

# Endeca® Discovery Framework

## Component Catalog





# Contents

<b>Preface.....</b>	<b>7</b>
About this guide.....	7
Who should use this guide.....	7
Conventions used in this guide.....	7
Contacting Endeca Customer Support.....	8
<b>Chapter 1: About the Discovery Framework component catalog.....</b>	<b>9</b>
About components.....	9
About component users.....	11
Modifying component appearance.....	11
Using page transitions.....	13
Page transition syntax.....	13
Creating page transitions using component IDs.....	14
Discovery Framework performance considerations.....	15
Obtaining more information about the Discovery Framework.....	15
<b>Chapter 2: About Endeca standard components.....</b>	<b>17</b>
Administrative components.....	17
Attribute Settings.....	17
Data Sources.....	21
Data Source Bindings.....	23
Framework Settings.....	23
Performance Metrics.....	25
Sample Endeca Portlet.....	26
Layout components.....	27
Tabbed Component Container.....	27
Personalization components.....	29
Bookmarks.....	29
Results and visualization components.....	33
Advanced Visualization.....	33
Chart.....	35
Cross Tab.....	47
Results Table.....	52
Metrics Bar.....	56
Filtering components.....	61
Breadcrumbs.....	61
Guided Navigation.....	64
Range Filter.....	70
Search Box.....	73
Find Similar.....	78
Record detail components.....	84
Record Details.....	85
<b>Chapter 3: About Liferay components.....</b>	<b>91</b>
Liferay component support.....	91
The Component Container.....	91
The Languages component.....	92
About Liferay Web Content Management components.....	92
The IFrame component.....	93
The Links component.....	94
The Web Content component.....	94
The Web Content Display component.....	95
The Web Content List component.....	95
The Web Proxy component.....	96
<b>Chapter 4: Component behavior with MDEX 7.....</b>	<b>97</b>

About the MDEX 7 Early Access release.....	97
Discovery Framework interaction with MDEX 7.....	97



---

## Copyright and disclaimer

Product specifications are subject to change without notice and do not represent a commitment on the part of Endeca Technologies, Inc. The software described in this document is furnished under a license agreement. The software may not be reverse engineered, decompiled, or otherwise manipulated for purposes of obtaining the source code. The software may be used or copied only in accordance with the terms of the license agreement. It is against the law to copy the software on any medium except as specifically allowed in the license agreement.

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of Endeca Technologies, Inc.

Copyright © 2003-2010 Endeca Technologies, Inc. All rights reserved. Printed in USA.

Portions of this document and the software are subject to third-party rights, including:

Corda PopChart® and Corda Builder™ Copyright © 1996-2005 Corda Technologies, Inc.

Outside In® Search Export Copyright © 2008 Oracle. All rights reserved.

Rosette® Globalization Platform Copyright © 2003-2005 Basis Technology Corp. All rights reserved.

Teragram Language Identification Software Copyright © 1997-2005 Teragram Corporation. All rights reserved.

### Trademarks

Endeca, the Endeca logo, Guided Navigation, MDEX Engine, Find/Analyze/Understand, Guided Summarization, Every Day Discovery, Find Analyze and Understand Information in Ways Never Before Possible, Endeca Latitude, Endeca Profind, Endeca Navigation Engine, and other Endeca product names referenced herein are registered trademarks or trademarks of Endeca Technologies, Inc. in the United States and other jurisdictions. All other product names, company names, marks, logos, and symbols are trademarks of their respective owners.

The software may be covered by one or more of the following patents: US Patent 7035864, US Patent 7062483, US Patent 7325201, US Patent 7424528, US Patent 7567957, US Patent 7617184, Australian Standard Patent 2001268095, Republic of Korea Patent 0797232, Chinese Patent for Invention CN10461159C, Hong Kong Patent HK1072114, European Patent EP1459206B1, and other patents pending.

Endeca Discovery Framework Component Catalog • November 2010

Version 1.3.1



# Preface

Endeca® Latitude applications guide people to better decisions by combining the ease of search with the analytic power of business intelligence. Users get self-service access to the data they need without needing to specify in advance the queries or views they need. At the same time, the user experience is data driven, continuously revealing the salient relationships in the underlying data for them to explore.

The heart of Endeca's technology is the MDEX Engine.™ The MDEX Engine is a hybrid between an analytical database and a search engine that makes possible a new kind of Agile BI. It provides guided exploration, search, and analysis on any kind of information: structured or unstructured, inside the firm or from external sources.

Endeca Latitude includes data integration and content enrichment tools to load both structured and unstructured data. It also includes the Discovery Framework, a set of tools to configure user experience features including search, analytics, and visualizations. This enables IT to partner with the business to gather requirements and rapidly iterate a solution.

## About this guide

This guide contains information about the components that make up the Endeca Discovery Framework.

The Discovery Framework enables rapid configuration of dashboard applications that offer the highly interactive Guided Navigation® user experience across a full range of structured and unstructured enterprise data.

The Discovery Framework is easy to deploy and ideal for the agile development of enterprise-quality applications. Due to component-based nature of the Discovery Framework, these applications are simple to control, adapt, and extend. It provides granular layout and configuration control to enable users to manage and personalize their own experiences.

The Discovery Framework consists of an enterprise-class portal framework and a library of UI components that embody best practices in Endeca applications. In addition, it includes a Component SDK, which is a packaged development environment for portlets, themes, layout templates, and other portal element. Endeca has modified Liferay's version of its Plugins SDK to include the Endeca enhancements, such as the `EndecaPortlet` core class.

## Who should use this guide

This guide is intended for end users and power users of the Endeca Discovery Framework on Windows or Linux.

## Conventions used in this guide

This guide uses the following typographical conventions:

Code examples, inline references to code elements, file names, and user input are set in `monospace` font. In the case of long lines of code, or when inline monospace text occurs at the end of a line, the following symbol is used to show that the content continues on to the next line: ↵

When copying and pasting such examples, ensure that any occurrences of the symbol and the corresponding line break are deleted and any remaining space is closed up.

## Contacting Endeca Customer Support

The Endeca Support Center provides registered users with important information regarding Endeca software, implementation questions, product and solution help, training and professional services consultation as well as overall news and updates from Endeca.

You can contact Endeca Standard Customer Support through the Support section of the Endeca Developer Network (EDeN) at <http://eden.endeca.com>.



## Chapter 1

---

# About the Discovery Framework component catalog

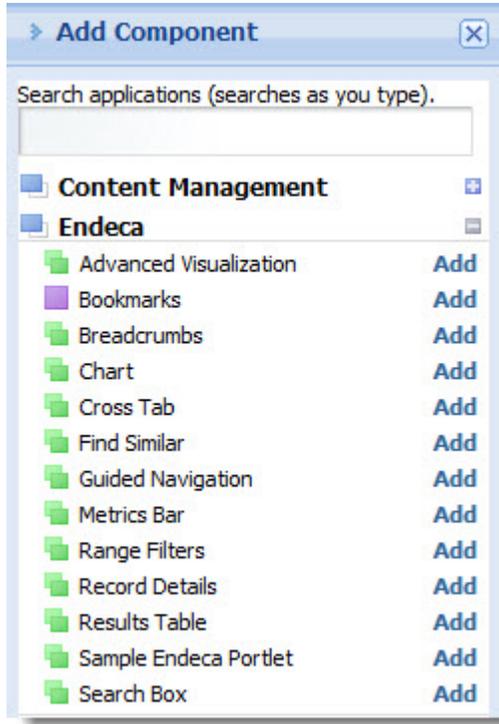
This guide provides an overview of the components available in the Discovery Framework 1.3.1.

## About components

The Discovery Framework is a portal application, built on the Liferay Portal, that you populate with portlet components.

Endeca standard components are Discovery Framework portlets provided by Endeca. These components contain the logic needed to retrieve and manipulate data from the MDEX Engine and render results. Each Endeca standard component is a self-contained unit that provides specific MDEX Engine features or other application support. By assembling these components together, you can rapidly develop robust search applications.

The set of Endeca standard components includes a selection of administrative components, such as **Data Sources** and **Performance Metrics**, that help you configure and control your application.



This catalog describes the way each Endeca-developed component works, and notes any interaction dependencies between components. It also provides basic information about the Liferay components that Endeca makes available by default. It does not include step-by-step instructions for using components.

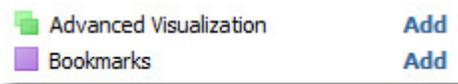
Components can be customized or developed to meet specific needs. This catalog does not cover this process, nor does it discuss the wide range of optional Liferay components that a user could add to the Discovery Framework.

### About instanceable and non-instanceable components

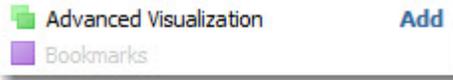
Liferay distinguishes between two kinds of components:

- *Instanceable* components can appear multiple times on a page. Each instance of the component has its own data source.
- *Non-instanceable* components can only appear once per page. If you place the component on additional pages, it will display the same data as the original instance.

Whether a component is instanceable or non-instanceable is indicated in the **Add Component** menu: a green icon identifies a component as instanceable, and a purple icon identifies a component as non-instanceable. In the image below, the **Advanced Visualization** component is instanceable. The **Bookmarks** component is not.



After adding the **Bookmarks** component to the page, it is no longer available, as you can see in the following image:



If you have multiple pages in your application, you could place **Bookmarks** on other pages, but the list of bookmarks in the component would be the same.

### About case sensitivity in the Discovery Framework

The Discovery Framework is case sensitive.

### About dimension search in the Discovery Framework

You must enable wildcard search on any dimension you want to do dimension search against.

## About component users

This catalog provides information about components from the perspective of two distinct Discovery Framework users. These users have different objectives and different access to component controls.

Several job roles might fit into each category.

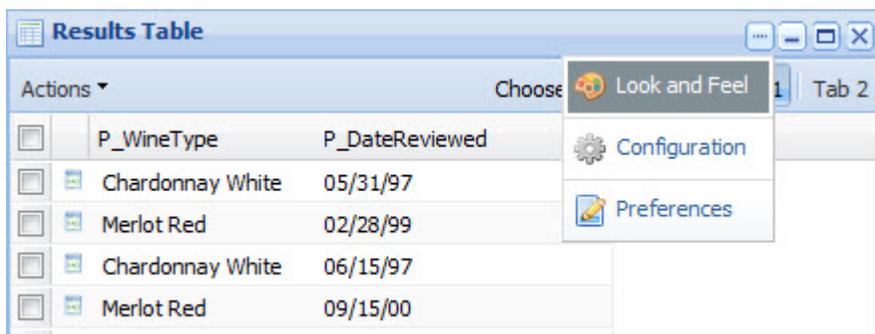
- **Power users** configure Discovery Framework content. One example would be a business analyst who configures the Discovery Framework for end users and determines what components and data they can access.
- **End users** are Discovery Framework content consumers. This includes executives seeking a dashboard view as well as others who need to drill through interactive visualizations and reports. Typically, the Discovery Framework is configured so that end users cannot access the edit controls found on the **Preferences** page of each component.



**Note:** Occasionally this catalog may mention tasks that other Discovery Framework users, such as application developers or data developers, perform. These tasks are not covered in detail in this guide.

## Modifying component appearance

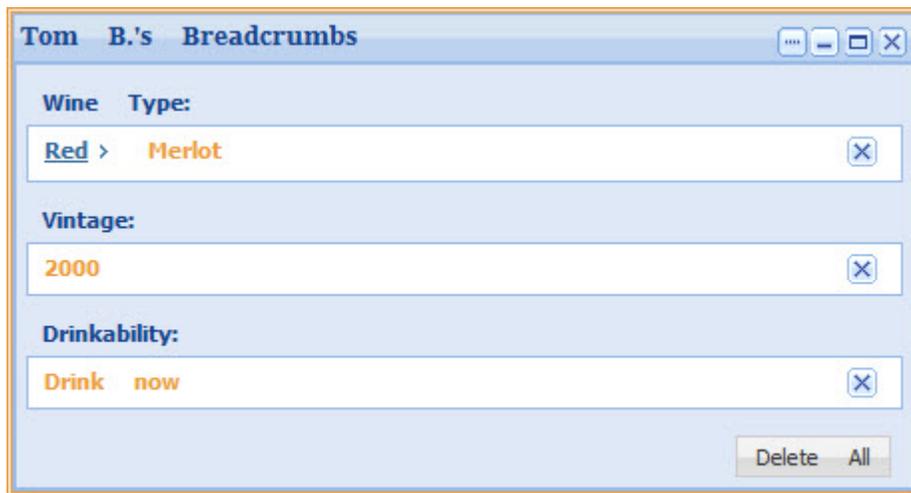
In any component, the power user can modify certain aspects of the component's appearance on the **Look and Feel** page.



For example, with slight modifications, the default appearance and text on the **Breadcrumbs** component:



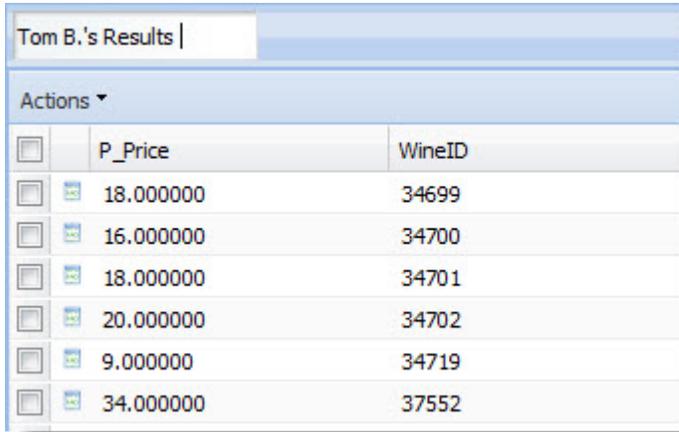
can be customized as follows:



### Changing component display names

The power user has the ability to modify the display name of any component simply by double-clicking the name in the component's header.

For example, the display name of the **Results Table** component can be customized as follows:



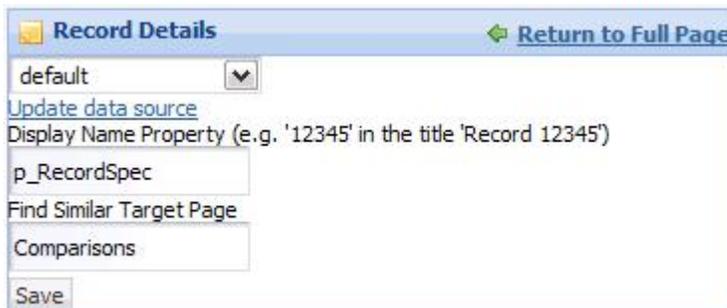
	P_Price	WineID
<input type="checkbox"/>	18.000000	34699
<input type="checkbox"/>	16.000000	34700
<input type="checkbox"/>	18.000000	34701
<input type="checkbox"/>	20.000000	34702
<input type="checkbox"/>	9.000000	34719
<input type="checkbox"/>	34.000000	37552

## Using page transitions

Page transitions allow a component on one page in your Discovery Framework application to pass data to a component on another page. In addition, page transitions let you specify the tab state on a tabbed component on that page.

For example, you might have a **Results Table** component on one page and a **Record Details** component on another page. The **Results Table** component could be configured to target the **Record Details** component through a page transition.

The page transitions feature is implemented by your component developer, but the power user can specify transition targets that will appear in the end user's application. In the example below, the power user set the target page for the **Record Details** component to a **Find Similar** component placed on the **Comparisons** page.



**Record Details** [Return to Full Page](#)

default

[Update data source](#)

Display Name Property (e.g. '12345' in the title 'Record 12345')

p\_RecordSpec

Find Similar Target Page

Comparisons

Save

## Page transition syntax

This topic describes the syntax for Discovery Framework page transitions.

Power users can target a page using a full context path or a relative context path. If the page transition target is a relative context path, the default context, `/web/guest`, is pre-pended to the target, as is shown in the following example:

```
Target: Analyze
Redirect: /web/guest/Analyze
```

The following example uses a full context path:

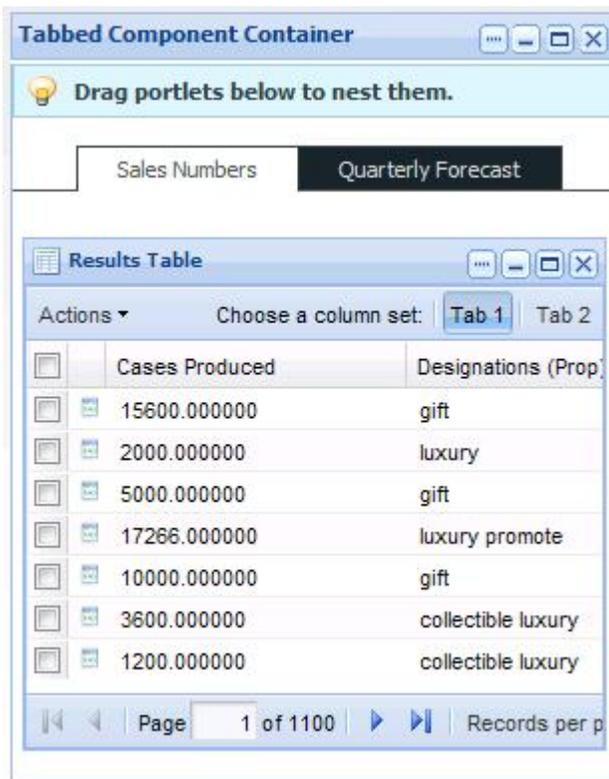
```
Target: /web/spend/Analyze
Redirect: /web/spend/Analyze
```

To set the tab state, the user appends a hash mark (#) followed by the tab name and number to the page name. In the following example, the **Sales Numbers** tabbed component is set to tab 1:

```
Target: /web/spend/Analyze#Sales Numbers[1]
Redirect: /web/spend/Analyze
```

To set multiple tabs, you delimit the additional tabs with a double colon (::). In the following example and image, the **Sales Numbers** tabbed component is set to tab 1, and the **Quarterly Forecast** tabbed component is set to tab 2:

```
Target: /web/spend/Analyze#Sales Numbers[1]::Quarterly Forecast[2]
Redirect: /web/spend/Analyze
```



## Creating page transitions using component IDs

In order to implement certain kinds of page transitions, you need to use component IDs, rather than names.

Because the double colon (::) is part of the page transition syntax, you should avoid using it in your tab titles. Likewise, you should avoid multiple tabbed component containers with duplicate titles. If you cannot avoid these naming features, then you have to use a component's ID rather than its name when defining a page transition target.

To find a component's ID, hover your mouse over the tab until the URL appears at the bottom of the screen, and then extract the `p_p_id_query` parameter from the URL. In the following example, the

tabbed component with ID `nested_tabs_INSTANCE_0CbE` is set to tab 2, while the tabbed component with ID `nested_tabs_INSTANCE_Ja6E` is set to tab 1.

```
Target: /web/spend/Analyze#nested_tabs_INSTANCE_0CbE[2]::nested_tabs_INSTANCE_Ja6E[1]
```

```
Redirect: /web/spend/Analyze
```

## Discovery Framework performance considerations

When planning and building Discovery Framework applications, keep in mind the following performance considerations.

- Like any Web-based application, the Discovery Framework is subject to browser limitations.
- The Discovery Framework is a rich Internet application that uses JavaScript to retrieve data in real time. Therefore, the amount of data you are accessing will affect performance.
- Attempting to process large numbers of dimensions or dimension values can result in information overload, particularly in components such as **Guided Navigation**.
- Within a Discovery Framework application, a large number of components on a single page can negatively impact performance. Rather than place many components on a single Discovery Framework page, you should consider using a multi-page application design. The number of pages in an application has no impact on performance.
- Endeca Analytics is a rich and complex language. To ensure that you understand the performance implications of the Analytics queries you are building, consult the *Analytics Considerations and Best Practices* article, which is available for download from the Endeca Developer Network ([EDeN](#)).
- Because the Discovery Framework leverages the MDEX Engine platform, it responds to many of the same performance strategies. For more detailed information, see the *MDEX Engine Performance Tuning Guide*, which is available for download or search on [EDeN](#).

## Obtaining more information about the Discovery Framework

In addition to this guide, the Discovery Framework documentation set contains two other guides.

- The *Discovery Framework Installation Guide* describes how to install and launch the Discovery Framework.
- The *Discovery Framework Extension Guide* describes how developers can extend the Discovery Framework.

In addition, there is a [forum](#) on the Endeca Developer's Network (EDeN) dedicated to the Discovery Framework. It hosts discussions about development, extension, deployment, and configuration for technical and business users of the Discovery Framework.





## Chapter 2

---

# About Endeca standard components

Endeca standard components are pre-packaged components that provide access to MDEX Engine functionality and other features useful to application development.

## Administrative components

The components in this section control administrative aspects of the Discovery Framework.

### Related Links

[Data Sources](#) on page 21

The **Data Sources** component allows you to view configured data sources and test the connection to them. In addition, you can reload updated configuration based on edits you have made on disk.

[Attribute Settings](#) on page 17

The **Attribute Settings** component allows the power user to create, edit, and delete attribute sets and change display names for any attribute in a selected data source.

[Framework Settings](#) on page 23

The **Framework Settings** component provides access to state, security, and other settings.

[Performance Metrics](#) on page 25

The **Performance Metrics** component displays information about component and MDEX Engine query performance.

[Sample Endeca Portlet](#) on page 26

The **Sample Endeca Portlet** component provides developers with a template from which they can build their own custom components.

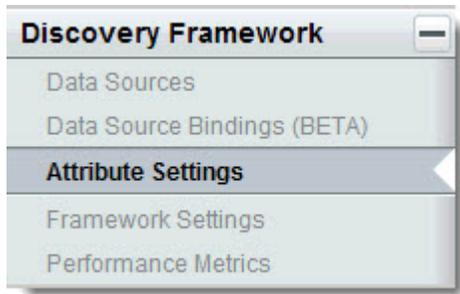
## Attribute Settings

The **Attribute Settings** component allows the power user to create, edit, and delete attribute sets and change display names for any attribute in a selected data source.

An attribute set is a collection of dimensions and properties associated with the named data source. They allow you to organize your properties and dimensions into named groups.



**Note:** This component appears in the Liferay **Control Panel** and is not accessible from the **Add Component** menu.

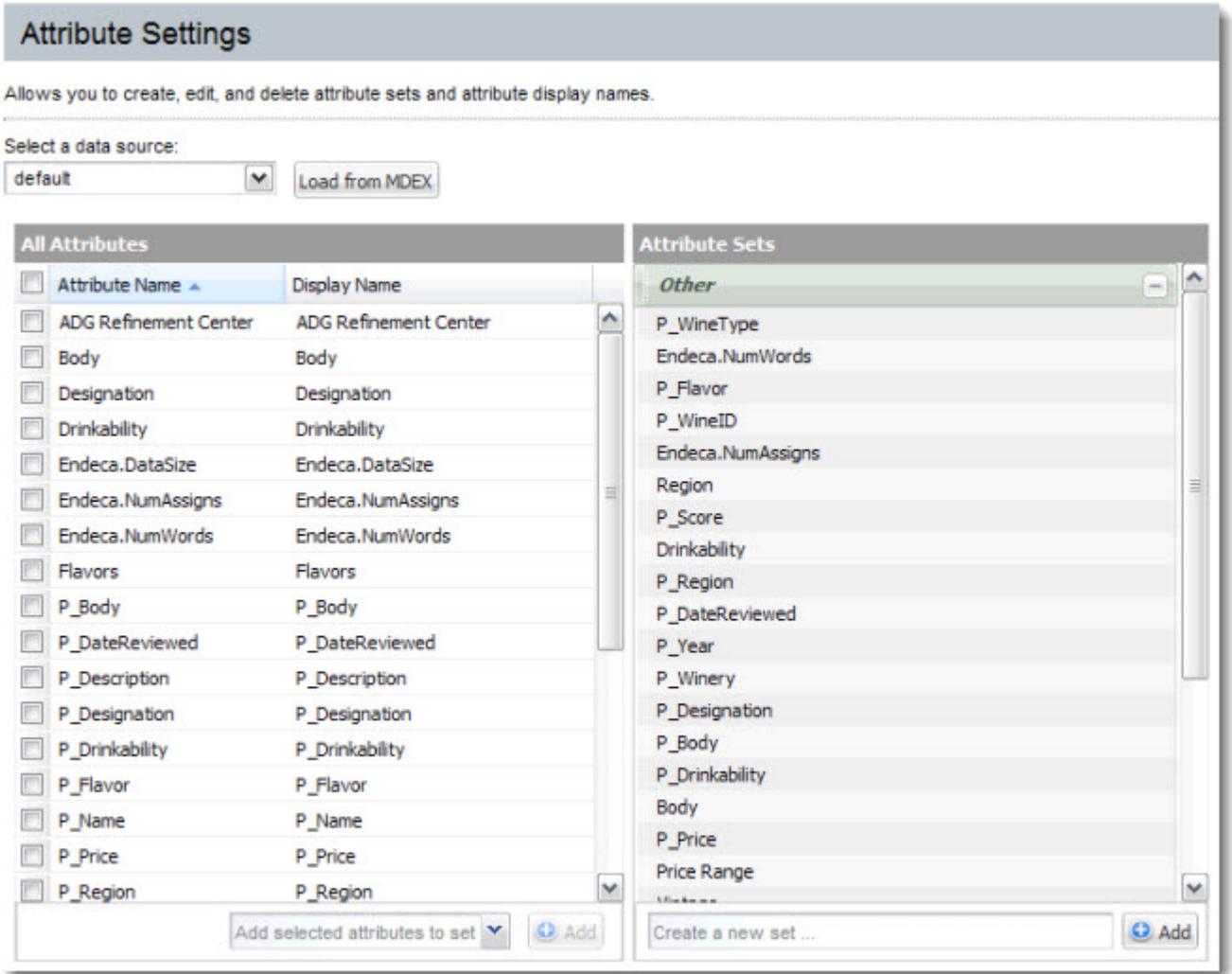


## Using Attribute Settings

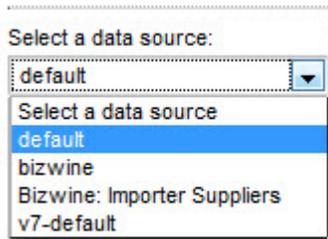
End users generally should not modify the information located on the **Attribute Settings** section of the **Control Panel**.

