

Oracle® Endeca Information Discovery Studio

Studio Administration and Customization Guide

Version 3.2.0 • January 2016

Copyright and disclaimer

Copyright © 2003, 2017, Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners. UNIX is a registered trademark of The Open Group.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Table of Contents

Copyright and disclaimer	2
Preface	7
About this guide	7
Who should use this guide	7
Conventions used in this document	7
Contacting Oracle Customer Support	8
Part I: Configuring and Monitoring Studio	
Chapter 1: Configuring Framework Settings	10
About the Framework Settings page	10
Configuring framework settings from the Control Panel	10
Configuring framework settings in portal-ext.properties	14
Chapter 2: Configuring Logging for Studio	15
About logging in Studio	15
About the log4j configuration XML files	15
About the Studio log files	16
Using the Control Panel to adjust logging verbosity	16
Chapter 3: Monitoring the Performance of Queries	18
Configuring the amount of metrics data to record	18
About the metrics log file	19
About the Performance Metrics page	20
Chapter 4: Viewing Summary Reports of Studio Usage	23
Enabling the logging of Studio system usage	23
About the System Usage page	25
Viewing the Number of Users by Date report	25
Viewing the Most/Least Frequently Accessed Summary Report	26
Chapter 5: Determining and Configuring the Locale Used in Studio	29
About locales and their effect on the Studio user interface	29
How Studio determines the locale to use	30
Locations where the locale may be set	30
Scenarios for selecting the locale	31
Setting the available locales for Studio	32
Setting the default locale used by Studio	33
Configuring the preferred locale for a Studio user	33
Including the locale in a URL	35

Part II: Controlling Access to Studio

Chapter 6: About Managing Users in Studio	37
About user roles	37
About the default user	38
Chapter 7: Creating and Editing Users in Studio	39
Configuring the type of user name for Studio	39
Creating a new user	40
Editing a Studio user	41
Preventing a user from creating applications	42
Deactivating, reactivating, and deleting Studio users	42
Chapter 8: Integrating with an LDAP System to Manage Users	44
About using LDAP	44
Configuring the LDAP settings and server	44
Configuring the Studio password policy when using LDAP	49
Preventing encrypted LDAP passwords from being stored in Studio	50
Assigning roles based on LDAP user groups	51
Chapter 9: Setting up Single Sign-On (SSO) for Studio	53
About single sign-on and Studio	53
Overview of the process for configuring SSO with Oracle Access Manager	53
Configuring the reverse proxy module in OHS	54
Reverse proxy configuration for WebLogic Server	54
Registering the Webgate with the Oracle Access Manager server	55
Testing the OHS URL	57
Configuring Studio to integrate with SSO via Oracle Access Manager	57
Configuring the LDAP connection for SSO	57
Configuring the Oracle Access Manager SSO settings	59
Completing and testing the SSO integration	60

Part III: Configuring Available Data for Applications

Chapter 10: Managing Endeca Server Connections	63
About Endeca Server connections	63
Using the Endeca Servers page to manage Endeca Server connections	63
About the Endeca Servers page	64
Displaying the Endeca Servers page	64
Adding an Endeca Server connection	66
Editing an Endeca Server connection	67
Deleting an Endeca Server connection	68
Testing an Endeca Server connection	68
Syntax for an Endeca Server connection definition	68
Escaping special characters in Endeca Server connection definitions	69
Basic Endeca Server connection properties	69
Configuring role-based security for viewing Endeca Server connection data	71

Connecting an Endeca Server connection to a secured Endeca Server	72
Connecting an Endeca Server connection to an Endeca Server cluster	74
Limiting who can connect an application to the Endeca Server connection	75
Setting up shared Endeca Server connections	77
Chapter 11: Configuring the Connection to the Provisioning Service	78
Chapter 12: Managing Available Sources of Application Data	81
About data sources	81
Displaying information about data sources on the Data Source Library	82
Adding and editing data sources in the Data Source Library	83
Adding and editing data sources	83
Selecting the data to use in a data source	86
Configuring the attributes in a data source	88
Configuring access to a data source in the Data Source Library	94
Removing a data source from the Data Source Library	95
Part IV: Managing Studio Applications	
Chapter 13: Configuring and Removing Applications	97
Configuring the application type	97
Configuring the visibility type for a page	98
Adding and removing application members	99
Assigning application roles to application members	100
Certifying an application	101
Making an application active or inactive	102
Removing applications	103
Chapter 14: Exporting and Importing Studio Application Pages	104
About exporting and importing application pages	104
Exporting pages from Studio	104
What is included in the export?	105
Completing the export	105
Importing pages into Studio	106
Ensuring that imported pages will work properly	106
Completing the import	106
Part V: Customizing Studio	
Chapter 15: Changing the Look and Feel of Studio	110
About customizing the Studio look and feel	110
Location of the Studio CSS and images	110
Updating the Studio CSS and images for a WebLogic Server instance	111
Chapter 16: Using a Custom Security Manager	112
Security Manager class summary	112
Creating a new Security Manager	113

Implementing a new Security Manager	113
Deploying a new Security Manager	114
Configuring Studio to use your Security Manager	114
Chapter 17: Developing Custom Components	115
Software and licensing requirements for component development	115
Configuring the Studio SDK for component development	116
Configuring Eclipse for component development	117
Developing a new component	118
Creating a new component	118
Importing the component project into Eclipse	119
Obtaining query results for custom components	119
Building and testing your new component	120
Adding and removing components from the Studio .ear file for WebLogic Server	120
Modifying the Studio SDK build properties for a component	121
Chapter 18: Working with QueryFunction Classes	122
Provided QueryFunction filter classes	122
Provided QueryConfig functions	129
Creating a custom QueryFunction class	135
Implementing a custom QueryFunction class	136
Deploying a custom QueryFunction class	137
Adding the custom QueryFunction .jar file to the custom component Eclipse build path	137

Preface

Endeca Information Discovery Studio is an industry-leading application composition environment and discovery experience that allows business users to easily upload and mash up multiple diverse data sources, and then quickly configure discovery applications - all within the context of an enterprise framework that maintains existing governance and enterprise definitions.

Studio includes world-class search, guided navigation, and filtering, as well as offering an array of powerful interactive visualizations, for rapid intuitive analysis that requires zero training.

About this guide

This guide provides information on configuring, administering, and customizing Oracle Endeca Information Discovery Studio.

Who should use this guide

This guide is intended for administrators who need to configure and monitor Studio, and developers who want to extend and customize Studio.

Conventions used in this document

The following conventions are used in this document.

Typographic conventions

The following table describes the typographic conventions used in this document.

Typeface	Meaning
User Interface Elements	This formatting is used for graphical user interface elements such as pages, dialog boxes, buttons, and fields.
Code Sample	This formatting is used for sample code phrases within a paragraph.
<i>Variable</i>	This formatting is used for variable values. For variables within a code sample, the formatting is <i>Variable</i> .
File Path	This formatting is used for file names and paths.

Symbol conventions

The following table describes symbol conventions used in this document.

Symbol	Description	Example	Meaning
>	The right angle bracket, or greater-than sign, indicates menu item selections in a graphic user interface.	File > New > Project	From the File menu, choose New, then from the New submenu, choose Project.

Contacting Oracle Customer Support

Oracle Customer Support provides registered users with important information regarding Oracle software, implementation questions, product and solution help, as well as overall news and updates from Oracle.

You can contact Oracle Customer Support through Oracle's Support portal, My Oracle Support at <https://support.oracle.com>.

Part I

Configuring and Monitoring Studio



Chapter 1

Configuring Framework Settings

Framework settings are used to configure state, security, and other settings for Studio.

[About the Framework Settings page](#)

[Configuring framework settings from the Control Panel](#)

[Configuring framework settings in portal-ext.properties](#)

About the Framework Settings page

The **Framework Settings** page on the **Control Panel** allows you to view and configure the framework settings.



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

Settings that have been configured in `portal-ext.properties` are displayed, but cannot be edited.

Configuring framework settings from the Control Panel

You use the fields on the **Framework Settings** page to modify the settings. You cannot modify settings that already have been configured in `portal-ext.properties`. If a setting has been configured directly in the `portal-ext.properties` file, then the field on the **Framework Settings** page is locked.

Framework Settings

Warning! Incorrect values for these settings can cause serious problems with your Studio application. Please do not change these settings unless you are sure of what you are doing.

You must restart Studio in order for changes to these settings to take effect.

Settings that are read-only on this page are controlled by the Studio properties file. To change these settings, edit the value in the properties file.

df.auditingLogging:

The auditing logging configuration. By default, this feature is disabled. Available options are: "DISABLED","ENABLED".

df.dataSourceDirectory:

The directory from which to load existing Endeca Server connection files into the database. Also used to store the keystore and certificate files for SSL-enabled data domains. Must be an absolute path. You may start this value with the token "\${eid.studio.home}" to represent the Studio home directory.

df.deepLinkPortletName:

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

Setting	Description
df.auditingLogging	<p>Whether to enable logging of Studio system usage.</p> <p>If set to DISABLED, then Studio does not log the system usage.</p> <p>If set to ENABLED, then Studio does log the system usage, and you can use the System Usage page on the Control Panel to view system usage reports.</p> <p>See Viewing Summary Reports of Studio Usage on page 22.</p>
df.dataSourceDirectory	The directory used to store keystore and certificate files for secured data domains.
df.deepLinkPortletName	The name of the deep link component.
df.defaultChartColorPalette	<p>The default set of colors to use to display charts in the Chart component.</p> <p>The value is a comma-separated list of between 16 and 30 hex color values.</p> <p>For reference, the default value is:</p> <pre>#57BCC1,#F3A900,#A5C500,#9C2E5B,#C4B25D,#0072B1,#229903, #D55E00, #F2D900,#A279CD,#ABDDE0,#AA7600,#D2E280,#6D2040,#E2D9AE, #00507C, #91CC81,#954200,#F9EC80,#71548F,#3D8387,#F9D480,#738A00, #CE97AD, #897C41,#80B9D8,#186B02,#EAAF80,#A99700,#D1BCE6</pre>
df.defaultCurrencyList	A comma-separated list of currency symbols to add to the ones currently available.
df.defaultExporter	The default exporter class.
df.exportBatchSize	<p>When exporting a large number of records, Studio splits the records into batches.</p> <p>This setting determines the number of records in each batch.</p> <p>The default value is 2000.</p>
df.healthCheckTimeout	<p>The time (in milliseconds) for query timeout when checking data domain availability on initialization.</p> <p>The default value is 10000.</p> <p>The timeout needs to be long enough to give an idle data domain time to wake up.</p>

Setting	Description
df.helpLink	<p>Used to configure the path to the Studio documentation for this release.</p> <p>Used for links from within Studio to specific information in the documentation.</p>
df.mapLocation	<p>The URL for the Oracle MapViewer eLocation service.</p> <p>The eLocation service is used for the text location search on the Map component, to convert the location name entered by the user to latitude and longitude.</p> <p>By default, this is the URL of the global eLocation service.</p> <p>If you are using your own internal instance, and do not have Internet access, then set this setting to "None", to indicate that the eLocation service is not available. If the setting is "None", Studio disables the text location search.</p> <p>If this setting is not "None", and Studio is unable to connect to the specified URL, then Studio disables the text location search.</p> <p>Studio then continues to check the connection each time the page is refreshed. When the service becomes available, Studio enables the text location search.</p>
df.mapTileLayer	<p>The name of the MapViewer Tile Layer.</p> <p>By default, this is the name of the public instance.</p> <p>If you are using your own internal instance, then you must update this setting to use the name you assigned to the Tile Layer.</p>
df.mapViewer	<p>The URL of the MapViewer instance.</p> <p>By default, this is the URL of the public instance of MapViewer.</p> <p>If you are using your own internal instance of MapViewer, then you must update this setting to connect to your MapViewer instance.</p>
df.maxExportRecords	<p>The maximum allowable number of records that can be exported from a component.</p> <p>The default value is 1000000.</p>
df.mdexCacheManager	<p>The fully-qualified class name to use for the MDEX Cache Manager.</p> <p>Changing this setting is currently experimental and unsupported, and should be used only for research purposes. This interface will change in upcoming releases.</p>
df.mdexSecurityManager	<p>The fully-qualified class name to use for the MDEX Security Manager.</p>

Setting	Description
df.provisioningServiceLimit	<p>The maximum number of data domains created by the Provisioning Service that can be present on the Endeca Server.</p> <p>Once this limit is reached, users cannot create a new application from a file upload or the Data Source Library until one or more of those data domains is removed.</p> <p>The default value is 0, which indicates that there is no limit.</p>
df.stringTruncationLimit	<p>The maximum number of characters to display for a string value.</p> <p>This value may be overridden when configuring the display of a string value in an individual component.</p> <p>The default value is 10000.</p>
df.versionPinningTimeout	<p>The time (in milliseconds) for which to pin the version of the Endeca Server data.</p> <p>This is used to help ensure that when users export data from an application, the same version of the data is used for the entire export.</p> <p>The default value is blank, which indicates to use the Endeca Server setting. Endeca Server uses a default value of 120000 milliseconds.</p>
df.wsConnectionTimeout	<p>The time in milliseconds before a connection to the Conversation, Configuration, Entity Configuration, or EQL Parser Web service times out.</p> <p>The default value is 300000.</p> <p>If these connections are timing out frequently, then some possible causes are:</p> <ul style="list-style-type: none"> • The Endeca Server is overloaded, and might benefit from upgraded hardware. • A problem in the networking hardware is causing bottlenecks.
df.wsIngestConnectionTimeout	<p>The timeout in milliseconds for responses to requests sent to the Data Ingest Web Service.</p> <p>The default value is 1680000.</p> <p>Because these requests can take a long time, this timeout should be longer than the regular timeout (df.wsConnectionTimeout).</p>

On the **Framework Settings** page, to change a setting:

1. Provide a new value in the setting configuration field.



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

If the setting is configured in `portal-ext.properties`, then you cannot change the setting from this page. You must set it in the file.

2. Click **Update Settings**.
3. To apply the changes, restart Studio.

Configuring framework settings in `portal-ext.properties`

By default, you configure settings from the **Framework Settings** page. You also can add one or more of the settings to the `portal-ext.properties` file.

Configuring settings in `portal-ext.properties` makes it easier to migrate settings across different environments. For example, after testing the settings in a development system, you can simply copy the properties file to the production system, instead of having to reset the production settings manually from the **Control Panel**.

In the file, the format for adding a setting is:

```
<settingname>=<value>
```

Where:

- `<settingname>` is the name of the setting from the **Framework Settings** page.
- `<value>` is the value of the setting.

For example, to set the default Endeca Server connection in the file, the entry would be:

```
df.maxExportRecords=50000
```

If a setting is configured in `portal-ext.properties`, you cannot edit it from the **Control Panel**. The field on the **Framework Settings** page is read only.

To move the configuration for a setting to the properties file after Studio has been started:

1. Stop the server.
2. Add the setting to `portal-ext.properties`.
3. Restart Studio.

On the **Framework Settings** page of the **Control Panel**, the setting is now read only. You can no longer edit the value from the **Control Panel**.



Chapter 2

Configuring Logging for Studio

Studio logging helps you to monitor and troubleshoot your Studio application.

[About logging in Studio](#)

[About the log4j configuration XML files](#)

[About the Studio log files](#)

[Using the Control Panel to adjust logging verbosity](#)

About logging in Studio

Studio uses the Apache log4j logging utility.

The Studio log files include:

- A main log file with most of the logging messages
- A second log file for performance metrics logging

You can also use the **Performance Metrics** page of the **Control Panel** to view performance metrics information.

For more information about log4j, see the [Apache log4j site](#), which provides general information about and documentation for log4j.

About the log4j configuration XML files

The primary log configuration is managed in `portal-log4j.xml`, which is packed inside the portal application file `WEB-INF/lib/portal-impl.jar`.

To override settings in `portal-log4j.xml`, you use the file `portal-log4j-ext.xml`, which is located in the portal application's `/WEB-INF/classes/META-INF/` directory.

Both files are in the standard log4j XML configuration format, and allow you to:

- Create and modify appenders
- Bind appenders to loggers
- Adjust the log verbosity of different classes/packages

By default, `portal-log4j-ext.xml` specifies a log verbosity of INFO for the following packages:

- `com.endeca`
- `com.endeca.portal.metadata`
- `com.endeca.portal.instrumentation`

It does not override any of the default log verbosity settings for non-Information Discovery components.



Note: If you adjust the logging verbosity, it is updated for both log4j and the Java Utility Logging Implementation (JULI). Code using either of these loggers should respect this configuration.

About the Studio log files

For Studio, one log file contains all of the log messages, and a second file is used only for metrics logging.

About the main Studio log file

In the Studio log file configuration, the main root logger prints all messages to:

- The console, which typically is redirected to the application server's output log. For WebLogic Server, the default server log file is `domainName\servers\serverName\logs\serverName.log`
- A file called `eid-studio.log`

The main logger does not print messages from the `com.endeca.portal.instrumentation` classes. Those messages are printed to the metrics log file.

Location of `eid-studio.log`

By default, the logger tries to create `eid-studio.log` in the root directory of the WebLogic domain.

About metrics logging

Studio also captures metrics logging, including all log entries from the `com.endeca.portal.instrumentation` classes.

The metrics log file, `eid-studio-metrics.log`, is in the same directory as `eid-studio.log`.

You also can view metrics data on the **Performance Metrics** page. For details on metrics logging, see [Monitoring the Performance of Queries on page 17](#).

Using the Control Panel to adjust logging verbosity

For debugging purposes in a development environment, you can use the **Control Panel** to dynamically adjust logging verbosity levels for any class hierarchy.



Note: When you adjust the logging verbosity, it is updated for both log4j and the Java Utility Logging Implementation (JULI). Code using either of these loggers should respect this configuration.

To adjust logging verbosity from the **Control Panel**:

1. From the administrator menu, choose **Control Panel**.
2. From the **Control Panel** menu, choose **Server Administration**.

3. In the **Server Administration** page, click the **Log Levels** tab.

Server Administration
Oracle Endeca Information Discovery Studio 3.1 (Build 30113701 / Tue, 24 Sep 2013 06:08:06 -0400)
Uptime: 00:00:03

Resources **Log Levels** Properties Data Migration File Uploads Mail OpenOffice Shutdown

Update Categories Add Category

Showing 1 - 20 of 196 results. Items per Page 20 Page 1 of 10 First Previous Next Last

Category	Level
com.endeca	INFO
com.endeca.portal.instrumentation	INFO
com.endeca.portal.metadata	INFO
com.germinus.easyconf	ERROR
com.liferay	ERROR

4. On the **Update Categories** tab, locate the class hierarchy you want to modify.
5. From the logging level drop-down list, select the logging level.



Note: When you modify a class hierarchy, all classes that fall under that class hierarchy also are changed.

6. When you have finished adjusting log levels, click **Save**.

You also can use the **Add Category** tab to set the verbosity for a specific class or package.



Monitoring the Performance of Queries

You can get access to performance metrics data both from the metrics log file and from the **Performance Metrics** page. A setting in `portal-ext.properties` controls the amount of metrics data to record.

[Configuring the amount of metrics data to record](#)

[About the metrics log file](#)

[About the Performance Metrics page](#)

Configuring the amount of metrics data to record

To configure the metrics you want to include, you use a setting in `portal-ext.properties`. This setting applies to both the `eid-studio-metrics.log` file and the **Performance Metrics** page.

The metrics logging can include:

- Endeca Server queries by data domain
- Portlet server executions by component. The server side code is written in Java.
It handles configuration updates, configuration persistence, and Endeca Server queries. The server side code generates results to send back to the client side code.
Server executions include component render, resource, and action requests.
- Component client executions for each component. The client side code lives in the browser and is written in JavaScript.
It issues requests to the server code, then renders the results as HTML. The client code also handles any dynamic events within the browser.

By default, only the Endeca Server queries and component server executions are included.

You use the `df.performanceLogging` setting in `portal-ext.properties` to configure the metrics to include. The setting is:

```
df.performanceLogging=<metrics to include>
```

Where `<metrics to include>` is a comma-separated list of the metrics to include. The available values to include in the list are:

Value	Description
QUERY	If this value is included, then the page includes information for Endeca Server queries.

Value	Description
PORTLET	If this value is included, then the page includes information on component server executions.
CLIENT	If this value is included, then the page includes information on component client executions.

In the default configuration, where only the Endeca Server queries and component server executions are included, the value is:

```
df.performanceLogging=QUERY,PORTLET
```

To include all of the available metrics, you would add the `CLIENT` option:

```
df.performanceLogging=QUERY,PORTLET,CLIENT
```

Note that for performance reasons, this configuration is not recommended.

If you make the value empty, then the log file and **Performance Metrics** page also are empty.

```
df.performanceLogging=
```

About the metrics log file

The `eid-studio-metrics.log` file contains the metrics logging information. It is located in the same directory as the `eid-studio.log` file.

The metrics log file contains the following columns:

Column Name	Description
Total duration (msec)	The total time for this entry (End time minus Start time).
Start time (msec since epoch)	The time when this entry started. For Endeca Server queries and server executions, uses the server's clock. For client executions, uses the client's clock.
End time (msec since epoch)	The time when this entry was finished. For Endeca Server queries and server executions, uses the server's clock. For client executions, uses the client's clock.
Session ID	The session ID for the client.
Page ID	If client instrumentation is enabled, the number of full page refreshes or actions the user has performed. Used to help determine how long it takes to load a complete page. Some actions that do not affect the overall state of a page, such as displaying attributes on an Available Refinements component, do not increment this counter.

Column Name	Description
Gesture ID	The full count of requests to the server.
Portlet ID	This is the ID associated with an individual instance of a component. It generally includes: <ul style="list-style-type: none"> • The type of component • A unique identifier For example, if a page includes two Chart components, the ID can be used to differentiate them.
Entry Type	The type of entry. For example: <ul style="list-style-type: none"> • PORTLET_RENDER - Server execution in response to a full refresh of a component • DISCOVERY_SERVICE_QUERY - Endeca Server query • CONFIG_SERVICE_QUERY - Configuration service query • SCONFIG_SERVICE_QUERY - Semantic configuration service query • LQL_PARSER_SERVICE_QUERY - EQL parser service query • CLIENT - Client side JavaScript execution • PORTLET_RESOURCE - Server side request for resources • PORTLET_ACTION - Server side request for an action
Miscellaneous	A URL encoded JSON object containing miscellaneous information about the entry.

About the Performance Metrics page

The **Performance Metrics** page on the **Control Panel** displays information about component and Endeca Server query performance.

It uses the same logging data recorded in `eid-studio-metrics.log`.

However, unlike the log file, the **Performance Metrics** page uses data stored in memory. Restarting Studio clears the **Performance Metrics** data.

For each type of included metric, the table at the top of the page contains a collapsible section.

