

IKM SQL to Endeca Server

Installation and Usage Guide

Version 3.1.0 Rev. A • December 2013

Copyright and disclaimer

Copyright © 2003, 2014, 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	4
About this guide	4
Who should use this guide	4
Conventions used in this guide	4
Contacting Oracle Customer Support	5
Chapter 1: Installing the IKM SQL to Endeca Server	6
Endeca Server version support	6
Prerequisite software	6
Prerequisite setup	6
Downloading the IKM SQL to Endeca Server	7
Installing the IKM SQL to Endeca Server	7
Package contents and directory structure	8
Chapter 2: Configuring the IKM SQL to Endeca Server	10
Importing the Endeca Server technology into ODI	10
Adding flexfield definitions	11
Creating an Endeca Server data server	13
Creating a physical schema	15
Creating a logical schema	15
Creating a model	17
Creating a datastore	19
Modifying column settings in a datastore	21
Importing the knowledge module	22
Creating an interface	23
Configuration reference for the IKM SQL to Endeca Server	27
Mapping source data types to Endeca Server data types	30
Chapter 3: Using the ODI Procedure to Write to Endeca Server	34
Configuring and running the ODI procedure	34
Configuration reference for the ODI procedure	38
Chapter 4: Uninstalling the IKM SQL to Endeca Server	41
Uninstalling the IKM SQL to Endeca Server	41

Preface

Oracle® Endeca Information Discovery Integrator is a powerful visual data integration environment that includes:

The Information Acquisition System (IAS) for gathering content from delimited files, file systems, JDBC databases, and Web sites.

Integrator ETL, an out-of-the-box ETL purpose-built for incorporating data from a wide array of sources, including Oracle BI Server.

In addition, Oracle Endeca Web Acquisition Toolkit is a Web-based graphical ETL tool, sold as an add-on module. Text Enrichment and Text Enrichment with Sentiment Analysis are also sold as add-on modules. Connectivity to data is also available through Oracle Data Integrator (ODI).

About this guide

This guide describes how to download, install, configure, and run the IKM SQL to Endeca Server. The IKM SQL to Endeca Server provides an integration module that enables writing source data to an Endeca Server target within ODI.

Who should use this guide

This guide is intended for data developers who are using Oracle Data Integrator to write data to an Endeca Server target.

Conventions used in this guide

The following conventions are used in this document.

Typographic conventions

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

Typographic conventions

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.
<Variable Name>	This formatting is used for variable values, such as <install path>.
File Path	This formatting is used for file names and paths.

Symbol conventions

The following table describes symbol conventions used in this document.

Symbol conventions

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>.



Chapter 1

Installing the IKM SQL to Endeca Server

This section describes the prerequisites to installing the IKM SQL to Endeca Server into Oracle Data Integrator, the installation procedure, and then describes the contents of the installation directory.

[Endeca Server version support](#)

[Prerequisite software](#)

[Prerequisite setup](#)

[Downloading the IKM SQL to Endeca Server](#)

[Installing the IKM SQL to Endeca Server](#)

[Package contents and directory structure](#)

Endeca Server version support

The IKM SQL to Endeca Server 3.1.0 supports writing data to Endeca Server 7.6.0.

Prerequisite software

Before you install the IKM SQL to Endeca Server into Oracle Data Integrator, you must have already installed the following software on your machine:

- Oracle Data Integrator 11g (11.1.1.x)



Note: Oracle Data Integrator has additional software prerequisites including WebLogic Server, Java Platform Standard Edition Development Kit (JDK), and Oracle Database 11g. For details about ODI requirements, see http://docs.oracle.com/html/E18558_01/fusion_requirements.htm.

- Endeca Server 7.6.0

Prerequisite setup

After you have installed the prerequisite software, you must perform the following before you install the IKM SQL to Endeca Server:

1. Start the ODI agent.
2. Start ODI Studio.
3. Connect to a master repository. (For more information about connecting, see [creating and connecting to ODI repositories](#).)

4. Connect to a work repository.
5. Create an ODI project.

At this point, you can install the IKM SQL to Endeca Server.

Downloading the IKM SQL to Endeca Server

You download the IKM SQL to Endeca Server from the Oracle Software Delivery Cloud.

To download the IKM SQL to Endeca Server:

1. Log in to <https://edelivery.oracle.com>.
2. Accept the terms and restrictions.
3. On the Media Pack Search page:
 - (a) From the **Select a Product Pack** drop-down list, select **Oracle Endeca**.
 - (b) From the **Platform** drop-down list, select the platform you are installing on.
 - (c) Click **Go**.

The list of available media packs displays. For the IKM SQL to Endeca Server, there is one media pack for both Windows and UNIX platforms.

4. Download the ZIP file for the IKM SQL to Endeca Server and save the file locally.

Installing the IKM SQL to Endeca Server

This process describes how to install the drivers for the knowledge module and also the XML definition of Endeca Server. Later in this guide, you import the knowledge module (i.e. the `ikm` directory contents). The knowledge module is project dependent and cannot be imported until you first create and configure a project.

To install the IKM SQL to Endeca Server:

1. In your local environment, locate the software that you downloaded from the Oracle Software Delivery Cloud.
2. Extract the ZIP file to the root of your hard drive.
For example, on Windows, extract the ZIP file to `C:\`.
This creates the files on disk at `C:\Oracle\Endeca\IKMSQLtoEndecaServer`.
3. If the ODI local agent is running, stop the process.
4. On the file system, browse to `<install path>\IKMSQLtoEndecaServer\3.1.0\lib` and copy all of the JAR files in the directory.
The destination location for the JAR files varies depending upon the type of ODI agent you are running.
5. If your ODI agent is a Java application running in WebLogic Server, do the following:
 - (a) Paste the JAR files into the `lib` directory of the WebLogic domain running the agent.
For example, `C:\Oracle\Middleware\user_projects\domains\odiagent\lib`.