## Configuring Attribute Settings

Because the **Attribute Settings** component does not display as an independent portlet, it is not configured in the same way as most Endeca standard components. Instead, you modify these settings in the **Control Panel**.



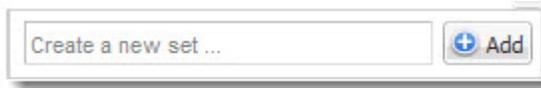
- **Data source controls:** Select from the available data sources in the drop-down list.



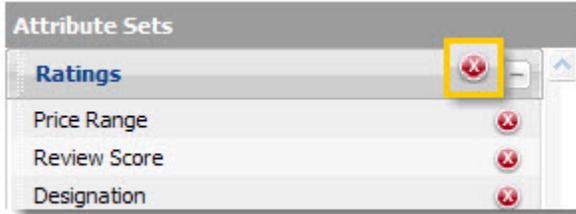
- **Load from MDEX:** Create new attributes and sets based on the contents of the MDEX Engine.



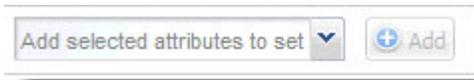
- **Create a new attribute set:** Create a new attribute set, and then select the attributes that populate it.



- **Delete an attribute set:** Delete an entire attribute set.



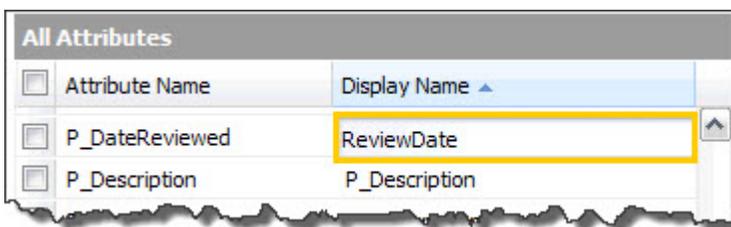
- **Add attributes to a set:** Check attributes in the **All Attributes** list and then add them to the selected set.



- **Remove attributes from a set:** Remove a single attribute from a set.



- **Change the global display name for an attribute:** Edit the display name of an attribute in the **All Attributes** list.



 **Note:** Attributes may be added to multiple sets, but there can only be a single display name for any given attribute.

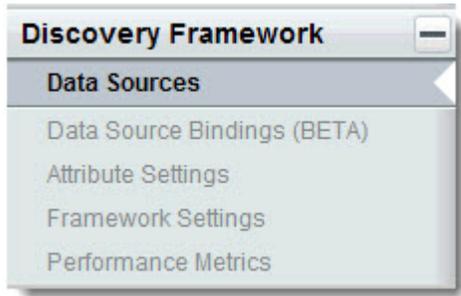
 **Note:** You cannot move an attribute between sets. To accomplish the same thing, first delete the attribute from its current set, and then add it to another set.

## Data Sources

The **Data Sources** component allows you to view configured data sources and test the connection to them. In addition, you can reload updated configuration based on edits you have made on disk.



**Note:** This component appears in the Liferay **Control Panel** and is not accessible from the **Add Component** menu.



Each Discovery Framework component that needs to query the MDEX Engine is tied to one particular data source. Changing any individual component's data source is done in that component's edit controls.



**Note:** For general information about Discovery Framework data sources, see the *Discovery Framework Installation Guide*. For information about data source configuration, security, and state management, see the *Discovery Framework Extension Guide*.

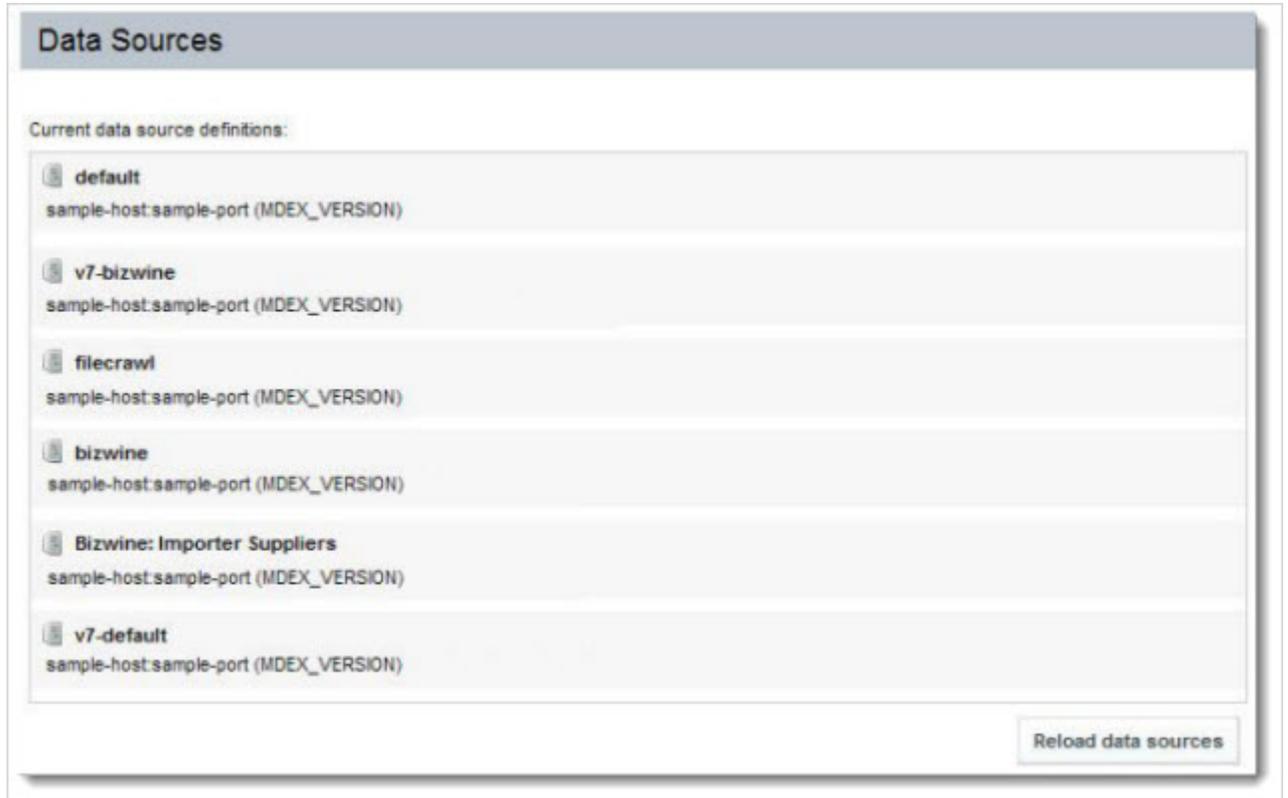
## Using Data Sources

End users do not see the **Data Sources** component.

## Configuring Data Sources

The **Data Source** component requires no configuration—you simply place it on the page.

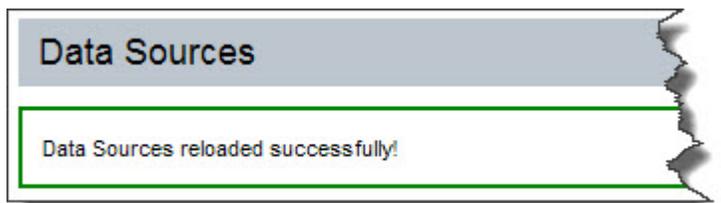
When you open the **Data Sources** component, you can see all of the data sources that have been configured for your application, as well as the version of the MDEX Engine they are using (`ENE_QUERY` for MDEX 6 or `DISCOVERY_SERVICE` for MDEX 7).



By clicking a data source's icon (highlighted below), you can test the data source's connection. The icon changes and mousing over it provides a confirmation message.



To reload updated configuration based on edits made on disk, you click the **Reload data sources** button. (For details on making these edits, see the *Discovery Framework Extension Guide*.)



**Note:** Configuration of data sources themselves takes place offline.

## Data Source Bindings

The **Data Source Bindings** component allows the power user to associate different configured data sources with selected components in a single operation, rather than on a per-component basis.



**Important:** The **Data Source Bindings** component is in Beta state in Discovery Framework 1.3.x, and is not supported.

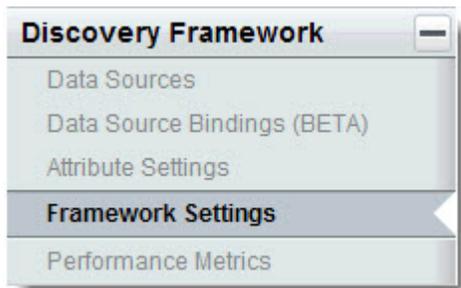
## Framework Settings

The **Framework Settings** component provides access to state, security, and other settings.



**Note:** This component appears in the Liferay **Control Panel** and is not accessible from the **Add Component** menu.

Many settings related to Discovery Framework can be adjusted from the **Discovery Framework Settings** section of the **Control Panel**.



The default values of these settings are created automatically upon first use. You cannot add or delete settings from the **Control Panel**—you can only edit them. Settings only appear after the feature(s) that use them have been executed at least once. For example, if you have never used the **Chart** component, the **Corda Server URL** settings will not appear.

## Using Framework Settings

End users generally should not modify the information located on the **Discovery Framework Settings** section of the **Control Panel**.

## Configuring Framework Settings

Because the **Framework Settings** component does not display as an independent portlet, it is not configured in the same way as most Endeca standard components. Instead, you modify these settings in the **Control Panel**.



**Note:** Take care when modifying these settings, as incorrect values can cause problems with your Discovery Framework application.

### Framework Settings

**Warning!** Incorrect values for these settings can cause serious problems with your Discovery Framework application. Please do not change these settings unless you are sure of what you are doing.  
You must restart the Discovery Framework in order for changes to these settings to take effect.

---

<b>df.container:</b>	liferay-async	<i>Controls asynchronous (AJAX) behavior of the framework. The only supported mode is "liferay-async", do not set a different value.</i>
<b>df.defaultDataSource:</b>	default	<i>The name of the data source to be used by default.</i>
<b>df.exportPortletName:</b>	endeca-results-export-portlet_WAR_endeca-results-export-portlet	<i>This is the default name of the export portlet that can used with p_p_id.</i>
<b>df.maxExportBaseRecords:</b>	50000	<i>The maximum allowable number of base records to export.</i>
<b>df.maxExportAnalyticsRecords:</b>	2000	<i>The maximum allowable number of analytics records to export.</i>
<b>df.defaultExporter:</b>	com.endeca.export.CSVExport	<i>The default exporter class.</i>
<b>df.dataSourceDirectory:</b>	\${liferay.home}/data/endeca-data-sources	<i>The directory on disk from which to load the Data Source definition files. This must be an absolute path. You may start this value with the token "\${liferay.home}" to represent the Liferay portal root.</i>
<b>df.mdexStateManager:</b>	com.endeca.portal.data.DefaultMDEXStateManager	<i>The fully-qualified class name to use for the MDEX State Manager. Only change this if you have extended the state manager with your own class.</i>
<b>df.mdexSecurityManager:</b>	com.endeca.portal.data.security.DefaultMDEXSecurityManager	<i>The fully-qualified class name to use for the MDEX Security Manager. Only change this if you have extended the security manager with your own class.</i>
<b>df.cordaServerRedirectorUrl:</b>	http://localhost:8080/endeca-corda-chart-portlet/ctRedirect	<i>The URL of the Corda Redirector.</i>

The **Framework Settings** component contains the following settings:

- **df.container:** Controls AJAX behavior of the Discovery Framework. Should not be edited.
- **df.defaultDataSource:** The name of the data source to use as the default.
- **df.exportPortletName:** The default name of the export portlet.
- **df.maxExportBaseErrors:** The maximum allowable number of non-Analytics records that can be exported.
- **df.maxExportAnalyticsRecords:** The maximum allowable number of Analytics records that can be exported.
- **df.defaultExporter:** The default exporter class.
- **df.dataSourceDirectory:** The directory on disk from which to load data source definition files.
- **df.mdexStateManager:** The fully-qualified class name to use for the MDEX State Manager.
- **df.mdexSecurityManager:** The fully-qualified class name to use for the MDEX Security Manager.
- **df.cordaServerRedirectorUrl:** The URL of the Corda Redirector, which can be used by the **Chart** component.

- **df.cordaServerExternalUrl** and **df.cordaServerInternalUrl**: The externally and internally accessible URLs of the Corda Server, which is used by the **Chart** component.



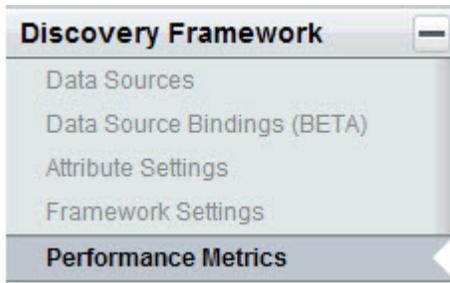
**Note:** If you do not see the **Discovery Framework Settings** in the **Control Panel**, it probably means you did not install the `endeca-framework-settings-portlet-<version>.war` file. Please review your installation settings.

## Performance Metrics

The **Performance Metrics** component displays information about component and MDEX Engine query performance.



**Note:** This component appears in the Liferay **Control Panel** and is not accessible from the **Add Component** menu.



The **Performance Metrics** component comes pre-configured and does not require a backing data source.

## Using Performance Metrics

This topic describes how an end user can use the **Performance Metrics** component.

The screenshot shows the "Performance Metrics" portlet interface. At the top, there is a header "Performance Metrics". Below it is a navigation bar with several tabs: "All MDEX Queries", "MDEX Queries: bizwine", "MDEX Queries: default", "MDEX Queries: v7-bizwine", "MDEX Queries: filecrawl", "MDEX Queries: v7-default", "Portlet Overview" (which is highlighted in orange), and "All Portlets". Below the navigation bar is a section titled "Portlet Executions:" containing a table with the following data:

Total number of portlet executions:	749
Total execution time of all portlets:	47372 ms
Average execution time of all portlets:	63 ms
Maximum portlet execution time:	1937 ms

At the bottom right of the portlet, there is a "Refresh" button.

The **Performance Metrics** component tracks total MDEX Engine queries as well as queries per data source. It also contains tabs tracking the overall performance of portlet components as well as the performance of each individual component.

It contains the following tabs:

- **All MDEX Queries**
- **MDEX Queries: <data source name>**
- **Portlet Overview**
- **All Portlets**

**All MDEX Queries** contains the following metrics:

- **All Data Queries** metrics for the total number of queries, total execution time of queries, average execution time of queries, and maximum query execution time.
- **All Cached Data Queries** metrics for the total number of queries, the total execution time of queries, average execution time of queries, maximum query execution time, and percentage of data queries served from the cache.

The **MDEX Queries** tab for each data source provides the same metrics as the **All MDEX Queries** tab at the data source level.

The **Portlet Overview** tab tracks the total number of component executions, the total execution time of all components, the average execution time of all components, and the maximum component execution time.

The **All Portlet** tab provides the same total metrics as the **Portlet Overview** tab, along with the same results broken out by component.



**Note:** This component provides a view into MDEX Engine query performance, which does not correlate directly to a page view, as there are often multiple MDEX Engine queries powering a single page.

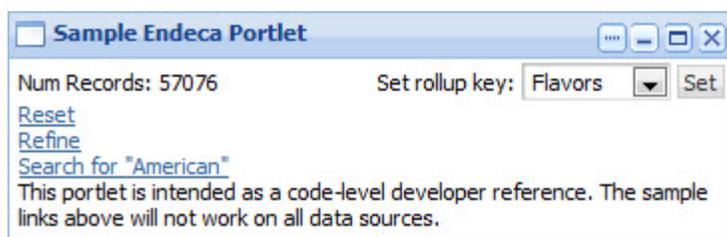
## Configuring Performance Metrics

The **Performance Metrics** component requires no configuration—you simply add it to the portal workspace.

## Sample Endeca Portlet

The **Sample Endeca Portlet** component provides developers with a template from which they can build their own custom components.

This component is provided as a starting point for developers who want to create their own components.



## Using Sample Endeca Portlet

The **Sample Endeca Portlet** component is not intended for use by end users.

## Configuring Sample Endeca Portlet

The **Sample Endeca Portlet** component is not intended for use by power users.

# Layout components

The components in this section provide additional control over Discovery Framework display.

## Related Links

[Tabbed Component Container](#) on page 27

The **Tabbed Component Container** allows developers to create a tabbed interface within a region of a page and then store different components on various tabs.

## Tabbed Component Container

The **Tabbed Component Container** allows developers to create a tabbed interface within a region of a page and then store different components on various tabs.

The **Tabbed Component Container** is the only Endeca-developed layout control. This component is available under the **Layout Control** section of the **Add Component** menu, and not the **Endeca** section.

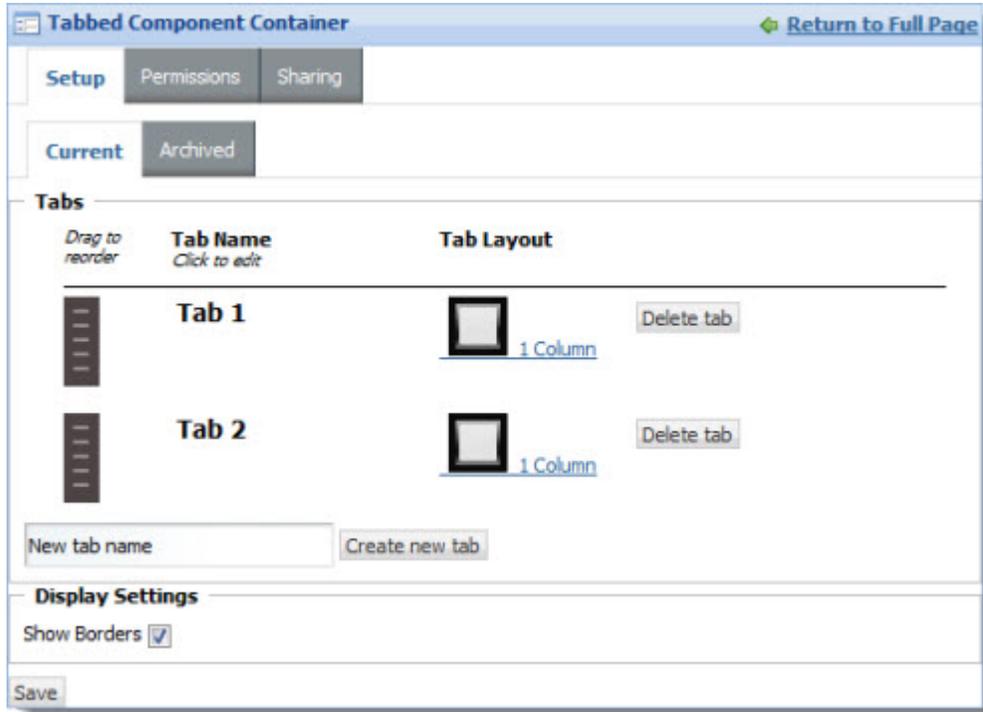
This component allows a user to create a tabbed interface within a region of a page, and then nest different components on the available tabs. For example, you might choose to put a **Results Table** component containing customer-based information on one tab, and another **Results Table** component containing product-based information on another tab.

## Using the Tabbed Component Container

End users do not access the **Tabbed Component Container** directly, though a power user might include one or more instances in the end users' application.

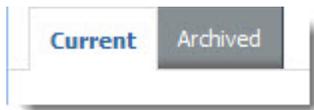
## Configuring the Tabbed Component Container

The power user can set the following preferences for the **Tabbed Component Container** component.



**Note:** Only the configuration tasks on the **Setup** tab are relevant to our purposes. You may disregard the other tabs.

- You can edit the current tabbed component container, or select an archived version (if available).



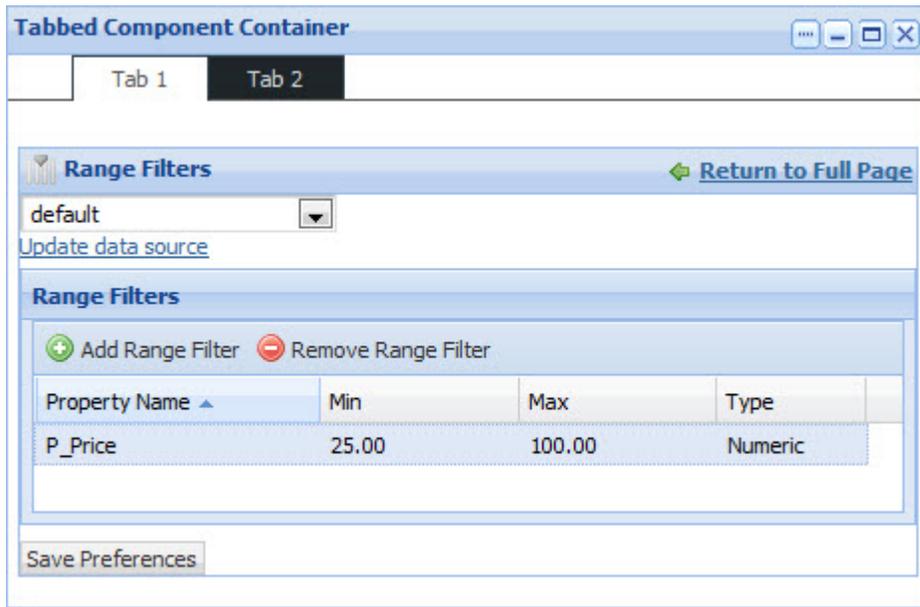
- Tabs:** Add or delete tabs, as well as change their layouts, display names, and order on the page.



- Display Settings:** Check **Show Borders** to draw the component container around the components on each tab.



Drag the configured **Tabbed Component Container** onto the portal work area, then drag other components onto the tabs and configure them as usual. In the example below, the **Range Filter** component has been dragged onto Tab 1.



## Personalization components

The components in this section allow users to customize their Discovery Framework application.

### Related Links

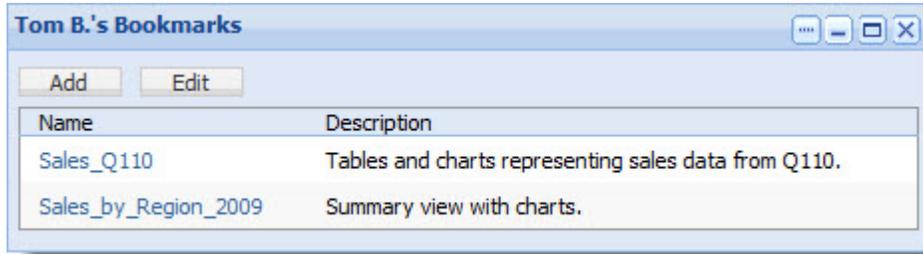
[Bookmarks](#) on page 29

**Bookmarks** is a personalization component that allows you to save a given navigation and component state so that you can return to it at a later time.

## Bookmarks

**Bookmarks** is a personalization component that allows you to save a given navigation and component state so that you can return to it at a later time.

Once a bookmark has been created, the user can click upon it to return to that state. Bookmarks appear in the are listed in the order they were created.



To share a bookmark, the user clicks on the bookmark to load it, then copies the browser URL and shares it with another user.



**Note:** The **Bookmarks** component does not require a backing data source.



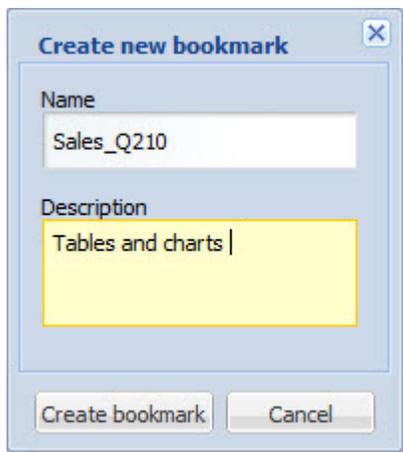
**Note:** The **Bookmarks** component supports the use of aggregated records. For details about configuring Discovery Framework data sources for aggregated records, see the *Discovery Framework Installation Guide*.

## Using Bookmarks

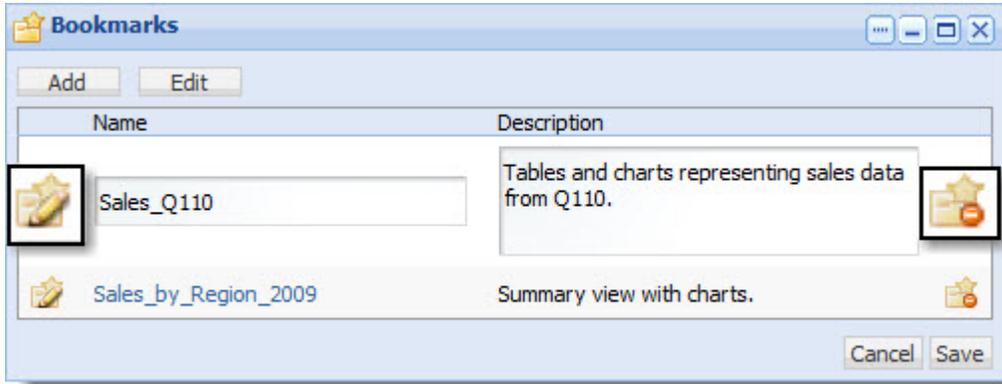
This topic describes how an end user can use the **Bookmarks** component.

The **Bookmarks** component allows the end user to create, update, and delete bookmarks.

To add a new bookmark, click **Add**. The **Create new bookmark** box, where you can fill in the **Name** and **Description**, displays. Click **Create bookmark** to add the bookmark.



To edit or delete a bookmark, click **Edit**. To edit the bookmark, click the icon to the left of its name (highlighted below). To delete it, click the icon to its right (highlighted below). Click **Save** to commit your changes.



### What Bookmarks saves for each component type

This topic describes what parts of the component state are saved in the bookmarks created by the **Bookmarks** component.