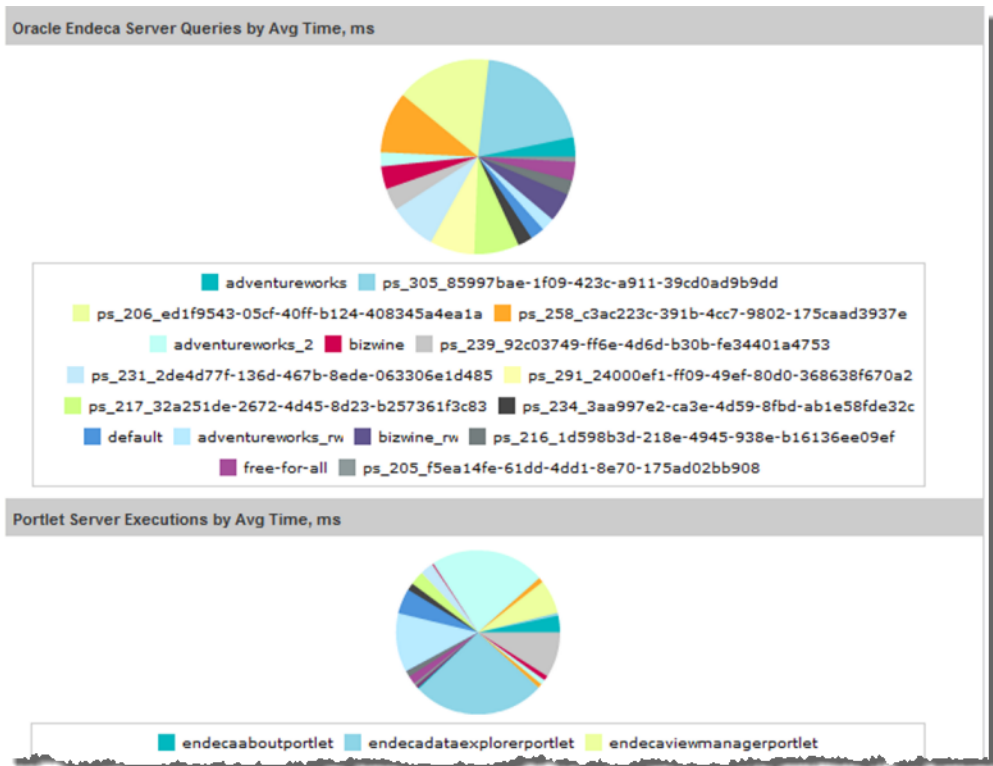
Performance Metrics

Performance Metrics					
Name ▲	Count	Total Time, ms	Avg Time, ms	Max Time, ms	
▼ Oracle Endeca Server Queries					
adventureworks	28	6980	249	2603	
adventureworks_2	40	7131	178	543	
adventureworks_rw	928	159285	171	4840	
bizwine	457	132479	289	2928	
bizwine_rw	531	195181	367	4281	
default	4111	734544	178	3245	
free-for-all	268	63290	236	2184	
ps_205_f5ea14fe-61d...	57	3814	66	649	
ps_206_ed1f9543-05...	83	100603	1212	9567	
ps_216_1d598b3d-21...	92	16810	182	3343	
ps_217_32a251de-26...	1	574	574	574	
ps_231_2de4d77f-13...	1	598	598	598	
ps_234_3aa997e2-ca...	10	1860	186	1052	
ps_239_92c03749-f6...	15	4264	284	1094	

For each data source or component, the table tracks:

- Total number of queries or executions
- Total execution time
- Average execution time
- Maximum execution time

For each type of included metric, there is also a pie chart summarizing the average query or execution time per data source or component.



Note: Endeca Server query performance does not correlate directly to a Studio application page, as a single Studio page often uses multiple Endeca Server queries.



Chapter 4

Viewing Summary Reports of Studio Usage

Studio provides basic reports to allow you to track Studio system usage.

[Enabling the logging of Studio system usage](#)

[About the System Usage page](#)

[Viewing the Number of Users by Date report](#)

[Viewing the Most/Least Frequently Accessed Summary Report](#)

Enabling the logging of Studio system usage

Studio can be configured to store application creation and usage information in the Studio database. To enable this logging, you must update a framework setting. If you are using a database other than those officially supported by Studio, then you may also need to update some custom SQL properties.

Framework setting

Usage logging is enabled if the framework setting `df.auditingLogging` is set to `ENABLED`.

About the usage logs

The usage logs are stored in the `ENDECA_AUDITING_LOG` table. If the logging is enabled, then Studio adds entries when users:

- Log in to Studio
- Navigate to an application
- Navigate to a different page in an application
- Create a data set from the **Data Source Library**
- Create an application from a shared Endeca Server connection

Updating custom SQL properties

When it generates these reports, Studio uses custom SQL functions to format date values.

If you are using one of Studio's officially supported databases, then Studio automatically uses the correct functions.

If you are using a different database, then you may need to add the following custom SQL functions for formatting date values.

- `custom.sql.function.year`

- `custom.sql.function.month`
- `custom.sql.function.day`
- `custom.sql.function.week`
- `custom.sql.function.quarter`
- `custom.sql.function.concat`

In `portal-ext.properties`, under the heading Custom SQL, there are commented out examples of these settings for each of the supported databases:

```
# Hypersonic
#
#custom.sql.function.year=TO_CHAR(? , 'YYYY')
#custom.sql.function.month=TO_CHAR(? , 'YYYY-MM')
#custom.sql.function.day=TO_CHAR(? , 'YYYY-MM-DD')
#custom.sql.function.week=CONCAT(YEAR(?), CONCAT(' W', WEEK(?)))
#custom.sql.function.quarter=CONCAT(YEAR(?), CONCAT(' Q', QUARTER(?)))
#custom.sql.function.concat=CONCAT(p1,p2)

#
# Oracle
#
#custom.sql.function.year=TO_CHAR(? , 'YYYY')
#custom.sql.function.month=TO_CHAR(? , 'YYYY-MM')
#custom.sql.function.day=TO_CHAR(? , 'YYYY-MM-DD')
#custom.sql.function.week=TO_CHAR(? , 'YYYY "W"WW')
#custom.sql.function.quarter=TO_CHAR(? , 'YYYY "Q"Q')
#custom.sql.function.concat=CONCAT(p1,p2)

#
# MySQL
#
#custom.sql.function.year=DATE_FORMAT(? , '%Y')
#custom.sql.function.month=DATE_FORMAT(? , '%Y-%m')
#custom.sql.function.day=DATE_FORMAT(? , '%Y-%m-%d')
#custom.sql.function.week=DATE_FORMAT(? , '%Y W%U')
#custom.sql.function.quarter=CONCAT(YEAR(?), CONCAT(' Q', QUARTER(?)))
#custom.sql.function.concat=CONCAT(p1,p2)
```

To customize the values for the custom SQL functions:

1. Copy one of these sets of values.
2. Remove the commenting.
3. Set the custom functions as needed.
4. Save the file.
5. Restart Studio.

About the System Usage page

The **System Usage** page of the **Control Panel** provides access to summary information from the ENDECA_AUDITING_LOG table.

The page displays the available usage reports. Each report allows you to provide criteria to customize the data you want to see. The reports are:

Report Name	Description
Number of Users by Date	<p>For a selected time frame, shows the number of user logins per unit of time.</p> <p>For example, the report could show the number of users who logged in per day for the last week.</p>
Most/Least Frequently Accessed Summary Report	<p>Provides access to the top or bottom values over a selected time period for:</p> <ul style="list-style-type: none"> • Number of user sessions • Number of user sessions per application • Number of times a data source from the Data Source Library was used • Number of applications created from shared Endeca Server connections

Viewing the Number of Users by Date report

The **Number of Users by Date** report displays the number of user logins during a selected time frame.

For the report, you can specify:

- The time frame for which to generate the report.

If you do not provide either a start or end date for the report, then Studio generates the report using all dates.

- The units of time for which to display the number

From the **System Usage** page of the **Control Panel**, to generate the **Number of Users by Date** report:

1. Use the **From** field to select the start date for the time period for which display the data.

2. Use the **To** field to select the end date for the time period for which to display the data.
3. From the **Date/Time Option** drop-down list, select the unit for which to display the counts.

You can display the number of logins:

- Per day
 - Per week
 - Per month
 - Per quarter
 - Per year
4. Click **Update Report**.

Viewing the Most/Least Frequently Accessed Summary Report

The **Most/Least Frequently Accessed Summary Report** displays the top or bottom values for a selected usage type.

By default, the report shows the most-frequently accessed applications from the last month.

For the report, you can specify:

- The type of information to display
- The time period for which to display the data.

If you do not provide either a start or end date for the report, then Studio generates the report using all dates.

- Whether to display the most or least frequently accessed applications
- The number of results to display

From the **System Usage** page of the **Control Panel**, to generate the **Most/Least Frequently Accessed Summary Report**:

1. From the **Report Name** drop-down list, select the type of data to display on the report.

The screenshot shows a web form titled "Most/Least Frequently Accessed Summary Report". It contains the following fields and options:

- Report Name:** A dropdown menu with "Total number of sessions per application" selected.
- From:** A date input field with "08/04/13" and a calendar icon.
- To:** A date input field with "09/04/13" and a calendar icon.
- Top/Bottom:** Two radio buttons, with "Top" selected.
- Results Limit:** A text input field with "10".
- Update Report:** A button at the bottom left.

The options are:

Access Type	Description
Total number of user sessions	The number of logins.
Total number of sessions per application	The number of times each application has been accessed.
Number of times each data source was used	The number of times each data source from the Data Source Library was used to create a data set.
Number of applications from pre-built Endeca Servers	The number of applications that have been created from a shared Endeca Server connection.

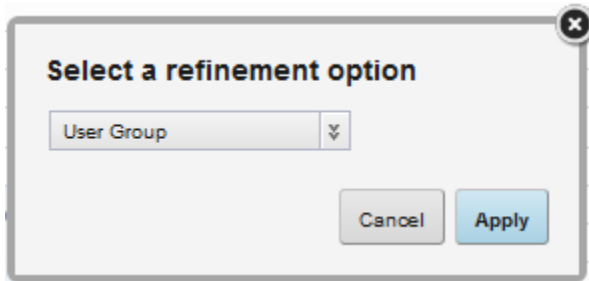
2. Use the **From** field to select the start date for the time period for which to display the data.
3. Use the **To** field to select the end date for the time period for which to display the data.
4. Under **Top/Bottom**, click a radio button to indicate whether to display the most accessed applications or the least accessed applications.
5. In the **Results Limit** field, type the number of results to display.
6. Click **Update Report**.
7. For reports based on use and applications, you can refine the report data.

If the report summary is based on the number of sessions per application, the report initially shows the number of times each application was accessed.

To further refine the data for a specific application:

- (a) Click the bar for the application.

- (b) On the refinement dialog, from the drop-down list, select whether to refine by page, user group, or user.



- (c) Click **Apply**.

If you refine by page, you can further refine by user group or user.

If you refine by user group, you can further refine by user.

If the report is based on the number of user sessions, the report initially shows a single bar with the total number of sessions for the selected time period.

To further refine the data:

- (a) Click the bar.
- (b) On the refinement dialog, from the drop-down list, select whether to refine by user group or user.
- (c) Click **Apply**.

If you refine by user group, you can refine the report to show the users from a specific user group.

When you refine the data, the refinements are displayed above the report. To remove the latest refinement, click the delete icon.



Chapter 5

Determining and Configuring the Locale Used in Studio

The Studio user interface and application data can be displayed in different locales.

[About locales and their effect on the Studio user interface](#)

[How Studio determines the locale to use](#)

[Setting the available locales for Studio](#)

[Setting the default locale used by Studio](#)

[Configuring the preferred locale for a Studio user](#)

[Including the locale in a URL](#)

About locales and their effect on the Studio user interface

The locale determines the language in which to display the Studio user interface. It can also affect the format of displayed data values.

Studio is configured with a default locale as well as a list of available locales.

Each user account also is configured with a preferred locale, and the user menu includes an option for users to select the locale to use.

In Studio, when a locale is selected:

- User interface labels are displayed using the locale
- Display names of attributes are displayed in the locale.
If there is not a version for that locale, then the default locale is used.
- Data values are formatted based on the locale.
- If the data contains locale-specific versions of attributes, then those locale-specific values are displayed on the application components.

If a locale-specific value is not available for the selected locale, then the default locale version is displayed.

The exception to this is the **Selected Refinements** component, which always displays the actual value that the user selected.

Supported locales in Studio

Studio supports the following languages:

- French

- German
- Italian
- Spanish
- Japanese
- Korean
- Simplified Chinese
- Traditional Chinese
- Portuguese-European

Note that this is a subset of the languages supported by Oracle Endeca Server.

How Studio determines the locale to use

When users enter Studio, it needs to determine the locale to use to display the user interface and data.

[Locations where the locale may be set](#)

[Scenarios for selecting the locale](#)

Locations where the locale may be set

The locale is set in different locations in Studio.

Studio can get the locale from the following locations:

- Studio cookie
- Browser locale
- Studio default locale
- User preferred locale, stored as part of the user account
- Locale selected using the **Change locale** option in the user menu, which is also available to users who have not yet logged in.
- Locale provided as part of a deep linking URL into Studio. For example:

```
http://localhost:8080/web/myapp/my-  
page?doAsUserId=zh_CN&deeplink=[{"default":{"queryFunctions":[{"class":"R  
efinementFilter","attributeValue":"1997","attributeKey":"Vintage"}]}}
```

Scenarios for selecting the locale

The locale used depends upon the type of user, the Studio configuration, and how the user entered Studio.

For the scenarios listed below. Studio determines the locale as follows:

Scenario	How the locale is determined
A new Studio user is created	<p>The locale for a new user is initially set to Use Browser Locale, which indicates to use the current browser locale.</p> <p>This value can be changed to a specific locale.</p> <p>If the user is configured with a specific locale, then that locale is used for the user unless they explicitly select a different locale or enter Studio with a URL that includes a supported locale.</p>
A non-logged-in user navigates to Studio	<p>For a non-logged-in user, Studio first tries to use the locale from the cookie.</p> <p>If there is no cookie, or the cookie is invalid, then Studio tries to use the browser locale.</p> <p>If the current browser locale is not one of the supported Studio locales, then Studio uses its configured default locale.</p>
A registered user logs in to Studio	<p>When a user logs in, Studio first checks the locale configured for their user account.</p> <ul style="list-style-type: none"> • If the user's locale is set to Use Browser Locale, then Studio tries to use the locale from the cookie. <p>If there is no cookie, or if the cookie is invalid, then Studio tries to use the browser locale.</p> <p>If the current browser locale is not a supported Studio locale, then Studio uses its configured default locale.</p> <ul style="list-style-type: none"> • If the user account is configured with a locale value other than Use Browser Locale, then Studio uses that locale, and also updates the cookie with that locale.
A non-logged-in user uses the user menu option to select a different locale	<p>When a non-logged-in user selects a locale, Studio updates the cookie with the new locale.</p> <p>Note that this locale change is only applied locally. It is not applied to all non-logged-in users.</p>
A logged-in user uses the user menu option to select a different locale	<p>When a logged-in user selects a locale, Studio updates both the user's account and the cookie with the selected locale.</p>

Scenario	How the locale is determined
A non-logged-in user navigates to Studio using a URL that includes a locale	<p>If the locale from the URL is supported by Studio, then Studio uses that locale and also updates the cookie with that locale.</p> <p>If the URL locale is not a supported Studio locale, then Studio tries to use the locale from the cookie.</p> <p>If there is no cookie, or if the cookie is invalid, then Studio tries to use the browser locale.</p> <p>If the current browser locale is not one of the supported Studio locales, then Studio uses its configured default locale.</p>
A logged-in user navigates to Studio using a URL that includes a locale	<p>If the URL locale is supported by Studio, then Studio uses that locale. Studio updates both the cookie and the user's account to reflect that URL.</p> <p>If the URL locale is not a supported Studio locale, then Studio gets the locale configured for the user's account.</p> <ul style="list-style-type: none"> • If the user's locale is set to Use Browser Locale, then Studio tries to use the locale from the cookie. <p>If there is no cookie, or if the cookie is invalid, then Studio tries to use the browser locale.</p> <p>If the current browser locale is not a supported Studio locale, then Studio uses its configured default locale.</p> <ul style="list-style-type: none"> • If the user's locale is a value other than Use Browser Locale, then Studio uses that locale and also updates the cookie with that locale.

Setting the available locales for Studio

Studio is configured with a list of available locales. This list is used to populate the drop-down list for configuring the Studio default locale, user default locale, and the available locales displayed for the **Change locale** option.

You can customize the setting to constrain the list. Supported locales are specified in `portal.properties`.

```
locales=de_DE, en_US, es_ES, fr_FR, it_IT, ja_JP, ko_KR, pt_PT, zh_CN, zh_TW
```

To reduce this list:

1. Copy this parameter into `portal-ext.properties`.
2. Update the list to remove the locales that you do not want to be available.

For example, to only support English, French, and Japanese, you would update it to:

```
locales=en_US, fr_FR, ja_JP
```


Setting the default locale used by Studio

Studio is configured with a default locale, which you can update from the **Control Panel**.

Note that if you have a clustered implementation, make sure to configure the same locale for all of the instances in the cluster.

To configure the default locale for an instance of Studio:

1. From the administrator menu, select **Control Panel**.
2. On the **Control Panel** menu, in the **Portal** section, click **Settings**.
3. On the **Portal** page, in the menu on the right, click **Display Settings**.
4. On the **Display Settings** page, from the **Locale** drop-down list, select the default locale for Studio.

Display Settings

Locale

United States - English ▼

Time Zone

(UTC) Coordinated Universal Time ▼

5. Click **Save**.

Configuring the preferred locale for a Studio user

Each user account is configured with a preferred locale. The default value for new users is **Use Browser Locale**, which indicates to use the current browser locale.

To configure the default locale for a user:

1. To display the setting for your own account:
 - (a) From the user menu, select **My Account**.

- (b) On the **My Account** page, in the menu on the right, click **Display Settings**.

Admin Admin

My Account

Display Settings

Locale

Time Zone

Greeting

2. To display the setting for another user:
- From the administrator menu, select **Control Panel**.
 - In the **Control Panel** menu, under **Portal**, click **Users**.
 - On the **Users** page, click the **Actions** button for the user you want to edit.
 - From the **Actions** menu, select **Edit**.
 - On the user edit page, in the menu on the right, click **Display Settings**.

Users

Display Settings

Locale

Time Zone

Greeting

- From the **Locale** drop-down list, select the preferred locale for the user.
- Click **Save**.

Including the locale in a URL

To include the locale in the URL, add the locale as a value of the `doAsUserLanguageId` parameter.

For example, to include the locale in a deep linking URL:

```
http://localhost:8080/web/myapp/my-  
page?doAsUserLanguageId=zh_CN&deeplink=[{"default":{"queryFunctions":[{"class":"Refi  
nementFilter","attributeValue":"1997","attributeKey":"Vintage"}]}}]
```

If the locale provided is not supported, then Studio reverts to the default locale.

Part II

Controlling Access to Studio



Chapter 6

About Managing Users in Studio

You can create users directly in Studio, or connect to an LDAP system.

[About user roles](#)

[About the default user](#)

About user roles

In Studio, each user is assigned a user role. The user role controls the functions that the user has access to.

The basic user roles are:

Role	Description
Power User	<p>For a new user, the default role is Power User. These users can:</p> <ul style="list-style-type: none">• View Studio applications, based on the application and page type and their application membership• Create Studio applications• Configure and manage applications for which they are an administrator• Edit their account information <p>They do not have access to Control Panel functions.</p>
Administrator	<p>Administrators have full access to Studio and Studio applications. They can:</p> <ul style="list-style-type: none">• View all Studio applications• Create Studio applications• Configure and manage all Studio applications• Use all of the Control Panel functions

Role	Description
User	<p>A user who does not have either the Power User or Administrator role has the User role. These users can:</p> <ul style="list-style-type: none"> • View Studio applications, based on the application and page type and their application membership. • Configure and manage applications for which they are an administrator • Edit their account information <p>They cannot create new applications, and they have no access to Control Panel functions.</p>

For information on using the **Application Configuration** page to configure application access and assign application roles to users, see the *Studio User's Guide*.

For information on using the **Control Panel** to configure application access, see [Configuring and Removing Applications on page 96](#).

About the default user

When you first install Studio, a default user is created.

The default user is an administrator and has full privileges to:

- View all Studio applications and pages, including private applications and pages
- Create Studio applications
- Configure and manage all Studio applications, including private applications
- View and use all of the **Control Panel** functions

To log in as the default user for the very first time, use the following user name and password:

Field	Value to Enter
Email address:	admin@oracle.com
Password:	Welcome123

You are immediately prompted to change the password. The new password must contain:

- At least 6 characters
- At least one non-alphabetic character



Chapter 7

Creating and Editing Users in Studio

The **Users** page on the Studio **Control Panel** provides options for creating and editing Studio users.

[Configuring the type of user name for Studio](#)

[Creating a new user](#)

[Editing a Studio user](#)

[Preventing a user from creating applications](#)

[Deactivating, reactivating, and deleting Studio users](#)

Configuring the type of user name for Studio

Each Studio user has both an email address and a screen name. By default, users log in to Studio using their email address.

To change the configuration so that users log in with their screen name:

1. From the administrator menu, select **Control Panel**.
2. On the **Control Panel** menu, click **Settings**.
3. In the **Settings** page menu to the right, click **Authentication**.
4. On the **General** tab, from the **How do users authenticate?** drop-down list, select the name used to log in.

Settings

Authentication

General | LDAP | Oracle Access Manager SSO | CAS | NTLM | Open SSO | SiteMinder

How do users authenticate?

By Email Address

Allow users to automatically login?

Allow users to request forgotten passwords?

Allow strangers to create accounts?

Allow strangers to create accounts with a company email address?

Require strangers to verify their email address?

ORACLE
oracle.com

Permissions
General

Authentication

Default User Associations
Reserved Credentials
Mail Host Names
Email Notifications

Identification
Addresses
Phone Numbers
Additional Email Addresses
Websites

Miscellaneous
Display Settings

Save Cancel

5. Click **Save**.

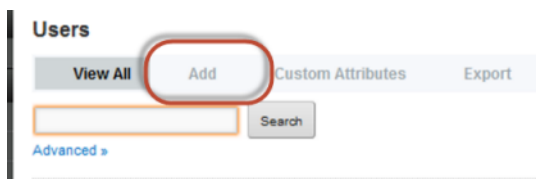
Creating a new user

Even if you are importing users from LDAP, you may still want to create a few users directly in Studio.

For example, for a small development instance, you may just need a few users to develop and test pages. Or if your LDAP users for a production site are all end users, you may need a separate user account for administering the site.

To add a new user:

1. From the administrator menu, select **Control Panel**.
2. On the **Control Panel** menu, click **Users**.
3. On the **Users** page, click **Add**.



4. On the **Details** page, to provide the minimum required information:
 - (a) In the **Screen Name** field, type the screen name for the user.
The screen name must be unique, and cannot match the screen name of any current active or inactive user.
 - (b) In the **Email Address** field, type the user's email address.
 - (c) In the **First Name** field, type the user's first name.
 - (d) In the **Last Name** field, type the user's last name.
5. Click **Save**.
The new user is added, and the configuration menu is updated to add the rest of the options.
6. To create the password for the user:
 - (a) On the user page, on the configuration menu to the right, click **Password**.
 - (b) On the **Password** page, enter the password to assign to the new user.
 - (c) To force the user to change the password the first time they log in, check the **Password Reset Required** checkbox.
 - (d) Click **Save**.
7. To add the user to an application:
 - (a) On the user page, from the list to the right, click **Applications**.
 - (b) Click the **Select** link.
 - (c) On the applications list, click the application to add the user to.
The user is made an application member.
 - (d) Click **Save**.

8. To manage the user roles for the user:
 - (a) On the user page, in the list to the right, click **Roles**.
By default, the new user has the Power User role.



Power Users can create new applications, but have no access to the Studio **Control Panel**. They can only view public applications, and private applications that they are members of. They can only edit applications for which they are an application administrator.

- (b) To configure the user to only have the User role, click the **Remove** link next to the Power User role.
If you remove the Power User role, then the user cannot create new applications in Studio.
 - (c) To make the user a Studio administrator, click the **Select** link, then in the **Roles** list, click **Administrator**.
 - (d) Click **Save**.
9. In order for the user to be able to configure an application they belong to, they must be an application administrator. On the **Roles** page for the user, to make the user an application administrator for an application that you added them to:
 - (a) Under **Application Roles**, click the **Select** link.
 - (b) If you added the user to more than one application, then in the application list, click the application you want to make them an administrator for.
 - (c) In the application roles list, click **Application Administrator**.
 - (d) Click **Save**.

Editing a Studio user

The **Users** page also allows you to edit a user's account.

From the **Users** page, to edit a user:

1. Click the **Actions** button next to the user.
2. In the **Actions** menu, click **Edit**.
3. After making your changes, click **Save**.

Preventing a user from creating applications

When you first create a new user, they are by default assigned the Power User role. This role allows the user to create Studio applications.

To prevent a user from being able to create applications, you must remove the Power User role.

To do this:

1. On the **Users** page, click the user name.
2. On the user configuration page, click the **Roles** option.

The **Regular Roles** section of the **Roles** page shows the Power User role assigned to the user.