- (b) Go to the `bin` directory of the WebLogic domain and open `setDomainEnv.cmd` (on Windows) or `setDomainEnv.sh` (on UNIX) in a text editor.
- (c) On Windows systems, locate the REM section and add a new Java option named `set JAVA_OPTIONS=-DUseSunHttpHandler=true` as the first lined in the file.

For example, here is a snippet of `setDomainEnv.cmd` with the new option:

```
...
@REM For additional information, refer to "Managing Server Startup and Shutdown for
Oracle WebLogic Server"
@REM (http://download.oracle.com/docs/cd/E23943_01/web.1111/e13708/overview.htm).
@REM *****
set JAVA_OPTIONS=-DUseSunHttpHandler=true

set WL_HOME=C:\Oracle\Middleware\wlserver_10.3
for %%i in ("%WL_HOME%") do set WL_HOME=%%~fsi

set BEA_JAVA_HOME=
...
```

- (d) On UNIX systems, locate the comments section and add `JAVA_OPTIONS=-DUseSunHttpHandler=true` and another line with `export JAVA_OPTIONS`.

For example, here is a snippet of `setDomainEnv.sh` with the new options:

```
...
#
# For additional information, refer to "Managing Server Startup and Shutdown for Oracle
WebLogic Server"
# (http://download.oracle.com/docs/cd/E23943_01/web.1111/e13708/overview.htm).
# *****
JAVA_OPTIONS=-DUseSunHttpHandler=true
export JAVA_OPTIONS
...
```

- (e) Save and close `setDomainEnv`.
6. If your ODI agent is a standalone agent, paste the JAR files into the `drivers` directory of ODI located in `<install path>\Oracle\Middleware\Oracle_ODI1\oracledi\agent\drivers` (or the equivalent path on UNIX).
 7. Restart the ODI agent.
 8. Start ODI Studio and connect to a Work repository with appropriate user name and password values.

Installation is complete.

Go on to import the Endeca Server technology into ODI.

Package contents and directory structure

The `IKMSQLtoEndecaServer` directory is the root directory for the IKM SQL to Endeca Server.

The extracted software has the following directory structure on disk:

```
IKMSQLtoEndecaServer
 3.1.0
  ikm
  lib
  procedure
  technology
```


tools

The contents of the `IKMSQLtoEndecaServer` directory are described here in detail.

Directory	Contents
<code><version>\ikm</code>	Contains knowledge module.
<code><version>\lib</code>	Contains the JAR files that implement the knowledge module, the JAR files for logging, and the JAR for the Web Service bulk load module.
<code><version>\procedure</code>	Contains a Web Service procedure to invoke Endeca Server Web Services.
<code><version>\technology</code>	Contains an XML definition of Endeca Server for ODI. (The Endeca Server technology.)
<code><version>\tools</code>	Contains the command line utility to create and load Endeca flexfields into ODI. Also contains a configuration file for the utility.



Chapter 2

Configuring the IKM SQL to Endeca Server

The following procedures describe how to configure and run the IKM SQL to Endeca Server in ODI. Perform the following procedures in order. After you complete the procedures, you can run an ODI Interface that integrates data into an Endeca Server.

Importing the Endeca Server technology into ODI

Adding flexfield definitions

Creating an Endeca Server data server

Creating a physical schema

Creating a logical schema

Creating a model

Creating a datastore

Modifying column settings in a datastore

Importing the knowledge module

Creating an interface

Configuration reference for the IKM SQL to Endeca Server

Mapping source data types to Endeca Server data types

Importing the Endeca Server technology into ODI

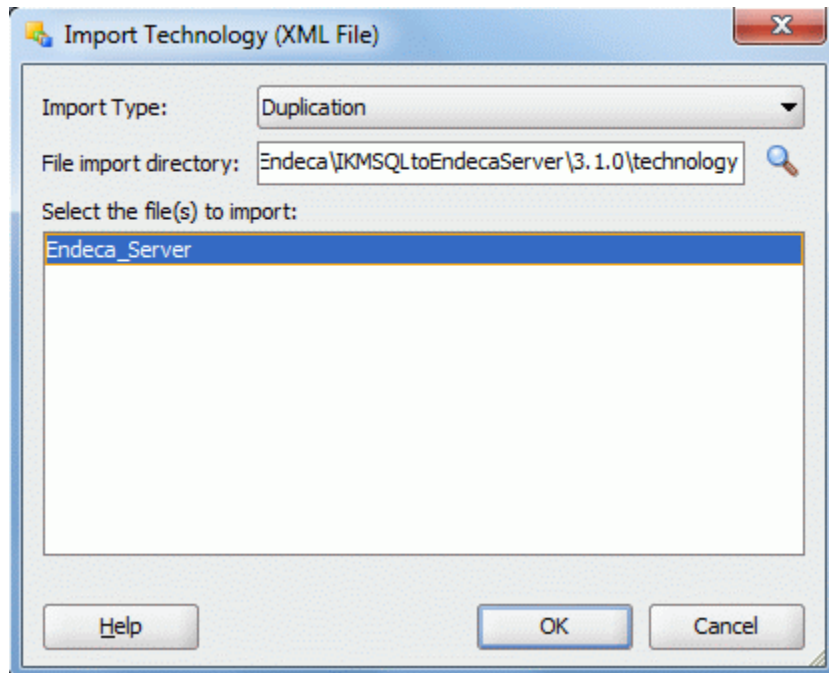
You use ODI Studio to import the Endeca Server technology into ODI.

To import the Endeca Server technology into ODI:

1. In ODI Studio, select the **Topology** tab.
2. Expand **Physical Architecture**.
3. Right-click **Technologies** and select **Import Technology**.
4. In the **Import Technology (XML File)** dialog, leave **Import Type** set to **Duplication**.
5. In **File to import**, browse to <install path>\IKMSQLtoEndecaServer\technology and click **Open**.

6. Select **Endeca_Server** and click **OK**.

For example:



The **Import Report** dialog displays with a row for Endeca Server.