**Note:** Administrative components are not saved in bookmarks, so they are not included here.

Component	Persisted States	Comments
<b>Advanced Visualization</b>	Developer-driven	<p>The data persisted in a bookmark for this component is specific to the SAP Dashboard Design (Xcelsius) movie being used. It is therefore the responsibility of the movie developer to define which elements are to be persisted. This requires two movie components to be defined via the Xcelsius Data Manager:</p> <ol style="list-style-type: none"> <li>1. Definition of name-value pairs (established in the Flash Variables data connection) for each stateful design element in the UI. For example, the Drilldown Chart example movie includes an "Average Price by:" drop-down control that drives the Analytics grouping used to render the chart. This is backed by a Flash Variable called "selectedGroupBy" whose value is bound to that of the drop-down selection, so that when a bookmark is loaded, the correct grouping is applied based on which grouping was selected when the bookmark was created. That is, if a user selected "Vintage" from the drop-down and then created a bookmark, then Vintage should be the selected drop-down value when the movie is loaded from a bookmark.</li> <li>2. An External Interface Connection (EIC) variable called "movieState" that is bound to a cell that contains an ampersand-delimited list of all variables defined in step one. For example, the Drilldown Chart example also includes a radio button control that can either be "yes" or "no" at any given point in time, whose corresponding Flash Variable is called "selectedRadioButton." In order to persist both the drop-down and radio button controls properly, the "movieState" EIC variable is bound to a cell that contains a string that looks like this:</li> </ol>

Component	Persisted States	Comments
		<p><code>selectedGroupBy=Vintage&amp;selectedRadionButton=yes</code></p> <p>The Xcelsius developer is responsible for setting up both sets of connections properly and binding them to appropriate cells in the spreadsheet that is leveraged by the visualization.</p>
<b>Bookmark</b>	None	Nothing about the <b>Bookmarks</b> component itself is stateful.
<b>Breadcrumbs</b>	Expanded multi-select dimensions	When multi-select dimension values are present in the <b>Breadcrumbs</b> component, they may be grouped into a collapse/expand control (depending on the number of values and the threshold set in the component's preferences). The state of each multi-select dimension's collapse/expand control is stateful and will be stored in a bookmark. For example, if you select Flavors of "Apple", "Berry" and "Cherry" in the sample wine data set, the collapse/expand control appears and is collapsed by default. If you expand it and then create a bookmark, the control will be expanded when that bookmark is loaded.
<b>Chart</b>	Metric, x-axis, and cross tab drop-down lists	When configuring a chart, the power user can define multiple metrics, x-axis, and cross tab groupings. When the chart is configured in this way, drop-down lists are provided for each. The user's selection for the <b>Metric</b> , <b>Group By (X-Axis)</b> , and <b>Cross Tab</b> drop-down lists are persisted with a bookmark.
<b>Find Similar</b>	None	
<b>Guided Navigation</b>	Expanded dimensions and collapsed dimension groups	Any dimension that is expanded so that its refinements are exposed is persisted in a bookmark, so that it will still be expanded when the bookmark is loaded. Dimension groups are, by default, expanded. If you collapse a dimension group, this is persisted in a bookmark, so that it will be collapsed when the bookmark is loaded.
<b>Metrics Bar</b>	None	
<b>Range Filter</b>	None	
<b>Record Details</b>	None	
<b>Results Table</b>	Sort state and number of records per page	If you have sorted on a column and create a bookmark, the sort will be applied when the bookmark is loaded. If you have selected a certain number of records per page and create a bookmark, the results table will display that number of records per page when you load the bookmark.
<b>Search Box</b>	None	
<b>Tabbed Component Container</b>	The tab in focus	If you create a bookmark when on a tab other than the first one, that tab will be reloaded properly when the bookmark is loaded.

## Configuring Bookmarks

The **Bookmarks** component requires no configuration—you simply add it to the portal workspace.

## Results and visualization components

The components in this section provide a more detailed view of your data.

### Related Links

[Advanced Visualization](#) on page 33

The **Advanced Visualization** component provides SAP Crystal Dashboard Design (Xcelsius) visualizations within your application.

[Chart](#) on page 35

The **Chart** component lets you access Corda-based Analytics charting of your data.

[Cross Tab](#) on page 47

The **Cross Tab** component provides a table that allows end users to perform comparisons and identify trends across several cross sections of data.

[Results Table](#) on page 52

The **Results Table** component provides a simple interface for displaying results. The **Results Table** component can show results from Analytics and non-Analytics queries.

[Metrics Bar](#) on page 56

The **Metrics Bar** component allows users to quickly view metrics that summarize various aspects of the underlying data.

## Advanced Visualization

The **Advanced Visualization** component provides SAP Crystal Dashboard Design (Xcelsius) visualizations within your application.

This integration provides Discovery Framework end users with dynamic insight into MDEX Engine powered data.

The **Advanced Visualization** component ships with three sample Flash movies in SWF format and the associated source files in XLF format.

### Dependencies

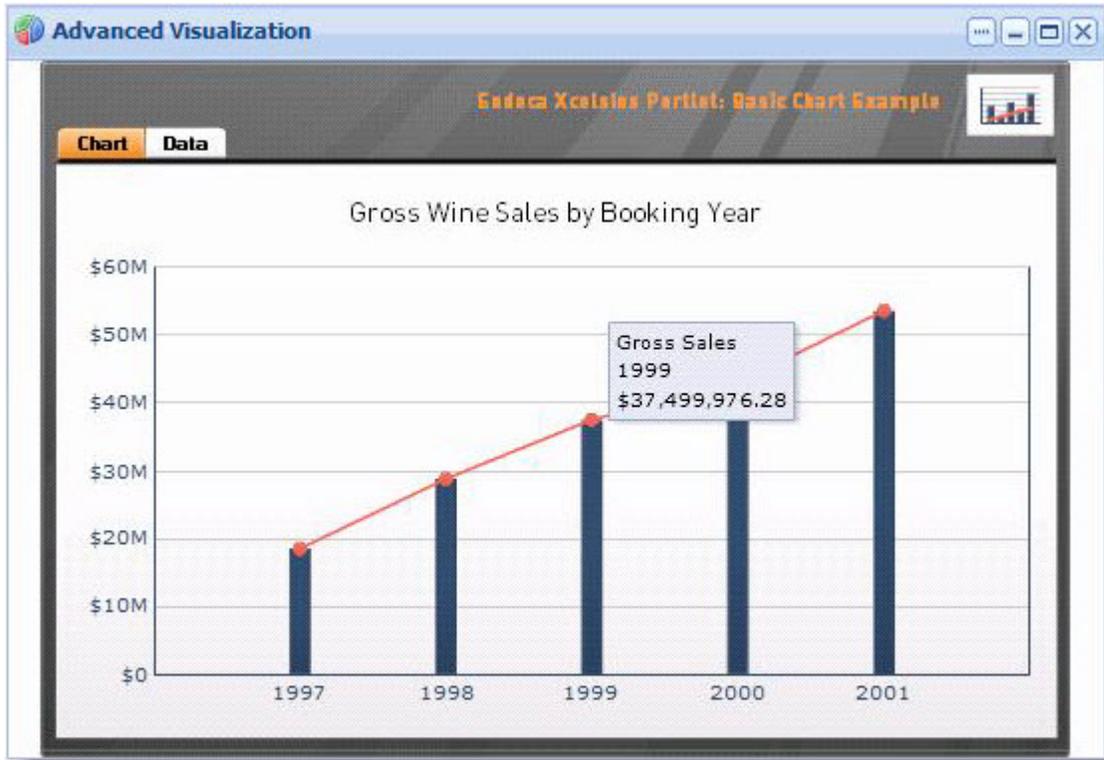
The **Advanced Visualization** component has the following dependencies:

- In order to use this component, you must purchase a license for SAP Crystal Dashboard Design 3.1. Once you have done so, you can download the software, along with a temporary license, from the Discovery Framework downloads page on [EDeN](#). To obtain a permanent license, contact Endeca Support.
- An experienced SAP Crystal Dashboard Design developer needs to create the Flash movies associated with your data. Once he or she has done so, it is easy to add the new movies to the Discovery Framework.
- This component requires the use of Endeca Analytics.

## Using Advanced Visualization

This topic describes how an end user can use the **Advanced Visualization** component.

The **Advanced Visualization** component provides SAP Crystal Dashboard Design interactive data visualization of your Discovery Framework data. Many different chart styles are available to your SAP Crystal Dashboard Design developer.



You can mouse over elements of the chart to see the related data. In addition, you can click the **Data** tab to see the underlying data and its state.

The screenshot shows the same "Advanced Visualization" window, but with the "Data" tab selected. The chart is hidden, and a "Raw Data" table is displayed. Below the table is a section titled "Data Source State" which is currently empty.

Booking Year	Gross (\$)
1997	\$18,568,161.04
1998	\$28,812,393.85
1999	\$37,499,976.28
2000	\$43,000,000.00
2001	\$53,519,333.05

## Configuring Advanced Visualization

The power user can set the following preferences for the **Advanced Visualization** component.

- **Data source controls:** Select from the available data sources in the drop-down list, and then click **Update data source** to bind your selection to the component.

- **Flash movie settings:** Choose from the available SAP Crystal Dashboard Design movies and set the component size.

- **Reload movies:** Load all of the available movies for the selected data source. A power user can add movies to the Discovery Framework and click **Reload movies** to recognize the newly-added movies.

[Reload movies](#)



**Note:** SAP Crystal Dashboard Design movies must be supplied by an SAP Crystal Dashboard Design developer.

## Chart

The **Chart** component lets you access Corda-based Analytics charting of your data.

The end user can mouse over the chart to reveal numeric data, and click on a chart element to filter results by the values displayed in the chart. In addition, the **Chart** component can be configured so that end users can switch the chart view by selecting drop-down choices. For example, an end user may switch the Metric or Group By that is summarized in the chart.

### Dependencies

The **Chart** component has the following dependencies:

- In order to use this component, you must install Corda. For details, see the *Discovery Framework Installation Guide*.
- This component requires the use of Endeca Analytics.

### Analytics statements supported in the Chart component

This topic describes the three types of Analytics statements that can be expressed in the **Chart** component.

The Analytics use cases are:

- **Zero GROUP BY statements, with one or more metrics** One or more metrics summarized by a single unspecified grouping (the name of which defaults to "Total").
- **One GROUP BY statement, with one or more metrics** One or more metrics summarized by a single specified categorization.
- **Two GROUP BY statement, with one metric** One metric summarized by a specified categorization, with a second GROUP BY statement specified as the *cross tab*. (Cross tabs are grouped aggregations such as cross-tabulated totals over one or more dimensions.)

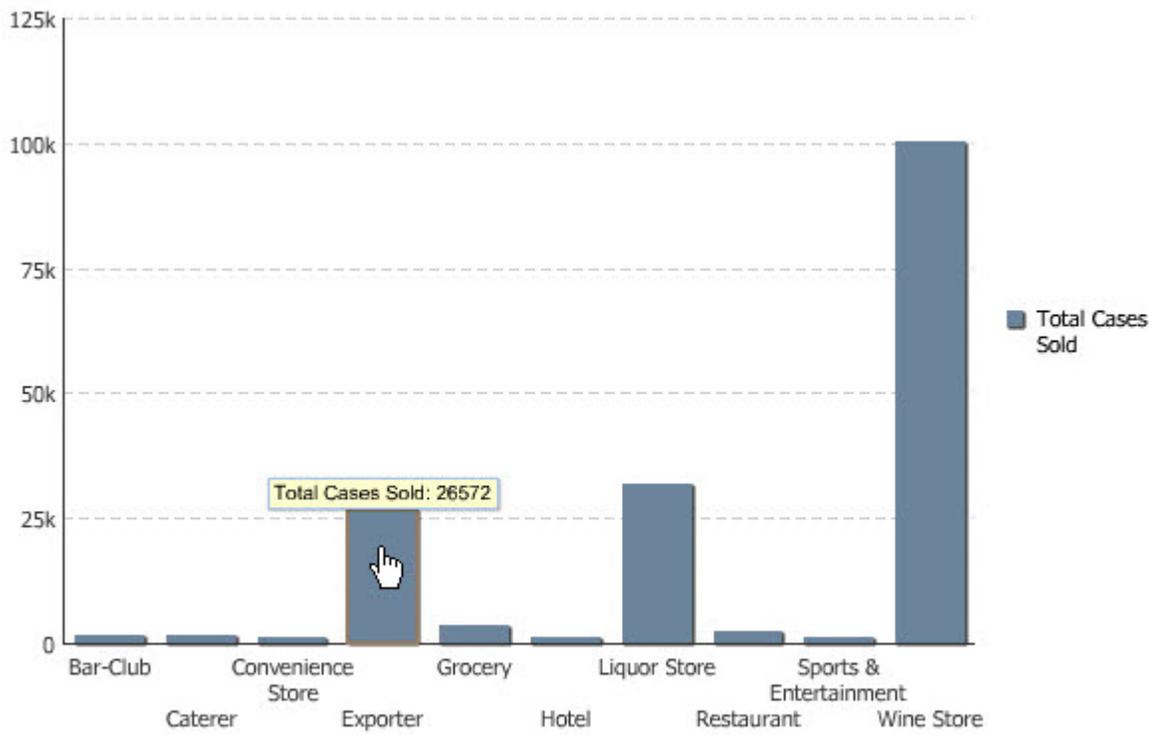
Various expressions of these types are shown in the chart style examples later in this section.

### Using Chart

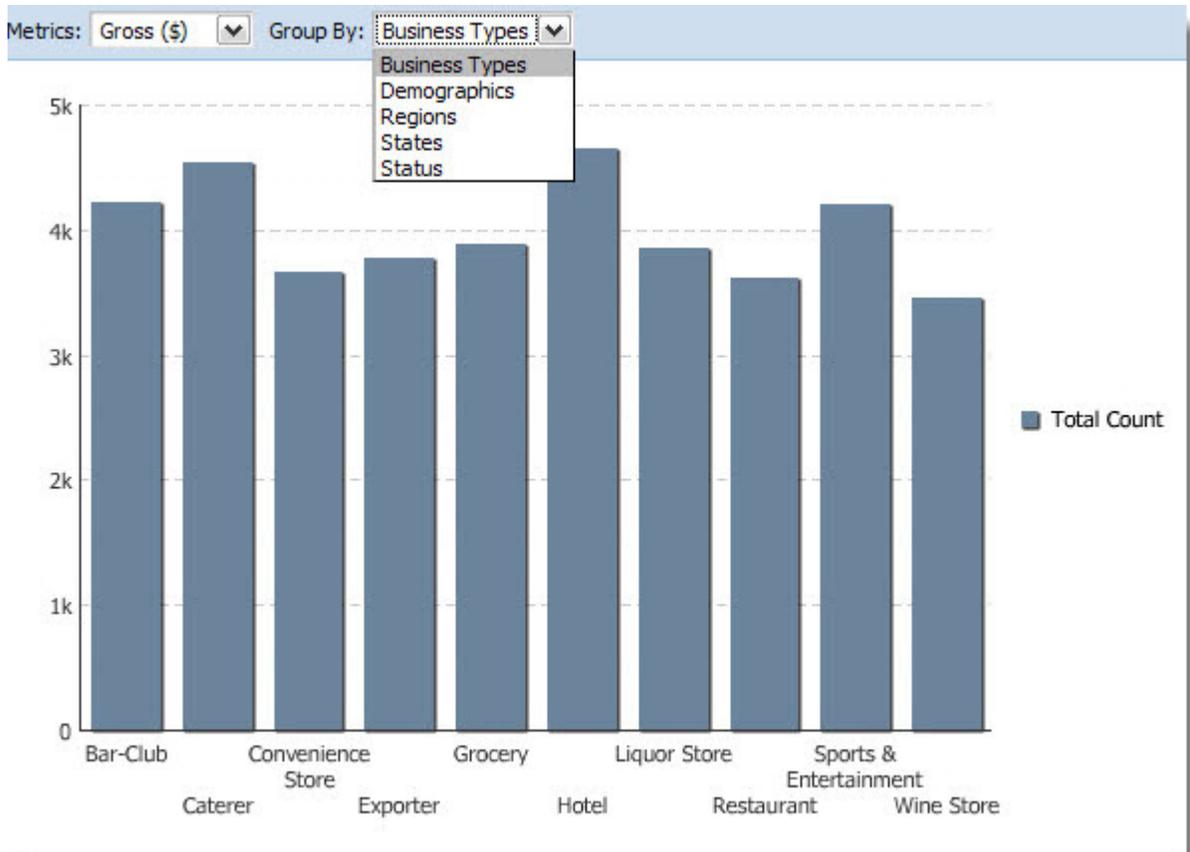
This topic describes how an end user can use the **Chart** component.

The power user who configures your Analytics statements can provide you with two kinds of charts: fixed and parametric. Examples of each are shown below.

The following chart is based on a fixed Analytics statement. You can mouse over chart elements to see the related data, or click to drill down. Drill down is supported for dimension GROUP BY statements but not for property GROUP BY statements.



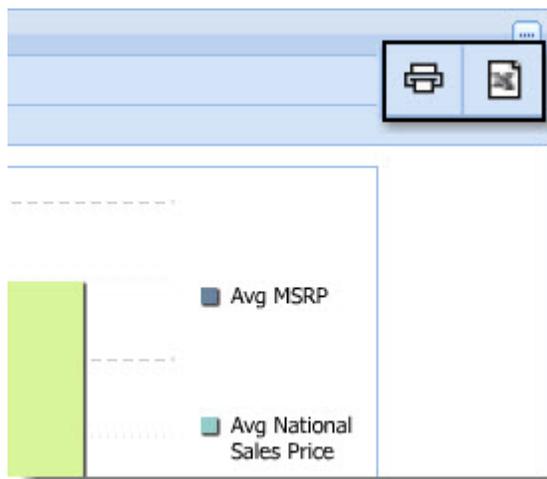
The following chart is based on a parametric Analytics statement. You can modify the chart display by selecting different values from the drop-down lists at the top of the chart.



In addition, the power user has the ability configure multiple charts, based on different Analytics statements and chart styles, that you can select from a drop down list.

### Printing and exporting

End users also have the ability to print and/or export the chart they have created. The end user accesses these features with the **Print** and **Export** buttons, shown below.



When the user selects **Print**, the chart displays in another browser window. The user prints as normal via the **File** menu or keyboard shortcut.

**Export** allows the user to export the chart to Excel.

## Configuring Chart

The power user can set the following preferences for the **Chart** component.

The power user can add a fixed Analytics statement to the **Chart** component, or configure a parametric Analytics statement that allows end users to see drop-down lists across the top of the chart.



**Note:** Unlike most components, the **Chart** component includes online help. This help, which mostly provides tips on working with Analytics statements, can be accessed from the same menu that contains the **Preferences** option. For more detailed information on the Endeca Analytics language, see the *Endeca Analytics Guide*.

**Chart Options**

Chart Height: 300    Chart Width: 600    Chart Select Label: Select Chart

HTML Height: 300    HTML Width: 600    Analytics Threshold: 1000000

Analytics Threshold Exceeded Message: Too many results to display. Filter results further to view chart

---

**Yearly Averages**

Chart Title: Yearly Averages    Style: Vertical Bar

```
RETURN YearlyAverages AS SELECT
AVG("Product MSRP ($)") AS "Avg MSRP",
AVG("Product Avg National Sales Price ($)") AS "Avg National Sales
...
```

Please see the help page for instructions on writing the statement. For example: RETURN "MyChart" as SELECT AVG({METRIC}) as {METRIC} GROUP BY {XAXIS}, {CROSSTAB} (values for {METRIC}, {XAXIS} and {CROSSTAB} are filled in from the Metric, X Axis and Cross tab options below).

**Metric Options (one per line):**    **X Axis Options (one per line):**    **Cross tab (one per line):**

Values in these fields are presented to the user in drop-down menus. Metric options are substituted for variable {METRIC} in the analytics statement, X Axis options for variable {XAXIS}, and Cross tab options for variable {CROSSTAB}. If these variables are not used in the statement, the drop-down menu(s) will have no effect.

**Metric Selector**

- Avg MSRP
- Avg National Sales Price
- Avg Sales Price

Save Preferences

- **Data source controls:** Select from the available data sources in the drop-down list, and then click **Update data source** to bind your selection to the component.

- **Chart options:** Define the height and width of the chart and set a name for the control that allows end users to choose from multiple charts when available. In addition, set a threshold for the number of Analytics results the query returns, and create a message to display if the query results exceed that threshold.

- **Chart Title and Style:** Name the chart and select the type of chart to display. The **Chart** component supports the following chart styles: Vertical Bar, Stacked Vertical Bar, Horizontal Bar, Line, Pie, and Line Bar Combo.

- **Chart configuration tabs:** Click the **+** to add additional chart tabs. You can configure a different chart on each tab for your end users.

- **Analytics Statement:** Type your Analytics statement here. Note that labels can be configured for metrics only, using the AS functionality in an Analytics statement. After the Analytics statement has been entered, use the **Test analytics query** button to test the validity of the statement before saving preferences.

- **Metric Options, X Axis Options, and Cross tab:** For parametric Analytics charts, provide options for the drop-down lists your end users will access. Cross tabs are grouped aggregations such as cross-tabulated totals over one or more dimensions.

- **Metric Selector:** Allows you to specify an ORDER BY in the Analytics statement without including it in the display. In the example below, if you did not want to include the statement's ORDER BY "Avg Sales Price" metric in the chart, you would un-check that metric in the list. Note that the **Metric Selector** does not appear until you have tested and saved the Analytics query.

**Metric Selector**

- Avg MSRP
- Avg National Sales Price
- Avg Sales Price

- **Enable Top-N Sort** and **Enable end user control:** When the **Enable Top-N sort** box is checked, the chart's results can be sorted in ascending or descending order. If the **Enable end user control** box is also checked, the end user can toggle the sort order.

**Enable Top-N Sort:**

Show   by

**Enable end user control**

## Chart styles provided by the Chart component

The following chart styles are supported by the **Chart** component.

- Vertical Bar
- Stacked Vertical Bar
- Horizontal Bar
- Line
- Pie
- Line Bar Combo

Examples of each of these chart styles appears in the following topics.

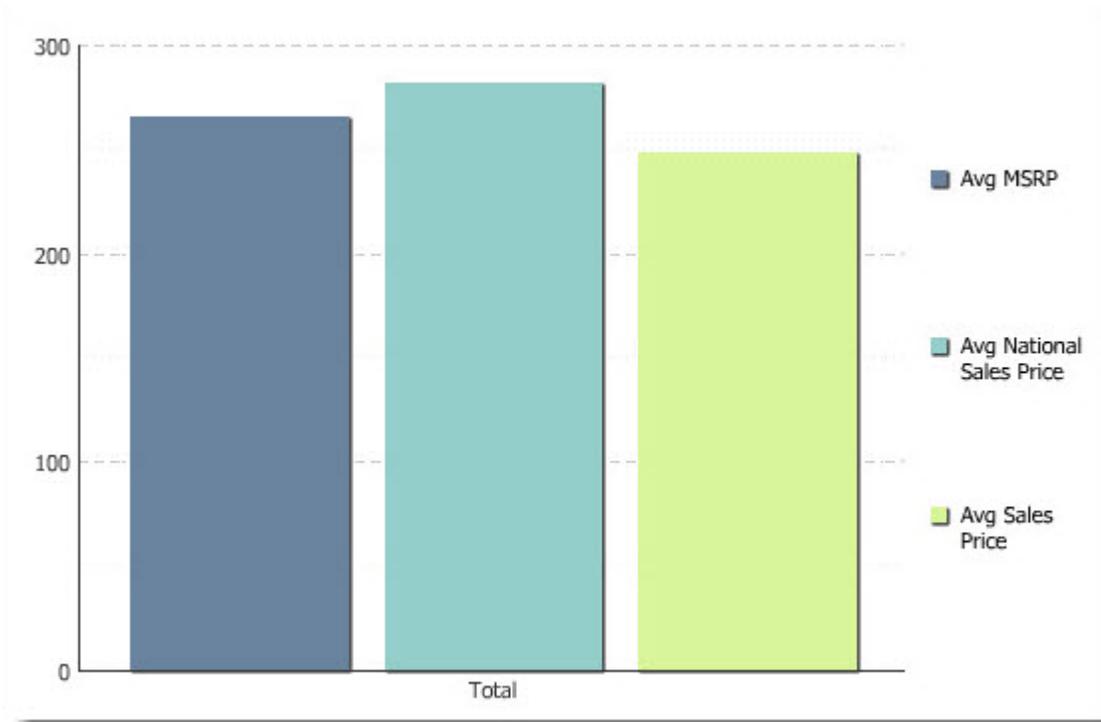
### Vertical bar chart

This topic contains an example of a vertical bar chart.

The following example uses a zero GROUP BY statement with three metrics:

```
RETURN YearlyAverages AS SELECT
  AVG("Product MSRP ($)") AS "Avg MSRP",
  AVG("Product Avg National Sales Price ($)") AS "Avg National Sales
Price",
  AVG("Unit Sale Price ($)") AS "Avg Sales Price"
GROUP
```

It produces a vertical bar chart displaying "Avg MSRP", "Avg National Sales Price" and "Avg Sales Price" metrics:



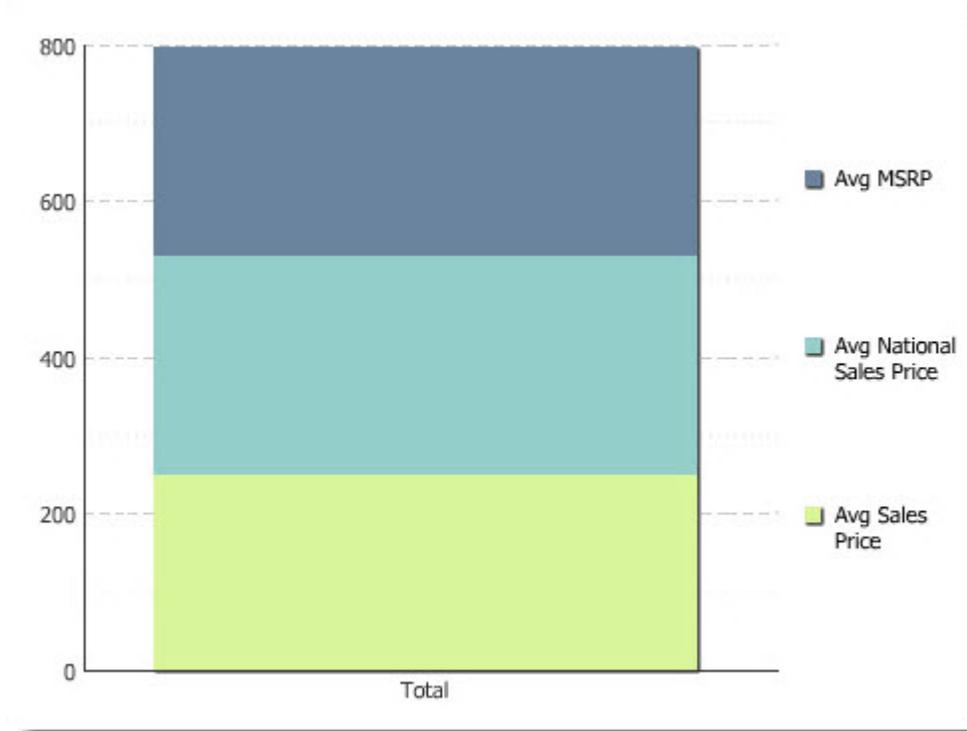
### Stacked vertical bar chart

This topic contains an example of a stacked vertical bar chart.