3. To remove the Power User role, click the **Remove** link.
4. To save the change to the user configuration, click **Save**.

On the **Discovery Applications** page, the user does not see the **New Application** button.

Deactivating, reactivating, and deleting Studio users

From the **Users** page of the **Control Panel**, you can make an active user inactive. You can also reactivate or delete inactive users.

Note that you cannot make your own user account inactive, and you cannot delete an active user.

From the **Users** page, to change the status of a user account:

1. To make an existing user inactive:
 - (a) In the users list, click the **Actions** button for the user you want to deactivate.
 - (b) From the **Actions** menu, select **Deactivate**.
Studio prompts you to confirm that you want to deactivate the user.
The user is then removed from the list of active users.
Note that inactive users are not removed from Studio.
2. To reactivate or delete an inactive user:
 - (a) Click the **Advanced** link below the user search field.
Studio displays additional user search fields.
 - (b) From the **Active** drop-down list, select **No**.
 - (c) Click **Search**.
The users list displays only the inactive users.

- (d) Click the **Actions** button for the use you want to reactivate.
- (e) To reactivate the user, from the **Actions** menu, select **Activate**.
- (f) To delete the user, from the **Actions** menu, select **Delete**.



Chapter 8

Integrating with an LDAP System to Manage Users

If you have an LDAP system, you can allow users to use those credentials to log in to Studio.

[About using LDAP](#)

[Configuring the LDAP settings and server](#)

[Configuring the Studio password policy when using LDAP](#)

[Preventing encrypted LDAP passwords from being stored in Studio](#)

[Assigning roles based on LDAP user groups](#)

About using LDAP

LDAP (Lightweight Directory Access Protocol) allows you to have users connect to your Studio application using their existing LDAP user accounts, rather than creating separate user accounts from within Studio. LDAP is also used when integrating with a single sign-on (SSO) system.

Configuring the LDAP settings and server

The LDAP settings on the **Control Panel** include whether LDAP is enabled and required for authentication, the connection to the LDAP server, and whether to support batch import or export to or from the LDAP directory. The method for processing batch imports is set in `portal-ext.properties`.

In `portal-ext.properties`, the setting `ldap.import.method` determines how to perform batch imports from LDAP. This setting is only applied if batch import is enabled. The available values for `ldap.import.method` are:

Value	Description
user	<p>Indicates to use user-based import. This is the default value.</p> <p>User-based batch import uses the import search filter configured in the Users section of the LDAP tab.</p> <p>For user-first import, Studio:</p> <ol style="list-style-type: none">1. Uses the user import search filter to run an LDAP search query.2. Imports the resulting list of users, including all of the LDAP groups the user belongs to. <p>The group import search filter is ignored.</p>

Value	Description
group	<p>Indicates to use group-based import.</p> <p>Group-based import uses the import search filter configured in the Groups section of the LDAP tab.</p> <p>For group-based import, Studio:</p> <ol style="list-style-type: none"> 1. Uses the group import search filter to run an LDAP search query. 2. Imports the resulting list of groups, including all of the users in those groups. <p>The user import search filter is ignored.</p>

The value you should use depends partly on how your LDAP system works. If your LDAP directory only provides user information, without any groups, then you have to use user-based import. If your LDAP directory only provides group information, then you have to use group-based import.

To configure the LDAP server and settings:

1. On the **Control Panel** menu, click **Settings**.
2. In the **Settings** page menu to the right, click **Authentication**.
3. Click the **LDAP** tab.

Settings

Authentication

General **LDAP** Oracle Access Manager SSO CAS NTLM Open SSO
 SiteMinder

Enabled

Required

Default Values

Apache Directory Server
 Fedora Directory Server
 Microsoft Active Directory Server
 Novell eDirectory
 OpenLDAP
 Oracle Internet Directory
 Other Directory Server

Reset Values

4. On the **LDAP** tab:
 - (a) To enable LDAP authentication, check the **Enabled** checkbox.
 - (b) To only allow users to log in using an LDAP account, check the **Required** checkbox.
 If this box is checked, then any users that you create manually in Studio cannot log in.

To make sure that users you create manually can log in, make sure that this box is not checked.

5. To populate the LDAP server configuration fields with default values based on a specific type of server:
 - (a) Under **Default Values**, click the radio button for the type of server you are using.
 - (b) Click **Reset Values**.
6. The **Connection** settings cover the basic connection to LDAP:

Connection

Base Provider URL
 ?

Base DN
 ?

Principal

Credentials

Field	Description
Base Provider URL	The location of your LDAP server. Make sure that the machine on which Studio is installed can communicate with the LDAP server. If there is a firewall between the two systems, make sure that the appropriate ports are opened.
Base DN	The Base Distinguished Name for your LDAP directory. For a commercial organization, it may look something like: <code>dc=companynamehere,dc=com</code>
Principal	The user name of the administrator account for your LDAP system. This ID is used to synchronize user accounts to and from LDAP.
Credentials	The password for the administrative user.

After providing the connection information, to test the connection to the LDAP server, click the **Test LDAP Connection** button.

- The **Users** section contains settings for finding users in your LDAP directory. The first couple of settings are filters for finding and identifying users.

Field	Description
Authentication Search Filter	<p>The search criteria for user logins.</p> <p>If you do not enable batch import of LDAP users, then the first time a user tries to log in, Studio uses this authentication search filter to search for the user in the LDAP directory.</p> <p>By default, users log in using their email address. If you have changed this setting, you must modify the search filter here.</p> <p>For example, if you changed the authentication method to use the screen name, you would modify the search filter so that it can match the entered login name:</p> <pre>(cn=@screen_name@)</pre>
Import Search Filter	<p>The search filter to use for batch import of users.</p> <p>This filter is used if:</p> <ul style="list-style-type: none"> You enable batch import of LDAP users In <code>portal-ext.properties</code>, <code>ldap.import.method</code> is set to <code>user</code> <p>Depending on the LDAP server, there are different ways to identify the user.</p> <p>The default setting <code>(objectClass=inetOrgPerson)</code> usually is fine, but to search for only a subset of users or for users that have different object classes, you can change this.</p>

8. Under **User Mapping**, map your LDAP attributes to the Studio user fields:

User Mapping

Screen Name

Password

Email Address:

Full Name

First Name

Last Name

Job Title

Group

After setting up the attribute mappings, to test the mappings, click **Test LDAP Users**.

9. Under **Groups**, map your LDAP groups.

Groups

Import Search Filter

Group Mapping

Group Name

Description

User

- (a) In the **Import Search Filter** field, type the filter for finding LDAP groups.

This filter is used if:

- You enable batch import of LDAP users

- In `portal-ext.properties`, `ldap.import.method` is set to `group`
- (b) Map the following group fields:
- Group Name
 - Description
 - User
- (c) To test the group mappings, click **Test LDAP Groups**.

The system displays a list of the groups returned by your search filter.

10. The **Import/Export** section is used to configure batch import and export of LDAP user data:

Import / Export

Import Enabled

Export Enabled

- (a) If the **Import Enabled** checkbox is checked, then batch import of LDAP users is enabled.

If the box is not checked, then Studio synchronizes each user as they log in.

It is recommended that you leave this box unchecked.

If you do enable batch import, then the import process is based on the value of `ldap.import.method`.

Note also that when using batch import, you cannot filter both the imported users and imported groups at the same time. For user-based batch import mode, you cannot filter the LDAP groups to import. For group-based batch import mode, you cannot filter the LDAP users to import.

- (b) If the **Export Enabled** checkbox is checked, then any changes to the user in Studio are exported to the LDAP system.

It is recommended that you leave this box unchecked.

11. To use the password policy from your LDAP system, instead of the Studio password policy, check the **Use LDAP Password Policy** checkbox.

Password Policy

Use LDAP Password Policy

12. To save the LDAP configuration, click **Save**.

Configuring the Studio password policy when using LDAP

When you are using LDAP, it is likely that you want user passwords to be managed outside of Studio. So if you are not using the LDAP password policy, then you may want to update the Studio password policy to prevent users from changing their password in Studio.

To update the password policy:

1. From the administrator menu, select **Control Panel**.
2. In the **Control Panel** menu, click **Password Policies**.

3. On the **Password Policies** page, click the **Actions** button, then click **Edit**.
4. On the **Password Policies** page:

Password Policies

View All **Add**

Name

Description

Default Password Policy

Changeable ?

Change Required ?

Minimum Age ?

- (a) To prevent users from being able to change passwords from within Studio, uncheck the **Changeable** checkbox.
 - (b) To prevent users from being prompted to change their password the first time they log in to Studio, uncheck the **Change Required** checkbox.
5. To save the changes, click **Save**.

Preventing encrypted LDAP passwords from being stored in Studio

By default, when you use LDAP for user authentication, each time a user logs in, Studio stores a securely encrypted version of their LDAP password. For subsequent logins, Studio can then authenticate the user even when it cannot connect to the LDAP system. For even stricter security, you can configure Studio to prevent the passwords from being stored.

To prevent Studio from storing the encrypted LDAP passwords:

1. Stop Studio.
2. Add the following settings to `portal-ext.properties`:

```
ldap.password.cache.hashd=false
ldap.auth.required=true
auth.pipeline.enable.liferay.check=false
```

3. Restart Studio.

Studio no longer stores the encrypted LDAP passwords for authenticated users. If the LDAP system is unavailable, Studio cannot authenticate previously authenticated users.

Assigning roles based on LDAP user groups

For LDAP integration, it is recommended that you assign roles based on your LDAP groups.

To ensure that users have the correct roles as soon as they log in, you create groups in Studio that have the same name as your LDAP groups, but in lowercase, then assign the correct roles to each group.

To create a user group and then assign a role to that group:

1. From the administrator menu, select **Control Panel**.
2. On the **Control Panel**, click **User Groups**.

The screenshot shows the 'User Groups' page. At the top, there are two buttons: 'View All' and 'Add'. Below them is a search bar with a 'Search' button. A 'Delete' button is also visible. Below the buttons is a table with two columns: 'Name' and 'Description'. The table is empty, and a message below it says 'No user groups were found.' At the bottom, it says 'Showing 0 results.'

3. On the **User Groups** page, to add a new group:
 - (a) Click **Add**.

The new group page is displayed.

The screenshot shows the 'User Groups' page with the 'Add' button selected. Below the buttons, there are two input fields: 'Name' and 'Description'. Each field has a small flag icon to its right. At the bottom, there are two buttons: 'Save' and 'Cancel'.

- (b) On the new group page, in the **Name** field, type the name of the group.

Make sure the name is the lowercase version of the name of a group from your LDAP system. For example, if the LDAP group is called SystemUsers, then the user group name would be systemusers.

To provide localized versions of the user group name, click the flag icon.

- (c) In the **Description** field, type a description of the group.

To provide localized versions of the group description, click the flag icon.

- (d) Click **Save**.

The group is added to the **User Groups** list.

4. To assign the group to a role:
 - (a) In the **Control Panel** menu, click the **Roles** option.
 - (b) On the **Roles** page, for the role you want to assign the group to, click the **Actions** button.
 - (c) In the menu, click **Assign Members**.
 - (d) Click the **User Groups** tab.
 - (e) To display the list of available groups to assign to the role, click the **Available** tab.
 - (f) Check the checkbox next to the group, then click the **Update Associations** button.

The group is added to the **Current** tab as a group assigned that user role.



Chapter 9

Setting up Single Sign-On (SSO) for Studio

Studio supports integrating with an SSO system.

About single sign-on and Studio

Overview of the process for configuring SSO with Oracle Access Manager

Configuring the reverse proxy module in OHS

Registering the Webgate with the Oracle Access Manager server

Testing the OHS URL

Configuring Studio to integrate with SSO via Oracle Access Manager

Completing and testing the SSO integration

About single sign-on and Studio

Integrating Studio with single sign-on (SSO) allows your users to be logged in to Studio automatically once they are logged in to your system.

Note that once Studio is integrated with SSO, you cannot create and edit users from within Studio. All users get access to Studio using their SSO credentials. This means that you can no longer use the default administrative user provided with Studio. You will need to make sure that there is at least one SSO user with an Administrator user role for Studio.

The supported method for integrating with SSO is to use Oracle Access Manager, with an Oracle HTTP Server in front of the Studio application server. If you wish to use another third-party SSO solution, it will require additional custom development, and you will need to contact Oracle Support to set up the consulting engagement.

The information in this guide focuses on the details and configuration that are specific to the Studio integration. For general information on installing Oracle Access Manager and Oracle HTTP Server, see the associated documentation for those products.

Overview of the process for configuring SSO with Oracle Access Manager

Here is an overview of the steps for using Oracle Access Manager to implement SSO in Studio.

1. Install Oracle Access Manager 11g, if you haven't already. See the Oracle Access Manager documentation for details.
2. Install Oracle HTTP Server (OHS) 11g. See the Oracle HTTP Server documentation for details.
3. Install OHS Webgate 11g. See the Webgate documentation for details.

4. Create an instance of OHS, and confirm that it is up and running. See the OHS documentation for details.
5. Configure the reverse proxy module for the Studio application server in Oracle HTTP Server. See [Configuring the reverse proxy module in OHS on page 54](#).
6. Install the Webgate module into the Oracle HTTP Server. See [Registering the Webgate with the Oracle Access Manager server on page 55](#).
7. In Studio, configure the LDAP connection for your SSO implementation. See [Configuring the LDAP connection for SSO on page 57](#).
8. In Studio, configure the Oracle Access Manager SSO settings. See [Configuring the Oracle Access Manager SSO settings on page 59](#).
9. Configure Studio's web server settings to use the OHS server. See [Completing and testing the SSO integration on page 60](#).
10. Disable direct access to the Studio application server, to ensure that all traffic to Studio is routed through OHS.

Configuring the reverse proxy module in OHS

You must configure your OHS instance to pass traffic back to Studio as a reverse proxy.

[Reverse proxy configuration for WebLogic Server](#)

Reverse proxy configuration for WebLogic Server

For WebLogic Server, you need to update the file `mod_wl_ohs.conf` to add the logout configuration for SSO.

Here is an example of the file with the `/eid/oam_logout_success` section added:

```
LoadModule weblogic_module    "${ORACLE_HOME}/ohs/modules/mod_wl_ohs.so"
<IfModule weblogic_module>
    WebLogicHost hostName
    WebLogicPort portNumber
</IfModule>

<Location /eid/oam_logout_success>
    PathTrim /eid/oam_logout_success
    PathPrepend /eid/c/portal
    DefaultFileName logout
    SetHandler weblogic-handler
</Location>

<Location />
    SetHandler weblogic-handler
</Location>
```

The `/eid/oam_logout_success` Location configuration is special for Studio. It redirects the default Webgate Logout Callback URL (`/eid/oam_logout_success`) to an application tier logout within Studio. With this configuration, when users sign out of SSO from another application, it is reflected in Studio.

Registering the Webgate with the Oracle Access Manager server

After you have installed the OHS Webgate, you use the remote registration (RREG) tool to register the OHS Webgate with the OAM server.

To complete the registration:

1. Obtain the RREG tarball (`rreg.tar.gz`) from the Oracle Access Manager server.
2. Extract the file to the OHS server.
3. Modify the script `rreg/bin/oamreg.bat` or `oamreg.sh`.

Correct the `OAM_REG_HOME` and `JAVA_HOME` environment variables.

`OAM_REG_HOME` should point to the extracted `rreg` directory created in the previous step.

You may not need to change `JAVA_HOME` if it's already set in your environment.

4. In the `input` directory, create an input file for the RREG tool. The file can include the list of resources secured by this Webgate.

You can omit this list if the application domain already exists.

Here is an example of an input file where the resources have not been set up for the application domain and host in Oracle Access Manager:

```
<?xml version="1.0" encoding="UTF-8"?>
<OAM11GRegRequest>
<serverAddress>http://oamserver.us.mycompany.com:7001</serverAddress>
<hostIdentifier>myserver-1234</hostIdentifier>
<agentName>myserver-1234-webgate</agentName>
<applicationDomain>Information Discovery Studio</applicationDomain>
<protectedResourcesList>
  <resource>/eid</resource>
  <resource>/eid/.../*</resource>
</protectedResourcesList>
<publicResourcesList>
  <resource>/public/index.html</resource>
</publicResourcesList>
<excludedResourcesList>
  <resource>/excluded/index.html</resource>
</excludedResourcesList>
</OAM11GRegRequest>
```

In this example, the resources have already been set up in Oracle Access Manager:

```
<?xml version="1.0" encoding="UTF-8"?>
<OAM11GRegRequest>
<serverAddress>http://oamserver.us.mycompany.com:7001</serverAddress>
<hostIdentifier>myserver-1234</hostIdentifier>
<agentName>myserver-1234-webgate</agentName>
<applicationDomain>Information Discovery Studio</applicationDomain>
</OAM11GRegRequest>
```

In the input file, the parameter values are:

Parameter Name	Description
serverAddress	The full address (<code>http://host:port</code>) of the Oracle Access Manager administrative server. The port is usually 7001.
hostIdentifier	The host identifier string for your host. If you already created a host identifier in the Oracle Access Manager console, use its name here.
agentName	A unique name for the new Webgate agent. Make sure it doesn't conflict with any existing agents in the application domain.
applicationDomain	A new or existing application domain to add this agent into. Each application domain may have multiple agents. An application domain associates multiple agents with the same authentication and authorization policies.

5. Run the tool:

```
./bin/oamreg.sh inband input/inputFileName
```

or

```
.\bin\oamreg.bat inband input\inputFileName
```

For example:

```
bin\oamreg.bat inband
input\my-webgate-input.xml
```

When the process is complete, you'll see the following message:

```
Inband registration process completed successfully! Output artifacts are created in the
output folder.
```

6. Copy the generated output files from the `output` directory to the OHS instance `config` directory (under `webgate/config/`).
7. Restart the OHS instance.
8. Test your application URL via OHS.

It should forward you to the SSO login form.

Check the OAM console to confirm that the Webgate is installed and has the correct settings.

Testing the OHS URL

Before continuing to the Studio configuration, you need to test that the OHS URL redirects correctly to Studio.

To test the OHS URL, use it to browse to Studio.

You should be prompted to authenticate using your SSO credentials.

Because you have not yet configured the Oracle Access Manager SSO integration in Studio, after you complete the authentication, the Studio login page is displayed.

Log in to Studio using an administrator account.

Configuring Studio to integrate with SSO via Oracle Access Manager

In Studio, you configure the LDAP connection and Oracle Access Manager connection settings.

[Configuring the LDAP connection for SSO](#)

[Configuring the Oracle Access Manager SSO settings](#)

Configuring the LDAP connection for SSO

The SSO implementation uses LDAP to retrieve and maintain the user information. For the Oracle Access Manager SSO, you configure Studio to use Oracle Internet Directory for LDAP.

In Studio, to configure the LDAP connection for SSO:

1. From the administrator menu, select **Control Panel**.
2. In the **Control Panel** menu, under **Portal**, click **Settings**.
3. In the **Settings** page menu to the right, click **Authentication**.
4. On the **Authentication** page, click the **LDAP** tab.
5. Check the **Enabled** checkbox. Do not check the **Required** checkbox.

- Under **Default values**, click the **Oracle Internet Directory** radio button, then click **Reset Values**.

Authentication

General	LDAP	Oracle Access Manager SSO	CAS	NTLM	Open SSO
		SiteMinder			

Enabled

Required

Default Values

- Apache Directory Server
- Fedora Directory Server
- Microsoft Active Directory Server
- Novell eDirectory
- OpenLDAP
- Oracle Internet Directory
- Other Directory Server

Reset Values

- Configure the LDAP connection, users, and groups as described in [Configuring the LDAP settings and server on page 44](#).
- To save the LDAP connection information, click **Save**.
- Configure the application roles for your user groups as described in [Assigning roles based on LDAP user groups on page 51](#).
- Make sure that the password policy is configured to not require users to change their password. See [Configuring the Studio password policy when using LDAP on page 49](#).

Configuring the Oracle Access Manager SSO settings

After you configure the LDAP connection for your SSO integration, you configure the Oracle Access Manager SSO settings.

The settings are on the **Oracle Access Manager SSO** tab on the **Authentication** page.

Authentication

General LDAP **Oracle Access Manager SSO** CAS NTLM Open SSO
StelMinder

Enabled

Import from LDAP ?

User Header
OAM_REMOTE_USER

Logout URL
http://OAMSERVER:14100/oam

To configure the SSO settings:

1. From the administrator menu, select **Control Panel**.
2. In the **Control Panel** menu, under **Portal**, click **Settings**.
3. In the **Settings** page menu to the right, click **Authentication**.
4. On the **Authentication** page, click the **Oracle Access Manager SSO** tab.
5. Check the **Enabled** checkbox.
6. Check the **Import from LDAP** checkbox.
7. Leave the default user header `OAM_REMOTE_USER`.

- In the **Logout URL** field, provide the URL to navigate to when users log out.
Make sure it is the same logout redirect URL you have configured for the Webgate:

The screenshot shows the configuration page for 'slc02xfd-webgate'. The 'Logout Redirect URL' field is highlighted with a red box and contains the URL 'http://appdev-ws4-rhel3-p15.us.or'. Other fields include 'Logout Callback URL' (/oam_logout_success), 'Logout Target URL', 'User Defined Parameters' (proxySSLHeaderVar=IS_SSL, URLInUTF8Format=true, client_request_retry_attempts=1, inactiveReconfigPeriod=10), 'Sleep for' (60), 'Cache Pragma Header' (no-cache), 'Cache Control Header' (no-cache), 'Debug' (unchecked), 'IP Validation' (unchecked), 'Deny On Not Protected' (checked), and 'Allow Management Operations' (unchecked).

For the logout URL, you can add an optional `end_url` parameter to redirect the browser to a final location after users sign out. To redirect back to Studio, configure `end_url` to point to the OHS host and port.

For example:

```
http://oamserver.us.mycompany.com:14100/oam/server/logout?end_url=http://
/studiohost.us.company.com:7777/
```

- To save the configuration, click **Save**.

Completing and testing the SSO integration

The final step in setting up the SSO integration is to add the OHS server host name and port to `portal-ext.properties`.

To complete and test the SSO configuration:

- In `portal-ext.properties`, add the following lines:

```
web.server.host=ohsHostName
web.server.http.port=ohsPortNumber
```

Where:

- `ohsHostName` is the fully qualified domain name (FQDN) of the server where OHS is installed. The name must be resolvable by Studio users.

For example, you would use `webserver01.company.com`, and not `webserver01`.

You need to specify this even if OHS is on the same server as Studio.

- *ohsPortNumber* is the port number used by OHS.

2. Restart Studio.

Make sure to completely restart the browser to remove any cookies or sessions associated with the Studio user login you used earlier.

3. Navigate to the Studio URL. The Oracle Access Manager SSO form is displayed.

4. Enter your SSO authentication credentials.

You are logged in to Studio.

As you navigate around Studio, make sure that the browser URL continues to point to the OHS server and port.

Part III

Configuring Available Data for Applications



Managing Endeca Server Connections

Each application is associated with an Endeca Server connection.

[About Endeca Server connections](#)

[Using the Endeca Servers page to manage Endeca Server connections](#)

[Syntax for an Endeca Server connection definition](#)

[Setting up shared Endeca Server connections](#)

About Endeca Server connections

An Endeca Server connection represents a pointer to a specific data domain on an Endeca Server.

Each Endeca Server connection is a JSON string that contains:

- Connection information for the Endeca Server
- The name of the specific data domain to connect to. The data domain must be configured to use data sets (referred to as collections in Endeca Server). Without data sets, the data cannot be used in Studio applications.
- Optional settings to restrict who can view the data
- Optional settings to restrict who can connect an application to that Endeca Server connection

Each Studio application is connected to a specific Endeca Server connection.

Using the Endeca Servers page to manage Endeca Server connections

The **Endeca Servers** page, available from the Studio **Control Panel**, provides access to Endeca Server connections.