7. Click **Close**.

Adding flexfield definitions

You must add flexfield definitions to ODI to define Endeca Server fields. As a convenience, the IKM SQL to Endeca Server provides a command line utility, named `metadata-import`, to create the flexfields into ODI. The `metadata-import` utility requires a few configuration settings for SDK, Java, and driver locations.

When you run `metadata-import`, it prompts you for a number of database and repository connection values. Optionally, you can set these connection values in `OdiRepository.properties`, and then `metadata-import` reads those values instead of prompting for them.

The `metadata-import` utility creates the following flexfields:

- Endeca Property IsValueSearchable
- Endeca Property IsSingleAssign
- Endeca Property IsTextSearchable
- Endeca Property IsUnique
- Endeca Property TextSearchWildcards
- Endeca Property Navigation Select
- Endeca Property ShowRecordCounts
- Endeca Property Navigation Sorting

- Endeca Property Attribute Group
- Endeca Property Search Interface
- Endeca Property Display Name
- Endeca Property Language

To add flexfield definitions to ODI for Endeca Server:

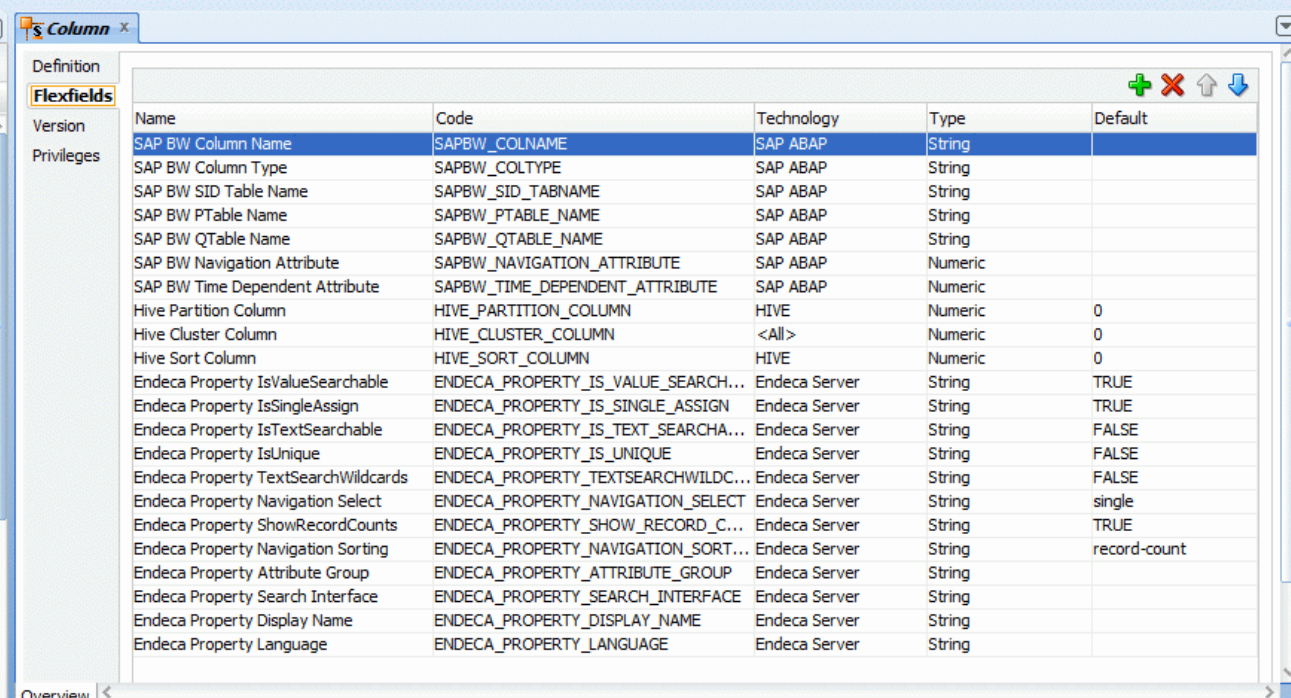
1. On the file system, navigate to `<install path>\IKMSQLtoEndecaServer\3.1.0\tools` directory.
2. Open `metadata-import` (the `.bat` file on Windows or the `.sh` file on UNIX) in a text editor and make the following changes:
 - (a) For the `ODI_SDK` setting, specify the full path to the Oracle ODI SDK.
For example, on Windows the default path is
`C:\Oracle\Middleware\Oracle_ODI1\oracledi.sdk.`
 - (b) For the `ODI_JDBC_DRIVER_JAR` setting, specify the full path to the JDBC driver for Oracle ODI.
For example, on Windows the default path is
`C:\Oracle\Middleware\Oracle_ODI1\modules\oracle.jdbc_11.1.1\ojdbc6dms.jar`
 - (c) For the `JAVA_LOCATION` setting, specify the full path to the Java installation.
For example, `C:\Java\jdk1.6.0_43.`
 - (d) Save and close `metadata-import`.
3. Also in `<install path>\IKMSQLtoEndecaServer\3.1.0\tools`, open `OdiRepository.properties` in a text editor and make the following changes:
 - (a) For the `ODI_MASTER_DRIVER` setting, specify the name of the JDBC driver for the ODI repository.
For example, the default driver is `oracle.jdbc.OracleDriver.`
 - (b) For the `ODI_MASTER_URL` setting, specify the JDBC connection URL for ODI repository.
 - (c) For the `ODI_MASTER_USER` setting, specify the user name for ODI repository database.
For example, `odi.`
 - (d) For the `ODI_WORK_REP` setting, specify the work repository name.
For example, `WORKREP1.`
 - (e) For the `ODI_SUPERVISOR` setting, specify the work repository login ID.
For example, `SUPERVISOR.`
 - (f) Save and close `OdiRepository.properties`.