This example uses an Analytics statement with two GROUP BY statements. One metric is summarized by a specified categorization, and the second GROUP BY is specified as the cross tab.

```
RETURN RegionalMargin AS SELECT
  SUM("Gross ($) * "Margin (%)") / SUM("Gross ($)") AS "DollarWeightedAvgMargin"
GROUP BY "Regions", "Demographics"
```

The stacked vertical bar chart below summarizes the “DollarWeightedAvgMargin” metric for each Region (which is the first GROUP BY and therefore the primary category) and presents a cross-tabulation with each Demographic (which is the second GROUP BY and therefore the cross tab):



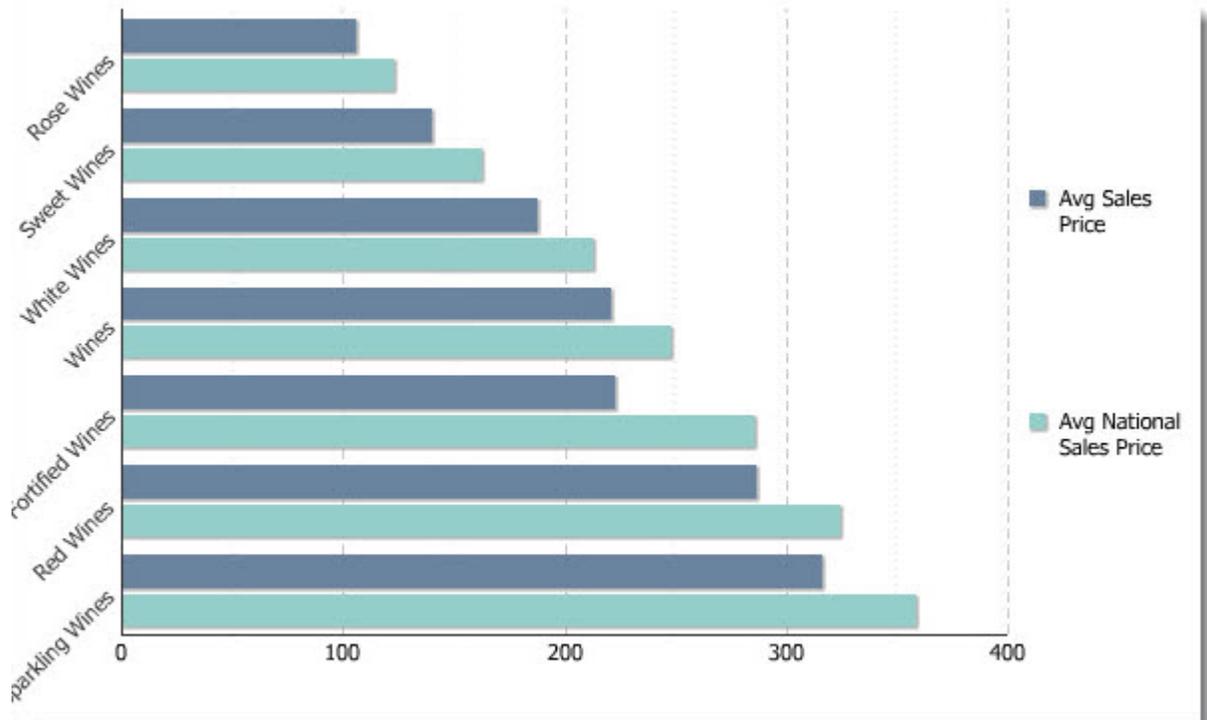
### Horizontal bar chart

This topic contains an example of a horizontal bar chart.

This example uses a single GROUP BY statement with two metrics:

```
RETURN NumCases AS SELECT
  AVG("Unit Sale Price ($)") AS "Avg Sales Price",
  AVG("Product Avg National Sales Price ($)") AS "Avg National Sales
Price"
GROUP BY "Varietals" ORDER BY "Avg National Sales Price"
```

The horizontal bar chart compares the "Avg Sales Price" metric to the "Avg National Sales Price" for each varietal:



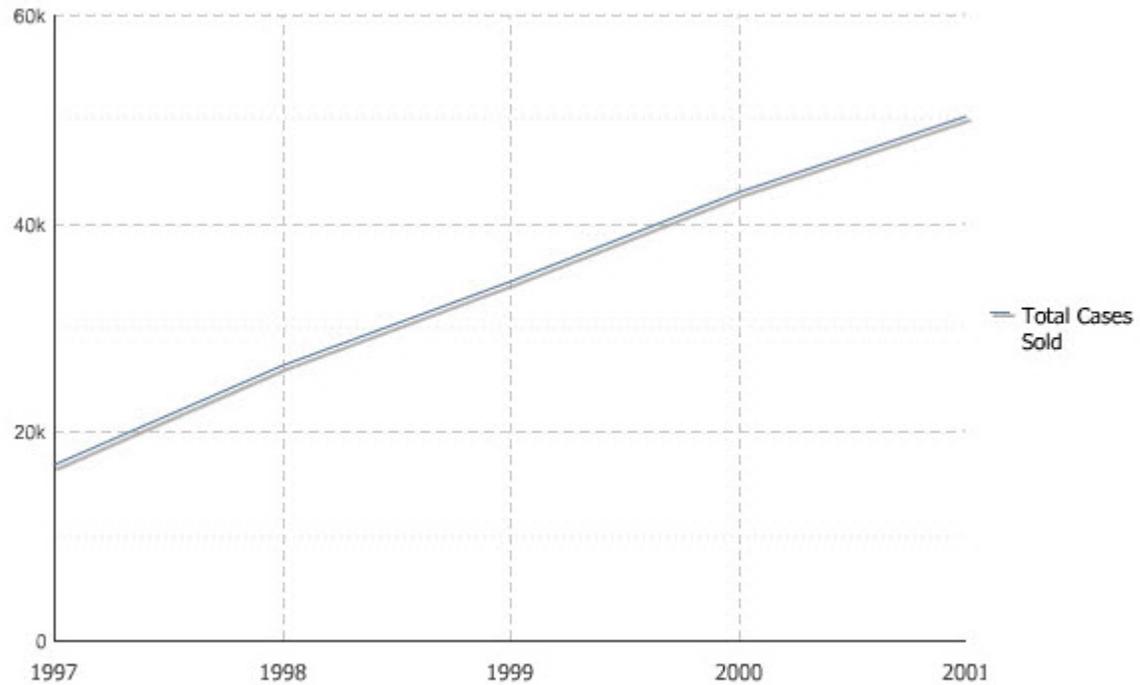
### Line chart

This topic contains an example of a line chart.

The following Analytics statement has one GROUP BY and a single metric:

```
RETURN NumCases AS SELECT
  SUM("Number of Cases Sold") AS "Total Cases Sold"
GROUP BY "Booking Year"
```

It produces a line chart summarizing the "Total Cases Sold" metric for each Booking Year:



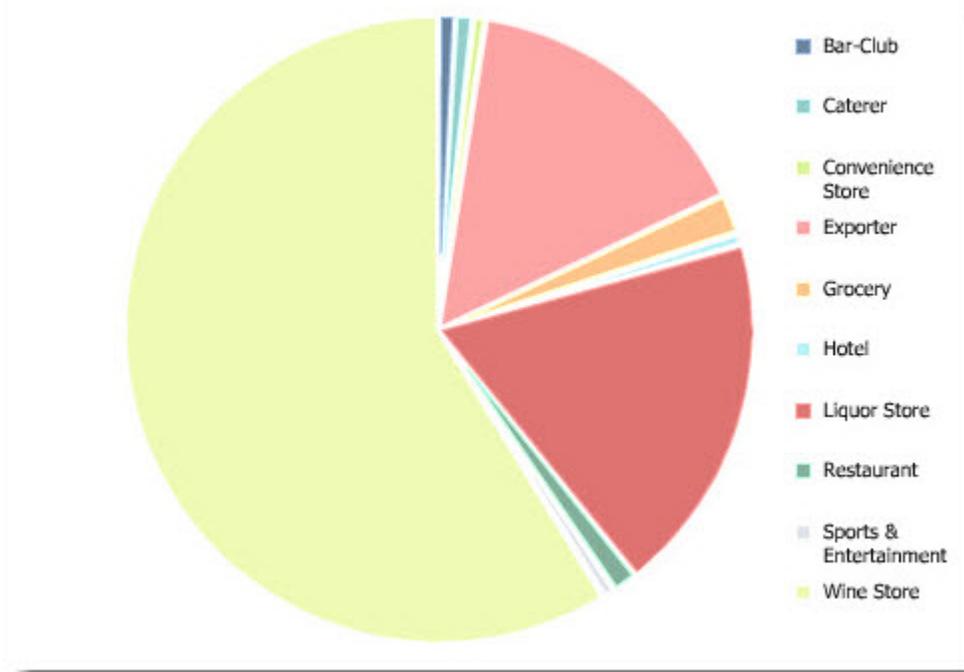
### Pie chart

This topic contains an example of the pie chart type.

The following Analytics statement contains one GROUP BY and a single metric:

```
RETURN NumCases AS SELECT
    SUM("Number of Cases Sold") AS "Total Cases Sold"
GROUP BY "Business Types"
```

It produces a pie chart summarizing the "Total Cases Sold" metric for each Business Type:



**★ Important:** The pie chart type only supports a single GROUP BY statement. If you have multiple GROUP BYs in your statements, the chart will not display or execute drill-down as expected.

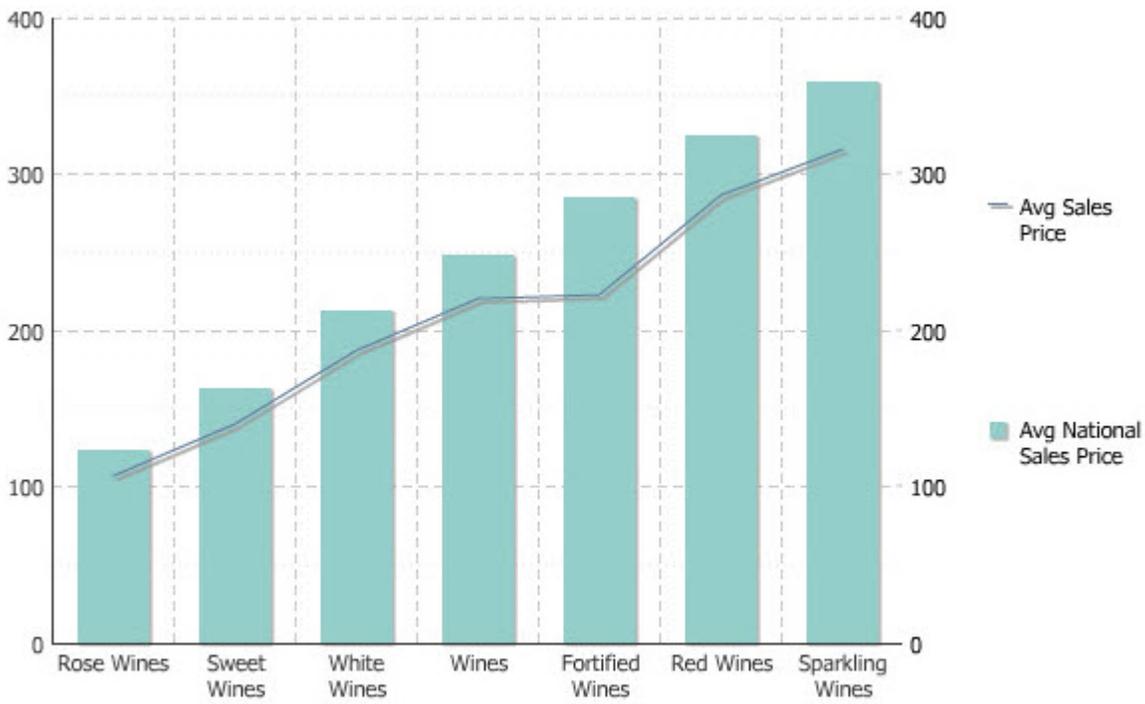
### Line bar combo chart

This topic contains an example of a line bar combo chart.

The following example contains one GROUP BY statement and two metrics:

```
RETURN NumCases AS SELECT
  AVG("Unit Sale Price ($)") AS "Avg Sales Price",
  AVG("Product Avg National Sales Price ($)") AS "Avg National Sales
Price"
GROUP BY "Varietals" ORDER BY "Avg National Sales Price"
```

It produces a line bar combo chart comparing the "Avg Sales Price" metric to the "Avg National Sales Price" for each varietal:



## Cross Tab

The **Cross Tab** component provides a table that allows end users to perform comparisons and identify trends across several cross sections of data.

A cross tab consists of a specific configuration of rows, columns, and summary cells that together make up the table body. The values displayed in the header rows and columns represent every possible grouping value of the specified data fields. Summary cells exist at the intersection of rows and columns. The value of a given summary cell is a metric corresponding to the intersection of those two groupings.

Cross Tab												
1997												
	First Quarter			Fourth Quarter			Second Quarter			Third Quarter		
	# Trans	avgMargin	avgGross	# Trans	avgMargin	avgGross	# Trans	avgMargin	avgGross	# Trans	avgMargin	avgGross
Austin Fyles	53	\$129.19	\$3,096.79	61	\$133.57	\$3,489.57	55	\$122.25	\$2,977.60	35	\$155.20	\$3,294.18
Jesenia Milman	71	\$133.95	\$3,246.36	95	\$125.46	\$3,386.29	80	\$118.19	\$2,886.56	60	\$126.33	\$3,144.19
Katharina Cafferky	44	\$149.95	\$3,728.25	60	\$140.72	\$4,545.75	52	\$157.50	\$4,084.33	27	\$120.89	\$3,408.31
Leida Gallander	55	\$123.77	\$3,368.03	88	\$127.51	\$3,042.26	65	\$119.46	\$3,379.75	52	\$110.20	\$2,996.39
Olevia Giardina	41	\$114.20	\$3,701.36	59	\$117.83	\$3,857.88	37	\$139.58	\$3,940.03	31	\$108.24	\$3,914.05
<b>Total</b>	<b>264</b>	<b>\$130.47</b>	<b>\$3,392.61</b>	<b>363</b>	<b>\$128.60</b>	<b>\$3,588.54</b>	<b>289</b>	<b>\$129.06</b>	<b>\$3,365.21</b>	<b>205</b>	<b>\$123.71</b>	<b>\$3,283.51</b>



**Note:** This component requires the use of Endeca Analytics.

## Using Cross Tab

The end user views the cross tab result table, but cannot edit or modify it.

## Configuring Cross Tab

The power user can set the following preferences for the **Cross Tab** component.

- **Data source controls:** Select from the available data sources in the drop-down list, and then click **Update data source** to bind your selection to the component. (If you are changing data sources, you must do so before entering your Analytics statement.)

- **Analytics Statement:** Enter an Analytics statement to return cross-tab values.

Analytics Statement:

```
return s1 as select count(1) as Transactions, avg("Margin ($)") as avgMargin, avg("Gross ($)") as avgGross group by "Booking Quarter", "Booking Year", "Shipping Companies"
```

Cell Limit 3000

Respect data source query state

Save Test analytics query...

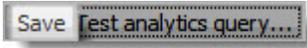


**Note:** Using fields with too many values in your GROUP BY clauses may cause your cross tab results table to become too large to be useful.

- **Cell Limit:** Set a threshold to prevent too large a grid from being rendered on the **Cross Tab** page. In general, you should not make this value, which represents the number of cells rendered in the cross tab results table, much larger than the default. A table with many thousands of cells is time-consuming to render and difficult for the end user to read.
- **Respect data source query state:** When this box is checked, the **Cross Tab** component is dependent on its data source query state, which means the metric values should update when the data source query state changes. When the box is unchecked, it is independent of the data source query state, so the metric values should remain constant regardless of the data source query state.

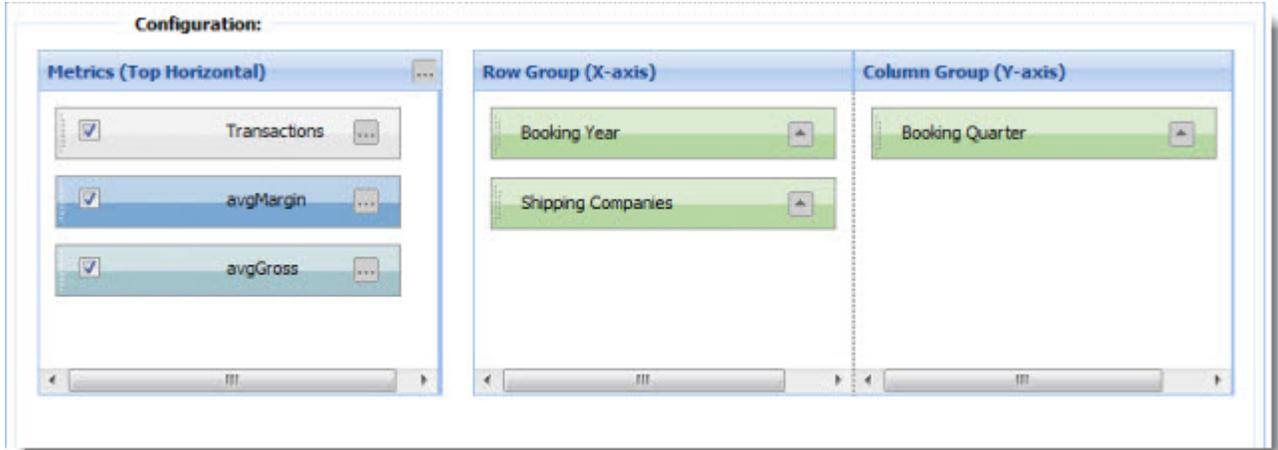
- **Test analytics query:** Validate the Analytics statements before loading the metrics into the component. (This step is required.)

After you test and debug your Analytics query, click **Save**.

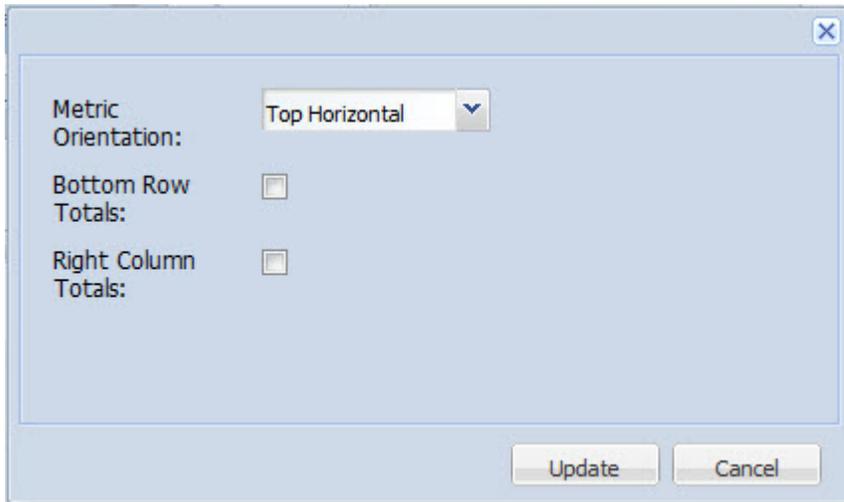


When you save, the **Configuration** section is populated with metric, row, and column information.

- **Configuration:** Adjust the table that your end users will see.



In the **Metrics** section, you can configure general metric options or an individual metric. Click the ... button in the top right corner of the panel to launch its formatting options, where you can change the axis on which metrics are displayed or add metric totals.



For any individual metric, click the . . . button on its label to access formatting options, where you can change the display name and format the data.

Metric Name: Transactions

Metric Display Name: Transactions

Format: Integer

**Formatter Options**

Thousand Separator: Comma (,)

Add percent (%) suffix:

OK Cancel

You can drag metrics to reorder them, and remove a metric from the cross tab result table display by unchecking it.

**Metrics (Top Horizontal)**

Transactions

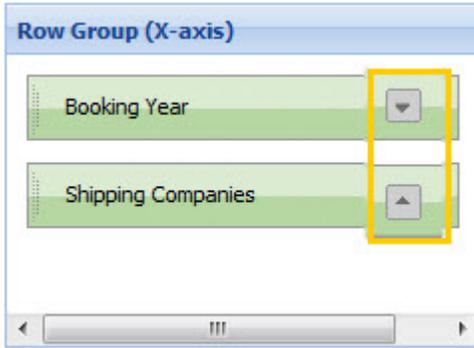
avgMargin

avgGross

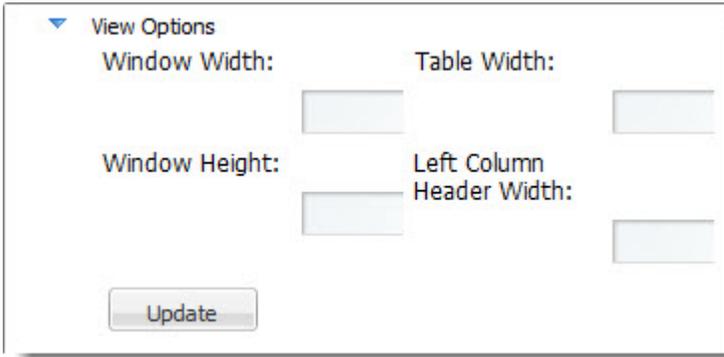


**Note:** To improve performance, if you do not want to display a metric, it is better to remove it from the Analytics statement than it is to uncheck it.

In the **Row Group** and **Column Group** section, you can drag items anywhere in the row and column panels. Each GROUP BY item contains a toggle that can be used to set ascending or descending order.



- **View Options:** Modify the dimensions (in pixels) of the table and its window.



 **Note:** The initial generated table will be fit into the size of the component, regardless of how many rows or columns there are. You can use these controls to make the table larger than the component space and navigable with scroll bars.

- **Preview table:** Preview the cross tab display as it will appear to the end user.

The screenshot shows a 'Cross Tab' window with a table of freight data. The table is organized by year (1997 and 1998) and then by company. The columns are '# Trans', 'avgMargin', 'avgGross', and '#'. The data is as follows:

		First Quarter			
		# Trans	avgMargin	avgGross	#
1997	A. & S. Transport	95	\$130.66	\$3,608.00	
	J. & B. Freight	30	\$118.34	\$2,634.60	
	Old Colony Freight	50	\$123.10	\$2,482.60	
	Red Freight	50	\$138.29	\$3,993.30	
	U.S. Express	39	\$138.77	\$3,847.60	
1998	A. & S. Transport	158	\$116.98	\$3,535.50	
	J. & B. Freight	38	\$127.31	\$4,222.70	
	Old Colony Freight	64	\$143.64	\$4,313.60	
	Red Freight	57	\$140.10	\$4,179.50	
	U.S. Express	71	\$124.50	\$4,274.90	
1999	A. & S. Transport	191	\$122.98	\$3,798.20	
	J. & B. Freight	73	\$121.56	\$3,370.00	
	Old Colony Freight	64	\$143.64	\$4,500.70	

## Results Table

The **Results Table** component provides a simple interface for displaying results. The **Results Table** component can show results from Analytics and non-Analytics queries.

It provides a list of records in table form and enables users to view a record's details. Users can page through, sort, and scroll across large tables, switch between tabs, and drill down on selected results. Pinned columns help users keep track of the data they are viewing.

The screenshot shows a 'Results Table' window with a table of wine records. The table has columns for P\_WineType, P\_Name, P\_Score, P\_Region, P\_Price, and P\_DateRe. The data is as follows:

P_WineType	P_Name	P_Score	P_Region	P_Price	P_DateRe
Zinfandel	Zinfandel Central	88	Other California	14.000000	10/15/96
White	Albarino Rias Bai	25	Spain	18.000000	06/15/93
White	Alsace Gentil	83	Alsace	9.000000	11/15/94
White	Alella Marques de	76	Spain	11.000000	04/15/94
White	Alsace Grand Cru	84	Alsace	16.000000	11/15/94
White	Chardonnay Nap.	82	Napa	11.000000	07/31/95
White	Chardonnay Nap.	81	Napa	18.000000	07/31/96
White	Sauvignon Blanc	79	New Zealand		05/15/94
White	Greece Laoutari	78	Greece	19.000000	04/30/97
White	Chardonnay Utiel	75	Spain	6.000000	04/30/95

At the bottom of the window, there is a pagination control showing 'Page 1 of 5708' and 'Records per page' with a refresh icon. The status bar indicates 'Displaying records 1 - 10 of 57076'.



**Note:** The **Results Table** component supports the use of aggregated records. For details about configuring Discovery Framework data sources for aggregated records, see the *Discovery Framework Installation Guide*.

### Dependencies

To see the details of a selected record, you need to have an instance of the **Record Details** component available in the application, either on the same page or on a page specified as the target.

### About attribute names

- Attributes (properties and dimensions) with parentheses in their names are not supported in the **Results Table** component.
- As a best practice, you should use NCName-compliant attribute names. If more user-friendly display names for attributes are required, you can provide them by using the display name feature of the **Attribute Settings** component.