[About the Endeca Servers page](#)

[Displaying the Endeca Servers page](#)

[Adding an Endeca Server connection](#)

[Editing an Endeca Server connection](#)

[Deleting an Endeca Server connection](#)

[Testing an Endeca Server connection](#)

About the Endeca Servers page

The **Endeca Servers** page displays the list of current Endeca Server connections, including Endeca Server connections created using the Provisioning Service from file uploads or the **Data Source Library**.

From the **Endeca Servers** page, you can:

- See whether each Endeca Server connection is available, read-only, and SSL-enabled
- Add a new Endeca Server connection
- Edit the definition of an Endeca Server connection
- Remove an Endeca Server connection
- Test each Endeca Server connection

Displaying the Endeca Servers page

The **Endeca Servers** page is available from the **Information Discovery** section of the **Control Panel** menu.

To display the **Endeca Servers** page:

1. From the administrator menu, select **Control Panel**.
2. In the **Information Discovery** section of the **Control Panel** menu, click **Endeca Servers**.

Each Endeca Server connection is represented by a block on the **Endeca Servers** page.

Endeca Servers

The screenshot displays the 'Endeca Servers' page with a '+ New Connection' button at the top right. The page lists several server connections, each with a header bar containing the connection name and ID, and a row of buttons for 'Test Connection', 'Edit', and 'Delete'. Below the header, the details for each connection are shown, including the Oracle Endeca Server host and port, and the Data Domain Name. The access type (Read-only or Read/write) is indicated on the right side of each entry, along with a lock icon. The 'unavailable' connection is highlighted in red and includes a red error icon and a message: 'Connection failure at 9/26/13 2:11 PM'.

Connection Name (ID)	Oracle Endeca Server	Data Domain Name	Access Type
adventure works (adventureworks)	busg102.us.oracle.com:7002	adventure_works	Read-only
adventure works 2 (adventureworks_2)	busg102.us.oracle.com:7002	adventure_works_2	Read-only
adventure works 2 rw (adventureworks_2_rw)	busg102.us.oracle.com:7002	adventure_works_2_rw	Read/write
Supplier Spend (federal)	busg106.us.oracle.com:7001	federal	Read-only
unavailable (unavailable)	appdev-x64-rhel5-p2.us.oracle.com:1234	unavailable	Read-only

The block contains the following information:

- Endeca Server connection name and ID
- Oracle Endeca Server host name and port
- Data domain name
- Whether the Endeca Server connection is read-write or read-only
- Whether the Endeca Server connection is SSL-enabled (indicated by a lock icon in the bottom right corner)
- Whether the Endeca Server connection is unavailable. Unavailable Endeca Server connections cannot be used by applications.

Note that an Endeca Server connection is also listed as unavailable when the associated data domain is currently idle.

The data domain wakes up automatically when you test the connection, and when users access an application that uses the Endeca Server connection.

However, Endeca Server connections that are tied to idle data domains are not available to users who are creating a new application. You need to wake up the data domain before users can create a new application from it.

Adding an Endeca Server connection

From the **Endeca Servers** page, you can add an Endeca Server connection. You must provide an Endeca Server connection ID and the Endeca Server connection definition.

To add a new Endeca Server connection:

1. Click the **New Connection** button.

The **Endeca Server Connection Definition** dialog is displayed. The definition text area contains an empty template definition with the key fields.

The screenshot shows a dialog box titled "Endeca Server Connection Definition". At the top, there is a "Connection ID:" label followed by an empty text input field. To the right of the input field is a red error message: "Connection ID is a required field." Below the input field is a text area containing a JSON template:

```
{
  "name": "",
  "server": "",
  "port": "",
  "dataDomainName": ""
}
```

 At the bottom of the dialog are three buttons: "Validate", "Cancel", and "Save".

2. In the **Connection ID** field, type the ID of the Endeca Server connection.
The ID must be unique, and can only contain letters, numbers, underscores, and hyphens.
3. In the text area, type the definition of the Endeca Server connection.

The definition must at least specify:

- The name of the server hosting the Endeca Server
- The port number for the Endeca Server
- The name of the data domain

For details on the definition syntax for an Endeca Server connection, see [Syntax for an Endeca Server connection definition on page 68](#).

Also remember that in order to be able to use a data domain in a Studio application, it must be configured with data sets (referred to as collections in Endeca Server).

4. To validate the definition, click **Validate**.

A message is displayed indicating whether the definition is valid and whether Studio was able to connect to the Endeca Server.

5. To save the new Endeca Server connection, click **Save**.

If the definition is not valid, then the Endeca Server connection is flagged as unavailable on the **Endeca Servers** page, and cannot be selected for an application.

Editing an Endeca Server connection

From the **Endeca Servers** page, you can edit the definition of an existing Endeca Server connection.

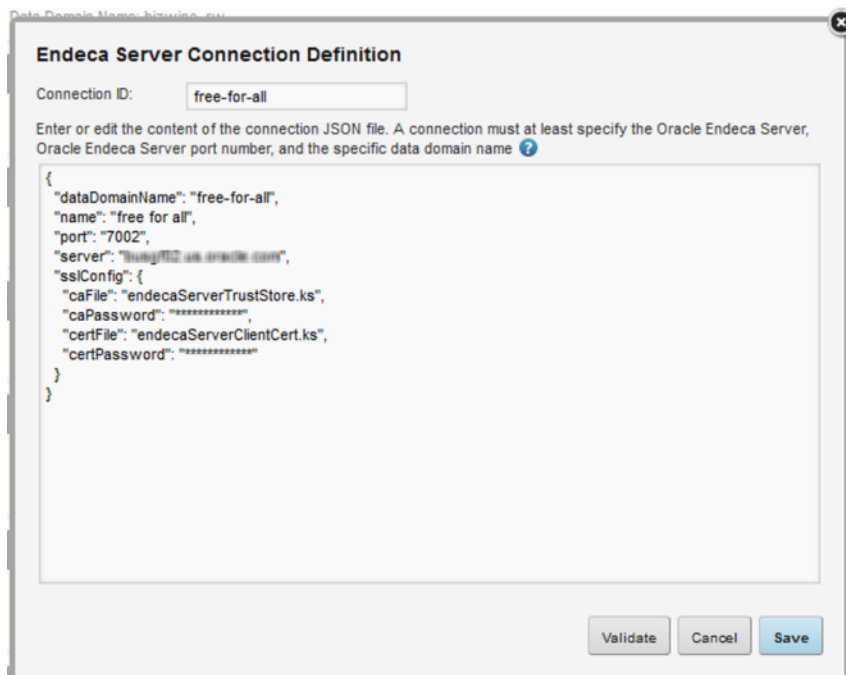
For Endeca Server connections you created manually, if you change the ID, you will need to update any application that uses the Endeca Server connection.

For Endeca Server connections generated by the Provisioning Service, you cannot change the ID, which is generated automatically. If you change the ID of the Endeca Server connection, then Studio can no longer connect to it. You would then have to remove and recreate the Endeca Server connection.

If the Endeca Server connection is connected to a secured Endeca Server, then when you edit the definition, you must replace the masking asterisks with the actual SSL passwords.

To edit an Endeca Server connection:

1. On the **Endeca Servers** page, click the **Edit** button for the Endeca Server connection.
2. On the **Endeca Server Connection Definition** dialog, make the changes to the Endeca Server connection.



If the Endeca Server connection is connected to a secured Endeca Server, then you must replace the masking asterisks with the actual SSL passwords. If you save the Endeca Server connection without replacing the passwords, then the asterisks are saved as the password value.

3. To validate that the updated definition is valid, click **Validate**.

A message is displayed indicating whether the definition is valid and whether Studio was able to connect to the Endeca Server.

4. To save the changes to the Endeca Server connection, click **Save**.

If the definition is not valid, then the Endeca Server connection is flagged as invalid on the **Endeca Servers** page, and cannot be selected for an application.

Deleting an Endeca Server connection

From the **Endeca Servers** page, you can delete an Endeca Server connection. Before deleting an Endeca Server connection, make sure it is not being used by an application. Do not use this option to delete Endeca Server connections generated by the Provisioning Service.

To delete an Endeca Server connection:

1. On the **Endeca Servers** page, click the **Delete** button for the Endeca Server connection you want to delete.

You are prompted to confirm that you want to delete the Endeca Server connection.

Do not use this option to delete an Endeca Server connection that was generated by the Provisioning Service, because Studio cannot perform the proper cleanup on the Provisioning Service. These types of Endeca Server connections are deleted automatically when the associated application is deleted from the **Discovery Applications** page.

2. To delete the Endeca Server connection, click **Delete**.

If any applications were using the deleted Endeca Server connection, then the components can no longer display data, and the applications will need to be updated to select a different Endeca Server connection.

Testing an Endeca Server connection

Each Endeca Server connection is connected to an Endeca Server data domain. The **Endeca Servers** page allows you to test the connections, so that you can verify that the Endeca Server connection is available.

The connection test can also be used to wake up an idle Endeca Server data domain.

To test the connection to the Endeca Server domain, click the **Test Connection** button for the Endeca Server connection. A message displays on the Endeca Server connection block indicating whether the test was successful.

Syntax for an Endeca Server connection definition

The Endeca Server connection definition JSON string must contain at minimum the connection information for the Endeca Server plus the name of the specific data domain on that server. Additionally, you can add security restrictions.

[Escaping special characters in Endeca Server connection definitions](#)

[Basic Endeca Server connection properties](#)

[Configuring role-based security for viewing Endeca Server connection data](#)

[Connecting an Endeca Server connection to a secured Endeca Server](#)

[Connecting an Endeca Server connection to an Endeca Server cluster](#)

Limiting who can connect an application to the Endeca Server connection

Escaping special characters in Endeca Server connection definitions

JSON requires you to use a backslash (\) to escape some special characters.

The characters that must be escaped are:

- Backspace
- Form feed
- New line
- Carriage return
- Tab
- Vertical tab
- Apostrophe or single quote
- Double quote
- Backslash

Basic Endeca Server connection properties

Each Endeca Server connection is assigned an identifier. The Endeca Server connection definition is written as a JSON string, and must contain the Endeca Server server name, port number, and the data domain name. Optionally, the definition can include a context path, name, and a description.

Endeca Server connection identifier

Every Endeca Server connection is assigned an identifier, used internally to manage the Endeca Server connection.

The identifier:

- Must begin with a letter (A-Z or a-z)
- Can contain letters, numbers, underscores, and hyphens. No other characters or spaces are permitted.

For Endeca Server connections generated by the Provisioning Service, the identifier is generated automatically and cannot be changed.

Host and data domain name settings (required, except for the context path)

Every Endeca Server connection definition must at a minimum contain the following settings for the Endeca Server host and data domain name:

Setting	Description
<code>server</code>	The name of the server on which the Endeca Server is running.
<code>port</code>	The port on which the Endeca Server is listening.

Setting	Description
dataDomainName	The name of the specific data domain on the Endeca Server.

Here is an example of the most basic Endeca Server connection definition, with just the host information and data domain name.

```
{
  "server": "server01.lab.acme.com",
  "port": "15000",
  "dataDomainName": "acmeDB"
}
```

If the Endeca Server is installed on a context path other than `endeca-server`, then you need to add a setting to provide the context path:

Setting	Description
contextPath	The context path for the Endeca Server. If this setting is not present, then the context path defaults to <code>endeca-server</code> , which is Endeca Server's default context path. To specify a root context, set the value to either "" or "/".

Here is an example of a basic Endeca Server connection definition with the context path specified.

```
{
  "server": "server01.lab.acme.com",
  "port": "15000",
  "dataDomainName": "acmeDB",
  "contextPath": "/acme/context/path/"
}
```

Endeca Server connection name and description settings

The Endeca Server connection definition can optionally contain name and description settings:

Setting	Description
name	The name of the Endeca Server connection. This is the value displayed on the Studio user interface, including on the Endeca Servers page and in Endeca Server connection drop-down lists. If you do not provide a value for <code>name</code> , then the ID is used.
description	A longer description of the Endeca Server connection. Used for logging and debugging.

For example:

```
{
  "server": "server01.lab.acme.com",
  "port": "15000",
```

```

"dataDomainName": "acmeDB",
"name": "Wine Transactions",
"description": "Transaction data for the Midwest Region"
}
    
```

Configuring role-based security for viewing Endeca Server connection data

You can also configure an Endeca Server connection to control who can view the data based on user roles.

The Endeca Server connection settings related to role-based security are:

Setting	Description
<code>securityEnabled</code>	<p>Whether to enable the security filters for queries to the Endeca Server connection.</p> <p>If set to "true", then the Endeca Server connection uses the filters configured under <code>securityFilters</code>.</p>
<code>securityFilters</code>	<p>Defines all of the security filters to be used by the Endeca Server connection. Each filter is given a name to map to the security roles.</p> <p>For security filters, <code>DataSourceFilters</code> are the only supported type of filter. For each filter, you specify:</p> <ul style="list-style-type: none"> <code>class</code> - the full path to the <code>DataSourceFilter</code> class. <code>filterString</code> - the EQL snippet containing the filter information. This is essentially the content of a <code>WHERE</code> clause for an EQL statement. <code>viewKey</code> - The key name (not the display name) of the data set against which to execute the EQL.
<code>rolePermissionsMultiOr</code>	<p>For users who have more than one security role, whether to use logical OR to combine the filters from each role into a single, combined security role filter.</p> <p>If set to "true", then logical OR is used, and users have access to data that matches at least one of the filters for their security roles.</p> <p>If set to "false" (the default value), then logical AND is used, and users only have access to data that matches all of the filters associated with all of their security roles.</p> <p>Note that if logical OR is used, it is only used to combine filters from different security roles. The filters from each individual role are still applied using logical AND before they are combined with the filters from the other roles.</p> <p>Data set base filters are also applied using logical AND.</p>

Setting	Description
rolePermissions	<p>Maps the user roles to the security filters.</p> <p>For each mapping, the format is:</p> <pre data-bbox="602 415 1448 449">"<role name>" : [<filter list>]</pre> <p>where:</p> <ul style="list-style-type: none"> • <role name> is the name of the user role. • <filter list> is a comma-separated list of filter names to apply for the specified role. Each name is in quotes. For example, ["filter1", "filter2", "filter3"].

Example of an Endeca Server connection with security filtering

In the following example, users with the role "French Wine" can only see data from the Bordeaux and Burgundy regions, while users with the role "Austrian Wine" can only see data from the Austria, Burgenland, and Steiermark regions.

Because `rolePermissionsMultiOr` is set to true, users who have both of these roles can view records from any of the five regions.

```

{
  "server": "server01.lab.acme.com",
  "port": "15000",
  "dataDomainName": "acmeDB",
  "name": "European Wines",
  "description": "Sales transactions for European wines",
  "securityEnabled": "true",
  "securityFilters": {
    "frenchFilter": {
      "class": "com.endeca.portal.data.functions.DataSourceFilter",
      "filterString": "Region='Bordeaux' OR Region='Burgundy'",
      "viewKey": "Wines"
    },
    "austrianFilter": {
      "class": "com.endeca.portal.data.functions.DataSourceFilter",
      "filterString": "Region='Austria' OR Region='Burgenland' OR Region='Steiermark'",
      "viewKey": "Wines"
    }
  },
  "rolePermissionsMultiOr": "true",
  "rolePermissions": {
    "French Wine": ["frenchFilter"],
    "Austrian Wine": ["austrianFilter"]
  }
}

```

Connecting an Endeca Server connection to a secured Endeca Server

When you install Endeca Server, the default option is to use SSL to secure it. To connect to a secured Endeca Server, you copy the Endeca Server certificate files to Studio. In the Endeca Server connection configuration, you then include the certificate file names and passwords.

To connect to a secured Endeca Server:

1. Stop Studio.

2. From the Endeca Server \$DOMAIN_HOME/config/ssl directory, copy the following files:
 - endecaServerClientCert.ks
 - endecaServerTrustStore.ks
3. Place the files into the endeca-data-sources directory in the Studio home directory.
4. In the Endeca Server connection definition, add the sslConfig setting, which contains the following settings:

Setting	Description
caFile	The name of the truststore file. For the default secured Endeca Server configuration, the file is endecaServerTrustStore.ks.
caPassword	The password for the truststore file. You need to obtain the password from whoever installed the Endeca Server and generated the certificates. Note that on the Data Source Definition dialog, once you save the Endeca Server connection, the value of caPassword is masked as *****. The value also is encrypted in the Studio database. When you edit the Endeca Server connection, you must re-type the actual password value before saving. Otherwise, Studio uses the masking asterisks as the password value.
certFile	The name of the keystore file. For the default secured Endeca Server configuration, the file is endecaServerClientCert.ks.
certPassword	The password for the keystore file. You need to obtain the password from whoever installed the Endeca Server and generated the certificates. Note that on the Data Source Definition dialog, once you save the Endeca Server connection, the value of certPassword is masked as *****. The value also is encrypted in the Studio database. When you edit the Endeca Server connection, you must re-type the actual password value before saving. Otherwise, Studio uses the masking asterisks as the password value.

For example:

```
"sslConfig": {
"caFile": "endecaServerTrustStore.ks",
"caPassword": "*****",
"certFile": "endecaServerClientCert.ks",
"certPassword": "*****"
}
```

5. Restart Studio.

Example of an Endeca Server connection connected to secured Endeca Server

The following Endeca Server connection connects to a secured Endeca Server.

```
{
  "server": "server01.lab.acme.com",
  "port": "7002",
  "dataDomainName": "acmeDB",
  "sslConfig": {
    "caFile": "endecaServerTrustStore.ks",
    "caPassword": "*****",
    "certFile": "endecaServerClientCert.ks",
    "certPassword": "*****"
  },
  "name": "Wine Transactions",
  "description": "Transaction data for the Midwest region"
}
```

Connecting an Endeca Server connection to an Endeca Server cluster

The Endeca Server can use a clustered configuration. When configuring an Endeca Server cluster to use with Studio, you should always have a load balancer in front of the cluster. For details on Endeca Server clustering, including how to set up a load balancer in front of the cluster, see the *Oracle Endeca Server Clustering Guide*.

Endeca Server connection definition settings for connecting to a cluster

When configuring the definition for an Endeca Server connection that is connecting to an Oracle Endeca Server cluster, the relevant settings are:

Endeca Server Connection Setting	Description
server	Required. The server for the load balancer.
port	Required. The port number for the load balancer.
dataDomainName	Required. The name of the data domain.

Endeca Server Connection Setting	Description
sslConfig	<p>Optional. If applicable, the SSL settings for connecting to the load balancer. Includes the following settings:</p> <ul style="list-style-type: none"> • caFile - The truststore file. • caPassword - The truststore password. <p>Note that on the Data Source Definition dialog, once you save the Endeca Server connection, the value of caPassword is masked as *****. The value also is encrypted in the Studio database.</p> <ul style="list-style-type: none"> • certFile - The name of the keystore file. • certPassword - The keystore password. <p>Note that on the Data Source Definition dialog, once you save the Endeca Server connection, the value of certPassword is masked as *****. The value also is encrypted in the Studio database.</p>

Example of an Endeca Server connection connected to an Endeca Server cluster

The following Endeca Server connection is configured to connect to a cluster.

```
{
  "server": "loadbalancer1.acme.com",
  "port": "7002",
  "dataDomainName": "acmeDB",
  "sslConfig": {
    "caFile": "truststore.ks",
    "caPassword": "*****",
    "certFile": "keystore.ks",
    "certPassword": "*****"
  },
  "name": "Sales Data"
}
```

Limiting who can connect an application to the Endeca Server connection

When configuring an Endeca Server connection, you can specify which users and roles can connect an application to it.

Users who are not allowed to bind an application to the Endeca Server connection do not see it in the list of available Endeca Server connections.

Note that these settings do not control who can view the data in an application. To control the displayed data, you would use filters or role-based security. The settings also do not control who can view applications that are linked to this Endeca Server connection. That access is based on the application type and membership.

Users who are Studio administrators (have the Administrator user role), can always connect to any Endeca Server connection, including Endeca Server connections created using the Provisioning Service. They are not bound by the restriction settings.

Only Studio administrators can create a new application from an Endeca Server connection created by the Provisioning Service. These Endeca Server connections are not available to any other users, not even to the user who created the original application.

When you create an Endeca Server connection from the **Endeca Servers** page, by default access to connect applications to the Endeca Server connection is not restricted.

To limit the access, you use the following settings:

Setting	Description
<code>restrictedToUsers</code>	<p>Comma-separated list of screen names of Studio users who can connect an application to the Endeca Server connection.</p> <p>For example:</p> <pre>restrictedToUsers:["jsmith", "mbrown"]</pre>
<code>restrictedToRoles</code>	<p>Comma-separated list of Studio user roles (not application roles) that can connect an application to the Endeca Server connection.</p> <p>For example:</p> <pre>restrictedToRoles:["Administrator", "Power User"]</pre>

Users can use the Endeca Server connection for an application if they are either in the user list or have a role in the role list. For example, if a user is in the `restrictedToUsers` list, then they can use the Endeca Server connection even if they do not have a role that is in the `restrictedToRoles` list. If a user has a role in the `restrictedToRoles` list, then they can use the Endeca Server connection even if they are not in the `restrictedToUsers` list.

Example of an Endeca Server connection that restricts who can connect an application to it

For the following Endeca Server connection, only users with the Administrator user role can connect an application to it.

The users `jsmith` and `rjones` can also connect an application to the Endeca Server connection, even if they do not have the Administrator user role.

```
{
  "server":"server01.lab.acme.com",
  "port":"15000",
  "dataDomainName":"acmeDB",
  "restrictedToUsers":["jsmith", "rjones"],
  "restrictedToRoles":["Administrator"]
}
```

Setting up shared Endeca Server connections

When you allow users to create applications from shared Endeca Server connections, you probably do not want them to make changes to the configuration of views and attribute groups.

Before users are using the Endeca Server connection to create applications, you need to restart the Endeca Server data domain as read-only.

To populate the views and groups before making the Endeca Server connection read-only, you can:

- Use Integrator ETL to ingest the views and attribute groups
- Use a dummy application in Studio to configure views and attribute groups

Using Integrator ETL to ingest views and attribute groups

For details on ingesting views and attribute groups into an Endeca Server data domain, see the *Integrator ETL User's Guide*.

If the data domain has more than one data set, you can also ingest refinement rules to link attributes.

After you finish ingesting the data:

1. Start the Endeca Server domain as read-only.
2. Use the **Endeca Servers** page to create the Endeca Server connection.

Using Studio to configure views and attribute groups

To use Studio to configure views and attribute groups for an Endeca Server domain:

1. In Studio, use the **Endeca Servers** page to create the Endeca Server connection to the Endeca Server domain.

Configure the connection to restrict who can create applications from that Endeca Server connection. You would probably want to restrict application either to yourself or to the Administrator user role.

2. Use that Endeca Server connection to create a new Studio application.

For information on how to create a new application from a shared Endeca Server connection, see the *Studio User's Guide*.

3. From that application, use the **Application Settings** page to configure the views and attribute groups.

If the Endeca Server domain contains more than one data set, then you can also configure refinement rules to link attributes.

For information on how to configure views, attribute groups, and refinement rules, see the *Studio User's Guide*.

4. Restart the Endeca Server domain as read-only.
5. From the **Endeca Servers** page, update the Endeca Server connection to remove the restriction on creating applications from it.



Chapter 11

Configuring the Connection to the Provisioning Service

In order for users to be able to create data sets using by uploading files or using the **Data Source Library**, you must first configure the connection to the Provisioning Service.

If the connection configuration is not valid, then when users create an application, the file upload and **Data Source Library** options are not displayed. You also cannot create available data sources in the **Data Source Library**.

The syntax for configuring the connection is similar to the syntax for defining an Endeca Server connection.

The general connection settings are:

Setting	Description
server	The name of the server on which the Provisioning Service is running.
port	The port on which the Provisioning Service is listening.

For example:

```
{
  "server": "ps.us.acme.com",
  "port": "7004"
}
```

If the Provisioning Service is installed on a context path other than `endeca-server`, then you need to add a setting to provide the context path:

Setting	Description
contextPath	The context path for the Provisioning Service. If this setting is not present, then the context path defaults to <code>eid-ps</code> , which is the Provisioning Service's default context path. To specify a root context, set the value to either "" or "/".