Remember, if you do not set the properties in `OdiRepository.properties`, the `metadata-import` utility prompts for them at runtime.
4. Start a command prompt and change to `<install path>\IKMSQLtoEndecaServer\3.1.0\tools`.
5. Run `metadata-import` and specify `OdiRepository.properties` as an argument.
For example:

```
C:\Oracle\Endeca\IKMSQLtoEndecaServer\3.1.0\tools>metadata-import.bat
OdiRepository.properties
```

6. After the script completes, confirm that the flexfields loaded correctly by doing the following:
 - (a) In ODI Studio, select the **Security** tab.
 - (b) Expand **SECURITY ADMIN**.
 - (c) Double click **Column**.
 - (d) On the **Column** tab, select **Flexfields**.

You see the following:



Name	Code	Technology	Type	Default
SAP BW Column Name	SAPBW_COLNAME	SAP ABAP	String	
SAP BW Column Type	SAPBW_COLTYPE	SAP ABAP	String	
SAP BW SID Table Name	SAPBW_SID_TABNAME	SAP ABAP	String	
SAP BW PTable Name	SAPBW_PTABLE_NAME	SAP ABAP	String	
SAP BW QTable Name	SAPBW_QTABLE_NAME	SAP ABAP	String	
SAP BW Navigation Attribute	SAPBW_NAVIGATION_ATTRIBUTE	SAP ABAP	Numeric	
SAP BW Time Dependent Attribute	SAPBW_TIME_DEPENDENT_ATTRIBUTE	SAP ABAP	Numeric	
Hive Partition Column	HIVE_PARTITION_COLUMN	HIVE	Numeric	0
Hive Cluster Column	HIVE_CLUSTER_COLUMN	<All>	Numeric	0
Hive Sort Column	HIVE_SORT_COLUMN	HIVE	Numeric	0
Endeca Property IsValueSearchable	ENDECA_PROPERTY_IS_VALUE_SEARCH...	Endeca Server	String	TRUE
Endeca Property IsSingleAssign	ENDECA_PROPERTY_IS_SINGLE_ASSIGN	Endeca Server	String	TRUE
Endeca Property IsTextSearchable	ENDECA_PROPERTY_IS_TEXT_SEARCHA...	Endeca Server	String	FALSE
Endeca Property IsUnique	ENDECA_PROPERTY_IS_UNIQUE	Endeca Server	String	FALSE
Endeca Property TextSearchWildcards	ENDECA_PROPERTY_TEXTSEARCHWILDC...	Endeca Server	String	FALSE
Endeca Property Navigation Select	ENDECA_PROPERTY_NAVIGATION_SELECT	Endeca Server	String	single
Endeca Property ShowRecordCounts	ENDECA_PROPERTY_SHOW_RECORD_C...	Endeca Server	String	TRUE
Endeca Property Navigation Sorting	ENDECA_PROPERTY_NAVIGATION_SORT...	Endeca Server	String	record-count
Endeca Property Attribute Group	ENDECA_PROPERTY_ATTRIBUTE_GROUP	Endeca Server	String	
Endeca Property Search Interface	ENDECA_PROPERTY_SEARCH_INTERFACE	Endeca Server	String	
Endeca Property Display Name	ENDECA_PROPERTY_DISPLAY_NAME	Endeca Server	String	
Endeca Property Language	ENDECA_PROPERTY_LANGUAGE	Endeca Server	String	

Creating an Endeca Server data server

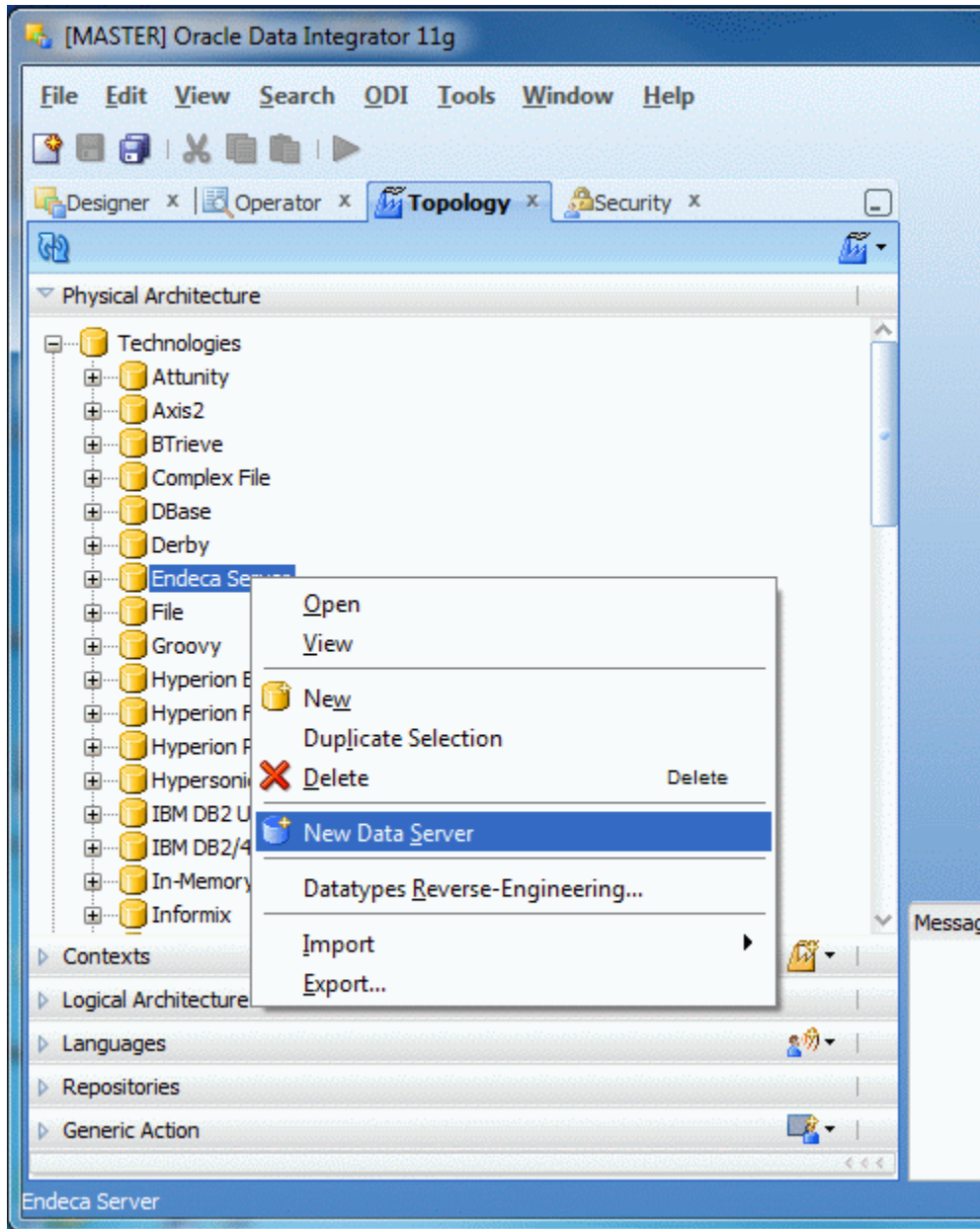
Next, you create an Endeca Server data server.