### Related Links

[Record Details](#) on page 85

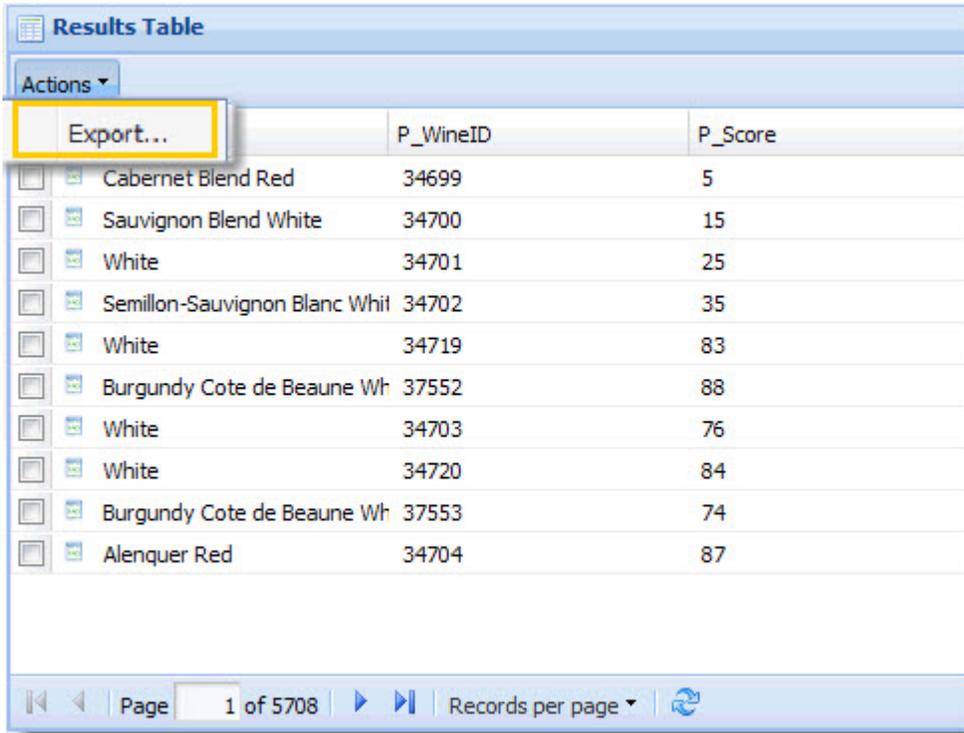
The **Record Details** component displays all of the properties and dimensions for a record in the **Results Table** component.

## Using Results Table

This topic describes how an end user can use the **Results Table** component.

The **Results Table** component provides end users with a list of records in table form and enables them to view record details for both Analytics and non-Analytics queries. The **Results Table** component enables end users to page through records, sort them by clicking the column headers and drill-down to view a record's details in the **Record Details** component.

In addition, the end user can export the record list to Excel, by clicking **Actions > Export**:



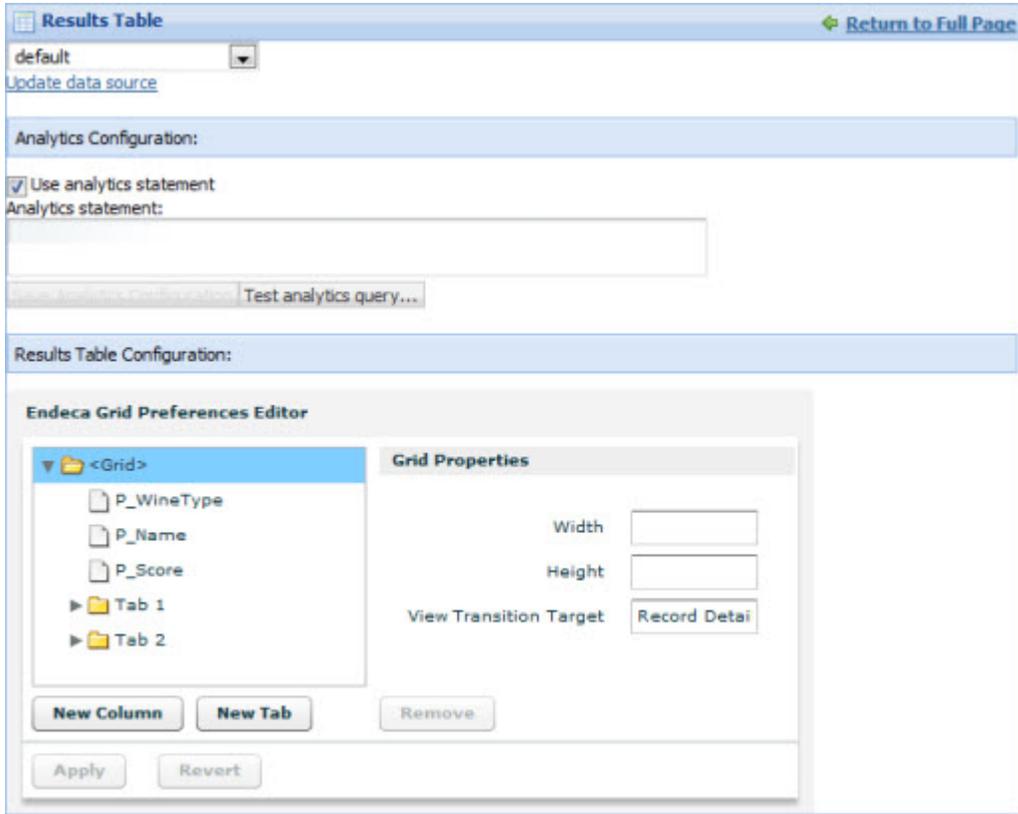
The screenshot shows a web interface for a 'Results Table'. At the top left, there is a tab labeled 'Results Table'. Below it is an 'Actions' dropdown menu with 'Export...' selected and highlighted by a yellow box. The table below has three columns: 'P\_WineID' and 'P\_Score'. The table contains 11 rows of wine data, each with a checkbox on the left. The footer of the table shows 'Page 1 of 5708' and 'Records per page' with a refresh icon.

	P_WineID	P_Score	
<input type="checkbox"/>	Cabernet Blend Red	34699	5
<input type="checkbox"/>	Sauvignon Blend White	34700	15
<input type="checkbox"/>	White	34701	25
<input type="checkbox"/>	Semillon-Sauvignon Blanc Whit	34702	35
<input type="checkbox"/>	White	34719	83
<input type="checkbox"/>	Burgundy Cote de Beaune Wh	37552	88
<input type="checkbox"/>	White	34703	76
<input type="checkbox"/>	White	34720	84
<input type="checkbox"/>	Burgundy Cote de Beaune Wh	37553	74
<input type="checkbox"/>	Alenquer Red	34704	87

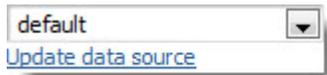
## Configuring Results Table

The power user can set the following preferences for the **Results Table** component.

You can view and sort matching results by any dimension or metric, rename and format fields for user-friendly display, and expose high volumes of information through multiple tabs.



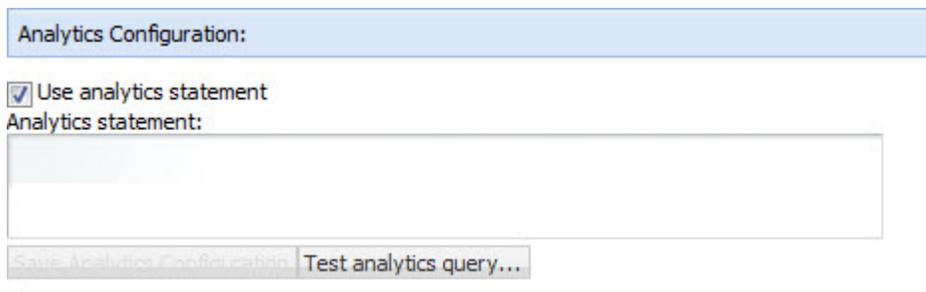
- **Data source controls:** Select from the available data sources in the drop-down list, and then click **Update data source** to bind your selection to the component.



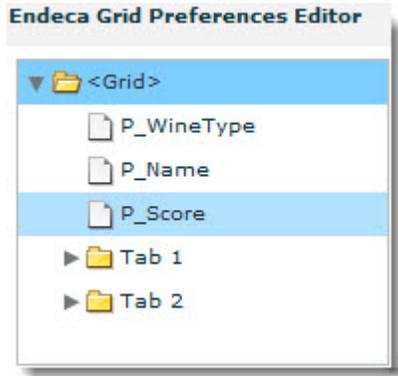
- **Analytics Configuration:** Provide table-based results display for Analytics statements.



**Note:** If **Use analytics statement** is not selected, the **Analytics statement** text field and the buttons beneath it do not appear in the **Preferences** display.



- **Grid Preferences Editor:** The grid control allows you to pin properties to ensure visibility when switching tabs, and to drag properties between tabs.



- **Grid Properties:** Set the dimensions of the grid and indicate where the results will display. If **View Transition Target** is blank, the record detail link triggers events on the same page, causing any **Record Detail** component on the same page to display the selected record. If **View Transition Target** specifies a target page, clicking the record detail link changes to the specified page, and causes any **Record Detail** component on the target page to display the selected record. The target can be specified as a relative path from `/web/guest` or an absolute path. The power user can set any column to be displayed exactly as-is ("Plain"), as HTML, as an integer, or as currency. Additionally, there are formatting options for showing or hiding the tool tip, setting the separator to use if the column has multiple values, and setting the width of the column.



- **New Column, New Tab, and Remove** buttons: Add and format columns for additional properties, add new tabs, or remove a tab or a column for a property.



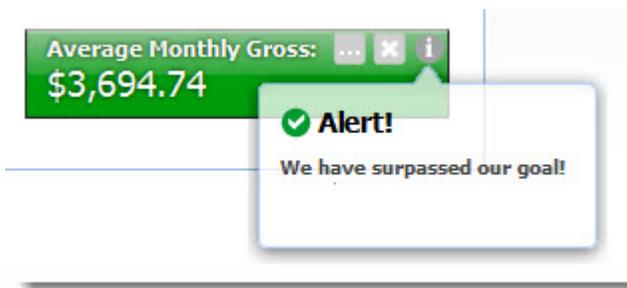
## Metrics Bar

The **Metrics Bar** component allows users to quickly view metrics that summarize various aspects of the underlying data.



The **Metrics Bar** component displays metrics based on Analytics statements written by the power user.

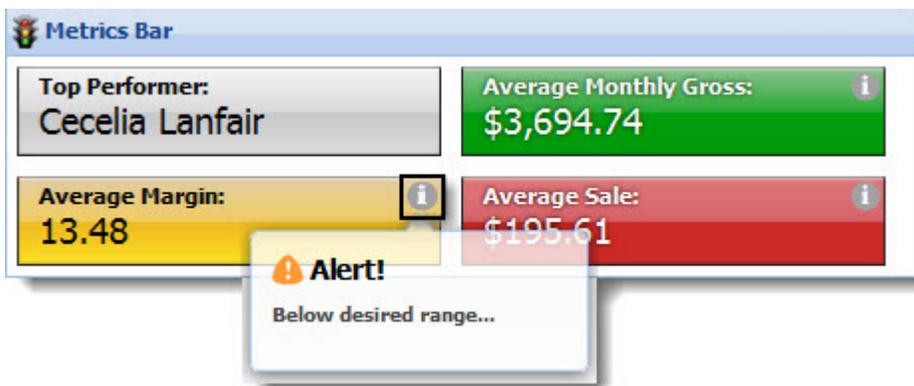
The power user is able to change the display format of the returned metric value. He or she can also set threshold violation indicators to alert the end users when a metric has reached a certain value or range of values (shown below):



## Using Metrics Bar

This topic describes how an end user can use the **Metrics Bar** component.

The following image shows a **Metrics Bar** component with four metric values on display. The end user can view the metric values and mouse over the **i** symbol (highlighted below) on each metric box to view any alerts or threshold violation indicators.



## Configuring Metrics Bar

The power user can set the following preferences for the **Metrics Bar** component.

- **Update data source:** Select from the available data sources in the drop-down list, and then click **Update data source** to bind your selection to the component.

- **Analytics Statement:** Enter an Analytics statement (or multiple statements) to return metric values.

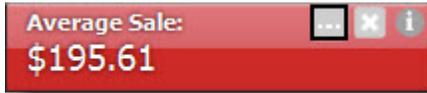
- **Respect data source query state:** When this box is checked, the **Metrics Bar** component is dependent on its data source query state, which means the metric values should update when the data source query state changes. When the box is unchecked, it is independent of the data source query state, so the metric values should remain constant regardless of the data source query state.

- **Test analytics query:** Validate the Analytics statements before loading the metrics into the component. The power user is required to test the Analytics statements before loading the statements into the component.

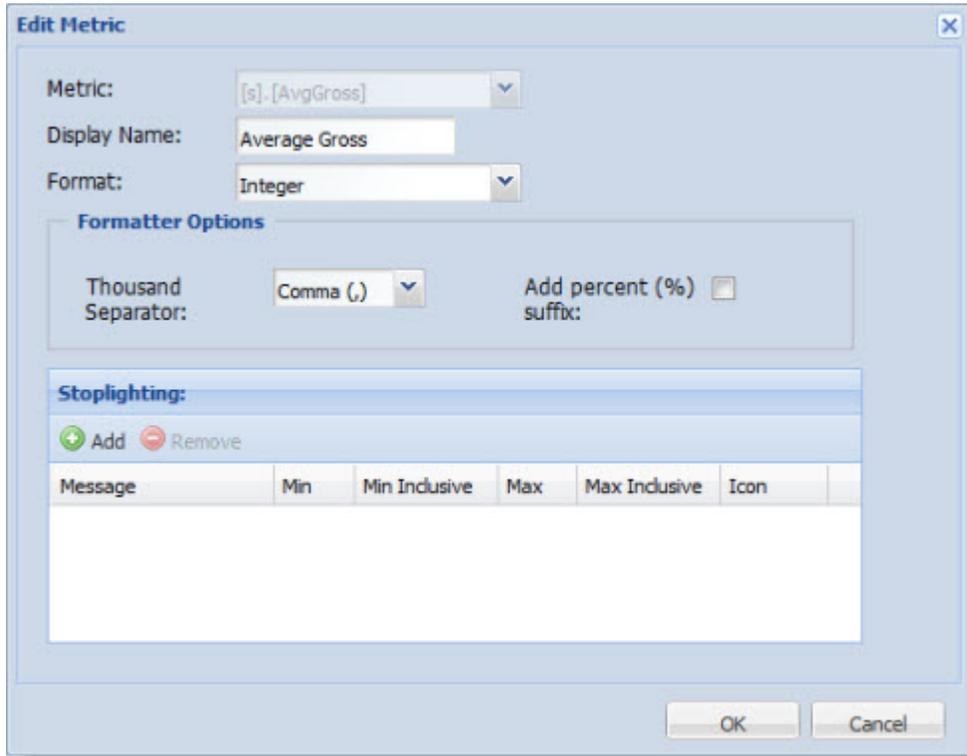
- **Load Metrics:** Save the Analytics statements and display the metric values.

## Configuring the metric boxes

To access the **Edit Metric** dialog box for an individual metric box, click ... (highlighted below):



The power user can edit the following metric box preferences:



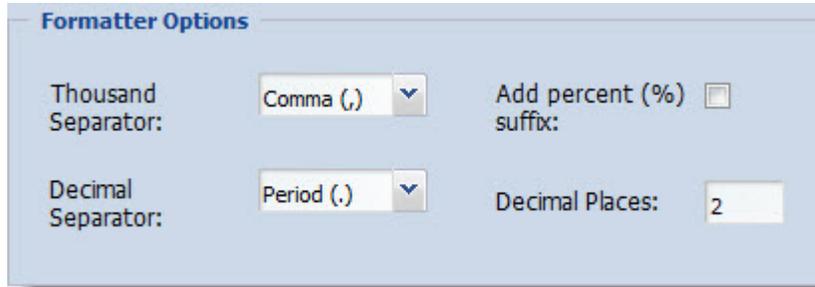
- **Edit Metric:** Select the metric to display (if applicable) and change the **Display Name** and **Format**. The **Metrics Bar** component supports decimal, currency, string, and integer formats.



- **Formatter Options:** Customize the metric's format.



**Note:** The options displayed change depending on the selected format.

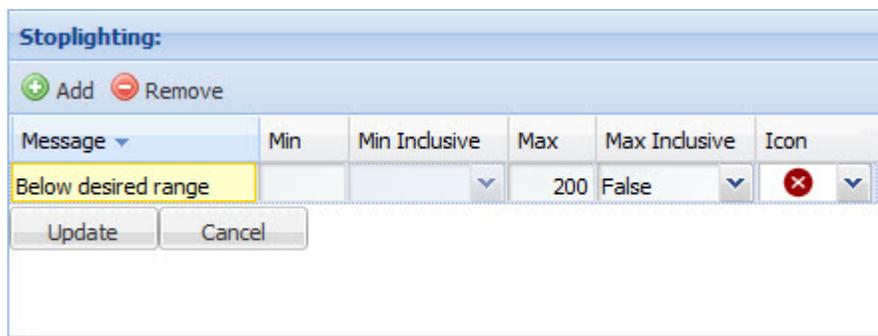


**Formatter Options**

Thousand Separator: Comma (,) Add percent (%) suffix:

Decimal Separator: Period (.) Decimal Places: 2

- **Stoplighting:** Configure threshold violation indicators. The power user determines a range of values for the indicator, chooses an icon and color, and creates an alert message to display if the metric falls within the set range. The power user can set the **Min** and **Max** values for the range, and can also set whether those values are included in the range using **Min Inclusive** and **Max Inclusive**.



**Stoplighting:**

+ Add - Remove

Message	Min	Min Inclusive	Max	Max Inclusive	Icon
Below desired range			200	False	⊗

Update Cancel

### Rearranging the metric boxes

The power user can customize the order of the metric boxes within the **Metric Bar** component by dragging and dropping the boxes into place (shown below).



Metrics Configuration (drag-and-drop to reorder):

Top Performer: Cecelia Lanfair

Average Margin: 13.48

Average Sale: \$195.61

Average Monthly Gross: \$3,694.74

The power user can also hide a metric box by clicking the **x** (highlighted below):



Average Margin: 13.48

To show a hidden metric box, select it from the drop-down menu and click the **+** (highlighted below):



## Filtering components

The components in this section allow you to search, navigate, and filter your data.

### Related Links

[Breadcrumbs](#) on page 61

The **Breadcrumbs** component provides breadcrumb navigation aid functionality.

[Guided Navigation](#) on page 64

The **Guided Navigation** component provides Endeca Guided Navigation® functionality.

[Range Filter](#) on page 70

The **Range Filter** component allows you to add and modify range filters.

[Search Box](#) on page 73

The **Search Box** component provides record and dimension search functionality.

[Find Similar](#) on page 78

The **Find Similar** component provides end users with the opportunity to find other records that share common attributes with a specific record of interest.

## Breadcrumbs

The **Breadcrumbs** component provides breadcrumb navigation aid functionality.

The **Breadcrumbs** component summarizes any Guided Navigation selections, keyword searches, or range filters specified by the end user in a vertical stack of selected values. If multiple selections from single dimension are made, they are displayed separately in sequential order. Breadcrumbs support multi-select AND and OR refinements, but not record filters.

When used with keyword search, the **Breadcrumbs** component features spelling correction, as displayed in the following example:



**Note:** The **Breadcrumbs** component supports the use of aggregated records. For details about configuring Discovery Framework data sources for aggregated records, see the *Discovery Framework Installation Guide*.

## Dependencies

The **Breadcrumbs** component requires a backing data source.

One of the following components must also be present: **Guided Navigation**, **Search Box**, or **Range Filter**.

## Related Links

[Search Box](#) on page 73

The **Search Box** component provides record and dimension search functionality.

[Guided Navigation](#) on page 64

The **Guided Navigation** component provides Endeca Guided Navigation® functionality.

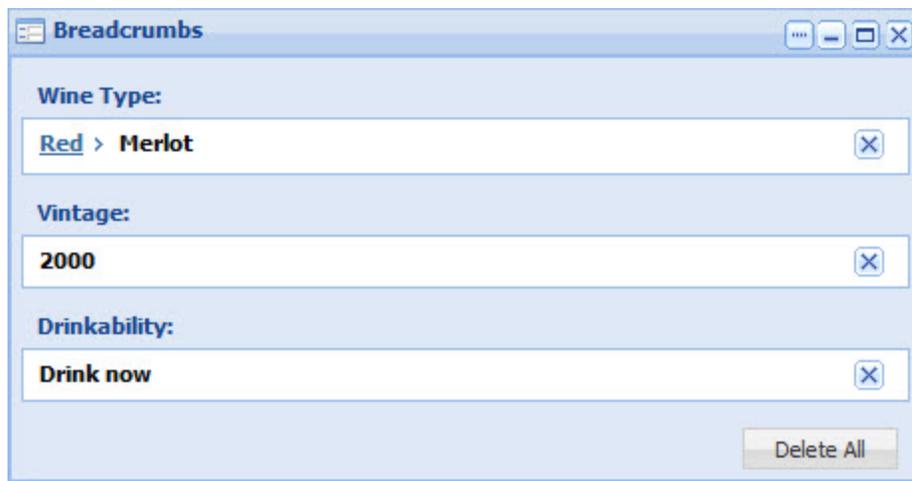
[Range Filter](#) on page 70

The **Range Filter** component allows you to add and modify range filters.

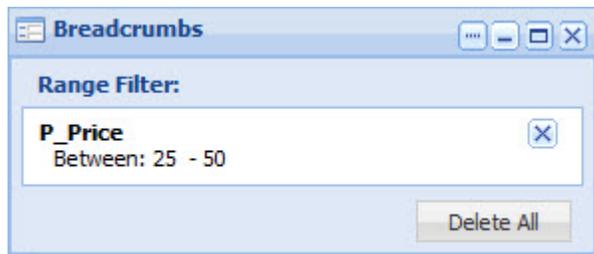
## Using Breadcrumbs

This topic describes how an end user can use the **Breadcrumbs** component.

The following example is based on **Guided Navigation** selections. Note the hierarchical selection in the Wine Type dimension.



The following example is based on a **Range Filter** selection.



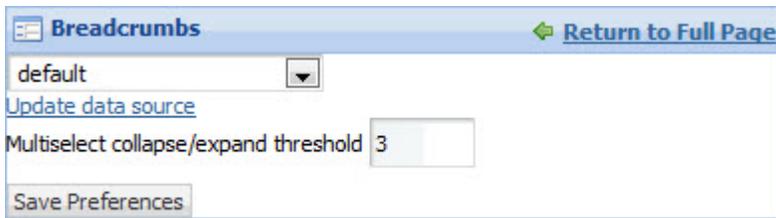
If a breadcrumb is based on a negative refinement (as enabled in the **Guided Navigation** component), that refinement is flagged with a red X icon. In the following **Guided Navigation** example, the refinement excludes Rose from the results.



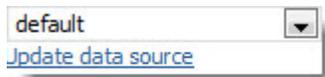
End users may remove individual selections or all selections within displayed breadcrumbs, and can also select a specific ancestor within the hierarchy for display.

## Configuring Breadcrumbs

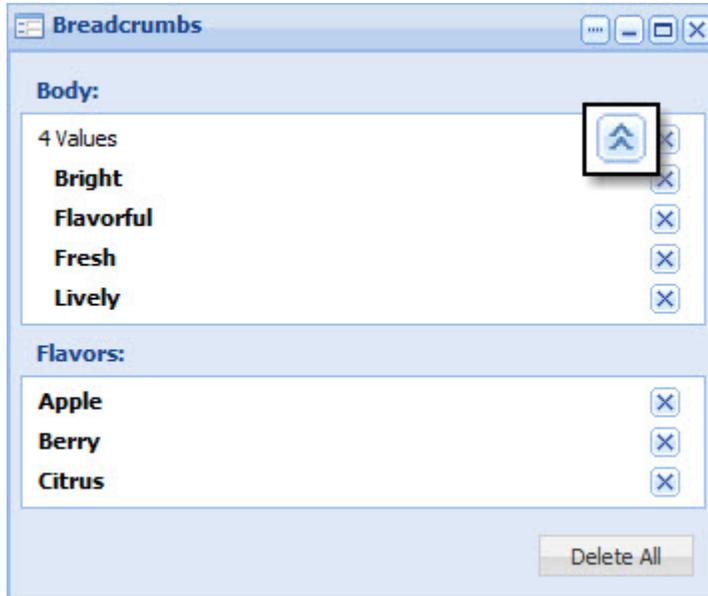
The power user can set the following preferences for the **Breadcrumbs** component.



- **Data source controls:** Select from the available data sources in the drop-down list, and then click **Update data source** to bind your selection to the component.



- **Multiselect collapse/expand threshold:** Set the number of dimension values after which the list can be collapsed. In the example below, dimension values for the Body dimension can be collapsed, because there are more than three selected. The dimension values for Flavors, on the other hand, cannot be collapsed, because there are only three selected.



## Guided Navigation

The **Guided Navigation** component provides Endeca Guided Navigation® functionality.



The **Guided Navigation** component provides contextual navigation across large volumes of structured MDEX Engine-specific data. End users can expand and close multiple dimension groups, view dimension values from multiple dimensions at a time, and select multi-OR and multi-AND dimensions. End users can also apply negative refinements to exclude dimension values from results if negative refinements are enabled by the power user.

The component shows the dimension groups that are configured in the MDEX Engine. It also displays base refinement statistics if they are enabled in the MDEX Engine.