For example:

```
{
  "server": "ps.us.acme.com",
  "port": "7004",
  "contextPath": "my-ps-path"
}
```

By default, the Provisioning Service has SSL enabled, and the configuration must include the `sslConfig` setting, which contains the following settings:

Setting	Description
<code>caFile</code>	<p>The name of the truststore file for the SSL connection to the Provisioning Service.</p> <p>This is the truststore file from the secured Endeca Server configuration. For the default configuration, the file is <code>endecaServerTrustStore.ks</code>.</p>
<code>caPassword</code>	<p>The password for the truststore file for the SSL connection to the Provisioning Service.</p> <p>This is the password generated during the Endeca Server installation.</p> <p>Note that once you save the Provisioning Service configuration, the value of <code>caPassword</code> is masked as <code>*****</code>. The value also is encrypted in the Studio database.</p> <p>When you edit the Provisioning Service connection, you must re-type the actual password value before saving. Otherwise, Studio uses the masking asterisks as the password value.</p>
<code>certFile</code>	<p>The name of the keystore file for the SSL connection to the Provisioning Service.</p> <p>This is the keystore file from the secured Endeca Server configuration. For the default configuration, the file is <code>endecaServerClientCert.ks</code>.</p>
<code>certPassword</code>	<p>The password for the keystore file for the SSL connection to the Provisioning Service.</p> <p>This is the password generated during the Endeca Server installation.</p> <p>Note that once you save the Provisioning Service configuration, the value of <code>certPassword</code> is masked as <code>*****</code>. The value also is encrypted in the Studio database.</p> <p>When you edit the Provisioning Service connection, you must re-type the actual password value before saving. Otherwise, Studio uses the masking asterisks as the password value.</p>

For example:

```
{
  "server": "ps.us.acme.com",
  "port": "7004",
  "contextPath": "my-ps-path",
  "sslConfig": {
    "caFile": "endecaServerTrustStore.ks",
    "caPassword": "*****",
    "certFile": "endecaServerClientCert.ks",
    "certPassword": "*****"
  }
}
```

The **Control Panel** includes a **Provisioning Service** page you use to configure the connection.

To configure the Provisioning Service connection:

1. From the administrator menu, select **Control Panel**.
2. In the **Control Panel** menu, click **Provisioning Service**.
3. On the **Provisioning Service** page, update the placeholder configuration with the connection information for your Provisioning Service.

Provisioning Service

Enter or edit the JSON string to configure the connection to the Provisioning Service. You must at least specify the server and port number. ?

```
{  
  "port": "7003",  
  "server": "tbug104.us.oracle.com"  
}
```

Save

4. Click **Save**.



Chapter 12

Managing Available Sources of Application Data

The **Data Source Library** allows administrators to configure available sources of data from Oracle BI servers or JDBC connections to use to create application data sets.

[About data sources](#)

[Displaying information about data sources on the Data Source Library](#)

[Adding and editing data sources in the Data Source Library](#)

[Configuring access to a data source in the Data Source Library](#)

[Removing a data source from the Data Source Library](#)

About data sources

Studio administrators can use the **Data Source Library** to create data sources that can be used to create data sets for an application.

Data sources can come from:

Data Source Type	Description
Oracle BI	Data from an Oracle BI (Business Intelligence) server. When users create a data set from an Oracle BI data source, they are prompted to provide Oracle BI credentials. Note that when Studio retrieves Oracle BI data, it uses the security configured in the RPD file, not the security used when users browse the data on the Oracle BI user interface. The permissions in the RPD file should match the security used on the Oracle BI user interface. If the permissions are not replicated in the RPD file, then users will have unrestricted access to the data.
JDBC connection	Data from a relational database.

When you create and edit a data source, you also provide the default configuration for the available attributes.

You can only create and manage data sources from the **Data Source Library** if:

- The Provisioning Service has been installed.
- The connection to the Provisioning Service has been configured. See *[Configuring the Connection to the Provisioning Service on page 77](#)*.

Displaying information about data sources on the Data Source Library

The **Data Source Library** is available from the **Information Discovery** section of the **Control Panel**.

The **Data Source Library** contains the list of data sources.

Data Source Library

Define and manage connections to various data sources from which Studio users can create data sets for their discovery applications.

For each data source, you define the data that can be obtained, set limits on the amount of data that can be uploaded into a data set, and control which users have access to the data source.

Data Source Name	Source Type	Total Size (estimated)	Max Per Upload	Access
▶ Airline Traffic	Oracle BI	1 records	1000000	Available to All
▶ HR Data	JDBC	107 records	1000000	Available to All

For each data source, the list initially displays:

- The name of the data source.
- The source type for the data source. Indicates whether the data is from Oracle BI or from a JDBC connection.
- An estimate of the total number of available records in the data source.
- The maximum number of records that can be added at a time from the data source to a data set.
- Whether the ability to create data sets from this data source is restricted to specific roles, users, or user groups.

To display additional information about a data source, click the expand icon.

Data Source Name	Source Type	Total Size (estimated)	Max Per Upload	Access
▼ Airline Traffic	Oracle BI	1 records	1000000	Available to All
Information on air traffic Host : busapp01b-us.oracle.com Port: 9703 Subject Area: X - Airlines Traffic Presentation Table: Traffic Facts				
▼ HR Data	JDBC	107 records	1000000	Available to All
Data from the HR database URL: jdbc:oracle:thin:@ohcloud-hdp-02.us.oracle.com:1521:emp				

The expanded information includes:

- The description of the data source
- The data source location (host and port for Oracle BI, URL for JDBC)
- For Oracle BI data sources, the subject area and presentation table

Adding and editing data sources in the Data Source Library

From the **Data Source Library**, you can add and edit data sources.

[Adding and editing data sources](#)

[Selecting the data to use in a data source](#)

[Configuring the attributes in a data source](#)

Adding and editing data sources

From the **Data Source Library**, you can add a new data source.

To add a new data source:

1. To add a new data source, on the **Data Source Library**, click **New Data Source**.
2. To edit an existing data source, click its edit icon.
3. On the data source configuration dialog, in the **Data source name** field, type the name of the new data source.

New Data Source (Step 1 of 3)

Provide a name and description for your data source, and set the number of records users can add to a data set at any one time.

Data source name:

Description (Optional):

Maximum records per upload: ?

Define Connection

Identify the source system and the necessary connections to pull data from.

Data source type: Oracle BI JDBC

Save & Exit Cancel Next

4. In the **Description** field, type a description of the data source.
5. In the **Maximum records per upload** field, type the maximum number of records that can be added at one time to a data set created from this data source.

When users create a data set or add records to a data set from this data source, if the number of records to add is greater than the maximum, the user must adjust the filters to reduce the number of records.

6. Under **Define connection**, click the radio button for the type of data source you are creating.

7. For an Oracle BI data source:

- In the **Host** field, type the host name for the Oracle BI server.
- In the **Port** field, type the port number for the Oracle BI server.
- In the **User name** field, type the user name to use to connect to the Oracle BI server.
- In the **Password** field, type the password to use to connect to the Oracle BI server.

8. For a JDBC data source:

- In the **URL** field, type the URL for the database.
- In the **Properties** field, if needed, update the additional parameters needed for the connection.

By default, the properties configure the JDBC connection to work in ANSI mode. This is required for some databases to support allowing users to filter the data when creating data sets from the data source.

The default properties also set `useLegacyDatetimeCode` to false. This property is needed in order for date/time values from MySQL to work properly.

If the data for this data source does not include any date/time values, then remove the `useLegacyDatetimeCode` property.

If the data for this data source does include date/time values, and uses MySQL on a Linux machine, then you must also add the following property:

```
serverTimezone=UTC
```

When you add properties, you must insert a line break between each property-value pair.

- (c) In the **User Name** field, type the user name to use to connect to the database.
- (d) In the **Password** field, type the password to use to connect to the database.

9. Click **Next**.

Studio attempts to connect to the data using the credentials provided. If the connection is successful, then:

- For an Oracle BI data source, the **Select Data Table** page is displayed.
- For a JDBC data source, the **Data Source Definition** page is displayed.

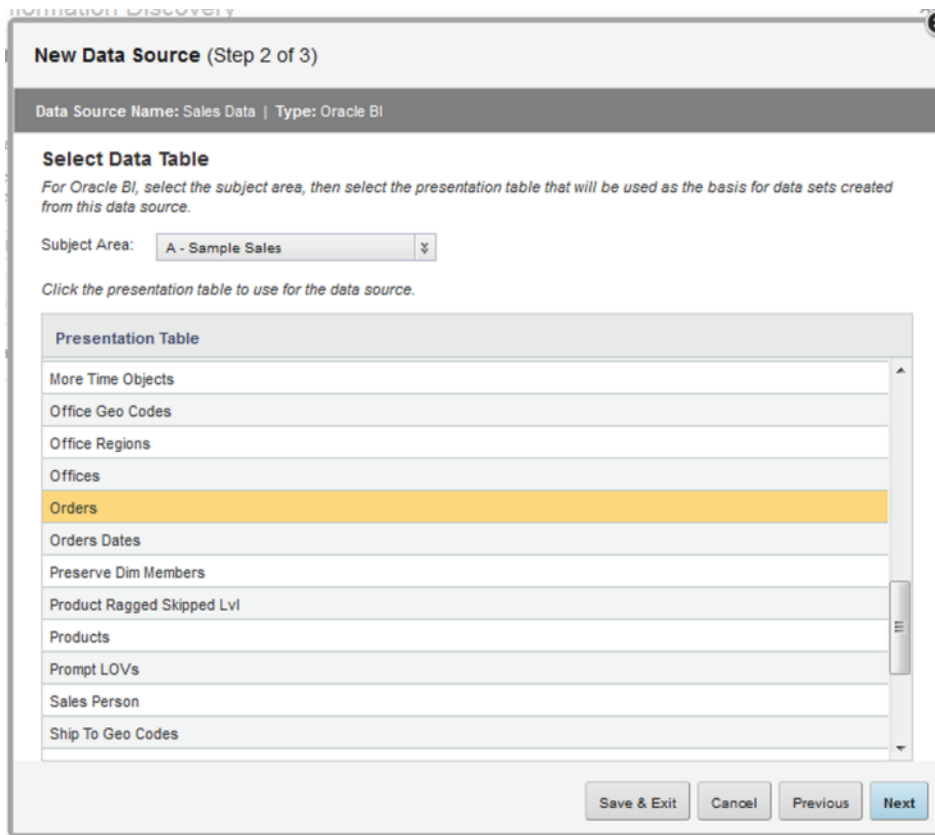
See [Selecting the data to use in a data source on page 86](#).

Selecting the data to use in a data source

After you configure the description and connection information for a data source, you select the specific data to include.

To select the data for a data source:

1. For an Oracle BI data source:



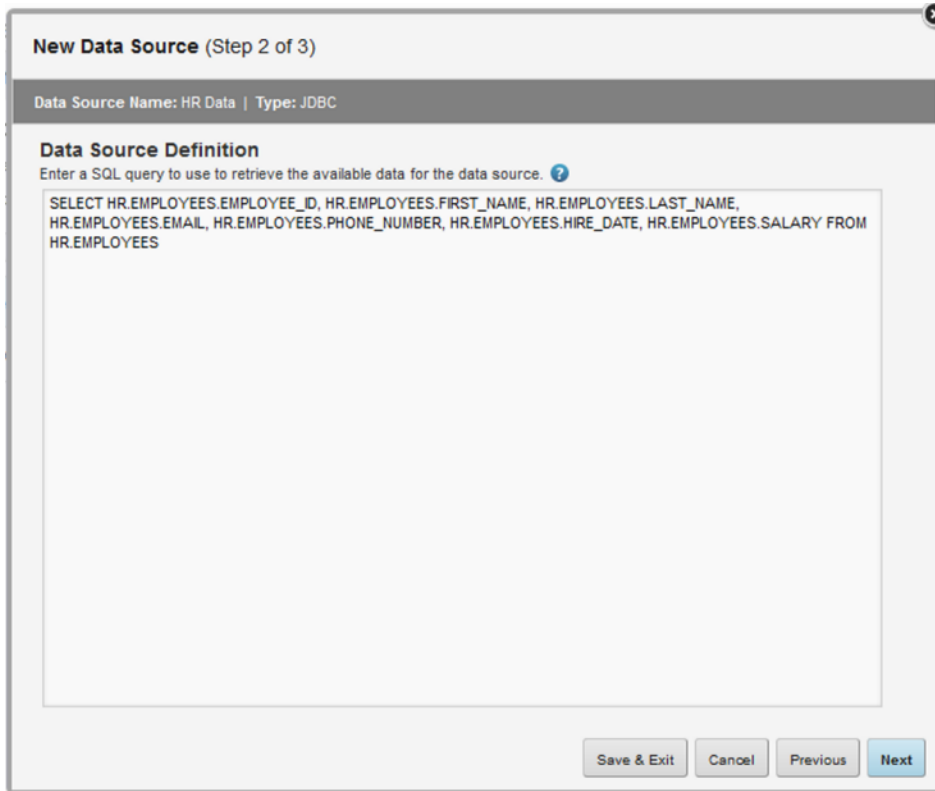
- (a) On the **Select Data Table** page of the data source dialog, from the **Subject Area** drop-down list, select the subject area to use.

The **Presentation Table** list is populated with the available tables from the selected subject area. The list only includes tables that contain data.

- (b) To select the presentation table to use for the data source, click the presentation table, then click **Next**.

The **Configure Data Source** page is displayed. See [Configuring the attributes in a data source on page 88](#).

2. For a JDBC connection data source:



- (a) On the **Data Source Definition** page, in the text area, type the SQL query to use to retrieve the data.

Do not include a trailing semicolon in the query. While direct SQL queries to a database do require the semicolon, it is not required for SQL queries over JDBC, and may cause errors.

When selecting the data, also note that data sources cannot include columns that have the following data types:

- BINARY_FLOAT
- BINARY_DOUBLE
- BLOB
- BFILE
- LONGBLOB
- LONG RAW
- MEDIUMBLOB
- RAW
- TIMESTAMP WITH TIME ZONE
- TIMESTAMP WITH LOCAL TIME ZONE
- UROWID

If columns using these data types are included, Studio returns a validation error.

(b) Click **Next**.

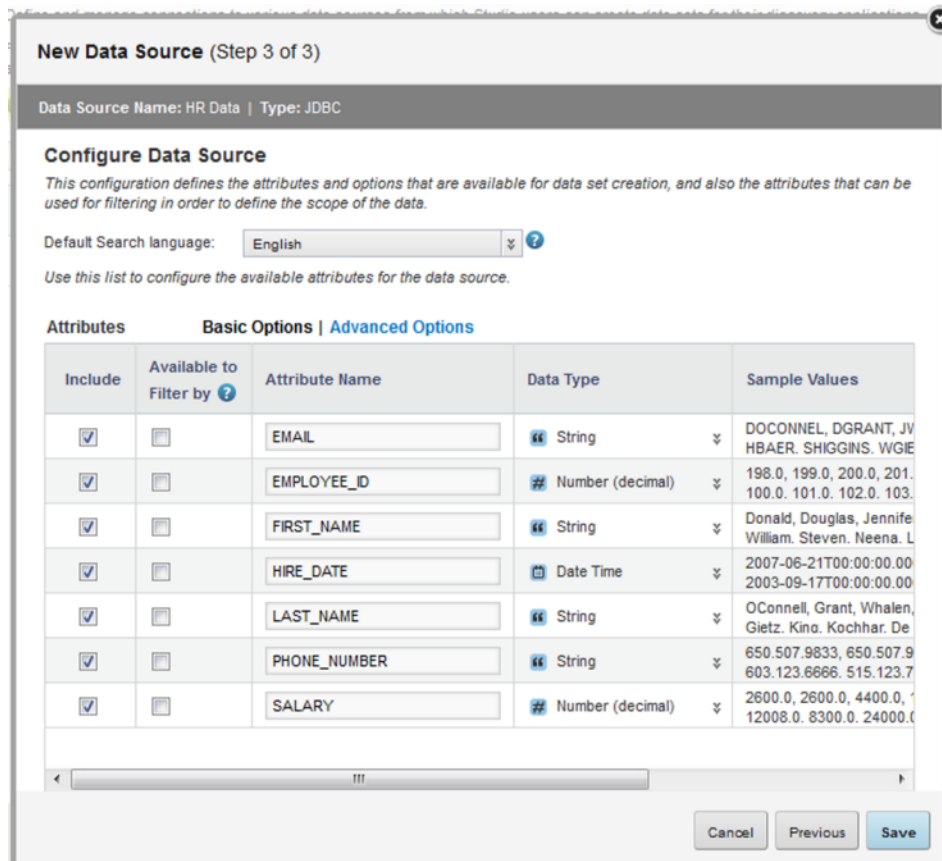
Studio validates the SQL before continuing to the **Configure Data Source** page. See [Configuring the attributes in a data source on page 88](#).

Configuring the attributes in a data source

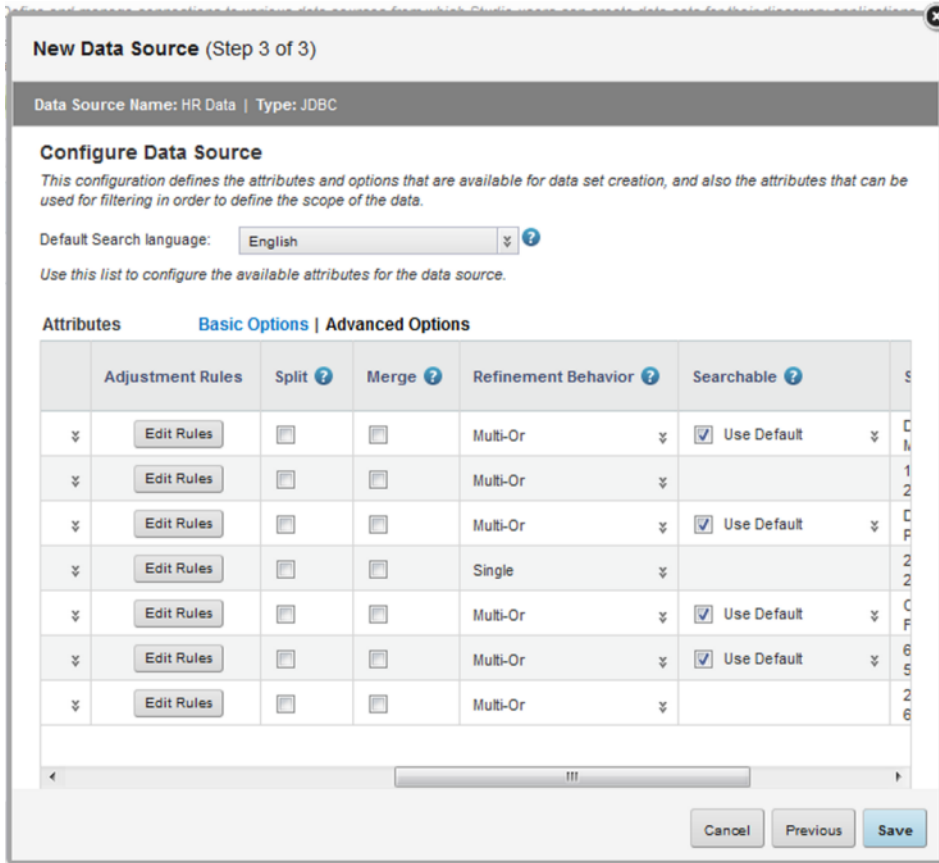
On the **Configure Data Source** page of the data source configuration dialog, you configure the attributes that are included in the data source.

The configuration for data source attributes includes most of the options available when users create new data sets. This allows you to provide default values that should in most cases be correct, so that users do not have to change them.

When the page is first displayed, it shows the **Basic Options** view.



To display additional configuration options for the attributes, click **Advanced Options**.



To configure the data source attributes:

1. From the **Default Search Language** drop-down list, select the default language to use for keyword searches against the data.
2. Use the **Include** column to determine whether each attribute is available to users when they create a data set.

If the checkbox is checked, then the column is included in the data source.

By default, the checkbox is checked for all columns for which the Provisioning Service can retrieve data.

If the data is not accessible, for example because the column is a constant or a reference to another column, then the checkbox is unchecked by default. If a checkbox is unchecked by default, it is recommended that you leave it unchecked.

3. To allow users to use the attribute to filter the data that is included in a data set, check the **Available to Filter by** checkbox.

If you do not specifically enable any of the attributes for filtering, then users cannot filter the data at all. If the source data contains a very large number of records, then it is recommended that you enable filtering for some of the attributes.

Note that only numeric, string, and date/time fields can be enabled for filtering. Users can never use time or duration fields for filtering.

You also should be careful about allowing users to filter by database columns that use CHAR or NCHAR as the original column type. If the columns contain values that are padded with spaces, users may have difficulty getting values to match when they go to filter the data. For example, if the value is actually "abc" followed by two spaces, there won't be a match when users type "abc" without the spaces.

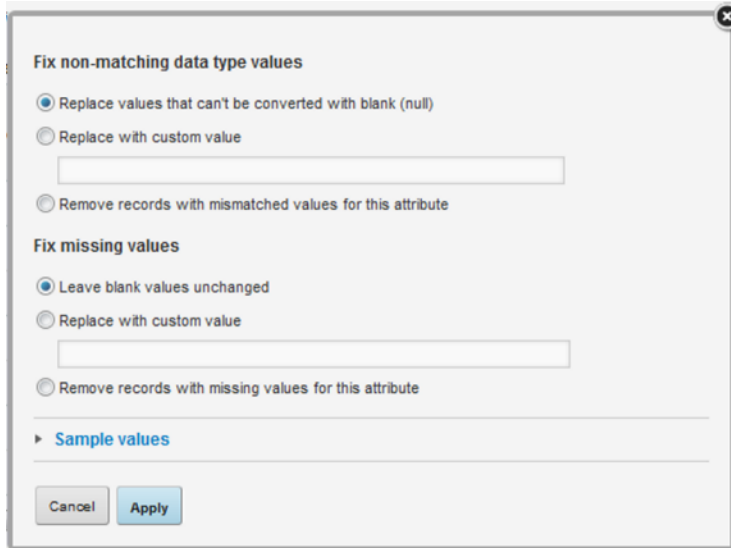
When you check the checkbox, an edit icon is added to the column.

Include	Available to Filter by	Attribute Name
<input checked="" type="checkbox"/>	<input type="checkbox"/>	EMAIL
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	EMPLOYEE_ID
<input checked="" type="checkbox"/>	<input type="checkbox"/>	FIRST_NAME
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	HIRE_DATE
<input checked="" type="checkbox"/>	<input type="checkbox"/>	LAST_NAME
<input checked="" type="checkbox"/>	<input type="checkbox"/>	PHONE_NUMBER
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	SALARY

To display a hint to users who are using the attribute to filter the data for a data set:

- (a) Click the edit icon.
 - (b) On the filter hint dialog, in the **Help text** field, type the hint.
 - (c) Click **Save**.
4. The **Attribute Name** column contains the default display name for the attribute. You can use the field to change the default name.
 5. The **Data Type** column indicates the data type for the attribute as specified in the source data. To use a different data type in the data source, from the drop-down list, select the data type.
Note that if an attribute is not initially identified as a date, you cannot change the data type to make it a date.
 6. In the **Advanced Options** view, you can use the **Adjustment Rules** column to provide rules for setting the attribute value if the original value is either invalid (does not match the data type) or empty.
 - (a) Click the **Edit Rules** button.
 - (b) Under **Fix non-matching data type values**, click a radio button to indicate how to handle values that cannot be converted to the selected data type. You can choose to either:
 - Replace the invalid values with a blank value
 - Replace the invalid values with a custom value

- Remove records that have invalid values



Note that because string values never have non-matching values, you cannot configure rules for adjusting invalid values.

- Under **Fix missing values**, click a radio button to indicate how to handle empty values.