To create an Endeca Server technology:

1. In ODI Studio, select the **Topology** tab.
2. Expand **Physical Architecture>Technologies**.

3. Right click **Endeca Server** and select **New Data Server**.

For example:



4. Select the **Definition** pane and specify a unique value for **Name**.
5. Select the **JDBC** pane and specify a value for **JDBC Url**.

This value must include an http prefix, the machine name, port number, and the context path for Endeca Server. For example: `http://<es server>:<es port>/<es context path>`. Also, leave the **JDBC Driver** field empty.

6. Click **Save**.

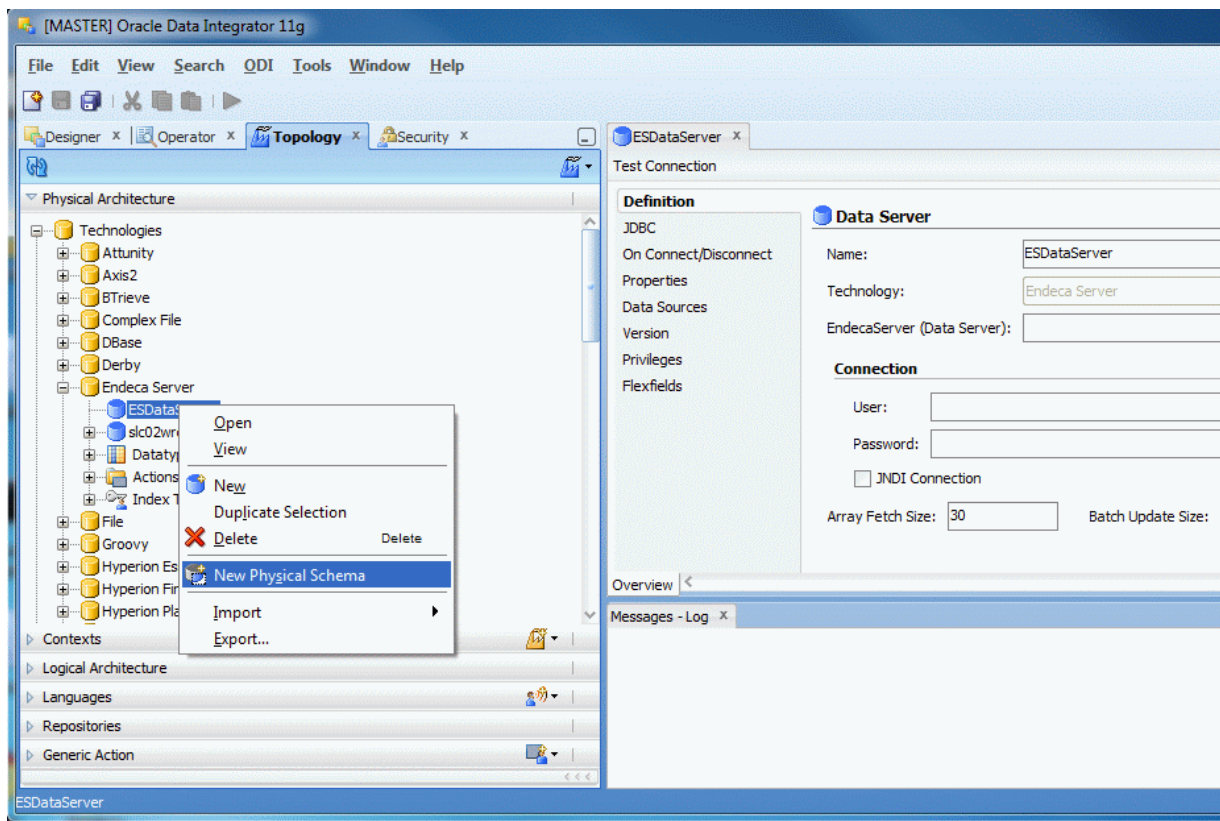
Creating a physical schema

After you have created an Endeca Server technology, you create a physical schema to associate with it.

To create a physical schema:

1. In ODI Studio, select the **Topology** tab.
2. Expand **Physical Architecture>Technologies>Endeca Server**.
3. Right click the Endeca Server you created earlier and select **New Physical Schema**.