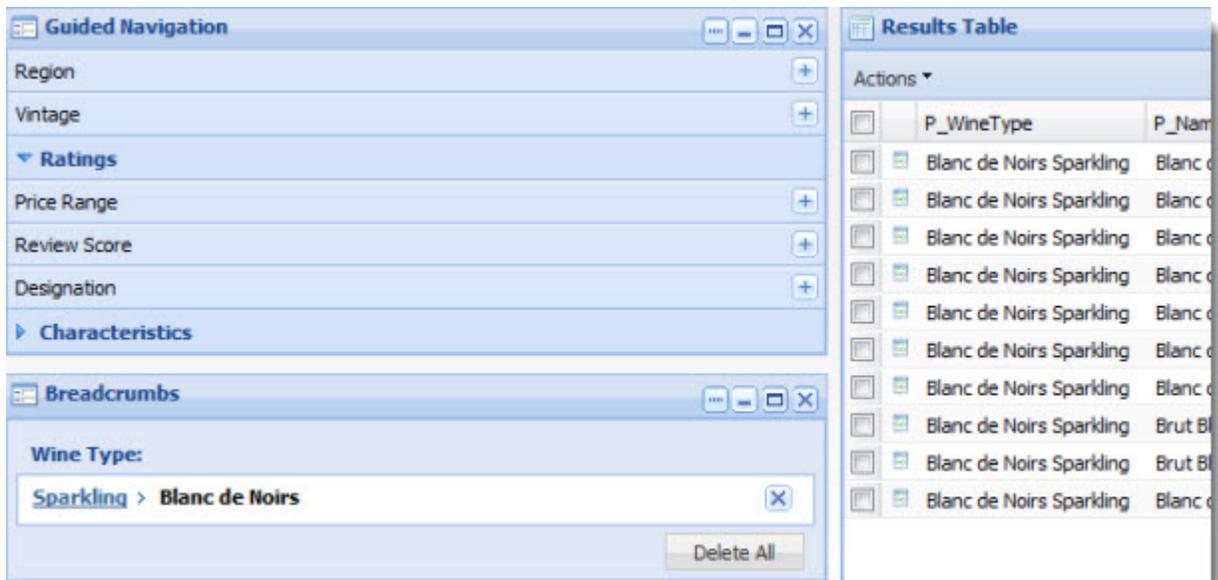
**Note:** Contracted or implicit dimension values and dead ends are not shown in this component.

The **Guided Navigation** component does not support multi-select for hierarchical dimensions, nor does it support tree view controls. Also, because aggregated records are not supported anywhere in the Discovery Framework, aggregated statistics are not supported in this component.

## Dependencies

The **Breadcrumbs** component is required in order to view or clear **Guided Navigation** component selections.

The image below shows the interaction between the **Guided Navigation**, **Breadcrumbs**, and **Results Table** components. The navigation state, which is displayed in the **Breadcrumbs** component, determines what is displayed in the **Results Table** component. If the user navigated to a different state in the **Guided Navigation** component, the other two components would dynamically update to reflect the change.



**Note:** The **Guided Navigation** component supports the use of aggregated records. Specifically, when the component is using a data source with a rollup key specified, its refinements display the number of aggregated records. For details about configuring Discovery Framework data sources for aggregated records, see the *Discovery Framework Installation Guide*.



**Note:** In order for the **Guided Navigation** component to generate aggregated record refinement statistics, the `--stat-abins` flag must be enabled on the MDEX Engine.



**Note:** The **Guided Navigation** component includes a threshold on the maximum number of dimension values that can be displayed at once within a dimension. The default value is 500. This threshold is in place for performance and usability reasons, so that the end user's experience can be managed for cases where dimensions include very long lists of dimension values. This threshold can be used in conjunction with dynamic ranking and type-ahead search in the **Guided Navigation** interface, so users can still access specific values by filtering for them.

## Related Links

[Breadcrumbs](#) on page 61

The **Breadcrumbs** component provides breadcrumb navigation aid functionality.

[Results Table](#) on page 52

The **Results Table** component provides a simple interface for displaying results. The **Results Table** component can show results from Analytics and non-Analytics queries.

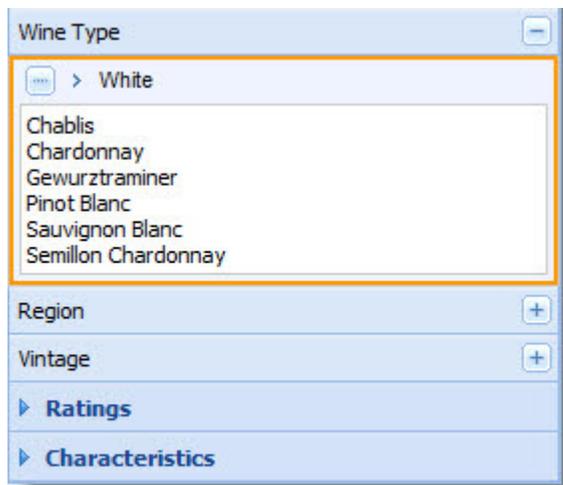
## Using Guided Navigation

This topic describes how an end user can use the **Guided Navigation** component.

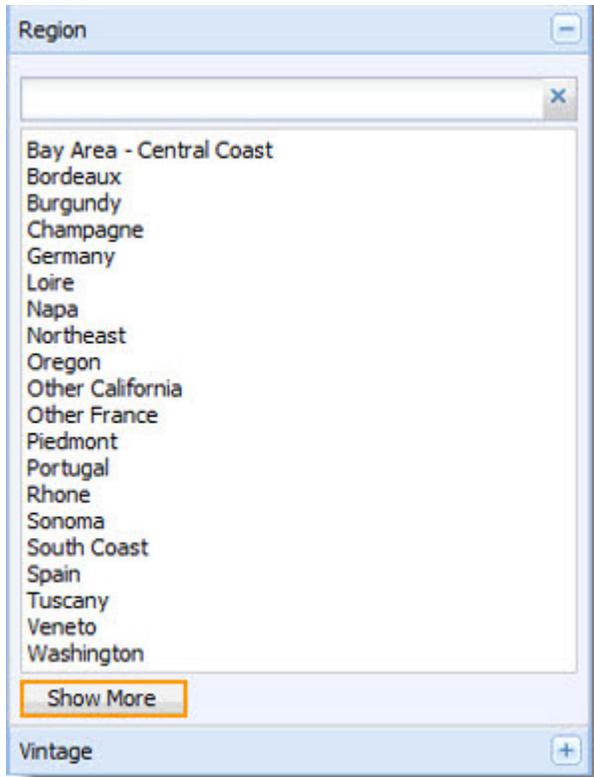


**Note:** The appearance of the Guided Navigation display varies depending on the dimension data and hierarchy.

The **Guided Navigation** component can feature dimension hierarchy:



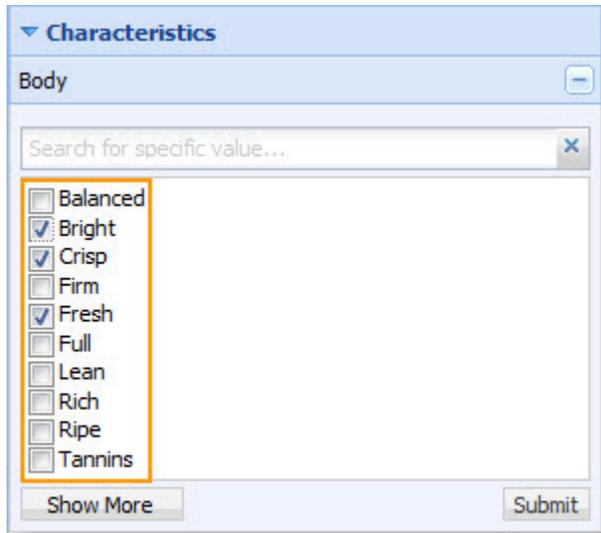
It features **Show More** functionality for large data sets:



It can include type-ahead:



It can incorporate multi-select OR dimensions through a check-box interface:



It can include negative as well as positive refinements, which allow you to exclude refinements from the results displayed.



## Configuring Guided Navigation

The power user can set the following preferences for the **Guided Navigation** component.

The edit controls on the **Guided Navigation** component allow you to configure the layout so only facets relevant to the end user are exposed.

default

[Update data source](#)

Disable type-ahead:

Maximum type-ahead suggestions: 10

Maximum refinements to show in a single dimension: 100

Dimension Display Name	Included <input checked="" type="checkbox"/>	Expanded <input type="checkbox"/>	Allow Negative <input checked="" type="checkbox"/>
ADG Refinement Center	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Body	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Designation	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Drinkability	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Flavors	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Price Range	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Region	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Targeted Page:

[Save Preferences](#) [Revert Changes](#)

- **Data source controls:** Select from the available data sources in the drop-down list, and then click **Update data source** to bind your selection to the component.

default

[Update data source](#)

- **Disable type-ahead:** Check to remove the **Type-ahead** search box from dimensions with many dimension values.

Disable type-ahead:



**Important:** Dimension search wildcarding must be enabled in the MDEX Engine in order for type-ahead to work in Discovery Framework components.

- **Maximum type-ahead suggestions:** Set the maximum total number of type-ahead suggestions the MDEX Engine will offer. If fewer results are available, the number returned will be less than this maximum. Note that this control is only visible if **Disable type-ahead** is not selected.

Maximum type-ahead suggestions: 10

- **Maximum refinements to show in a single dimension:** Set the maximum number of refinements per dimension to be displayed. This can help prevent your end users from being overwhelmed by an excessive number of results.
- **Dimension Configurations:** Includes the following settings:

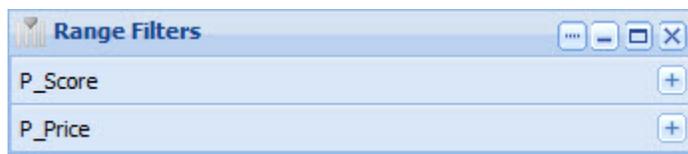
Dimension Display Name	Included <input checked="" type="checkbox"/>	Expanded <input type="checkbox"/>	Allow Negative <input checked="" type="checkbox"/>
Flavors	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Price Range	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Region	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Review Score	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Vintage	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Wine Type	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

- Search for a dimension by name.
  - Select dimensions to include. Dimensions not selected are hidden from the end user's view.
  - Select dimensions that will appear expanded by default in the end user's view. (The end user may choose to collapse them.)
  - Select dimensions that can be negative as well as positive refinements. A negative refinement, when selected, is excluded from results.
- **Targeted Page:** Set the target page for any page transition. Page transitions allow a component on one page in your Discovery Framework application to pass data to a component on another page.



## Range Filter

The **Range Filter** component allows you to add and modify range filters.

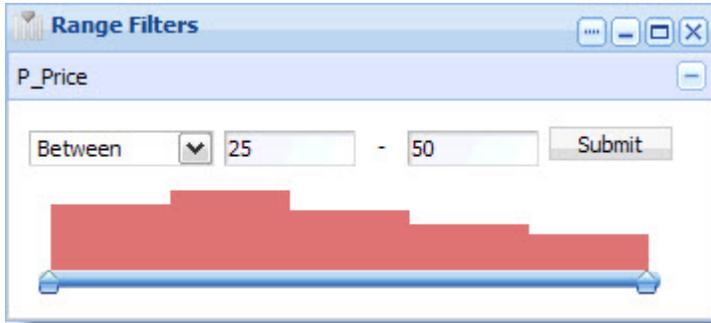


**Note:** The **Range Filter** component provides numeric and date filtering, but not geocode filtering. You cannot use the **Range Filter** component to filter on dimensions.

## Using Range Filter

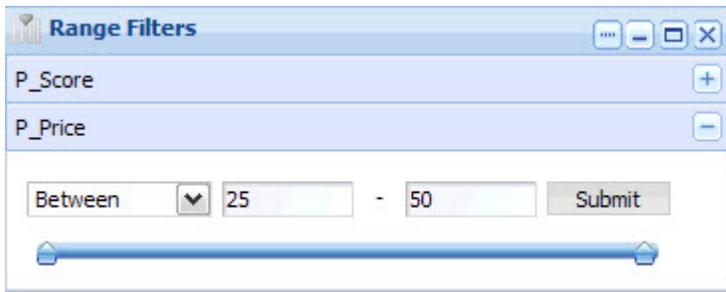
This topic describes how an end user can use the **Range Filter** component.

If the power user has configured a range filter for a property, the end user sees something like this:

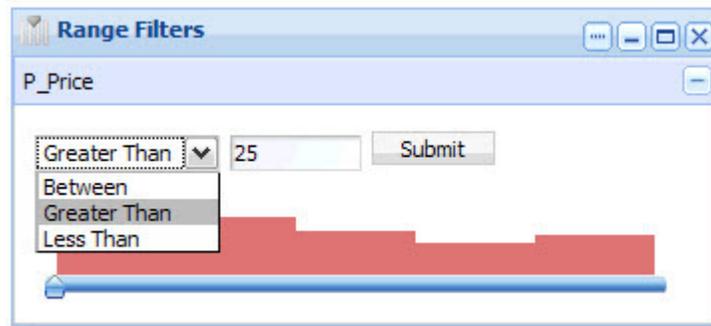


The histogram gives the end user a high-level view of the data. The end user can use the slider bar to refine the filter within the defined range.

It is also possible for the power user to configure the range filter to display the slider bar with no histogram:



The end user can also select the **Greater Than** and **Less Than** filters from the drop down list:



## Configuring Range Filter

The power user can add and remove range filters on existing properties in the data set in the **Range Filters** component.

default [Update data source](#)

**Range Filters**

+ Add Range Filter - Remove Range Filter

Property Name ▲	Min	Max	Type	<input type="checkbox"/> Enable Histogram	# Bin
P_Score	60.00	50.00	Numeric	Yes	30
P_Price	25.00	50.00	Numeric	Yes	5

[Save Preferences](#)

- **Data source controls:** Select from the available data sources in the drop-down list, and then click **Update data source** to bind your selection to the component.

default [Update data source](#)

- **Range Filters:** Establish range filters for existing properties. When adding a range filter, you can set the following features:

**Range Filters**

+ Add Range Filter - Remove Range Filter

Property Name ▲	Min	Max	Type	<input type="checkbox"/> Enable Histogram	# Bin
P_Score			Numeric ▼	<input type="checkbox"/>	

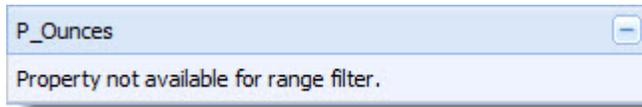
[Update](#) [Cancel](#)

- The name of the property this filter is based on.



**Note:** Property names are case sensitive. In addition, you need to know the exact name of the property for which you want to create a filter (you can consult the **Endeca Attribute Settings** component if you are unsure of a property's name). If you attempt to add a range filter based on a property that does not exist, when you return to the full page, you

will see the following error message:



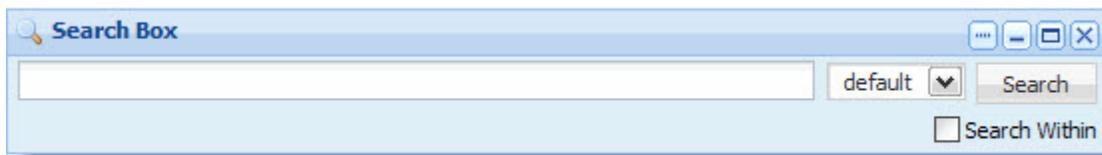
- Values for both **Min** and **Max** to provide a range, or a single value to allow **Greater Than** or **Less Than** filtering by end users.
- The type of range filter: Numeric or Date.
- **Enable Histogram** provides the end user with an interactive histogram control. If **Enable Histogram** is not selected, the end user sees a slider bar with no associated histogram.
- **# Bins** sets the number of bars to display in the histogram.

### Min and Max date-time format

**Min** and **Max** values must be given in milliseconds-since-epoch format. The **Range Filter** does not accept formatted date strings (such as "11/22/2009") or values in standard seconds-since-epoch format.

## Search Box

The **Search Box** component provides record and dimension search functionality.



It allows end users to submit keyword searches based on available properties in the data source he or she selects from the drop-down list.

Power users can choose from multiple search interfaces and match modes when configuring this component to further control the information that is displayed to the end user. The **Search Box** component can also be configured to provide type-ahead suggestions populated with matching dimension values.

### Dependencies

The **Search Box** component works in conjunction with the **Breadcrumbs** component, the **Results Table** component, and the **Guided Navigation** component.

### Related Links

[Breadcrumbs](#) on page 61

The **Breadcrumbs** component provides breadcrumb navigation aid functionality.

[Results Table](#) on page 52

The **Results Table** component provides a simple interface for displaying results. The **Results Table** component can show results from Analytics and non-Analytics queries.

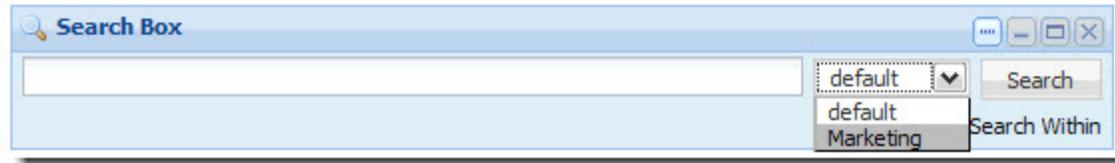
[Guided Navigation](#) on page 64

The **Guided Navigation** component provides Endeca Guided Navigation® functionality.

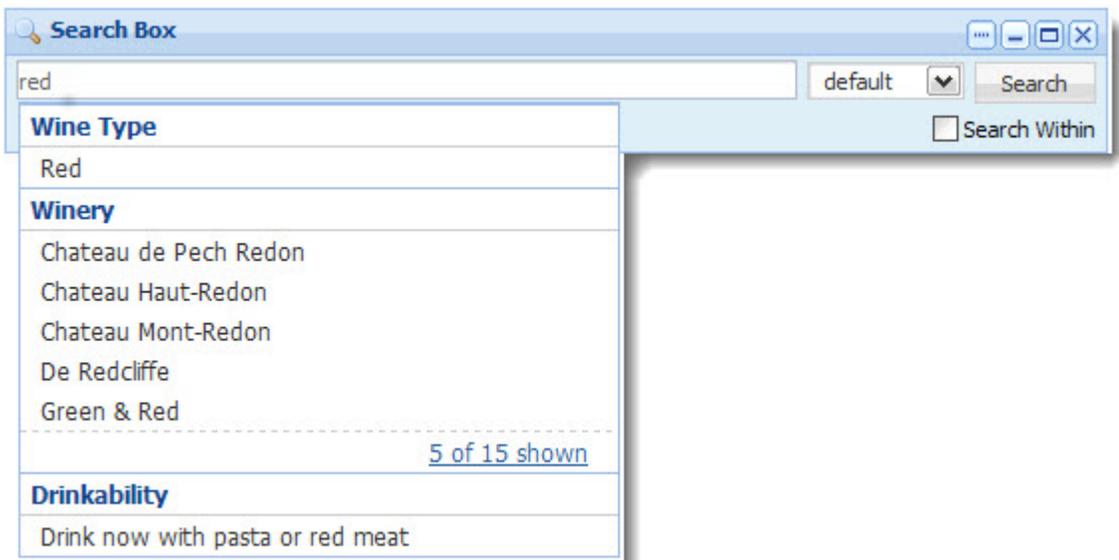
## Using Search Box

This topic describes how an end user can use the **Search Box** component.

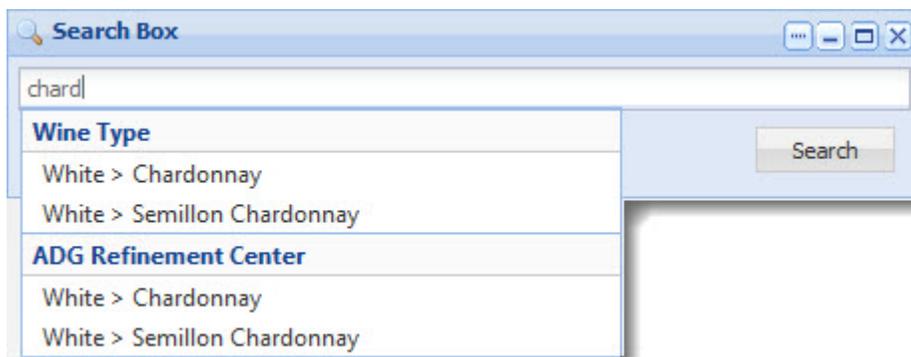
The end user can select a search configuration (based on a data source and a set of other criteria) from a list that the power user enables.



The end user can then enter a search term. In the example below, the end user typed the term *red*. The Wine Type, Winery, and Drinkability dimensions, along with matching dimension values, are made available as suggestions.



If the power user has enabled type ahead, the MDEX Engine will start offering dimension value suggestions before the user has typed a complete search term.

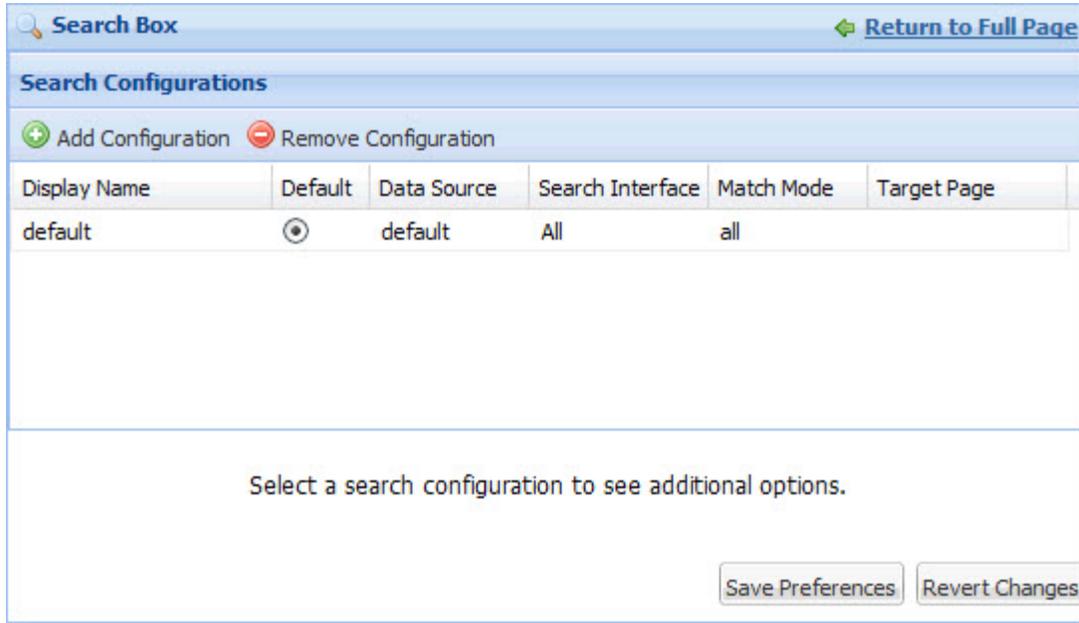


To restrict the results displayed to those within the current navigation state, the end user can check **Search Within**.

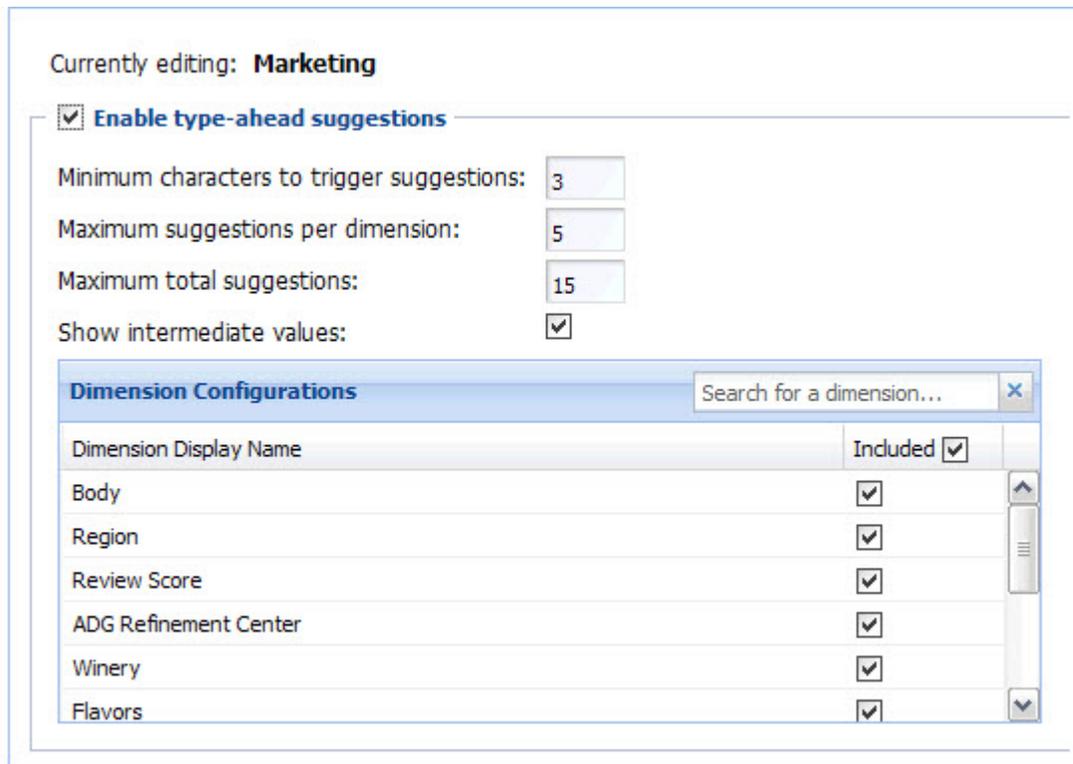
## Configuring Search Box

The power user can set the following preferences for the **Search Box** component.