You can choose to either:

- Leave the blank values as is
- Provide a custom value to use wherever a value is missing
- Remove records that have empty values

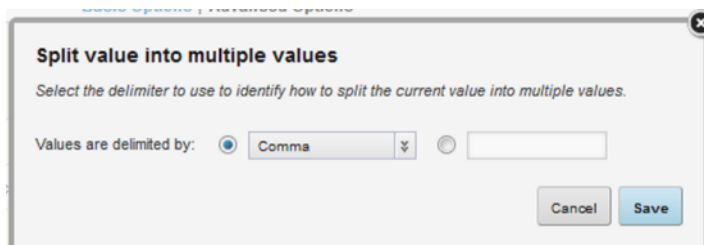
- To display a sample list of values for the attribute, expand the **Sample Values** section at the bottom of the dialog.

- In the **Advanced Options** view, the **Split** and **Merge** columns allow you to create multi-value attributes.

The **Split** option is used when the attribute value is actually a list of values. For example, for a Color attribute, you can indicate that "blue;red;white" are actually three separate values delimited by a semicolon.

To split the value for an attribute:

- Check the **Split** checkbox for the attribute.
- On the split value dialog, under **Values are delimited by**, specify the delimiter used to separate the values.



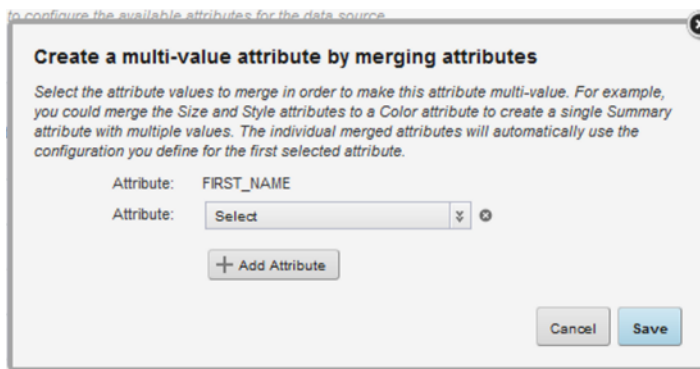
Select the delimiter from the drop-down list. If the delimiter used is not in the drop-down list, click the other radio button, then type the delimiter in the field.

- (c) To save the split configuration, click **Save**.
- (d) If you need to change the split configuration, click the edit icon in the **Split** column.
- (e) To not split the attribute value, uncheck the **Split** checkbox.

The **Merge** option is used to combine values from other attributes into the current attributes. For example, you could add the values from Color2 and Color3 to the Color1 attribute in order to generate the full list of colors.

To merge values from multiple attributes:

- (a) Check the **Merge** checkbox for the attribute.
- (b) On the merge dialog, from the **Attribute 2** drop-down list, select the first attribute to merge with the original attribute.



The merged attributes must be of the same type as the original attribute.

- (c) To add the attribute, click the **Add Attribute** button.
- (d) To remove an attribute, click its delete icon. You cannot remove the original attribute.
- (e) To save the merge configuration, click **Save**.

On the **Configure Data Source** page, you can no longer edit the merged attributes. They automatically inherit all of the configuration options you specify for the original attribute.

The **Merge** checkbox is also checked for those attributes.

- (f) If you need to change the merge configuration, click the edit icon in the **Merge** column.
- (g) To remove the merge, uncheck the **Merge** checkbox.

8. In the **Advanced Options** view, from the **Refinement Behavior** drop-down list, select how refinement works for the attribute.

The refinement behavior indicates whether users can refine by multiple values, and whether multiple values use AND or OR.

The available options are:

Option	Description
Multi-Or	<p>Indicates that end users can refine by more than one value at a time.</p> <p>For multi-or, a record matches if it has at least one of the selected values.</p> <p>So if an end user selects the values Red, Green, and Blue, then matching records only need to have one of those values (Red or Green or Blue).</p> <p>For most attributes, this is the default and the recommended value. Date attributes are always multi-or.</p>
Multi-And	<p>Indicates that end users can refine by more than one value at a time.</p> <p>For multi-and, a record matches only if it has all of the selected attribute values. Multi-and should only be used with multi-value attributes.</p> <p>So if an end user selects the values Red, Green, and Blue, then matching records must have all of those values (Red and Green and Blue).</p>
Single	<p>Indicates that end users can only refine by one value at a time.</p> <p>The Single refinement behavior is recommended for string attributes with long values, and numeric and geocode attributes with a large number of unique values.</p> <p>It is not recommended for multi-value attributes, including attributes that you have split or merged.</p>

- In the **Advanced Options** view, the **Searchable** column indicates whether the attribute can be used for text searches.

Text searches are when end users use the **Search Box** component to search for a specific search term.

The **Searchable** checkbox is only displayed and checked by default for all string attributes.

Other types of attributes do not support text search.

To exclude the attribute from text searches, uncheck the **Searchable** checkbox.

By default, the text search for the attribute uses the default search language. If the attribute values are in a different language than the default, then use the drop-down list to select the search language.


- To save the new or updated data source, click **Save**.

Configuring access to a data source in the Data Source Library

By default, any user who can create or manage an application can create a data set from a data source in the **Data Source Library**. You can, however, restrict access to the data source to only allow certain roles, users, or user groups to use the data source.

In the **Data Source Library** list, the **Access** column indicates whether access to the data source is restricted:

- If set to **Available to All**, then any user can use this data source to create a new application, and any application administrator can use this data source to add a new data set to their application.
- If set to **Restricted**, then only specific users, user groups, or user roles can use this data source to create a data set.

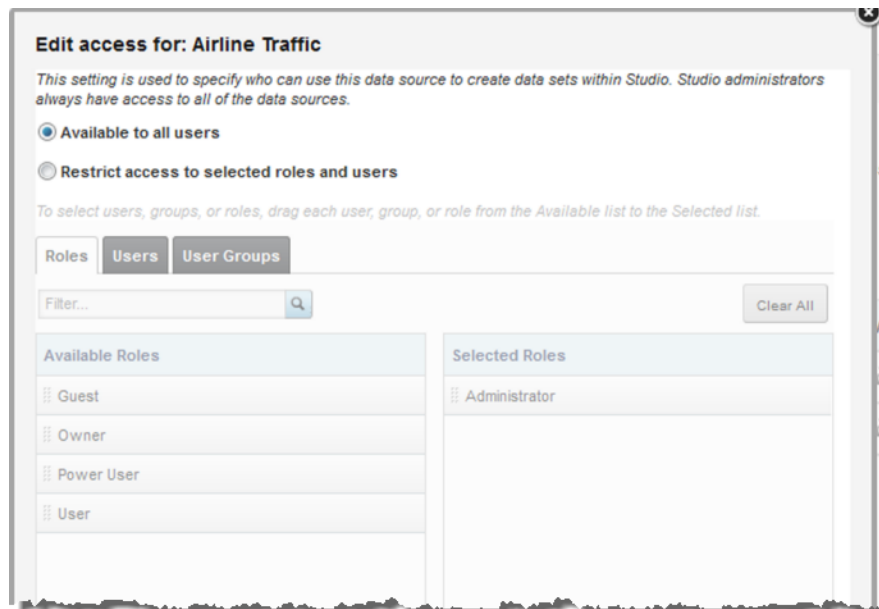
Access
 Available to All  
 Available to All  

Note that this only controls whether a user can create an application from the data source. It does not control access to the specific data. For Oracle BI data sources, access to the data is controlled using the permissions in the RPD file.

To configure access to a data source:

1. On the **Data Source Library**, click the value in the **Access** column.

The data source access dialog is displayed.



2. On the data source access dialog, to restrict access to a data source, click the **Restrict access to selected roles and users** radio button.
The **Roles**, **Users**, and **User Groups** tabs are enabled.
3. On the **Roles** tab, by default, the Administrator role is added to the **Selected Roles** list and cannot be removed. To add and remove other roles:
 - (a) To add a role to the list of roles that can use the data source, drag the role from the available list to the selected list.
You can use the filter field to find a specific role.
 - (b) To remove a role from the selected list, click its delete icon.
 - (c) To clear the selected list, click the **Clear All** button.
4. On the **Users** tab:
 - (a) To add a user to the list of users who can use the data source, drag the user from the available list to the selected list.
You can use the filter field to find a specific user.
 - (b) To remove a user from the selected list, click its delete icon.
 - (c) To clear the selected list, click the **Clear All** button.

Note that if a user has a role or belongs to a user group that you have granted access to, then they will have access to this data source, even if they are not in the list of users that you have granted access to.
5. On the **User Groups** tab:
 - (a) To add a user group to the list of groups that can use the data source, drag the group from the available list to the selected list.
You can use the filter field to find a specific user group.
 - (b) To remove a user group from the selected list, click its delete icon.
 - (c) To clear the selected list, click the **Clear All** button.
6. To save the changes to the data source access, click **Save**.

Removing a data source from the Data Source Library

To remove a data source from the **Data Source Library**, click its delete icon.

When you remove a data source from the **Data Source Library**, any data sets based on that data source are not changed. However, users can no longer add or reload data from the data source into the data set.

When you delete a data source from the **Data Source Library**, Studio prompts you to confirm the deletion.

Part IV

Managing Studio Applications



Configuring and Removing Applications

The Studio **Control Panel** provides options for Studio administrators to configure and remove applications.

[Configuring the application type](#)

[Configuring the visibility type for a page](#)

[Adding and removing application members](#)

[Assigning application roles to application members](#)

[Certifying an application](#)

[Making an application active or inactive](#)

[Removing applications](#)

Configuring the application type

The application type determines whether the application is visible to users on the **Discovery Applications** page.

The application types are:

Application Type	Description
Public	<p>The application is visible to all logged-in users, and all logged-in users can select the application in order to view public pages.</p> <p>Application members can also see private pages.</p> <p>Membership must be granted by an application administrator.</p> <p>Applications created by Studio administrators from the Applications page on the Control Panel are by default public applications.</p>
Private	<p>The application is visible only to application members.</p> <p>Membership must be granted by an application administrator.</p> <p>Applications created from the Discovery Applications page are by default private applications.</p>

If you change the application type, then the page visibility type for all of the application pages changes to match the application type.

From the **Control Panel**, to change the application type for an application:

1. In the **Control Panel** menu, click **Applications**.

2. On the **Applications** page, click the **Actions** link for the application, then select **Edit**.
3. On the **Applications** page, from the **Type** drop-down list, select the application type.

Applications

View All **Add**

Group ID: 10182

Name:

Description:

Type:

Data Source:

Maintain existing component preferences:

Active:

Certified:

Tags: or

4. To save the change, click **Save**.

Configuring the visibility type for a page

Whether users can have access to pages within an application, particularly pages for applications they're not a member of, is based on the page visibility type.

The page visibility type works similarly to the application type. It determines whether users can view the page without logging in or being an application member.

The page visibility types are:

Page Type	Description
Public	<p>A public page is visible to all logged-in users, including users who are not members of the application.</p> <p>When non-logged-in users navigate to the URL, they are prompted to log in before they can view the page.</p>
Private	<p>A private page is only visible to logged-in users who are members of the application.</p> <p>When non-logged-in users navigate to the page URL, they are prompted to log in.</p> <p>When they log in, if they are not a member of the application, then they cannot view the page.</p>

By default, the page visibility type is the same as the application visibility type.

From the **Control Panel**, to select a different visibility type for an application page:

1. In the **Control Panel** menu, click **Applications**.
2. For the application containing the page you want to configure, click the **Actions** link, then click **Manage Pages**.
3. In the page list at the left, click the page name.
4. Click the **Page** tab for the selected page.

The screenshot shows the 'Manage Pages' configuration interface. At the top, there are tabs for 'Pages' and 'Export / Import', and buttons for 'View Pages' and 'Publish to Remote'. Below this, there are links for 'Expand All' and 'Collapse All', and a breadcrumb 'Edit Page: Sales Discovery Dashboard » Visualizations'. A tree view on the left shows 'Sales Discovery Dashboard' with sub-items 'Wine' and 'Visualizations'. The main configuration area has a 'Page' tab and 'Children' sub-tab. The configuration fields include:

- Name: Visualizations
- HTML Title: (empty)
- Type: Portlet
- Page Visibility Type: Private
- Hidden: (checkbox, unchecked)
- Friendly URL: http://appdev-x2k8-p3.us.oracle.com:8080/eid/web/sales-discovery-dashboard /visualizations
- Query String: (empty)
- Icon: (Browse... button, No file selected)
- Use Icon: (checkbox, unchecked)
- Target: (empty)

5. From the **Page Visibility Type** drop-down list, select the visibility type.
6. To save the change, click **Save**.

Adding and removing application members

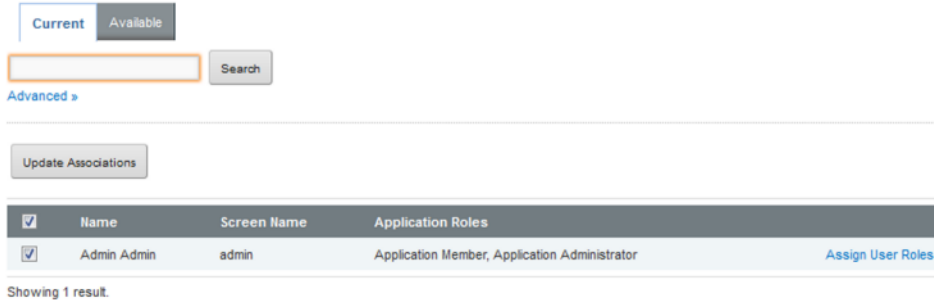
From the **Control Panel Applications** page, Studio administrators can add and remove members from any Studio application.

From the **Control Panel**, to manage the membership for an application:

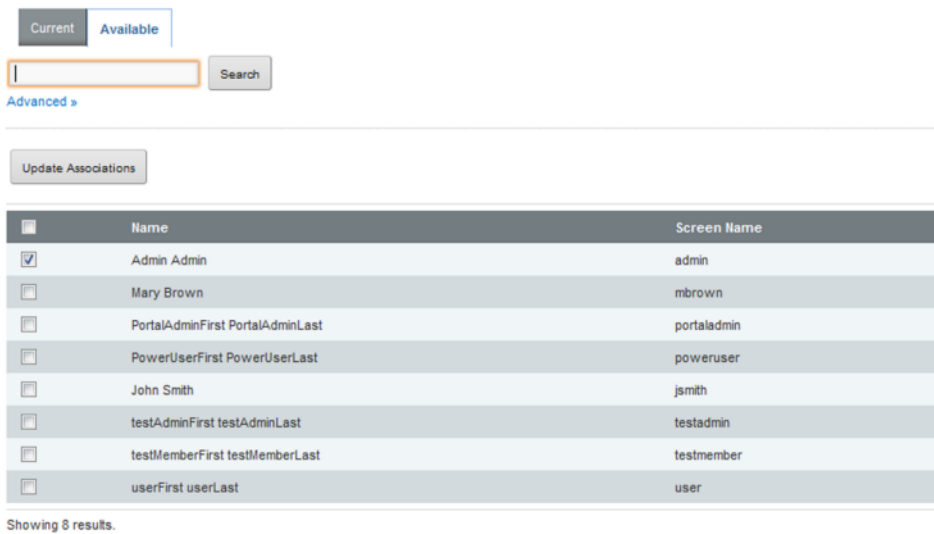
1. In the **Control Panel** menu, click **Applications**.

- For the application you want to manage membership for, click the **Actions** button, then click **Assign Members**.

On the membership page, the **Current** tab lists the current application members.



The **Available** tab lists all of the users. For current members, the checkbox is checked.



- To add a user as a new member, on the **Available** tab, check the user's checkbox.
- To remove a member, on either the **Current** or **Available** tab, uncheck the user's checkbox.
- To save the membership changes, click **Update Associations**.

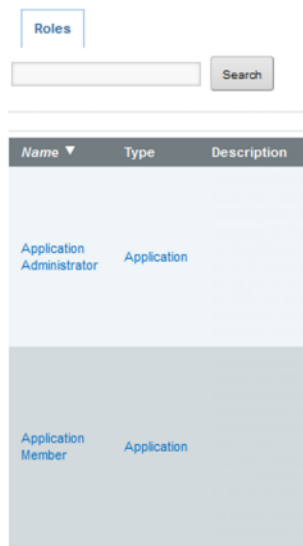
Assigning application roles to application members

From the **Control Panel Applications** page, Studio administrators can change any application's membership to determine whether members are an application members or administrators.

From the **Control Panel**, to assign application roles to application members:

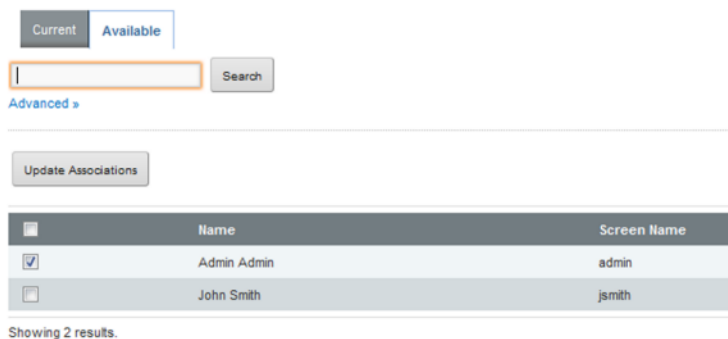
- In the **Control Panel** menu, click **Applications**.
- For the application you want to update the member roles for, click the **Actions** button, then click **Assign User Roles**.

- On the **Roles** page, click the role you want to assign.



On the users page, the **Current** tab lists the users who currently have the selected role.

The **Available** tab lists all of the application members. Members who already have that role have the checkbox checked.



- On the **Available** tab, to assign the selected role to a user, check the checkbox.
- On the **Current** or **Available** tab, to remove the role from the user, uncheck the checkbox.
- To save the membership changes, click **Update Associations**.

Certifying an application

Studio administrators can certify an application. On the **Discovery Applications** page, certified applications are displayed in the **Certified Applications** list. Applications that are not certified are displayed in the **Community Applications** list.

Certifying an application can be used to indicate that the application content and functionality has been reviewed and the application is approved for use by all users who have access to it.

Note that only Studio administrators can certify an application. Application administrators cannot change the certification status.

From the **Control Panel**, to certify an application:

1. In the **Control Panel** menu, click **Applications**.
2. Click the **Actions** link for the action, then click **Edit**.
3. On the application configuration page, to certify the application, check the **Certified** checkbox.

Applications

[View All](#) [Add](#)

Group ID: 10182

Name:

Description:

Type:

Data Source:

Maintain existing component preferences:

Active:

Certified:

Tags: [Add Tags](#) or [Select Tags](#)

[Save](#) [Cancel](#)

4. To save the change, click **Save**.

Making an application active or inactive

By default, a new application is marked as active. From the **Control Panel**, Studio administrators can control whether an application is active or inactive. Inactive applications are not displayed on the **Discovery Applications** page.

Note that this option only available to Studio administrators.

To change the application status:

1. In the **Control Panel** menu, click **Applications**.
2. Click the **Actions** link for the action, then click **Edit**.

- On the application configuration page, to make the application inactive, uncheck the **Active** checkbox.

Applications

View All Add

Group ID 10182

Name Sales Discovery Dashboard

Description Dashboard for exploring recent sales data.

Type Private

Data Source free for all

Maintain existing component preferences

Active

Certified

Tags Add Tags or Select Tags

Save Cancel

If the application is inactive, then to make the application active, check the **Active** checkbox.

- To save the change to the application status, click **Save**.

Removing applications

From the **Control Panel**, you can delete applications created from shared Endeca Server connections.



Important: Do not use the option on the **Control Panel Applications** page to remove an application connected to data created from a file upload or the **Data Source Library**. If you do this, Studio cannot properly clean up the data. Always delete these applications from the **Discovery Applications** page.

To remove an application:

- In the **Control Panel** menu, click **Applications**.
- On the **Applications** page, click the **Actions** link for the application you want to remove.
- In the **Actions** menu, click **Delete**.



Chapter 14

Exporting and Importing Studio Application Pages

To back up your application, or to migrate applications between environments, you can export and import Studio application pages.

[About exporting and importing application pages](#)

[Exporting pages from Studio](#)

[Importing pages into Studio](#)

About exporting and importing application pages

The export and import process is mostly used to migrate application pages between environments.

Your company may use multiple environments for its development process. These environments might include:

- Development, for creating new content
- Testing, to test content that is ready to go to production
- Staging, to stage new content. In some cases, the testing and staging systems are combined into a single environment.
- Production, the site available to end users

When developing new content, instead of having to recreate the content on each environment, you can export the pages from one environment, and then import them into another environment.

You also can use the export function to back up a set of pages.

The pages are exported to a LAR file.

Note that you can only export and import pages on the same version of Studio.

Exporting pages from Studio

From Studio, you export selected pages into a LAR file.

[What is included in the export?](#)

[Completing the export](#)

What is included in the export?

When you export pages, the exported material only includes the pages and the components. The export does not contain any data.

Completing the export

Studio administrators export application pages using the **Manage Pages** option on the **Applications** page of the **Control Panel**.

To export pages from Studio:

1. From the administrator menu, select **Control Panel**.
2. On the **Control Panel** menu, click **Applications**.
3. On the **Applications** page, for the application you want to export pages from, click the **Actions** button, then click **Manage Pages**.
4. On the **Applications** page for the selected application, click the **Export/Import** tab.
5. Click the **Export** tab. The **Export** tab contains the options for exporting the pages.

Export the selected data to the given LAR file name.

Sales_Discovery_Dashboard-2

What would you like to export?

Pages

Portlets

Setup

Archived Setups

User Preferences

Data

Range:

All

Date Range ?

Last 12 Hours ▾

Permissions ?

Categories ?

[More Options »](#)

Export

6. In the field, set the name of the LAR file to export the pages to.
7. In addition to the default settings, make sure to check the **User Preferences** checkbox, so that the exported pages include the complete component configuration.

Note that if you also choose to export permissions, remember that you must have the same users and user groups on the destination environment as on the source environment.

8. Click **Export**.

You are prompted to save the resulting LAR file.

Importing pages into Studio

From Studio, you use LAR files to import pages that were exported from another Studio instance.

Ensuring that imported pages will work properly

Completing the import

Ensuring that imported pages will work properly

When importing pages into Studio, make sure that both environments are based on the same version of Studio.

In addition, to ensure that your imported pages will work correctly on the new environment, make sure that:

- You import the pages into the same application as you exported them from.
If you import a page into a different application, then any links between pages or deep links from external sites may not work.
- The data domains used by the page components are also configured on the destination environment.
For those data domains, the attribute group configuration also needs to be the same.
If the application you import the pages into is connected to a different data domain, then the components on the imported pages are updated to connect to that domain. The component configuration will need to be updated.
- If you are importing permissions, the same users are configured on the destination environment.

Completing the import

Studio administrators import applications using the **Manage Pages** option on the **Applications** page of the **Control Panel**.

To import pages from a LAR file into Studio:

1. From the administrator menu, select **Control Panel**.
2. On the **Control Panel**, click **Applications**.
3. On the **Applications** page, for the application you want to import pages into, click the **Actions** button, then click **Manage Pages**.
4. On the **Applications** page for the selected application, click the **Export/Import** tab.

5. Click the **Import** tab.

The **Import** tab contains the options for importing pages from a LAR file.

6. To search for and select the file to import, click the **Browse** button.
7. By default, if a page in the selected LAR file has the same friendly URL as an existing page in the destination environment, then it replaces that page. To instead add the page as a new page:
 - (a) On the **Import** tab, click the **More Options** link.

Under the **Pages** checkbox, the **Page Merge Strategy** setting is displayed.

By default, the **Replace pages with same Friendly URL** radio button is selected.

- (b) To add new pages instead of replacing existing pages, click the **Add As New** radio button.

When Studio adds the new page, if an existing page has the same friendly URL, then Studio adds an index number to the end of the new page's friendly URL to make it unique.

8. To delete any pages on the destination environment that do not exist in the LAR file, check the **Delete Missing Pages** checkbox.

For example, a LAR file contains the pages Welcome, Dashboard, and Search. The destination environment contains a page called Charts.

If the **Delete Missing Pages** checkbox is checked, then when the LAR file is imported, the Charts page would be removed from the destination environment. The destination environment would only contain Welcome, Dashboard, and Search.