For example:



4. Select the **Definition** pane.
5. Leave all the default values and click **Save**.

Creating a logical schema

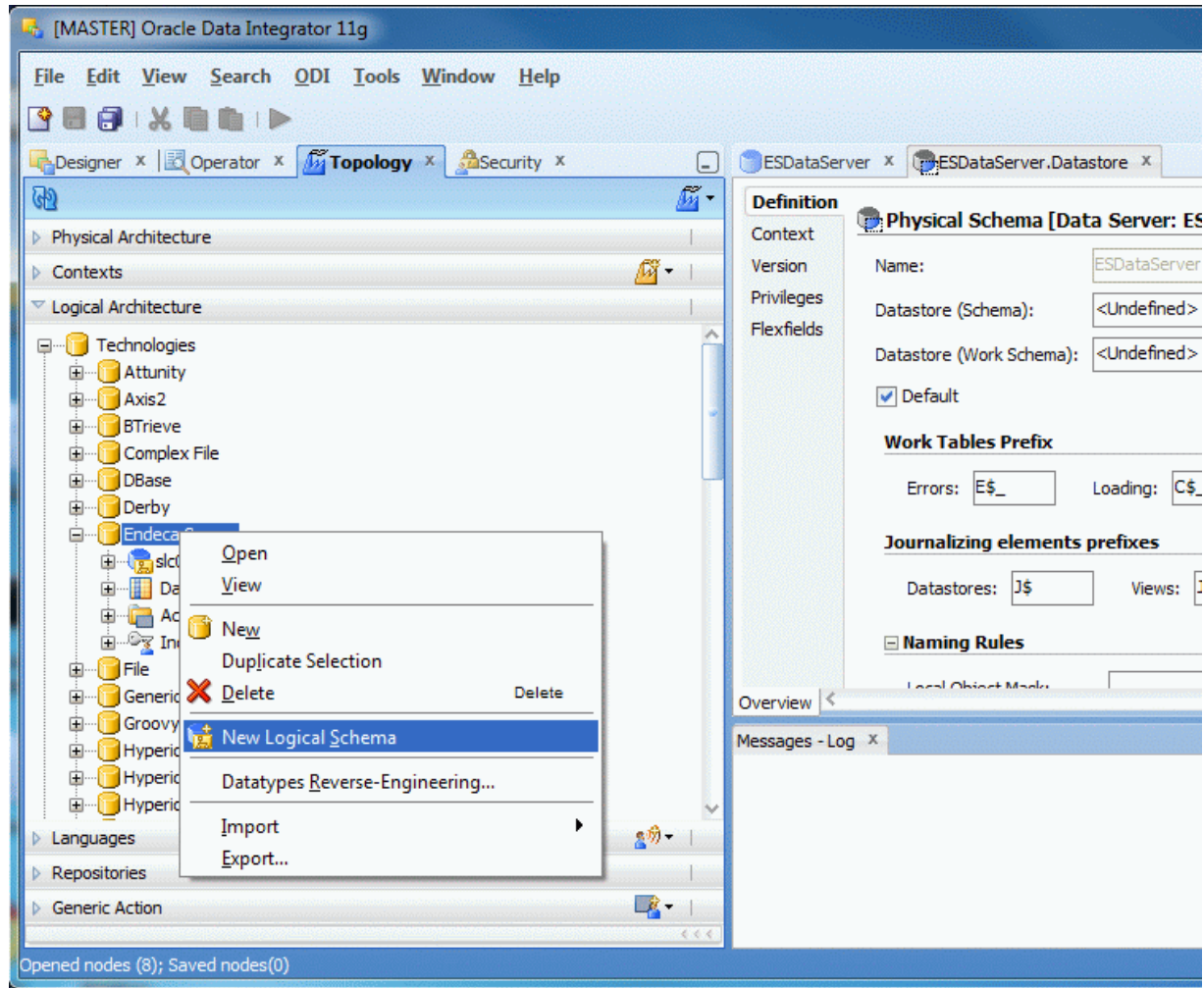
After you have created an Endeca Server, you create a logical schema to associate with it.

To create a logical schema:

1. In ODI Studio, select the **Topology** tab.
2. Expand **Logical Architecture>Technologies>Endeca Server**.

3. Right click **Endeca Server** and select **New Logical Schema**.

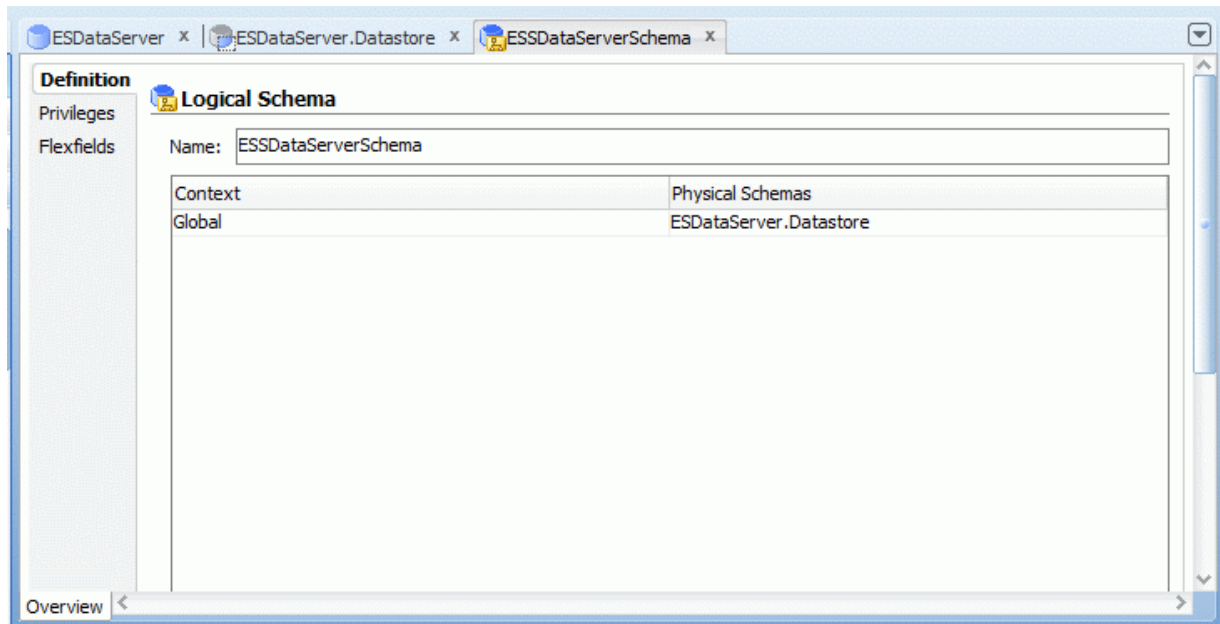
For example:



4. Select the **Definition** pane and specify a unique name.

5. From the **Physical Schemas** list, select a schema for this context.
(This is typically the schema you created in the previous procedure.)