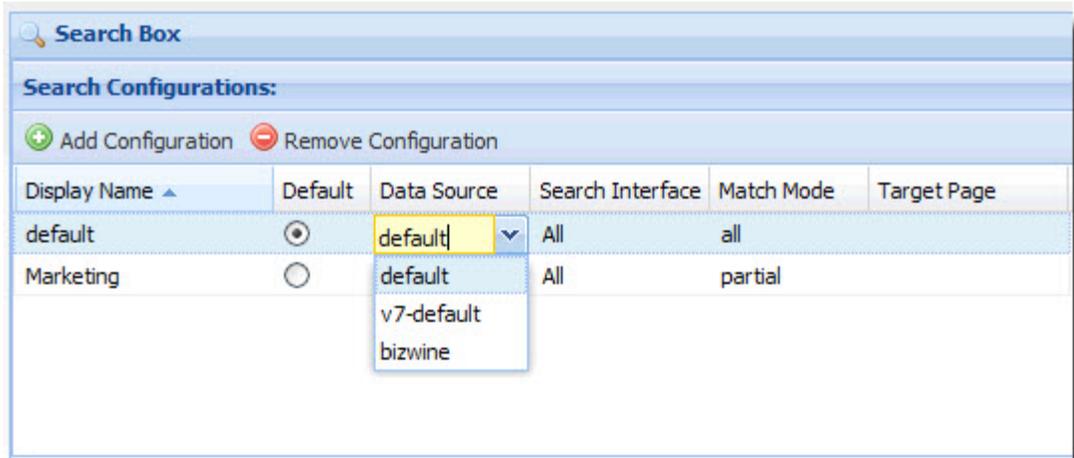
When you first open **Search Box** preferences, you see a list of existing search configurations:



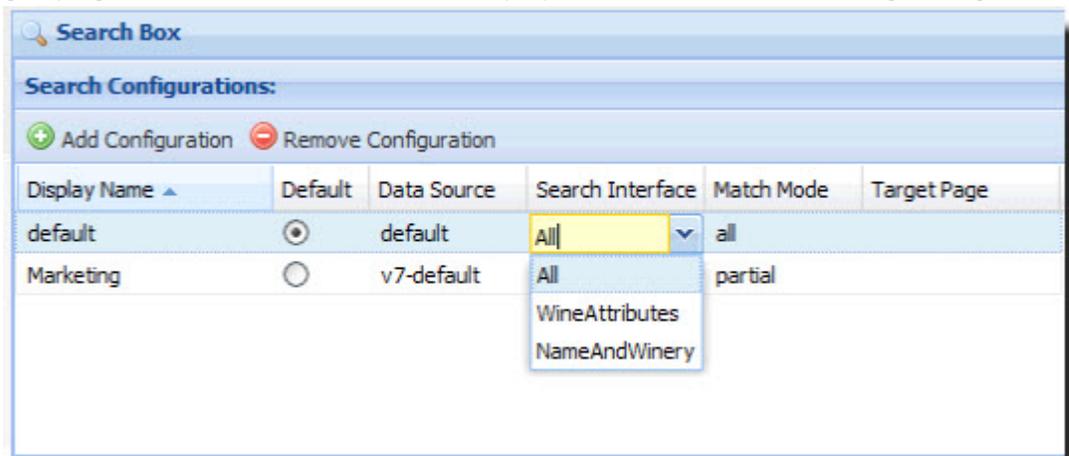
When you select a search configuration, or click **Add Configuration**, **Search Box** preferences updates to include options for type-ahead configuration:



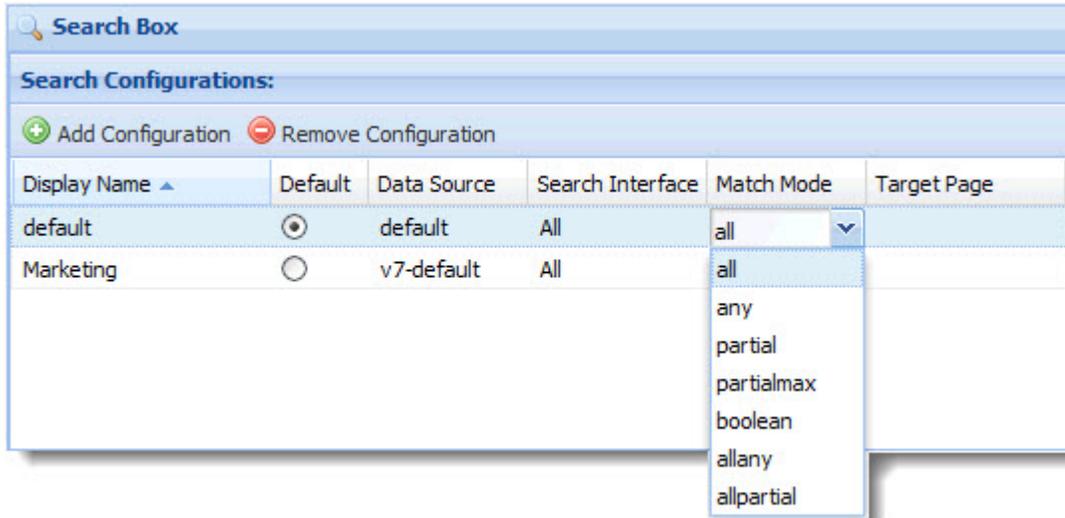
- **Search Configurations:** You can add, remove, or edit configurations that are based on the following criteria:
  - **Display Name:** The name used in breadcrumbs.
  - **Default:** Choose which search configuration is the default.
  - **Data Source:** Select one of the available data sources to associate with this search configuration.



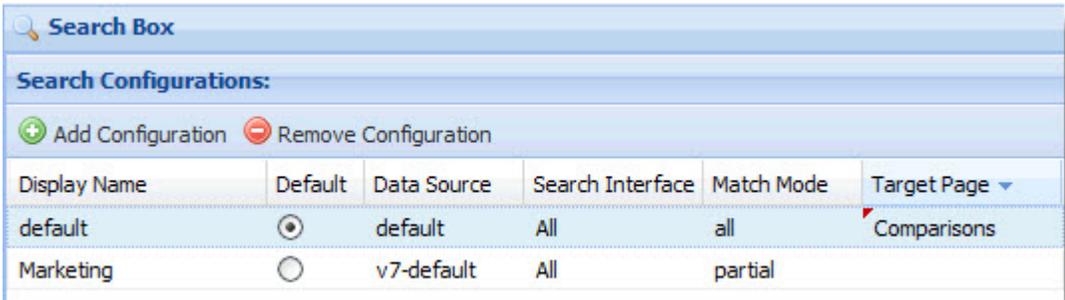
- **Search Interface:** Select from the search interfaces defined in this data source. The search interface limits the end user's search, and allows you to control record search behavior for groupings of some number of dimensions, properties, attributes, and ranking strategies.



- **Match mode:** Select a match mode from the drop-down list. Match modes allow you to specify a less restrictive matching strategy than the default conjunctive `MatchAll`.



- **Target Page:** Optionally, set the target page for any page transition. Page transitions allow a component on one page in your Discovery Framework application to pass data to a component on another page. In the example below, data will be sent to a component on the **Comparisons**



~

In addition, by clicking in the label of any column, you can sort the contents of that column and remove columns from the **Search Configurations** display.

- **Enable type-ahead suggestions:** Check to enable the MDEX Engine to start offering suggestions before the user has typed a complete search term. If **Enable type-ahead suggestions** is not selected, none of the remaining controls appear in the **Search Box** preferences.

**Enable type-ahead suggestions**

 **Important:** Dimension search wildcarding must be enabled in the MDEX Engine in order for type-ahead to work in Discovery Framework components.

- **Minimum characters to trigger suggestions:** Set the minimum number of characters needed before type-ahead suggestions are offered.

Minimum characters to trigger suggestions:

- **Maximum suggestions per dimension:** Set the maximum number of type-ahead suggestions the MDEX Engine will offer for each dimension.

Maximum suggestions per dimension:

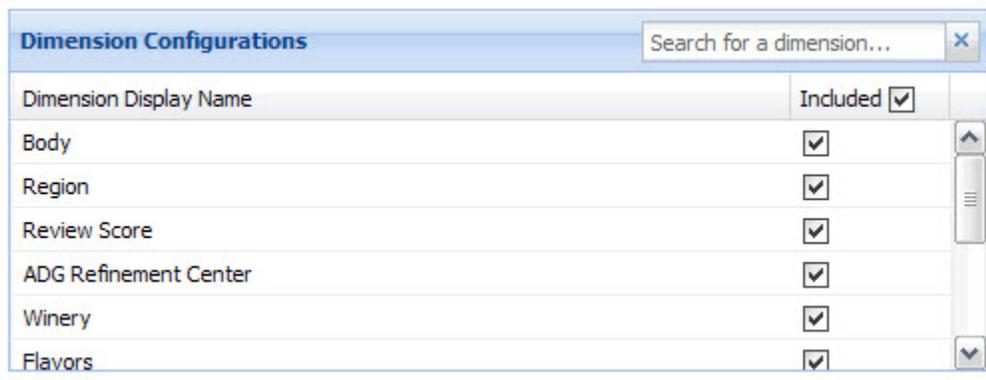
- **Maximum total suggestions:** Set the maximum total number of type-ahead suggestions the MDEX Engine will offer. If fewer results are available, the number returned will be less than this maximum.

Maximum total suggestions:

- **Show intermediate values:** Check this box to display the full path for hierarchical results.

Show intermediate values:

- **Dimension Configuration:** Check **Included** to include all dimensions in the type-ahead results. Alternatively, the power user can search for a dimension, or select dimensions individually, to customize which dimensions are included.



## Find Similar

The **Find Similar** component provides end users with the opportunity to find other records that share common attributes with a specific record of interest.



**Note:** In order to use the **Find Similar** component, the **Record Details** component must be configured.

Users can view specific records of interest in the **Record Details** component and then click **Find Similar** to search for records with similar attributes. The image below shows the relationship between the **Find Similar** and **Record Details** components.

**Record Details**

Property value: 90401 Find Similar

P_Body:	Full
P_Body:	Full-Bodied
P_Body:	Lingering
P_DateReviewed:	12/15/00
P_Designation:	Brut
P_Designation:	Reserve
P_Drinkability:	Drink now through 2003
P_Flavor:	Apple
P_Flavor:	Hazelnut
P_Flavor:	Lemon
P_Name:	Brut Champagne Royale Reserve
P_Price:	30.000000
P_Region:	Champagne

**Find Similar**

[Return to Record Details](#)

**Wine ID:** 90401

Please select...

- Wine Type > Sparkling > Brut
- Region > Champagne
- Winery > Philipponnat
- Vintage > Non - Vintage
- Price Range > \$30 to \$40
- Review Score > 90 to 100
- Body > Full
- Body > Full-Bodied



**Note:** The **Find Similar** component supports the use of aggregated records. For details about configuring Discovery Framework data sources for aggregated records, see the *Discovery Framework Installation Guide*.

## Dependencies

The **Find Similar** component works in conjunction with the **Record Details** component.

## Related Links

[Record Details](#) on page 85

The **Record Details** component displays all of the properties and dimensions for a record in the **Results Table** component.

## Using Find Similar

This topic describes how an end user uses the **Find Similar** component.



**Note:** The **Find Similar** component must be used in conjunction with the **Record Details** component.

Once an end user has selected a record to view in the **Record Details** component, he or she clicks **Find Similar** to populate the **Find Similar** component with the record's attributes. The user then selects which attributes to search by, as shown below:

The user can employ the search box within the **Find Similar** component to narrow the list of available attributes or to locate a specific attribute, as shown below:

When the user clicks **Submit**, the **Results Table** component displays all records with matching attributes:

<input type="checkbox"/>	P_WineType	P_WineID	P_Score	P_Region
<input type="checkbox"/>	Merlot Red	76877	82	Bay Area - Central Coast
<input type="checkbox"/>	Merlot Red	76878	89	Bay Area - Central Coast
<input type="checkbox"/>	Merlot Red	76879	83	Bay Area - Central Coast
<input type="checkbox"/>	Sauvignon Blanc White	79904	87	Bay Area - Central Coast
<input type="checkbox"/>	Merlot Red	76880	82	Bay Area - Central Coast
<input type="checkbox"/>	Sauvignon Blanc White	79956	88	Bay Area - Central Coast
<input type="checkbox"/>	Red Syrah	80689	87	Bay Area - Central Coast
<input type="checkbox"/>	Sauvignon Blanc White	79957	90	Bay Area - Central Coast
<input type="checkbox"/>	Chardonnay White	74160	85	Bay Area - Central Coast
<input type="checkbox"/>	Pinot Noir Red	78282	89	Bay Area - Central Coast

The **Breadcrumbs** component is also populated with the search dimensions. The user can further refine these results using the **Guided Navigation** and **Search Box** components.

## Configuring Find Similar

The power user can set the following preferences for the **Find Similar** component.

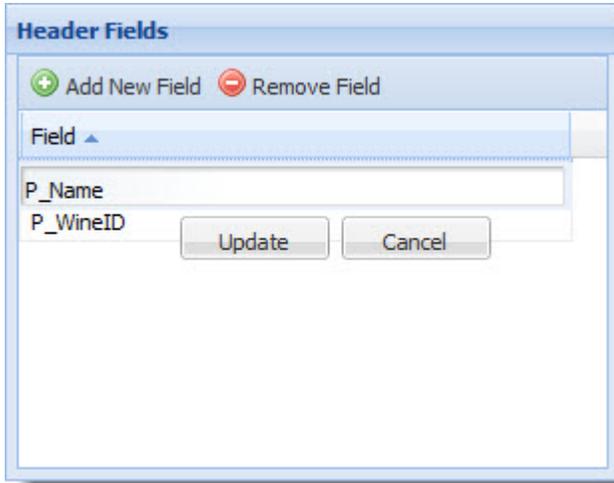
The power user can update the backing data source, add identifying headers, and customize the component's interface.

- **Data source controls:** Select from the available data sources in the drop-down list, and then click **Update data source** to bind your selection to the component.

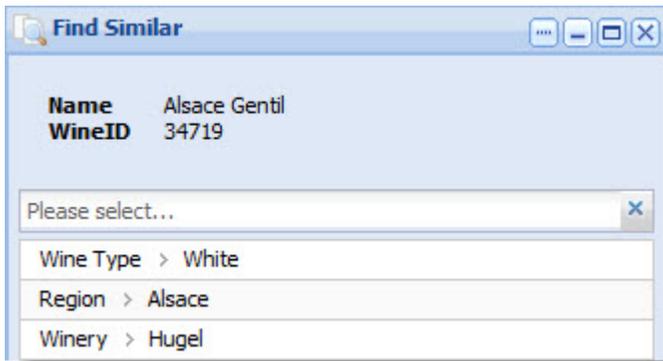
- **Header Fields:** Display property or dimension values that identify the starting record in the header of the component.



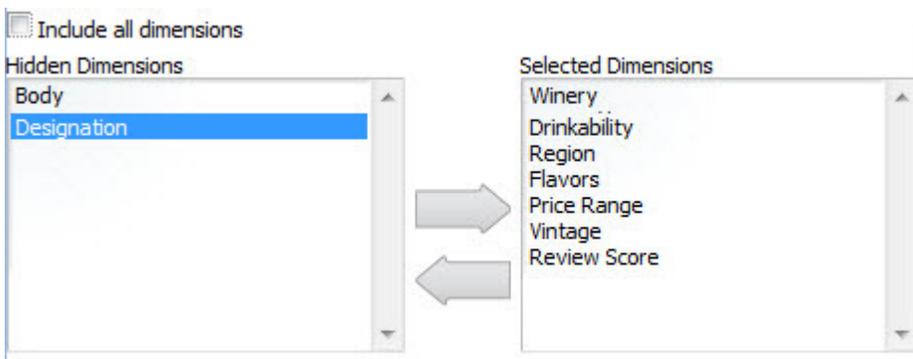
**Note:** To configure the display name of the property or dimension, you must use the **Attribute Settings** component.



The image below shows the **Find Similar** component with **Header Fields** configured:



- **Include all dimensions:** When this box is checked, all dimensions are included in the **Find Similar** component. When this box is unchecked, the **Hidden Dimensions** and **Selected Dimensions** lists display, and the user can select which dimensions to hide or include.



- **Calculate projected results:** Show how many similar results will be returned with the selected dimensions.



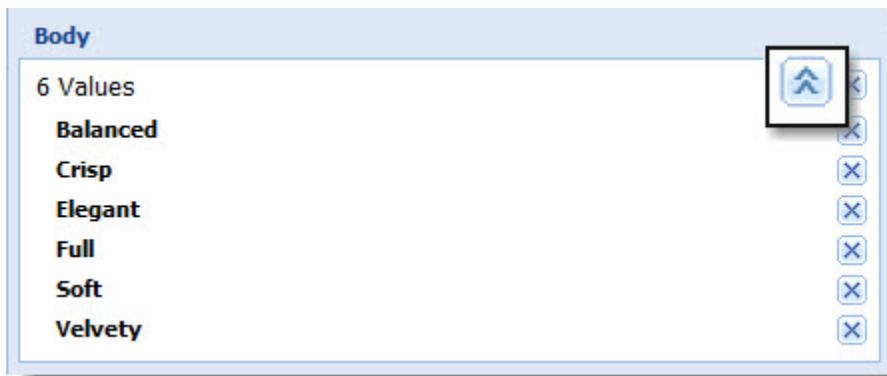
- **Show "Return to Record Details" link:** Clears the dimensions listed in the **Find Similar** component and returns the user to the **Record Details** component.



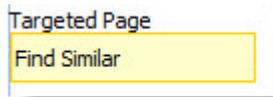
- **Multiselect collapse/expand threshold:** Sets the number of dimension values after which the list can be collapsed.



In the example below, six values are selected for the Body dimension. Because the threshold is set at 3, the user has the option to collapse the list (highlighted below).



- **Targeted Page:** Specify a separate page to display the **Find Similar** results.



**Note:** The target page must have a **Results Table** component.

## Record detail components

The components in this section allow you to focus on the details in your data.

### Related Links

[Record Details](#) on page 85

The **Record Details** component displays all of the properties and dimensions for a record in the **Results Table** component.

## Record Details

The **Record Details** component displays all of the properties and dimensions for a record in the **Results Table** component.

The **Record Details** component shows all properties first, followed by all dimensions, in the order they are returned from the Presentation API (that is, alphabetically). The user cannot change this order. However, the user can edit the render configuration part of the pipeline to control which properties and dimensions are displayed.

The example below shows the relationship between the **Results Table** and **Record Details** components. Selecting a record in the **Results Table** component and clicking the **View record detail** icon (highlighted below) populates the **Record Details** component with details about the selected record.

The screenshot displays two components from the Endeca interface. The top component is the **Results Table**, which shows a list of wine records. The selected record is **Pinot Gris White** with a score of 88 and a price of 15.000000. The bottom component is the **Record Details** view for the selected record, showing various properties and their values.

P_WineType	P_WineID	P_Score	P_Region	P_DateRevis	P_Price
Burgundy Macon White	52930	87	Burgundy	08/31/95	15.000000
Riesling White	69611	88	Australia	05/31/97	18.000000
Loire White	69217	90	Loire	06/30/99	45.000000
<b>Pinot Gris White</b>	<b>59401</b>	<b>88</b>	<b>Oregon</b>	<b>11/15/96</b>	<b>15.000000</b>
Albarino White	72176	89	Spain	11/15/98	13.000000
Other Italy DOC White	84280	91	Other Italy	01/31/00	30.000000

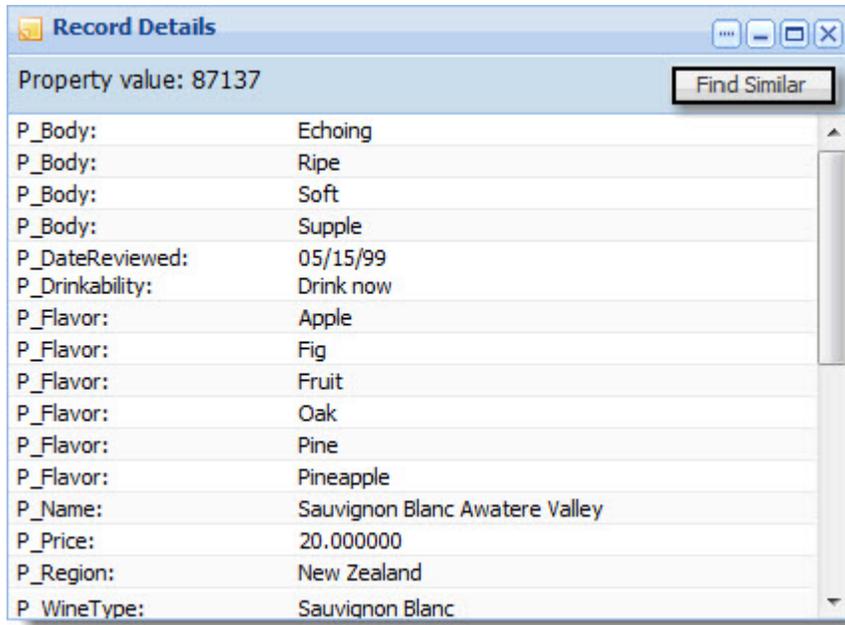
  

Property value: 59401		Find Similar
P_Body:	Ripe	
P_DateReviewed:	11/15/96	
P_Designation:	Rose	
P_Flavor:	Almond	
P_Flavor:	Melon	
P_Flavor:	Rose Petal	
P_Flavor:	Spice	
P_Name:	Pinot Gris Willamette Valley	
P_Price:	15.000000	

### Dependencies

The **Record Details** component must be used with the **Results Table** component, because only the **Results Table** provides the required detail links. However, as long as a page transition is specified, the two components do not have to reside on the same page.

The **Record Details** component also works in conjunction with the **Find Similar** component. When a record is displayed in **Record Details**, the user can click the **Find Similar** button (highlighted below) to search through records with similar attributes.



**Note:** The **Record Details** component supports the use of aggregated records. For details about configuring Discovery Framework data sources for aggregated records, see the *Discovery Framework Installation Guide*.

## Related Links

[Results Table](#) on page 52

The **Results Table** component provides a simple interface for displaying results. The **Results Table** component can show results from Analytics and non-Analytics queries.

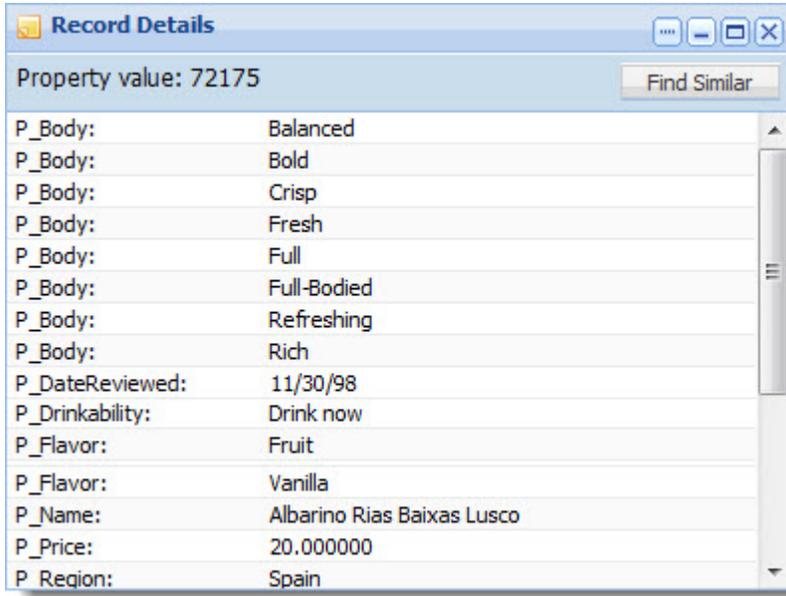
[Find Similar](#) on page 78

The **Find Similar** component provides end users with the opportunity to find other records that share common attributes with a specific record of interest.

## Using Record Details

This topic describes how an end user can use the **Record Details** component.

You can use the **Record Details** component to view all properties associated with the selected record.



The user can also use the **Record Details** component to find records with similar attributes to the one on display. When a record is displayed in the **Record Details** component, the user can click **Find Similar** to search through records with similar attributes to the record of interest. The relationship between the **Record Details** and **Find Similar** components is shown below:

The image shows two overlapping windows from a software application. The top window, titled "Record Details", displays a list of properties for a record with ID 72175. The properties include P\_Body (Balanced, Bold, Crisp, Fresh, Full, Full-Bodied, Refreshing, Rich), P\_DateReviewed (11/30/98), P\_Drinkability (Drink now), P\_Flavor (Fruit, Vanilla), and P\_... (partially visible). A "Find Similar" button is located in the top right corner of this window. The bottom window, titled "Find Similar", shows a search result for "Albarino Rias Baixas Lusco" from Spain, with Wine ID 72175. It includes a "Return to Record Details" link and a list of similar items with expandable details: Region > Spain, Winery > Lusco do Mino, Vintage > 1997, Price Range > \$20 to \$30, Review Score > 80 to 90, and Body > Balanced, Bold, Crisp, Fresh, Full.



**Note:** To make use of this functionality, the **Find Similar** component must be added to some page in the Discovery Framework.

## Configuring Record Details

The power user can set the following preferences for the **Record Details** component.

- **Data source controls:** Select from the available data sources in the drop-down list, and then click **Update data source** to bind your selection to the component.

- **Display Name Property:** Set the property to be used as the display name in the end user's view of the record.

- **Find Similar Target Page:** Specify a separate page to display the **Find Similar** component.



**Note:** The target page must have the **Find Similar** component added to it. It is also useful, but not necessary, to have a **Results Table** component added to the target page as well.





## Chapter 3

# About Liferay components

The Discovery Framework includes a number of Liferay components in its installation by default. These components can be integrated with Endeca components to allow you to build richer applications.

## Liferay component support

The level of documentation and support Endeca provides for the Liferay components included in the Discovery Framework differs from that provided for our own standard components.

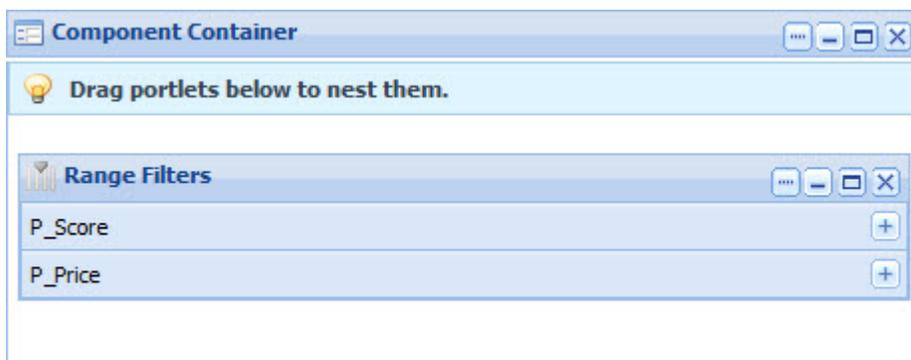
Because these components were not developed by Endeca, Endeca cannot control their interface or guarantee that they will be available in subsequent versions. Endeca provides only high-level documentation for Liferay components.

For more detailed information, consult the Liferay documentation and forums, available at <http://www.liferay.com>.

## The Component Container

The **Component Container** allows power users to organize components by grouping them together in a single container.