If the checkbox is not checked, then when the LAR file is imported, the destination environment would contain Welcome, Dashboard, Search, and Charts.

Note that if you are importing the page into the same application that was displayed when you navigated to the **Control Panel**, then Studio disables this option.

9. To ensure that the import includes all of the component configuration, check the **User Preferences** checkbox.
10. After selecting the import options, to complete the import, click the **Import** button.

Part V

Customizing Studio



Chapter 15

Changing the Look and Feel of Studio

Users with CSS expertise can customize the look and feel of the Studio application.

[About customizing the Studio look and feel](#)

[Location of the Studio CSS and images](#)

[Updating the Studio CSS and images for a WebLogic Server instance](#)

About customizing the Studio look and feel

Studio allows you to customize the Studio CSS and use your own images.

Note that updating the CSS is not recommended, and you should only attempt to do this type of customization if you are very familiar with cascading style sheets.

When replacing images, if you only want to replace the image and don't want to have to update the CSS, then you should make sure that the images are the same size.

If you have a clustered instance of Studio, also be sure to make the same changes on all of the Studio instances.

Location of the Studio CSS and images

The CSS and images that control the Studio look and feel are located in the `html/css/eid-default` directory.

For WebLogic Server deployments, the directory is embedded in `endeca-portal-<versionNumber>.war`, which in turn is embedded in `endeca-portal-weblogic-<versionNumber>.ear`.

The directory contains:

- `endeca-skin.css` – The main CSS file for Studio. The CSS selectors are split among multiple CSS files. This file contains pointers to those files. This is the minified version of the file.
- `endeca-skin-unminified.css` - The unminified version of `endeca-skin.css`.
- `endeca-skin-idNumber.css` - The CSS files containing the CSS selectors. *idNumber* is a generated identifier.

This files are all minified.

- `endeca-skin-idNumber-unminified.css`

These are the unminified versions of the `endeca-skin-idNumber.css` files.

- `images/extjs-default-images` – Contains third-party images used on the Studio user interface.
- `images/liferay-default-images` – Contains third-party images used on the Studio user interface.

- `images/oracle-default-images` – Contains our custom images to display the user interface. The images are grouped by function.

Note that while this directory contains both minified and unminified versions of the same CSS files, by default, Studio uses the minified version. If you update the unminified version, then you also must configure Studio to use that version.

Updating the Studio CSS and images for a WebLogic Server instance

Before you can update the files, you need to extract the `css/eid-default` directory from the `.ear` file.

To extract and update files in WebLogic Server:

1. Stop Studio.
2. Open `endeca-portal-weblogic-<versionNumber>.ear`.
3. In the `.ear` file, navigate to `endeca-portal-<versionNumber>.war`.
4. Open `endeca-portal-<versionNumber>.war`.
5. In the `.war` file, navigate to `html/css/eid-default`.
6. Update the CSS and images as needed.
7. Resave `endeca-portal-<versionNumber>.war`, then re-save this `.war` file in `endeca-portal-weblogic-<versionNumber>.ear`.
8. By default, Studio uses the minified version of the CSS files. If you updated the unminified version, you must change the Studio configuration to use the unminified version.

To do this, add the following property to `portal-ext.properties`:

```
css.minifier.enabled=false
```

9. Redeploy `endeca-portal-weblogic-<versionNumber>.ear` in Weblogic Server.

Once the file finishes deploying, the CSS and images are updated.

If needed, clear the browser cache in order to see the changes on the Studio user interface.



Chapter 16

Using a Custom Security Manager

If you may require more than the default role-based security supported for Endeca Server connections, you can create a custom Security Manager to customize how Studio filters data from Endeca Server.

[Security Manager class summary](#)

[Creating a new Security Manager](#)

[Implementing a new Security Manager](#)

[Deploying a new Security Manager](#)

[Configuring Studio to use your Security Manager](#)

Security Manager class summary

A Security Manager is a concrete class that implements `com.endeca.portal.data.security.MDEXSecurityManager`.

For additional details about `MDEXSecurityManager`, see the *Studio API Reference*.

Class Summary Item	Item value or Description
Abstract base class	<code>com.endeca.portal.data.security.MDEXSecurityManager</code>
Default implementation class	<code>com.endeca.portal.data.DefaultMDEXSecurityManager</code>
Description	Handles pre-execution query modification based on the user, role, or group-based security configuration of filters.

Class Summary Item	Item value or Description
Default implementation behavior	<p>The default Security Manager implementation uses the following properties:</p> <ul style="list-style-type: none"> • <code>securityEnabled</code>. If the value is not present, then <code>securityEnabled</code> defaults to <code>false</code>. • <code>securityFilters</code>. <code>DataSourceFilters</code> are the only supported type of <code>securityFilter</code>. • <code>rolePermissions</code> <p>These properties are defined in Endeca Server connections in order to apply role-based security filters to queries issued to the Endeca Server. See Configuring role-based security for viewing Endeca Server connection data on page 71.</p> <p>For each Endeca Server connection, the Security Manager maintains an internal map of security filters to always apply to queries issued during that user's session.</p>

Creating a new Security Manager

The Studio SDK includes Windows and Linux batch scripts for creating a new Security Manager.

To create a new Security Manager project:

1. From the Studio Media Pack for Windows or Linux, download the Studio SDK.
2. Unzip the Studio SDK file into a separate directory.
3. In a terminal, change your directory to the `endeca-extensions` directory within the Studio SDK's root directory (normally called `components`).
4. Run one of the following commands:
 - On Windows: `.\create-mdexsecuritymanager.bat <your-security-manager-name>`
 - On Linux: `./create-mdexsecuritymanager.sh <your-security-manager-name>`

This command creates a `your-security-manager-name` directory under `endeca-extensions`. This directory is an Eclipse project that you can import directly into Eclipse, if you use Eclipse as your IDE.

This directory also contains a sample implementation that you can use to help understand how the Security Manager can be used. The sample implementation is essentially identical to the default implementation of the Security Manager used by Studio.

Implementing a new Security Manager

Your Security Manager must implement the `applySecurity` method.

```
public void applySecurity(PortletRequest request, MDEXState mdexState, Query query) throws
MDEXSecurityException;
```

The Query class in this signature is `com.endeca.portal.data.Query`. This class provides a simple wrapper around a Conversation Service request.

Deploying a new Security Manager

Before you can use the new Security Manager, you must deploy it to Studio.

The `your-security-manager-name` directory you created contains an ant build file.

For WebLogic, you must add it to the deployed `.ear` file, so that it will be deployed automatically the next time you deploy the file, for example when installing a production instance after you have completed testing on a development instance.

To deploy a custom Security Manager in WebLogic:

1. Run the ant build script for the custom security manager project to generate a JAR archive named `<your-security-manager-name>-mdexsecuritymanager.jar`, located in your security manager project directory.
2. Add the `.jar` file to the `APP-INF/lib` directory within the WebLogic `.ear` file.

Configuring Studio to use your Security Manager

In order to use your Security Manager, you must specify a new class for Studio to pick up and use in place of the default Security Manager implementation.

To configure Studio to use your new Security Manager class:

1. From the administrator menu, section **Control Panel**.
2. In the **Information Discovery** section of the **Control Panel** navigation panel, select **Framework Settings**.
3. Change the value of the `df.mdexSecurityManager` property to the full name of your class, similar to following example:

```
df.mdexSecurityManager = com.endeca.portal.extensions.YourSecurityManagerClass
```
4. Click **Update Settings**.
5. Restart Studio so the change can take effect. You may also need to clear any cached user sessions.



The Studio SDK is a packaged development environment that you can use to create or modify components.

[Software and licensing requirements for component development](#)

[Configuring the Studio SDK for component development](#)

[Configuring Eclipse for component development](#)

[Developing a new component](#)

[Modifying the Studio SDK build properties for a component](#)

Software and licensing requirements for component development

To develop custom components, you need the following software and licenses.

Software requirements

In addition to the Studio SDK, component development requires the following software:

- Eclipse
- JDK 1.5 or above
- Apache Ant 1.7.1 or higher

Ext JS license requirement

Studio uses [Ext JS](#) in its components and in the default components created using the Studio SDK.

The Oracle Endeca Information Discovery license does not bundle licensing for Ext JS.

Therefore, customers developing components with Ext JS must either purchase their own development licenses from Ext JS, or remove Ext JS and develop components without using that Javascript framework.

Obtaining .jar files for JavaScript minification

By default, when you compile a custom component, the JavaScript minification is not used.

While components do build successfully without JavaScript minification, for performance purposes you may want to enable it.

In order to be able to use minification to build components, you must obtain the following .jar files for version 2.4.2 of YUI Compressor:

- `yuicompressor-2.4.2.jar`

- jargs-1.0.jar
- rhino-1.6R7.jar

To obtain the files:

1. Go to <https://github.com/yui/yuicompressor/downloads>.
2. From the YUI Compressor downloads page, download `yuicompressor-2.4.2.zip`.
3. From the .zip file, extract the following files to your machine:
 - `build/yuicompressor-2.4.2.jar`
 - `lib/jargs-1.0.jar`
 - `lib/rhino-1.6R7.jar`

Make sure to note where you have placed these files, as you will be updating a configuration file to reflect their location.

About obtaining junit.jar for component unit tests

If you are planning to create unit tests for your custom components, you will need to first obtain `junit.jar`.

The Studio SDK can use JUnit for unit tests, but does not come with the `junit.jar` file.

Configuring the Studio SDK for component development

Before you can start developing components, you must configure the Studio SDK.

To download, install, and configure the Studio SDK:

1. From the Studio Media Pack for Windows or Linux, download the Studio SDK.
2. Unzip the file into a separate directory.



Note: Do not install the Studio SDK in a directory path that contains spaces.

3. Within the Studio SDK directory:

- (a) Create the following file:

```
components/build.<user>.properties
```

In the file name, `<user>` is the user name that you use used to log in to the current machine.

- (b) Within that file, add the following property:

```
portal.base.dir=<absolute_path_to_portal>
```

The value `<absolute_path_to_portal>` is the path to the `endeca-portal` directory for the Studio instance.



Note: On Windows, backslashes in paths must be escaped. For example, use:

```
portal.base.dir=C:\\my_folder\\endeca-portal
```

instead of:

```
portal.base.dir=C:\my_folder\endeca-portal
```

- (c) In the `shared/` directory, create a `shared.properties` file .
- (d) In `shared.properties`, add the following property:

```
portal.base.dir=<absolute_path_to_portal>
```

The value `<absolute_path_to_portal>` is the path to the `endeca-portal` directory for the Studio instance.



Note: On Windows, you must escape backslashes in paths. For example, use:

```
portal.base.dir=C:\\my_folder\\endeca-portal
```

instead of:

```
portal.base.dir=C:\my_folder\endeca-portal
```

4. To enable JavaScript minification when building your components:
 - (a) If you haven't already, obtain the required YUI Compressor .jar files. See [Obtaining .jar files for JavaScript minification on page 115](#).
 - (b) Once the files are in place, from the Studio SDK directory, open the file `components\build-common-plugin.xml`.
 - (c) In the file, find the `Minify JavaScript` section.
 - (d) Find the following line:

```
<available file="${project.dir}/../portal/lib/development/liferay-yuicompressor.jar"/>
```

- (e) Update the path and file name to reflect where you placed `yuicompressor-2.4.2.jar`.
- (f) Next, find the following lines:

```
<arg path="${project.dir}/../portal/lib/development/liferay-rhino.jar;
${project.dir}/../portal/lib/development/jargs.jar;
${project.dir}/../portal/lib/development/liferay-yuicompressor.jar"/>
<arg line="com.liferay.yahoo.platform.yui.compressor.Bootstrap"/>
```

- (g) In the first line, update the paths and file names to reflect where you placed `rhino-1.6R7.jar`, `jargs-1.0.jar`, and `yuicompressor-2.4.2.jar`.
- (h) In the second line, remove `liferay` from the value.

The value is then:

```
<arg line="com.yahoo.platform.yui.compressor.Bootstrap"/>
```

Configuring Eclipse for component development

Before using the Studio SDK to develop Studio components in Eclipse, you need to create two Eclipse classpath variables.



Note: Depending on your version of Eclipse, the steps below may vary slightly.

To configure the Eclipse classpath variables for Studio component development:

1. In Eclipse, go to **Window>Preferences>Java>Build Path>Classpath Variables**.
2. Create two new variables:

Name	Path
DF_GLOBAL_LIB	Path to the application server global library.
DF_PORTAL_LIB	Path to the Studio Web application library.

Once these variables have been created, you can import into Eclipse the components generated using the Studio SDK.

Developing a new component

After you have configured the Studio SDK and Eclipse, you then develop, build, and test the new component.

[Creating a new component](#)

[Importing the component project into Eclipse](#)

[Obtaining query results for custom components](#)

[Building and testing your new component](#)

[Adding and removing components from the Studio .ear file for WebLogic Server](#)

Creating a new component

New Studio components are extensions of the `EndecaPortlet` class.

To create a new component:

1. At a command prompt, navigate to the Studio SDK directory, and from there to `components/portlets`.
2. Run the command:

```
create.bat <component-name-no-spaces> "<ComponentDisplayName>"
```

For example:

```
create.bat johns-test "John's Test Component"
```

In the command, the first argument is the component name. The component name:

- Cannot have spaces.
- Cannot include the string `-ext`, because it causes confusion with the `ext` plugin extension. For example, `my-component-extension` would not be a valid name.
- Has the `-portlet` automatically appended to the name. For example, if you set the name to `johns-test`, the name will actually be `johns-test-portlet`.

The second argument is intended to be a more human-friendly display name. The display name can have spaces, but if it does, it must be enclosed in quotation marks.

Importing the component project into Eclipse

Before beginning component development, you have to import the component project you just created into Eclipse.

To import the component project into Eclipse:

1. Within Eclipse, choose **File>Import>General>Existing Projects into Page**.
2. As the root directory from which to import, select the directory where you installed the Studio SDK. You should see multiple projects to import.
3. Import the components you need to work with.

If your components depend on shared library projects located within the `/shared` directory, import those as well.



Note: It takes some time for projects to build after they are imported.

After you import the component project into Eclipse, you can begin the actual component development.

Obtaining query results for custom components

When developing a custom component, use the `QueryState` and `QueryResults` classes to issue a request to and obtain results from the Endeca Server.

To specify the types of results the component needs, you must add the relevant `QueryConfigs` to the `QueryState`. For example:

```
QueryState query = getDataSource(request).getQueryState();
CollectionBaseView defaultBaseView = EndecaPortletUtil.getDefaultCollection(request);
query.addFunction(new NavConfig(), defaultBaseView, request.getLocale());
QueryResults results = getDataSource(request).execute(query);
```

You can then get the underlying Conversation Service API results in order to obtain the data required by your component.

```
Results discoveryResults = results.getDiscoveryServiceResults();
```

Before executing the query, you can also make other local modifications to your query state by adding filters or configurations to your query. For example:

```
String viewKey = request.getParameter(VIEW_KEY_PARAM);
DataSource ds = getDataSource(request);
QueryState query = ds.getQueryState();
SemanticView sView = ds.getCollectionOrSemanticView(viewKey, request.getLocale());
query.addFunction(new ResultsConfig(), sView, request.getLocale());
ExpressionBase expression = getDataSource(request).parseLQLEExpression("Region = 'Midwest'");
query.addFunction(new SelectionFilter(expression), sView, request.getLocale());
QueryResults results = getDataSource(request).execute(query);
```

To persist `QueryState` changes to the user's session, which also updates the associated components, use `setQueryState`. For example:

```
String viewKey = request.getParameter(VIEW_KEY_PARAM);
```

```
DataSource ds = getDataSource(request);
QueryState query = ds.getQueryState();
SemanticView sView = ds.getCollectionOrSemanticView(viewKey, request.getLocale());
query.addFunction(new ResultsConfig(), sView, request.getLocale());
ExpressionBase expression = getDataSource(request).parseLQLEExpression("Region = 'Midwest'");
query.addFunction(new SelectionFilter(expression), sView, request.getLocale());
ds.setQueryState(query);
```

For details on the `QueryConfig` and `QueryFunction` classes, see [Working with QueryFunction Classes on page 121](#), and the [Studio API Reference](#).

Building and testing your new component

Next, you can build your new component in Eclipse and verify that it is available in Studio.

To build your new component in Eclipse:

1. In your new project, open the `build.xml` file at the top level.
2. In the outline view, right-click the deploy task and select **Run as...>Ant Build**.



Note: This step is only necessary if you do not have **Build Automatically** checked in the Eclipse **Project** menu.

3. If Studio is not already running, start Studio and log in.
4. Look at the Studio logs to confirm that the component loaded successfully.
5. To test your new component within Studio:
 - (a) From within a Studio application, click **Add Component**.
Your component should be included in the list of available components.
 - (b) To add the new component to the Studio page, drag and drop it from the **Add Component** menu.

Adding and removing components from the Studio .ear file for WebLogic Server

If you have installed Studio on Oracle WebLogic Server, then you can also add the component to the deployed `.ear` file, so that it will be deployed automatically the next time you deploy the file, for example when installing a production instance after you have completed testing on a development instance.

To add components to and remove components from the `.ear` file for WebLogic Server:

1. To add a custom component to the `.ear` file:
 - (a) Copy your component to the `<StudioHome>/deploy` directory.
 - (b) After the component has been processed and moved to the `<StudioHome>/weblogic-deploy` directory, undeploy the `.ear` file.
 - (c) Add the processed component `.war` file to the root of the zipped `.ear` file.
 - (d) In the `.ear` file, add an entry for the new component to `META-INF/application.xml`.
2. To remove a component from the `.ear` file:
 - (a) Remove the component `.war` file from the root of the `.ear` file.

- (b) In the .ear file, remove the component entry from META-INF/application.xml.

Modifying the Studio SDK build properties for a component

The `build.xml` file in the root directory of each component created by the Studio SDK contains properties that control how to build the component.

By default, these properties are:

```
<property name="shared.libs" value="endeca-common-resources,endeca-discovery-taglib" />
<property name="endeca-common-resources.includes" value="**/*" />
<property name="endeca-common-resources.excludes" value="" />
```

The properties control the following behavior:

Property	Description
<code>shared.libs</code>	<p>Controls which projects in the <code>shared/</code> directory to include in your component.</p> <p>These shared projects are compiled and included as <code>.jar</code> files where appropriate.</p>
<code>endeca-common-resources.includes</code>	<p>Controls which files in the <code>shared/endeca-common-resources</code> project are copied into your component.</p> <p>The default value is <code>**/*</code>, indicating that all of the files are included.</p> <p>These files provide AJAX enhancements (<code>preRender.jspf</code> and <code>postRender.jspf</code>).</p>
<code>endeca-common-resources.excludes</code>	<p>Controls which files from the <code>shared/endeca-common-resources</code> project are excluded from your component.</p> <p>By default, the value is <code>"</code>, indicating that no files are excluded.</p> <p>If your component needs to override any of these files, you must use this build property to exclude them. If you do not exclude them, your code will be overwritten.</p>

The `includes` and `excludes` properties can be specified for any shared library. For example:

```
<property name="endeca-discovery-taglib.includes" value="**/*" />
<property name="endeca-discovery-taglib.excludes" value="" />
```



Chapter 18

Working with QueryFunction Classes

When developing custom components, you can use Studio's set of `QueryFunction` classes to filter and query data. You can also create and implement your own `QueryFunction` classes.

[Provided QueryFunction filter classes](#)

[Provided QueryConfig functions](#)

[Creating a custom QueryFunction class](#)

[Implementing a custom QueryFunction class](#)

[Deploying a custom QueryFunction class](#)

[Adding the custom QueryFunction .jar file to the custom component Eclipse build path](#)

Provided QueryFunction filter classes

Studio provides the following filter classes. Filters are used to change the current query state.

The available filter classes are:

- `DataSourceFilter`
- `RefinementFilter`
- `NegativeRefinementFilter`
- `RangeFilter`, including the following date/time-specific range filters that extend `RangeFilter`:
 - `DateRangeFilter`
 - `TimeRangeFilter`
 - `DurationRangeFilter`
- `DateFilter`
- `LastNDateFilter`
- `GeoFilter`
- `SearchFilter`

In addition to the information here, for more details on the `QueryFunction` filter classes, see the *Studio API Reference*.

DataSourceFilter

Uses an EQL snippet to provide the filtering. `DataSourceFilter` refinements are not added to the **Selected Refinements** component.

The available properties are:

Property	Description
filterString	<p>The EQL snippet containing the filter information.</p> <p>For a <code>DataSourceFilter</code>, this would be the content of a <code>WHERE</code> clause for an EQL statement.</p> <p>For details on the EQL syntax, see the <i>Oracle Endeca Server EQL Guide</i>.</p>

For example, to filter data to only show records from the Napa Valley region with a price lower than 40 dollars:

```
ExpressionBase expression = dataSource.parseLQLEExpression("Region='Napa Valley' and P_Price<40");
DataSourceFilter dataSourceFilter = new DataSourceFilter(expression);
```

RefinementFilter

Used to filter data to include only those records that have the provided attribute values. `RefinementFilter` refinements are added to the **Selected Refinements** component.

The properties for a `RefinementFilter` are:

Property	Description
attributeValue	<p>String</p> <p>The attribute value to use for the refinement.</p> <p>For a managed attribute, this is the value spec.</p>
attributeKey	<p>String</p> <p>The attribute key. Identifies the attribute to use for the refinement.</p>
multiSelect	<p>AND OR NONE</p> <p>For multi-select attributes, how to do the refinement if the filters include multiple values for the same attribute.</p> <p>If set to <code>AND</code>, then matching records must contain all of the provided values.</p> <p>If set to <code>OR</code>, then matching records must contain at least one of the provided values.</p> <p>If set to <code>NONE</code>, then multi-select is not supported. Only the first value is used for the refinement.</p> <p>This setting must match the refinement behavior configured for the attribute in the data set. For information on using the Views page to view and configure the refinement behavior for an attribute, see the <i>Studio User's Guide</i>.</p>

In the following example, the data is refined to only include records that have a value of 1999 for the Year attribute.

```
RefinementFilter refinementFilter = new RefinementFilter("1999", "Year");
```

NegativeRefinementFilter

Used to filter data to exclude records that have the provided attribute value. `NegativeRefinementFilter` refinements are added to the **Selected Refinements** component.

The properties for a `NegativeRefinementFilter` are:

Property	Description
<code>attributeValue</code>	String The attribute value to use for the refinement.
<code>attributeKey</code>	String The attribute key. Identifies the attribute to use for the refinement.
<code>attributeType</code>	BOOLEAN STRING DOUBLE LONG GEOCODE DATETIME TIME DURATION The type of value to use for the refinement. The default is <code>STRING</code> . If the attribute is a standard attribute of a type other than string, then you must provide the type.
<code>attributeValueName</code>	String Optional. The value to display on the Selected Refinements component for the refinement. If you do not provide a value for <code>attributeValueName</code> , then the Selected Refinements component displays the value of <code>attributeValue</code> . You may want to provide a separate display value if the selected attribute is a managed attribute for which the value names are different from the actual stored value.
<code>ancestors</code>	String List Optional. The display names of the ancestor values to display on the Selected Refinements component. You would most likely want to provide ancestor values when selecting a managed attribute value from a value hierarchy.

Property	Description
isAttributeSingleAssign	<p>Boolean.</p> <p>If set to <code>true</code>, then the attribute can only have one value.</p> <p>If set to <code>false</code>, then the attribute is multi-value.</p> <p>For information on using the View Manager to see whether an attribute is multi-value, see the <i>Studio User's Guide</i>.</p>

In the following example, the data is refined to only include records that do NOT have a value of Washington for the Region attribute. Because Region is a string attribute, no other configuration is needed.

```
NegativeRefinementFilter negativeRefinementFilter
= new NegativeRefinementFilter("Region", "Washington");
```

In the following example, the data is refined to only include records that do NOT have a value of 1997 for the P_Year attribute, which is a single-assign attribute. Because P_Year is not a string attribute, the attribute type LONG is specified.

```
NegativeRefinementFilter negativeRefinementFilter
= new NegativeRefinementFilter("P_Year", "1997", PropertyType.LONG, true);
```

In the following example, the data is refined to only include records that do NOT have Caterer as the value for the Outlet attribute, which is a single-assign attribute. The values for Outlet are stored as codes, so a display name to use for the refinement is provided. Also, Outlet is a hierarchical attribute, and the refinement indicates that Caterer is a subcategory of Nonstore Retailers under the category Retail Sales.

```
List<String> ancestors = new ArrayList<String>();
ancestors.add("Retail Sales");
ancestors.add("Nonstore Retailers");
NegativeRefinementFilter negativeRefinementFilter
= new NegativeRefinementFilter("Outlet", "454210", "Caterer", ancestors, true);
```

RangeFilter

Used to filter data to include only those records that have attribute values within the specified range. RangeFilter refinements are added to the **Selected Refinements** component.