For example:



6. Click **Save**.

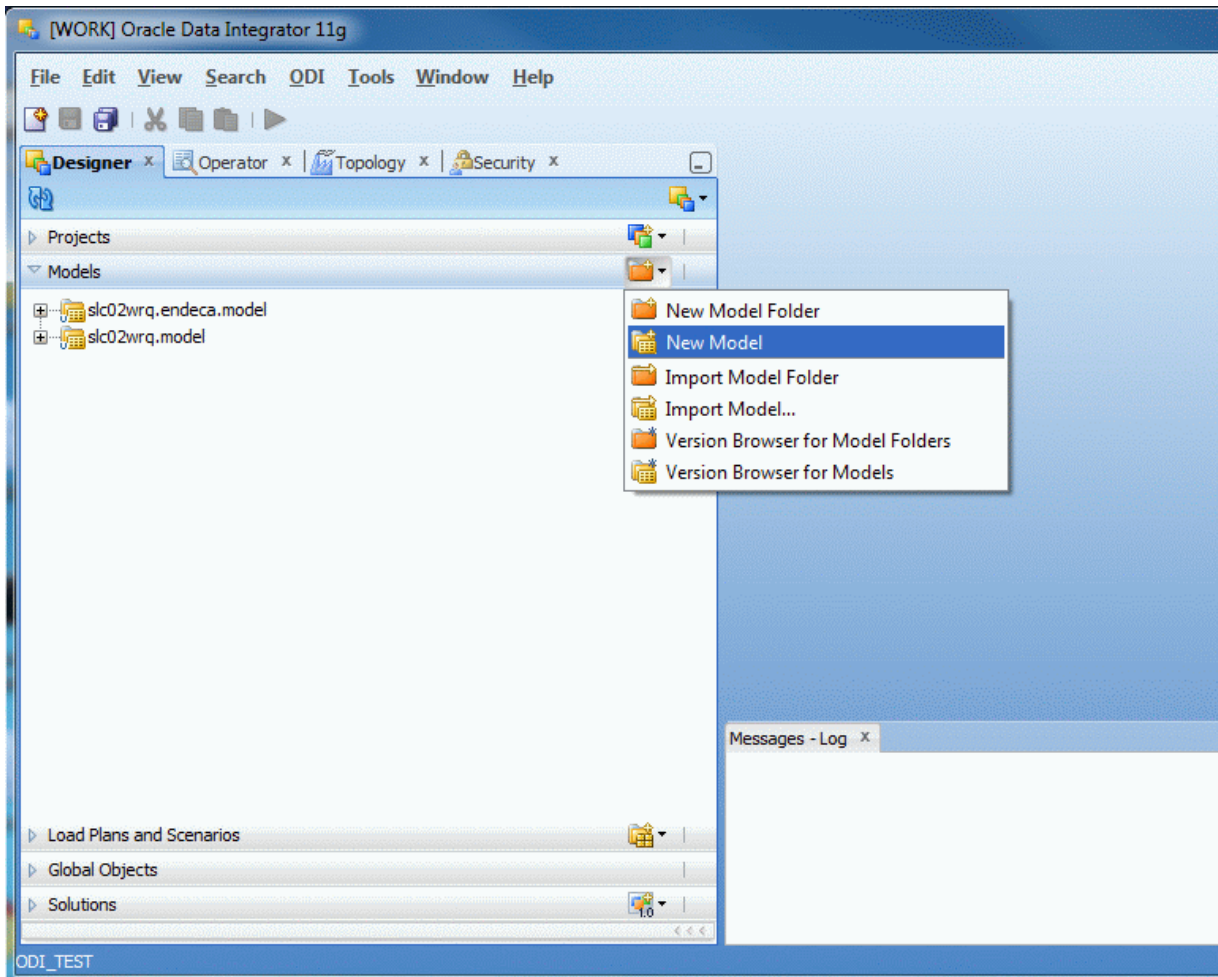
Creating a model

Next, you create a model.

To create a model:

1. In ODI Studio, select the **Designer** tab.
2. Expand **Models**.

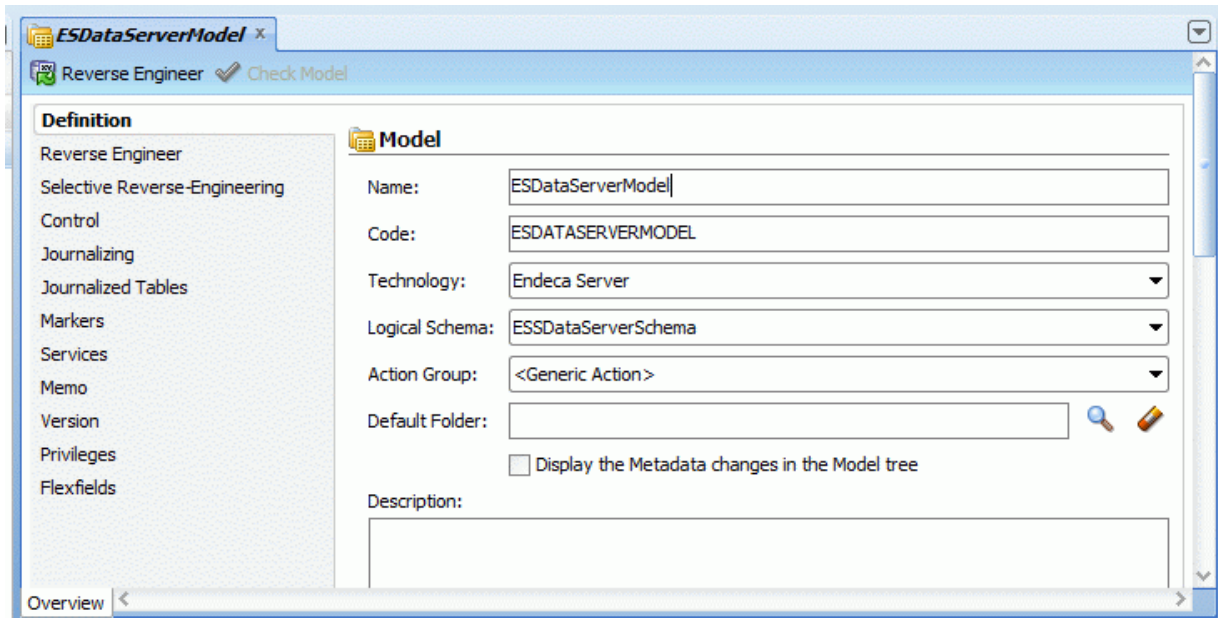
3. Select **New Model**.
For example:



