - **Content Analysis** — Performs an [iKnow content analysis](#). This option is available only if the pivot table is based on a subject area that includes unstructured data.

- **Entity Analysis** — Performs an [iKnow entity analysis](#). This option is available only if the pivot table is based on a subject area that includes unstructured data.

For information on these options, see the chapter “[Using the Pivot Analysis Window](#)” in the *DeepSee End User Guide*.

If you do not choose an option, the user can select any option. If you do choose an option, the user sees only that option.

- **Type** — Select **auto** (which is shown as the  button in this case).
- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

11.16 Adding a Control to Display Another Dashboard

To add a control that displays another dashboard, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — Leave this blank.
- **Action** — Select **View Dashboard**.
- **Dashboard** — Click the Search button , click the dashboard, and click **OK**.
- **Type** — Select **button** or **auto** (which is shown as a button in this case).
- **Control Label or Icon** — Optionally type the text that you want to display on the control. Or select an icon to display next to the control.
- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

Note: You can also configure a pivot table widget so that you can double-click a cell and drill to another dashboard. See the chapter “[Adding Pivot Table Widgets](#).”

11.16.1 Passing Values

To pass values from a dashboard to a web page, include the following tokens in the URL:

\$\$\$VALUELIST

DeepSee replaces this token with a comma-separated set of values for the selected rows of the widget, as follows:

- For a scorecard widget, the values are taken from the property that is configured with **Value Column**.
- For a pivot table widget that displays a listing, the values are taken from the listing IDs.
- For a pivot table widget that displays a KPI, the values are taken from the property that is configured with **Value Column**, if any. Otherwise the values are taken from the first column.
- In other scenarios, \$\$\$VALUELIST does not have values.

The values are assumed not to contain commas.

As a simple example, the URL could be as follows:

```
http://www.google.com/search?q=$$$VALUELIST
```

\$\$\$CURRVALUE

DeepSee replaces this token with the current value of the widget, as follows:

- For a scorecard widget, the value is taken from the property that is configured with **Value Column**.
- For a pivot table widget that displays a listing, the value is taken from the first selected cell that was displayed before the listing was shown.
- For other pivot table widgets, the value is taken from the first selected cell.
- In other scenarios, \$\$\$VALUELIST does not have values.

Note: \$\$\$VALUELIST and \$\$\$CURRVALUE are not macros. They are replaced at run time, not at compile time.

11.17 Adding a Control to Display a Web Page

To add a control that displays a web page, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select either **Widget** (the default) or **Dashboard**.
- **Target** — Leave this blank.
- **Action** — Select **Navigate**.
- **URL** — Type the URL.
- **Type** — Select **button** or **auto** (which is shown as a button in this case).
- **Control Label or Icon** — Optionally type the text that you want to display on the button. Or select an icon to display next to the control.
- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

11.18 Adding a Control to Perform a Custom Action

To add a control that performs a custom action, add a control as described [earlier in this chapter](#) and configure it as follows:

- **Location** — Select **Widget** (the default), **Dashboard**, or **Onclick Event**.
- **Target** — Leave this blank.
- **Action** — Select the custom action. The custom actions are at the end of the list.

For information on creating custom actions, see the *DeepSee Implementation Guide*.

- **Type** — Select an option as follows:
 - If **Location** is **Widget** or **Dashboard**, select **button** or **auto** (which is shown as a button in this case).
 - Otherwise, ignore **Type**.
- **Control Label or Icon** — Optionally type the text that you want to display on the button. Or select an icon to display next to the control.
- **Control Tooltip** — Optionally type a tooltip to display when the user hovers the cursor over the control.

11.18.1 Specifying the Value of a Row

In some cases, a custom action might require information about the context — that is, the row that the user selected before launching the action. (Other actions operate the same way no matter what the context is.) When you use custom actions, make sure that you know whether this context is required.

If a custom action requires context, do the following to specify which column contains the value to pass to the actions:

1. Identify the [column](#) or [property](#) to use as the value.
2. Reconfigure that column or property and select the **Value Column** option.

Do not select the **Value Column** option for more than one property or column in the same widget; if you do, the result is indeterminate.

11.19 Hiding a Control

Sometimes it is useful to add a control, specify a default value, and hide it. To hide a control, for **Type**, select **Hidden**.