The properties for a RangeFilter are:

Property	Description
attributeKey	<p>String</p> <p>The attribute key. Identifies the attribute to use for the filter.</p>

Property	Description
rangeOperator	LT LTEQ GT GTEQ BTWN GCLT GCGT GCBTWN The type of comparison to use. <ul style="list-style-type: none"> • LT - Less than • LTEQ - Less than or equal to • GT - Greater than • GTEQ - Greater than or equal to • BTWN - Between. Inclusive of the specified range values. • GCLT - Geocode less than • GCGT - Geocode greater than • GCBTWN - Geocode between
rangeType	DECIMAL INTEGER DATE GEOCODE TIME DURATION The type of value that is being compared.
value1	Numeric The value to use for the comparison. For BTWN, this is the low value for the range. For the geocode range operators, the origin point for the comparison.
value2	Numeric For a BTWN, this is the high value for the range. For GCLT and GCGT, this is the value to use for the comparison. For GCBTWN, this is the low value for the range.
value3	Numeric Only used for the GCBTWN operator. The high value for the range.

In the following example, the data is refined to only include records where the value of P_Score is a number between 80 and 100:

```
RangeFilter rangeFilter
= new RangeFilter("P_Score", RangeType.NUMERIC, RangeOperator.BTWN, "80", "100");
```

There are also date/time-specific range filters that extend `RangeFilter`:

- `DateRangeFilter`
- `TimeRangeFilter`
- `DurationRangeFilter`

DateFilter

Used to filter date values. Using a `DateFilter`, you can filter by subsets of the date/time value. For example, you can filter a date attribute to include all records with a specific year or specific month.

The properties for a `DateFilter` are:

Property	Description
<code>dateFilters</code>	<p>A list of <code>DateFilterDimension</code> objects that represent the date filters to apply.</p> <p>Each <code>DateFilterDimension</code> object consists of:</p> <ul style="list-style-type: none"> • <code>DatePart</code> constants identify each date part • Integer values to represent the values for each date part <p>The filter only filters down to the most specific date part provided.</p>

In the following example, the data is refined to only include records where `SalesDate` is June 15, 2006. The filter only provides the year, month, and day. Even if records have different hour-minute-second values for `SalesDate`, as long as they are within June 15, 2006, they still match this filter:

```
DateFilterDimension dfd = new DateFilterDimension();
dfd.addDatePartFilter(DatePart.YEAR, 2006);
dfd.addDatePartFilter(DatePart.MONTH, 6);
dfd.addDatePartFilter(DatePart.DAY_OF_MONTH, 15);
DateFilter dateFilter = new DateFilter("SalesDate", dfd);
```

LastNDateFilter

Used to filter the date to include records with a date attribute with a value in the last n years, months, or days.

The properties for a `LastNDateFilter` are:

Property	Description
<code>attributeKey</code>	The key name of the attribute.
<code>ticksBack</code>	The number of years, months, or days within which to include records in the results.
<code>datePart</code>	<p>The date part to use for the filtering. The possible values are:</p> <ul style="list-style-type: none"> • YEAR • MONTH • DAY_OF_MONTH • HOUR • MINUTE • SECOND

In the following example, the data is refined to only include records with `SalesDate` values from the last 3 years:

```
LastNDateFilter lastNDateFilter = new LastNDateFilter("SalesDate", 3, DatePart.YEAR);
```

GeoFilter

Used filter data to include records with a geocode value within a specific distance of a specific location.

The properties for a `GeoFilter` are:

Property	Description
<code>attributeKey</code>	The key name for the geocode attribute.
<code>rangeOperator</code>	The comparison operator.
<code>value1</code>	A geocode value to use as the starting point.
<code>radius</code>	The number of miles or kilometers within which to search.
<code>locationName</code>	The name of a location to use as the starting point.
<code>unit</code>	The unit of distance (mi or km) for the comparison.

SearchFilter

Used to filter the data to include records that have the provided search terms. `SearchFilter` refinements are added to the **Selected Refinements** component.

The properties for a `SearchFilter` are:

Property	Description
<code>searchInterface</code>	String Either the name of the search interface to use, or the name of an attribute that is enabled for text search.
<code>terms</code>	String The search terms.
<code>matchMode</code>	ALL PARTIAL ANY ALLANY ALLPARTIAL PARTIALMAX BOOLEAN The match mode to use for the search.
<code>enableSnipping</code>	Boolean Whether to enable snipping. Optional. If not provided, the default is <code>false</code> .

Property	Description
snippetLength	<p>Integer</p> <p>The number of characters to include in the snippet.</p> <p>Required if <code>enableSnipping</code> is true.</p> <p>To enable snipping, set <code>enableSnipping</code> to true, and provide a value for <code>snippetLength</code>.</p>

In the following example, the filter uses the "default" search interface to search for the terms "California" and "red". The matching records must include all of the search terms. Snipping is supported, with a 100-character snippet being displayed.

```
SearchFilter.Builder builder = new SearchFilter.Builder("default", "California red");
builder.matchMode(MatchMode.ALL);
builder.enableSnipping(true);
builder.snippetLength(100);
SearchFilter searchFilter = builder.build();
```

Provided QueryConfig functions

Studio provides the following `QueryConfig` functions, used to manage the results returned by a query. These are more advanced functions for component development.

Each `QueryConfig` function generally has a corresponding function in `DiscoveryServiceUtils` to get the results.

`QueryConfig` functions are most often used to obtain results that are specific to a component. Because of this, `QueryConfig` functions should never be persisted to the application data domain using `setQueryState()`, as this would affect all of the components that are bound to the same data. Instead, `QueryConfig` functions should only be added to a component's local copy of the `QueryState` object.

The available `QueryConfig` functions are:

- `AttributeValueSearchConfig`
- `BreadcrumbsConfig`
- `ExposeRefinement`
- `LQLQueryConfig`
- `NavConfig`
- `RecordDetailsConfig`
- `ResultsConfig`
- `ResultsSummaryConfig`
- `SearchAdjustmentsConfig`
- `SearchKeysConfig`
- `SortConfig`

In addition to the information here, for more details on the `QueryConfig` functions, see the *Studio API Reference*.

AttributeValueSearchConfig

Used for typeahead in search boxes. For example, used in Guided Navigation to narrow down the list of available values for an attribute.

`AttributeValueSearchConfig` has the following properties:

Property	Description
<code>searchTerm</code>	String The term to search for in the attribute values.
<code>maxValuesToReturn</code>	int (optional) The maximum number of matching values to return. If you do not provide a value, then the default is 10.
<code>attribute</code>	String (optional) The attribute key for the attribute in which to search. Use the <code>attribute</code> property to search against a single attribute. To search against multiple attributes, use <code>searchWithin</code> .
<code>searchWithin</code>	List<String> (optional) A list of attributes in which to search for matching values.
<code>matchMode</code>	ALL PARTIAL ANY ALLANY ALLPARTIAL PARTIALMAX BOOLEAN (optional) The match mode to use for the search.
<code>relevanceRankingStrategy</code>	String (optional) The name of the relevance ranking strategy to use during the search.

The following example searches for the term "red" in the WineType attribute values:

```
AttributeValueSearchConfig attributeValueSearchConfig
= new AttributeValueSearchConfig("red", "WineType");
```

BreadcrumbsConfig

Used to return the refinements associated with the query. Allows you to specify whether to display the full path for hierarchical attribute values.

BreadcrumbsConfig has the following property:

Property	Description
returnFullPath	<p>Boolean (optional)</p> <p>For a hierarchical managed attribute, whether to return the full path to the selected value.</p> <p>The default is <code>true</code>, indicating to return the full path.</p> <p>To not return the full path, set this to <code>false</code>.</p>

This example returns the refinements, but does not return the full path for hierarchical managed attributes:

```
BreadcrumbsConfig breadcrumbsConfig = new BreadcrumbsConfig(false);
```

ExposeRefinement

Affects results from a `NavConfig` function. Used to implement available refinements. Controls whether to display available attributes within groups, and whether to display available refinements for attributes.

`ExposeRefinement` has the following properties:

Property	Description
dimValId	<p>String</p> <p>The ID of the selected attribute value.</p> <p>You would provide an attribute value ID if you were displaying the next level of available values in a managed attribute hierarchy.</p>
dimensionId	<p>String</p> <p>The name of the attribute.</p> <p>You must provide at least one <code>dimValId</code> or <code>dimensionId</code>.</p>
ownerId	<p>String (optional)</p> <p>The ID of the associated <code>NavConfig</code> instance.</p> <p>If not provided, then uses the first <code>NavConfig</code> instance.</p>
dimExposed	<p>Boolean (optional)</p> <p>Whether to display the available values for the attribute, to the number specified in <code>maxRefinements</code>.</p> <p>The default is <code>true</code>.</p>

Property	Description
exposeAll	Boolean (optional) Whether to display the complete list of available values. For example, on the Available Refinements component, would indicate whether the "More..." link is selected. The default is <code>false</code> .
maxRefinements	Integer (optional) The maximum number of available values to display. The default is <code>1000</code> .
groupKey	String (required) The name of a group.
groupExposed	boolean (optional) Whether to display all of the attributes in the specified group. The default is <code>true</code> .

The following example shows the available attributes for the Flavors attribute within the Characteristics group.

```
ExposeRefinement exposeRefinement = new ExposeRefinement("/", "Flavors", "Characteristics");
```

LQLQueryConfig

Executes an EQL query on top of the current filter state.

LQLQuery has the following property:

Property	Description
LQLQuery	AST The EQL query to add. To retrieve the AST from the query string, call <code>DataSource.parseLQLQuery</code> .

The following example retrieves the average of the P_Price attribute grouped by Region:

```
Query query
= dataSource.parseLQLQuery("return mystatement as select avg(P_Price) as avgPrice group by Region",
true);
LQLQueryConfig lqlQueryConfig = new LQLQueryConfig(query);
```

NavConfig

Used to retrieve a navigation menu, such as in the **Available Refinements** component.

NavConfig has the following properties:

Property	Description
exposeAllRefinements	<p>Boolean</p> <p>Whether to display all of the available values for the attributes.</p> <p>Determines the initial state of the menu. The associated <code>ExposeRefinement</code> function is then applied.</p> <p>The default is false.</p>
List<RefinementGroupConfigs>	<p>List of groups for which to return the available attributes.</p> <p>If no <code>RefinementGroupConfigs</code> are specified, no attribute groups or attributes are returned.</p>

The following example returns attributes in the Source and Characteristics groups:

```
List<RefinementGroupConfig> refinementGroups = new ArrayList<RefinementGroupConfig>();
RefinementGroupConfig source = new RefinementGroupConfig();
source.setName("Source");
source.setExpose(true);
refinementGroups.add(source);
RefinementGroupConfig characteristics = new RefinementGroupConfig();
characteristics.setName("Characteristics");
characteristics.setExpose(true);
refinementGroups.add(characteristics);
NavConfig navConfig = new NavConfig();
navConfig.setRefinementGroupConfig(refinementGroups);
```

RecordDetailsConfig

Sends an attribute key-value pair to assemble the details for a selected record. The complete set of attribute-value pairs must uniquely identify the record.

RecordDetailsConfig has the following property:

Property	Description
recordSpecs	<p>List<RecordSpec></p> <p>Each new <code>RecordDetailsConfig</code> is appended to the previous <code>RecordDetailsConfig</code>.</p>

The following example sends the value of the P_WineID attribute:

```
List<RecordSpec> recordSpecs = new ArrayList<RecordSpec>();
recordSpecs.add(new RecordSpec("P_WineID", "37509"));
RecordDetailsConfig recordDetailsConfig = new RecordDetailsConfig(recordSpecs);
```

ResultsConfig

Used to manage the returned records. Allows for paging of the records.

ResultsConfig has the following properties:

Property	Description
recordsPerPage	Long The number of records to return at a time.
offset	Long (optional) The position in the list at which to start. The very first record is at position 0. For example, if recordsPerPage is 10, then to get the second page of results, the offset would be 10.
columns	String[] (optional) The columns to include in the results. If not specified, then the results include all of the columns.
numBulkRecords	Integer (optional) The number of records to return. Overrides the value of recordsPerPage.

The following example returns a selected set of columns for the third page of records, where each page contains 50 records:

```
ResultsConfig resultsConfig = new ResultsConfig();
resultsConfig.setOffset(100);
resultsConfig.setRecordsPerPage(50);
String[] columns = {"Wine_ID", "Name", "Description", "WineType", "Winery", "Vintage"};
resultsConfig.setColumns(columns);
```

ResultsSummaryConfig

Gets the number of records returned from a query.

```
ResultsSummaryConfig resultsSummaryConfig = new ResultsSummaryConfig();
```

SearchAdjustmentsConfig

Returns "Did you mean" and auto-correction items for a search.

```
SearchAdjustmentsConfig searchAdjustmentsConfig = new SearchAdjustmentsConfig();
```

SearchKeysConfig

Returns the list of available search interfaces.

```
SearchKeysConfig searchKeysConfig = new SearchKeysConfig();
```

SortConfig

Used to sort the results of a query. Used in conjunction with `ResultsConfig`.

`SortConfig` has the following properties:

Property	Description
<code>ownerId</code>	String (optional) The ID of the <code>ResultsConfig</code> that this <code>SortConfig</code> applies to. If not provided, uses the default <code>ResultsConfig</code> ID. If you configure a different ID, then you must provide a value for <code>ownerId</code> .
<code>property</code>	String The attribute to use for the sort.
<code>ascending</code>	Boolean Whether to sort in ascending order. If set to <code>false</code> , then the results are sorted in descending order.

For example, with the following `SortConfig`, the results are sorted by the `P_Score` attribute in descending order:

```
SortConfig sortConfig = new SortConfig("P_Score", false);
```

Creating a custom QueryFunction class

The Studio SDK directory includes scripts for creating new `QueryFunction` classes.



Note: Before you can create `QueryFunction` classes, you must install the Studio SDK, which is a separate download. See [Configuring the Studio SDK for component development on page 116](#).

To create a new `QueryFilter` or `QueryConfig` class:

1. In a terminal window, change to the `endeca-extensions` subdirectory of the Studio SDK's root directory (normally called `components`).
2. Run the appropriate command to create the `QueryFilter` or `QueryConfig` class.

To create a `QueryFilter` class:

Operating System	Command Syntax
Windows:	<code>.\create-queryfilter.bat <your-query-filter-name></code>
Linux:	<code>./create-queryfilter.sh <your-query-filter-name></code>

To create a `QueryConfig` class:

Operating System	Command Syntax
Windows:	<code>.\create-queryconfig.bat <your-query-config-name></code>
Linux:	<code>./create-queryconfig.sh <your-query-config-name></code>

The command creates in the `endeca-extensions` directory a new directory for the `QueryFilter` or `QueryConfig` class:

- For a `QueryFilter`, the directory is `<your-query-filter-name>-filter`.
- For a `QueryConfig`, the directory is `<your-query-config-name>-config`.

This directory is an Eclipse project that you can import directly into Eclipse, if you use Eclipse as your IDE.

It contains an empty sample implementation of a `QueryFilter` or `QueryConfig`. This has no effect on `QueryState` in its original form.

The skeleton implementation creates source files that:

- Extend either `QueryFilter` or `QueryConfig`.
- Create stubs for the `applyToDiscoveryServiceQuery`, `toString`, and `beforeQueryStateAdd` methods. `applyToDiscoveryServiceQuery` and `toString` are required methods that you must implement. `beforeQueryStateAdd` is an optional method to verify the query state before the function is added. This method is used to prevent invalid query states such as duplicate refinements.
- Create a no-argument, protected, empty constructor. The protected access modifier is optional, but recommended.
- Create a private member variable for logging.

Implementing a custom QueryFunction class

After you create your new `QueryFunction` class, you then implement it.

To implement your new `QueryFunction`, you must:

- Add private filter or configuration properties.
- Create getters and setters for any filter properties you add.
- Define a no-argument constructor (protected access modifier optional, but recommended).
- Implement the `applyToDiscoveryServiceQuery` method.

This method is called with the following arguments:

- The Conversation Service query
- A `stateName` string

Your custom function should use the Conversation Service API to apply itself to the conversation service query argument. See the *Endeca Server API Reference* for details.

The `stateName` argument provides the value to use for state name references in Conversation Service filters or content element configs that your custom function adds to the query.

- Implement the `toString` method, which is used to compare `QueryFunction` instances for equality.

`toString` should be consistent and deterministic in order to accurately determine if two instances of your custom `QueryFunction` are identical or distinct.

- Optionally, implement the `beforeQueryStateAdd(QueryState state)` method to check the current query state before the function is added.

Deploying a custom QueryFunction class

Before you can use your new `QueryFunction`, you must deploy it to Studio.

The directory that you created for the new `QueryFilter` or `QueryConfig` contains an ant build file.

The ant `deploy` task places a `.jar` file containing the custom `QueryFunction` into the `endeca-portal/tomcat-<version>/lib/ext` directory. Put the new `QueryFunction.jar` into the container's global classpath.

To deploy the new `QueryFunction`:

1. Run the ant build.
2. Restart Studio.

After you deploy your custom `QueryFunction`, you can use it in any component.

Adding the custom QueryFunction .jar file to the custom component Eclipse build path

If you are using Eclipse as your IDE, you need to add the new `.jar` file to the build path of your custom component.

To add the new `.jar` file to the Eclipse build path for your component:

1. Right-click the project, then select **Build Path>Configure Build Path**.
2. Click the **Libraries** tab.
3. Click **Add Variable**.
4. Select **DF_GLOBAL_LIB**.

You should have added this variable when you set up the Studio SDK.

5. Click **Extend**.
6. Open the `ext/` directory.
7. Select the `.jar` file containing your custom `QueryFunction`.
8. Click **OK**.

After adding the `.jar` file to the build path, you can import the class, and use your custom `QueryFilter` or `QueryConfig` to modify your `QueryState`.

Index

A

- application roles, assigning to members 100
- applications
 - application roles, assigning to members 100
 - application type, configuring 97
 - certifying 101
 - making active or inactive 102
 - members, adding 99
 - members, removing 99
 - removing 103
- application type, configuring for an application 97
- AttributeValueSearchConfig QueryConfig function 130

B

- BreadcrumbsConfig QueryConfig function 130

C

- components
 - adding to WebLogic .ear file 120
 - building and testing 120
 - build properties, modifying 121
 - creating 118
 - importing projects into Eclipse 119
 - Performance Metrics 20
 - query results, obtaining 119
 - removing from WebLogic .ear file 120
- Control Panel
 - adjusting logging levels 16
 - Data Source Library 81
 - Endeca Servers page 64
 - Provisioning Service page 78
- CSS
 - file location 110
 - updating for WebLogic 111

D

- DataSourceFilter QueryFunction class 122
- Data Source Library
 - attributes, configuring 88
 - data sources, adding 83
 - data sources, configuring access 94
 - data sources, editing 83
 - data sources, removing 95
 - data sources, selecting data 86
 - displaying 82
- data sources
 - access, configuring 94
 - adding 83
 - attributes, configuring 88
 - editing 83

- removing 95
- selecting data 86
- types of 81

DateFilter QueryFunction class 127

E

- Eclipse
 - configuring classpath variables 117
 - importing component projects 119
 - QueryFunctions, adding custom .jar file 137
- Endeca Server clustering, Endeca Server connection definition for 74
- Endeca Server connections
 - adding 66
 - connecting to an Endeca Server cluster 74
 - connecting to a secured Endeca Server 72
 - deleting 68
 - description setting 70
 - editing 67
 - escaping special characters 69
 - host settings 69
 - identifier, configuring 69
 - idle data domains, handling of 65
 - name setting 70
 - restricting who can bind applications to 75
 - role-based security for viewing data 71
 - shared, setting up 77
 - syntax 68
 - testing 68
- Endeca Servers page
 - about 64
 - displaying 64
 - Endeca Server connection, adding 66
 - Endeca Server connection, deleting 68
 - Endeca Server connection, editing 67
 - Endeca Server connection, testing 68
- ExposeRefinement QueryConfig function 131

F

- framework settings
 - about 10
 - configuring from the Control Panel 10
 - configuring in portal-ext.properties 14

G

GeoFilter QueryFunction class 128

I

- images
 - file location 110
 - updating for WebLogic 111

J

JSON Endeca Server connection syntax 68

L

LastNDateFilter QueryFunction class 127

LDAP integration

- assigning roles to groups 51
- configuring password policy 49
- configuring settings 44
- configuring the server connection 44
- preventing passwords from being stored 50

locales

- configuring available 32
- configuring Studio default 33
- configuring user preferred 33
- effect of selection on Studio 29
- including in a URL 35
- list of supported 29
- locations where set 30
- scenarios for determining 31

log files

- metrics log file 19
- Studio log file 16

logging

- adjusting verbosity from the Control Panel 16
- configuration XML files 15
- log files 16

look and feel, customizing 110

LQLQueryConfig QueryConfig function 132

M

Most/Least Frequently Accessed Summary Report, viewing 26

N

NavConfig QueryConfig function 132

NegativeRefinementFilter QueryFunction class 124

Number of Users by Date report, viewing 25

P

pages

- exporting 104
- importing 106
- visibility type, configuring 98

Performance Metrics

- about 20
- configuring 18

Provisioning Service, configuring the connection to 78

Q

QueryConfig functions

- AttributeValueSearchConfig 130

BreadcrumbsConfig 130

ExposeRefinement 131

LQLQueryConfig 132

NavConfig 132

RecordDetailsConfig 133

ResultsConfig 133

ResultsSummaryConfig 134

SearchAdjustmentsConfig 134

SearchKeysConfig 134

SortConfig 135

QueryFunction classes

- creating custom 135
- deploying custom 137
- implementing custom 136
- jar files, adding to Eclipse 137

QueryFunction filter classes

- DataSourceFilter 122
- DateFilter 127
- GeoFilter 128
- LastNDateFilter 127
- NegativeRefinementFilter 124
- RangeFilter 125
- RefinementFilter 123
- SearchFilter 128

R

RangeFilter QueryFunction class 125

RecordDetailsConfig QueryConfig function 133

RefinementFilter QueryFunction class 123

ResultsConfig QueryConfig function 133

ResultsSummaryConfig QueryConfig function 134

roles

- about 37
- assigning to groups 51

S

SearchAdjustmentsConfig QueryConfig function 134

SearchFilter QueryFunction class 128

SearchKeysConfig QueryConfig function 134

Security Manager

- class summary 112
- configuring Studio to use 114
- creating 113
- deploying 114
- implementing 113

single sign-on

See SSO

SortConfig QueryConfig function 135

SSO

- about 53
- configuring Oracle Access Manager settings in Studio 59
- configuring portal-ext.properties 60
- configuring the LDAP connection 57
- overview of the integration process 53
- registering the Webgate 55

- reverse proxy configuration, WebLogic Server 54
- testing the OHS URL 57
- Studio SDK
 - components, modifying build properties 121
 - configuring 116
 - configuring Eclipse for 117
 - Ext JS license requirement 115
 - software requirements 115
- syntax for Endeca Server connections 68
- System Usage
 - logging, enabling 23
 - Most/Least Frequently Accessed Summary Report 26
 - Number of Users by Date report 25
 - reports, about 25

U

- users
 - creating 40
 - deactivating 42
 - deleting 42
 - editing 41
 - reactivating 42
 - removing the Power User role 42

V

- visibility type, configuring for a page 98