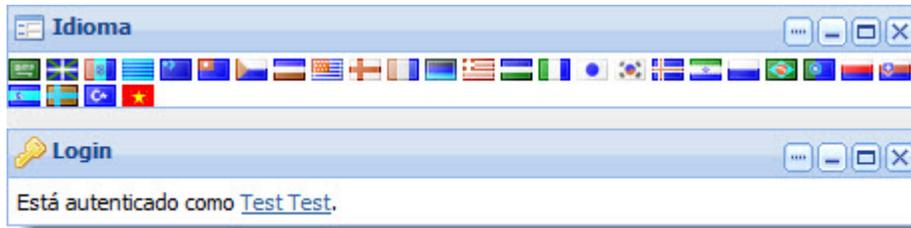
The power user can choose from several layout templates in order to customize the display to the needs of the application.



## The Languages component

The **Language** component can be used to change the locale of the server.

To select an alternate language in the **Language** component, click the flag icon associated with your target language. In the example below, we have chosen Spanish.



The Discovery Framework displays the component messages from your resource bundle in your target language. In addition, because the portal itself is also localized, menus and other portal controls also appear in your target language.



**Note:** For more information about localizing the Discovery Framework, see the *Discovery Framework Extension Guide*.

## About Liferay Web Content Management components

Liferay Web Content Management (WCM) components can be integrated into your application to provide document management and publishing capabilities.

These components are often used together to configure, display, and save links to Web content.

### Related Links

[The IFrame component](#) on page 93

The **IFrame** component provides an inline frame that can embed a separate Web document.

[The Links component](#) on page 94

The **Links** component allows users to save and manage their own Web content links in folders.

[The Web Content component](#) on page 94

The **Web Content** component allows the power user to configure and manage the administrative aspects of Liferay Web Content Management.

[The Web Content Display component](#) on page 95

The **Web Content Display** component allows you to request and display articles in your application.

[The Web Content List component](#) on page 95

The **Web Content List** component displays a list of all Web content articles that are available in the application.

[The Web Proxy component](#) on page 96

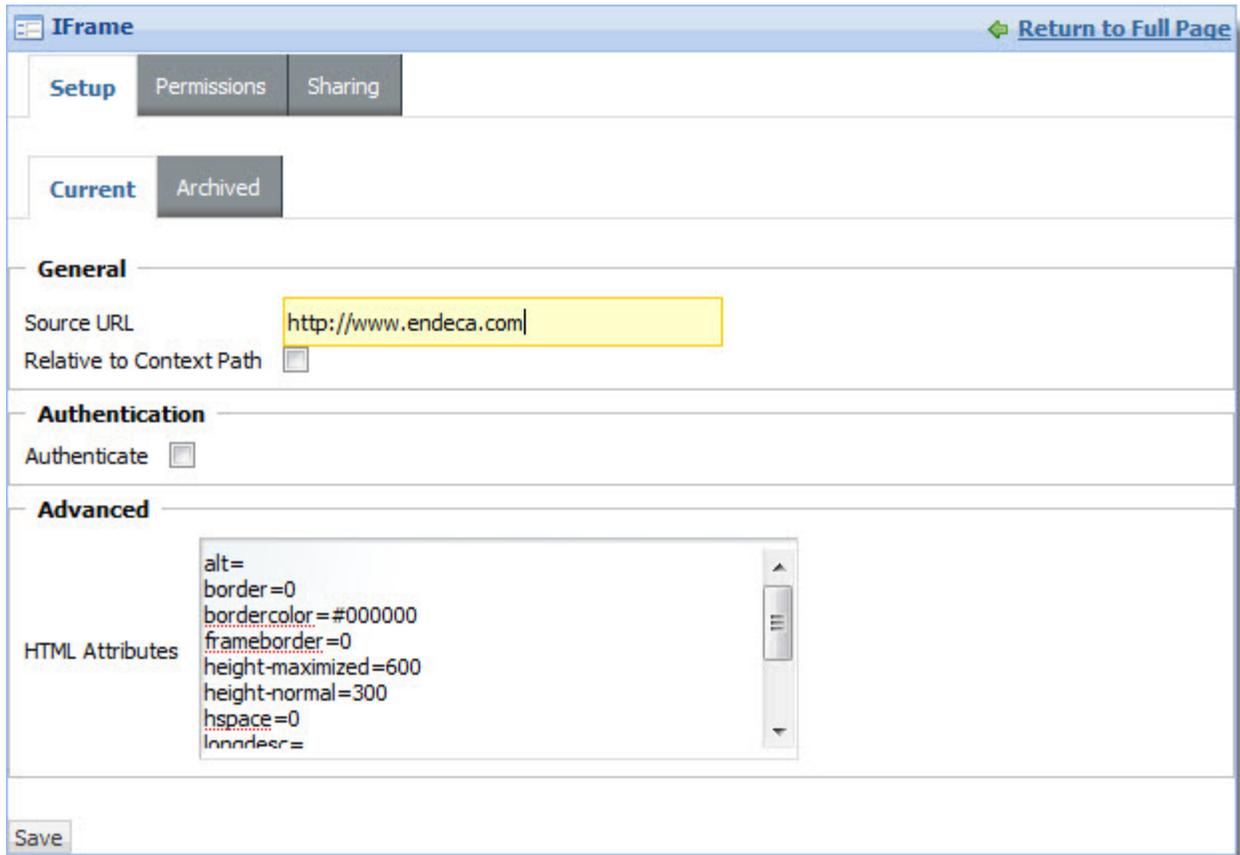
The **Web Proxy** component makes it possible to show any Web site as if it were a component.

## The IFrame component

The **IFrame** component provides an inline frame that can embed a separate Web document.

The power user can configure access and authentication, as well as modify HTML attributes that control the appearance of the output.

In the example below, the source URL is pointed at the Endeca website.

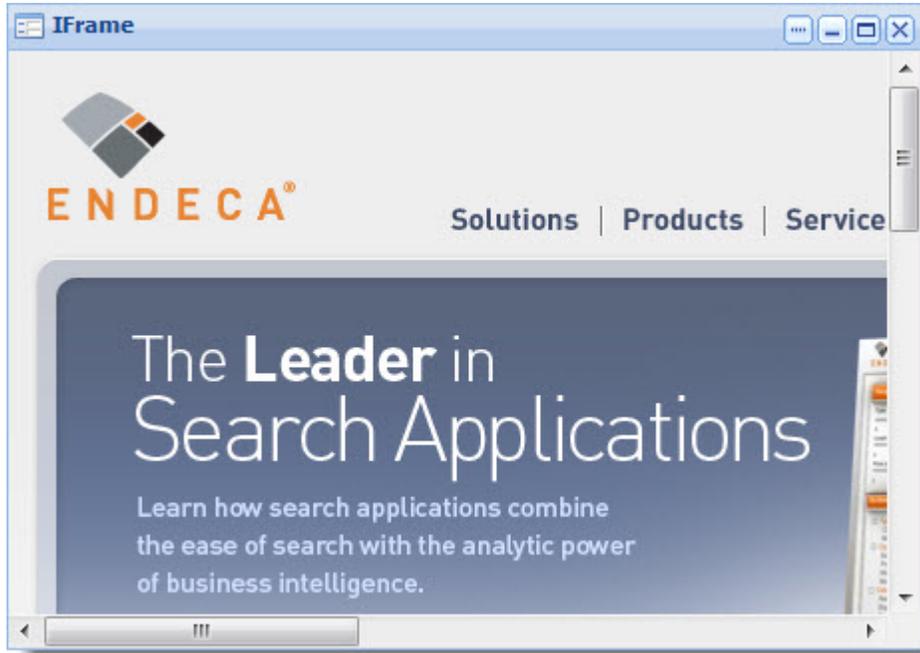


The screenshot shows the configuration interface for the IFrame component. At the top, there are tabs for 'Setup', 'Permissions', and 'Sharing', with 'Setup' selected. Below this are tabs for 'Current' and 'Archived', with 'Current' selected. The 'General' section contains a 'Source URL' field with the value 'http://www.endeca.com' and a 'Relative to Context Path' checkbox. The 'Authentication' section has an 'Authenticate' checkbox. The 'Advanced' section has an 'HTML Attributes' text area containing the following code: 

```
alt=  
border=0  
bordercolor=#000000  
frameborder=0  
height-maximized=600  
height-normal=300  
hspace=0  
innder=
```

 A 'Save' button is located at the bottom left of the interface.

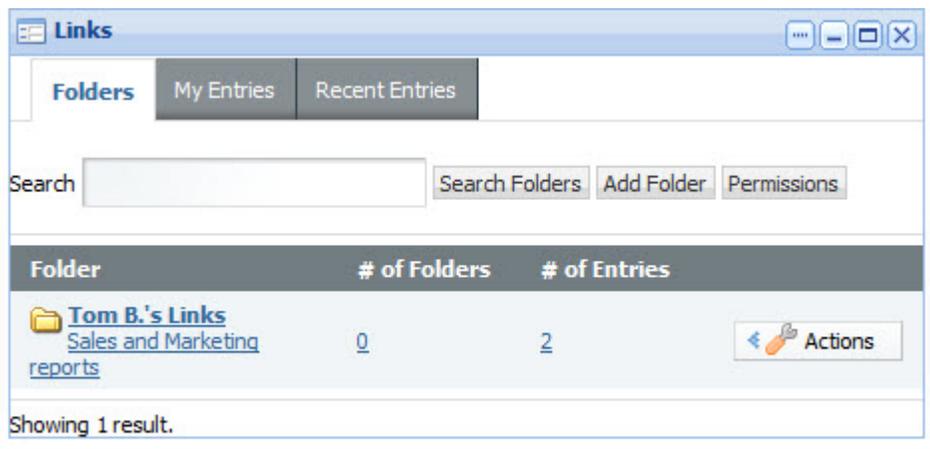
The end user sees the following:



 **Note:** Click [here](#) for Liferay documentation on this component.

## The Links component

The **Links** component allows users to save and manage their own Web content links in folders. These links can be tagged for later search and shared with other users.

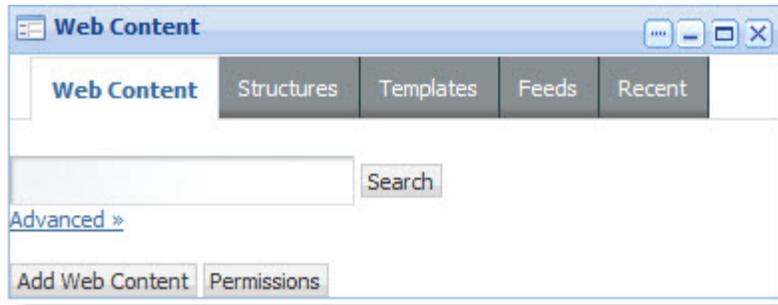


## The Web Content component

The **Web Content** component allows the power user to configure and manage the administrative aspects of Liferay Web Content Management.

This includes the following:

- Enabling users to write and publish articles to the site.
- Creating article templates.
- Controlling article-creation workflow and versioning.
- Managing article search and metadata.



The content is displayed in the **Web Content Display** component.



**Note:** Click [here](#) for Liferay documentation on this component.

## The Web Content Display component

The **Web Content Display** component allows you to request and display articles in your application.

The appearance of the articles can be controlled by WCM templates, as configured in **Web Content** component.



## The Web Content List component

The **Web Content List** component displays a list of all Web content articles that are available in the application.

The article list is automatically updated when new articles are made available, and can be sorted by various criteria.

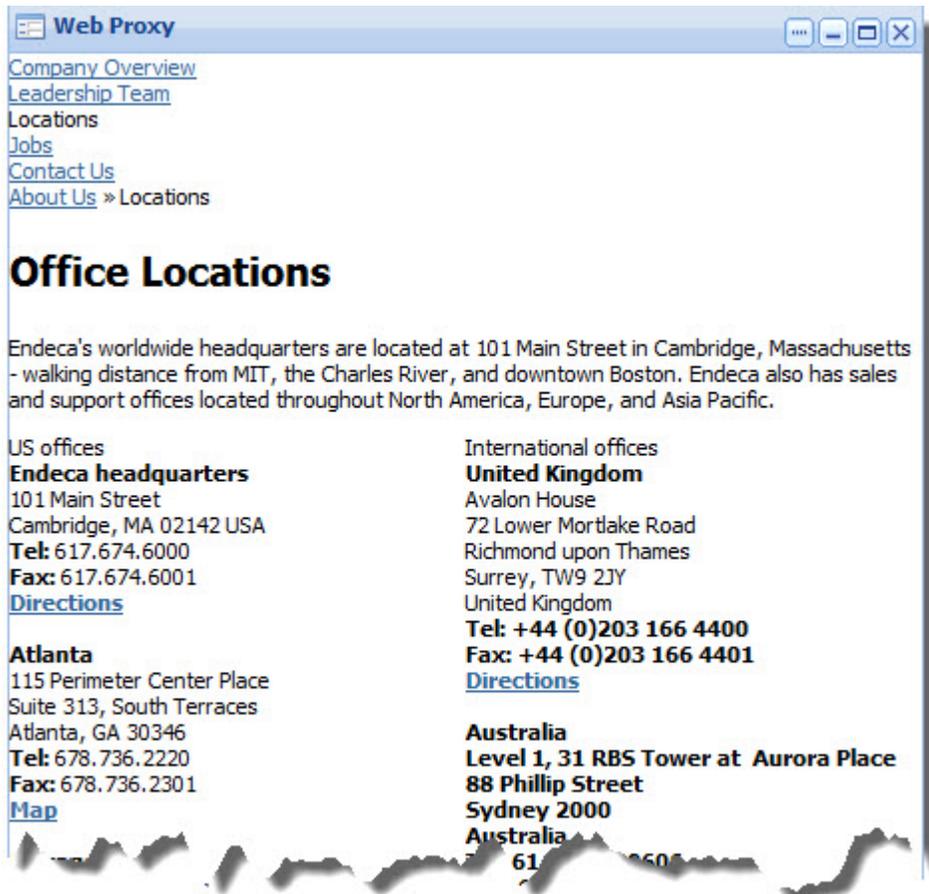


Name	Display Date	Author
<a href="#">The Supply and Demand Daily Update</a>	<a href="#">6/8/10 7:07 PM</a>	<a href="#">SD Inc</a>
<a href="#">ACME NEWS</a>	<a href="#">6/8/10 7:11 PM</a>	<a href="#">ACME</a>

Showing 2 results.

## The Web Proxy component

The **Web Proxy** component makes it possible to show any Web site as if it were a component. The power user can configure the appearance and authentication of the content.



[Company Overview](#)  
[Leadership Team](#)  
 Locations  
[Jobs](#)  
[Contact Us](#)  
[About Us](#) » Locations

## Office Locations

Endeca's worldwide headquarters are located at 101 Main Street in Cambridge, Massachusetts - walking distance from MIT, the Charles River, and downtown Boston. Endeca also has sales and support offices located throughout North America, Europe, and Asia Pacific.

<p>US offices</p> <p><b>Endeca headquarters</b>            101 Main Street            Cambridge, MA 02142 USA  <b>Tel:</b> 617.674.6000  <b>Fax:</b> 617.674.6001  <a href="#">Directions</a></p> <p><b>Atlanta</b>            115 Perimeter Center Place            Suite 313, South Terraces            Atlanta, GA 30346  <b>Tel:</b> 678.736.2220  <b>Fax:</b> 678.736.2301  <a href="#">Map</a></p>	<p>International offices</p> <p><b>United Kingdom</b>            Avalon House            72 Lower Mortlake Road            Richmond upon Thames            Surrey, TW9 2JY            United Kingdom  <b>Tel: +44 (0)203 166 4400</b>  <b>Fax: +44 (0)203 166 4401</b>  <a href="#">Directions</a></p> <p><b>Australia</b>  <b>Level 1, 31 RBS Tower at Aurora Place</b>  <b>88 Phillip Street</b>  <b>Sydney 2000</b>  <b>Australia</b>  <b>61 2 9500 2600</b></p>
--	---



## Chapter 4

# Component behavior with MDEX 7

---

This section describes the interaction of Endeca standard components with MDEX 7 Early Access edition. It lists the components that do not currently work with MDEX 7, and calls out any differences in behavior between Discovery Framework against MDEX 6 and MDEX 7.

## About the MDEX 7 Early Access release

This release of Discovery Framework includes limited capabilities for integration of Discovery Framework with MDEX Engine Release 7.

These features are in an Early Access state. The interfaces and behavior of these Early Access features may change in later releases of MDEX 7 and of the Discovery Framework, based on information gathered during the Early Access period. These capabilities are not supported for use in production.



**Important:** These capabilities are only available to users who have access to MDEX 7 Early Access edition. If you do not have access to MDEX 7 Early Access edition, you can safely ignore this section.

## Discovery Framework interaction with MDEX 7

Keep the following guidelines in mind when using the Discovery Framework 1.3.x with MDEX 7 Early Access edition.

- Performance is not expected to be optimal, as MDEX 7 is not optimized.
- Stability is not guaranteed, as MDEX 7 is not guaranteed to be stable.
- Dimension and property names must be NCName-compliant.
- All dimension and property names display their raw NCNames, which may not be user-friendly.

### Component interaction with MDEX 7

This table lists MDEX 7 interaction on a component-by-component basis.

Component	MDEX 7 Behavior Notes
<b>Advanced Visualization</b>	Chart legends and axes contain full dimension value specs instead of dimension value names, because Analytics result records return dimension value specs instead of dimension value names.
<b>Attribute Settings</b>	Display names do not work.
<b>Bookmarks</b>	The behavior of this component is the same on MDEX 6 and MDEX 7.
<b>Breadcrumbs</b>	In MDEX 7, the search configuration display name is not displayed. Instead, the user sees <b>search interface match mode</b> .
<b>Chart</b>	Chart legends and axes contain full dimension value specs instead of dimension value names, because Analytics result records return dimension value specs instead of dimension value names.
<b>Data Source Bindings</b>	The behavior of this component is the same on MDEX 6 and MDEX 7.
<b>Data Sources</b>	The behavior of this component is the same on MDEX 6 and MDEX 7.
<b>Discovery Framework Settings</b>	The behavior of this component is the same on MDEX 6 and MDEX 7.
<b>Find Similar</b>	This component is not supported on MDEX 7, because MDEX 7 does not support record detail queries.
<b>Guided Navigation</b>	<ul style="list-style-type: none"> <li>• Dimensions (and navigable properties) are not grouped. There are no groupings, because MDEX 7 does not support dimension groups.</li> <li>• Users cannot navigate to or past an inert dimension value, because MDEX 7 does not support inert dimensions.</li> <li>• Although multi-select works, it does so using a different mechanism than that used in MDEX 6. MDEX 7 does not support the ingest of multi-select dimensions.</li> <li>• Users cannot properly refine by more than one value of a multi-select dimension in type-ahead search results. MDEX 7 dimension search results do not indicate whether or not a matching dimension value is multi-select.</li> <li>• Power users see all properties in the component preferences view (excluding system properties) if navigable properties are enabled. Power users must edit component preferences to hide undesired properties. In addition, power users have the option to display only dimensions.</li> <li>• By default, when you open the <b>Guided Navigation</b> component, it displays only dimensions. The display of navigable properties (a feature of MDEX 7) can be enabled in the component's preferences by checking <b>Show navigable properties</b>. The <b>Show navigable properties</b> checkbox only appears when the <b>Guided Navigation</b> component is backed by an MDEX 7 data source.</li> <li>• <b>Maximum refinements to show in a dimension</b> is limited to 100 dimension values, due to differing dynamic ranking behavior in MDEX 7.</li> <li>• The number of results from type-ahead is not shown, because in MDEX 7, dimension search does not return the total number of results.</li> </ul>
<b>Metrics Bar</b>	This component is not yet supported on MDEX 7.

Component	MDEX 7 Behavior Notes
<b>Performance Metrics</b>	The behavior of this component is the same on MDEX 6 and MDEX 7.
<b>Range Filters</b>	This component is not supported on MDEX 7, because range filters are not yet supported.
<b>Record Details</b>	This component is not supported on MDEX 7, because record details are not yet supported.
<b>Results Table</b>	<ul style="list-style-type: none"> <li>• Detail view does not work, because MDEX 7 does not yet support record detail queries.</li> <li>• <b>Export to Excel</b> does not work, because MDEX 7 does not support bulk export.</li> <li>• When the <b>Results Table</b> component is powered by an Analytics query, dimension value specs instead of dimension value names are shown in columns, because Analytics result records return dimension value specs instead of dimension value names.</li> </ul>
<b>Sample Endeca Portlet</b>	The aggregate records drop-down option is not displayed, because MDEX 7 does not support aggregate records.
<b>Search Box</b>	<ul style="list-style-type: none"> <li>• In type-ahead search, intermediate values are not shown. For example, if you search for <b>Merlot</b> it is impossible to display <b>Red &gt; Merlot</b>. Because there is no way to retrieve ancestor values from a dimension search result, the <b>Show intermediate values</b> checkbox is not available in the preferences editor when an MDEX 7 data source is selected.</li> <li>• If multi-select dimension values are already in the navigation state and a type-ahead search result is clicked for that dimension, the existing values are replaced. MDEX 7 dimension search results do not indicate whether a matching dimension value is multi-select.</li> <li>• In preferences, all searchable properties are shown, as well as search interfaces. You cannot distinguish search interfaces from searchable properties.</li> <li>• The number of results from type-ahead is not shown, because in MDEX 7, dimension search does not return the total number of results.</li> </ul>



# Index

## A

### about

- Advanced Visualization component 33
- Bookmarks 29
- Breadcrumbs 61
- Chart 36
- Cross Tab 47
- Data Sources component 21
- Find Similar 78
- Framework Settings 23
- Guided Navigation 64
- Metrics Bar 57
- Performance Metrics 25
- Range Filters 70
- Record Details 85
- Results Table 52
- Sample Endeca Portlet component 26
- Search Box 73
- Tabbed Component Container 27

### about this guide 11

### additional documentation 15

### Advanced Visualization component

- about 33
- editing 35
- using 33
- what is saved in a bookmark 31

### Analytics statements

- supported by the Chart component 36

### appearance

- changing a component's 11

## B

### Bookmarks

- editing 33

### Bookmarks component

- about 29
- using 30
- what is saved for each standard component 31
- what is saved in a bookmark 32

### Breadcrumbs component 61

- editing 63
- using 62
- what is saved in a bookmark 32

## C

### case sensitivity

- in the Discovery Framework 11

### characteristics

- of Liferay components 91

### chart

- horizontal bar chart type 43
- line bar combo chart type 46
- line chart type 44

### Chart

- pie chart type 45
- vertical bar chart type 41

### Chart component

- about 36
- Analytics statements 36
- editing 39
- provided chart styles 41
- using 36
- what is saved in a bookmark 32

### Component Container 91

### component IDs

- with view transitions 14

### component state

- what is saved in Bookmarks 31

### component users 11

### component, changing appearance 11

### components

- about 9
- Data Source Bindings 23

### configuring 48

- Find Similar 81
- Metrics Bar 58

### Cross Tab component

- about 47
- configuring 48
- using 48

## D

### Data Source Bindings component 23

### Data Sources component

- about 21
- using 21

### different users for the Discovery Framework 11

### dimension search

- enabling wildcard for 11

### Discovery Framework

- case sensitivity in 11
- obtaining more information 15

### Discovery Framework, performance considerations 15

## E

### editing

- Advanced Visualization component 35
- Bookmarks 33
- Breadcrumbs component 63
- Chart 39

editing (*continued*)

- Framework Settings 23
- Guided Navigation 68
- Performance Metrics 26
- Range Filters 72
- Record Details 89
- Results Table 54
- Sample Endeca Portlet component 27
- Search Box component 75
- Tabbed Component Container 28

Endeca Attribute Settings

- about 17
- editing 19
- using 18

Endeca standard components

- about
- example
  - stacked vertical bar chart 42

## F

Find Similar component

- about 78
- configuring 81
- using 80
- what is saved in a bookmark 32

Framework Settings

- editing 23
- using 23

Framework Settings component 23

## G

Guided Navigation component

- about 64
- editing 68
- using 66
- what is saved in a bookmark 32

## H

horizontal bar chart example 43

## I

IFrame component 93

instanceable components 10

## L

Languages component 92

Liferay components

- about 91
- Component Container 91
- IFrame 93
- Languages 92
- Links 94
- Web Content 95

Liferay components (*continued*)

- Web Content Display 95
- Web Content List 95
- Web Proxy 96

Liferay WCM 92

line bar combo chart example 46

line chart example 44

Links component 94

## M

MDEX 7 Early Access, about 97

MDEX 7 Early Access, Discovery Framework interaction with 97

Metrics Bar component

- configuring 58
- using 57
- what is saved in a bookmark 32

Metrics Bar component, about 57

more information about the Discovery Framework 15

## N

non-instanceable components 10

## P

page transitions 13

performance considerations 15

Performance Metrics

- about 25
- editing 26
- using 25

pie chart example 45

## R

Range Filter component

- about 70
- editing 72
- using 70
- what is saved in a bookmark 32

Record Details

- about 85

Record Details component

- editing 89
- using 86
- what is saved in a bookmark 32

Results Table component

- about 52
- editing 54
- using 53
- what is saved in a bookmark 32

## S

Sample Endeca Portlet component

- about 26

Sample Endeca Portlet component (*continued*)

  editing 27

  using 27

Search Box

  about 73

  dependencies 73

  editing 75

  using 74

Search Box component

  what is saved in a bookmark 32

stacked vertical bar chart type 42

## T

tab transitions 13

Tabbed Component Container

  about 27

  editing 28

  using 27

  what is saved in a bookmark 32

transition syntax 13

## U

user roles 11

using

  Advanced Visualization component 33

  Bookmarks component 30

  Breadcrumbs component 62

  Chart component 36

  Cross Tab 48

using (*continued*)

  Data Source components 21

  Find Similar 80

  Framework Settings 23

  Guided Navigation 66

  Metrics Bar 57

  Performance Metrics 25

  Range Filter 70

  Record Details 86

  Results Table 53

  Sample Endeca Portlet component 27

  Search Box component 74

  Tabbed Component Container 27

## V

vertical bar chart example 41

view transitions

  with component IDs 14

## W

Web Content component 95

Web Content Display component 95

Web Content List component 95

Web Content Management in Liferay

  about 92

Web Proxy component 96

what is a component 9

wildcard

  enabling for dimension search 11