4. Select the **Definition** pane and specify a unique value for **Name**.
5. From the **Technology** list, select **Endeca Server**.

- From the **Logical Schema** list, select the logical schema that you created earlier.

For example:



- Click **Save**.

Creating a datastore

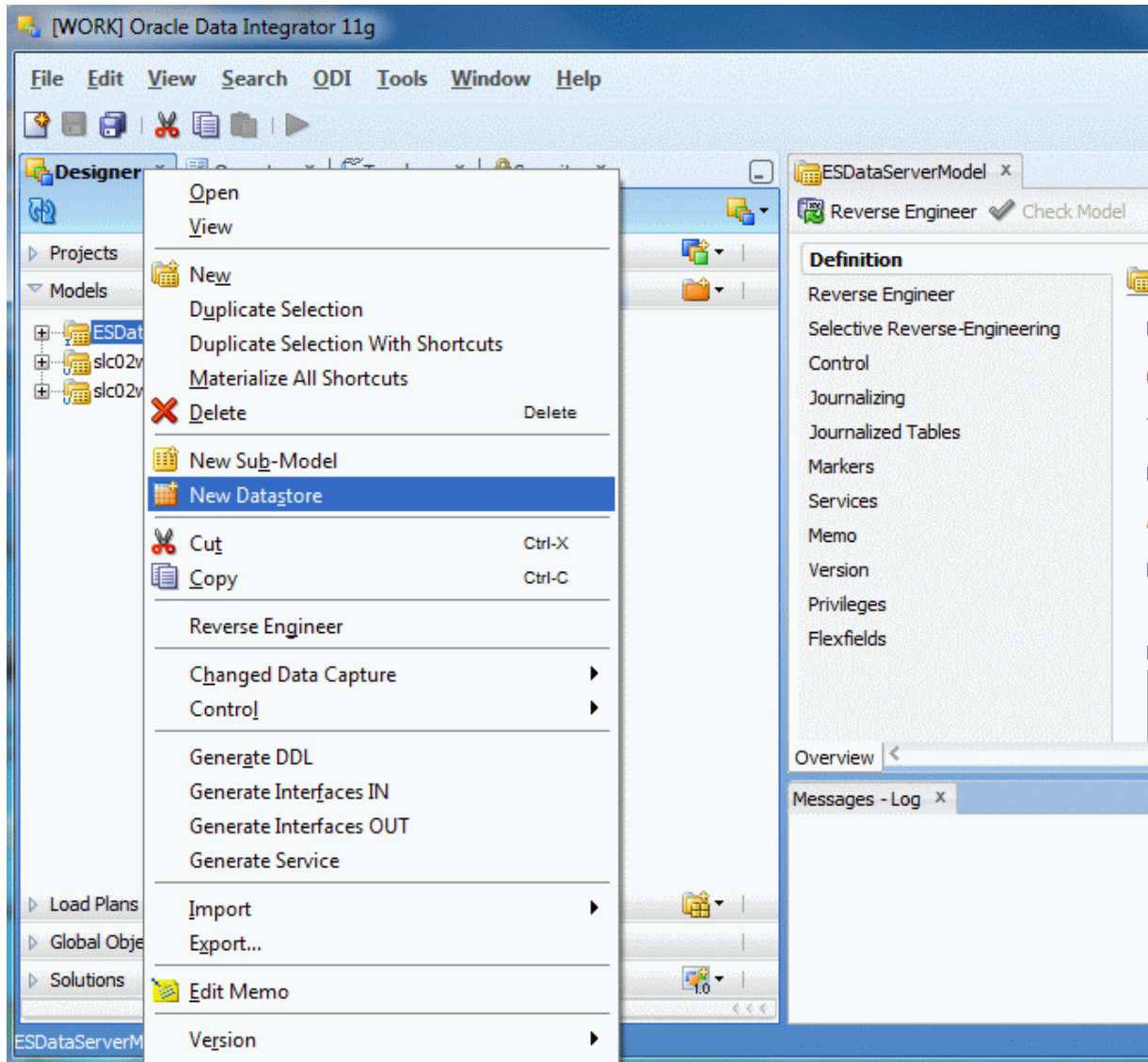
Next, you create a datastore.

To create a datastore:

- In ODI Studio, select the **Designer** tab.
- Expand **Models**.

- Right click the model you created earlier and select **New Datastore**.

For example:



- Select the **Definition** pane and specify a unique value for **Name**.
- Select the **Columns** pane.
- Click the "+" (Add) icon.
 - In the **Name** field, specify a unique name for the column value.
 - In the **Type** field, select an Endeca data type from the list.
- Repeat step 6 as necessary to create all columns required in your datastore.

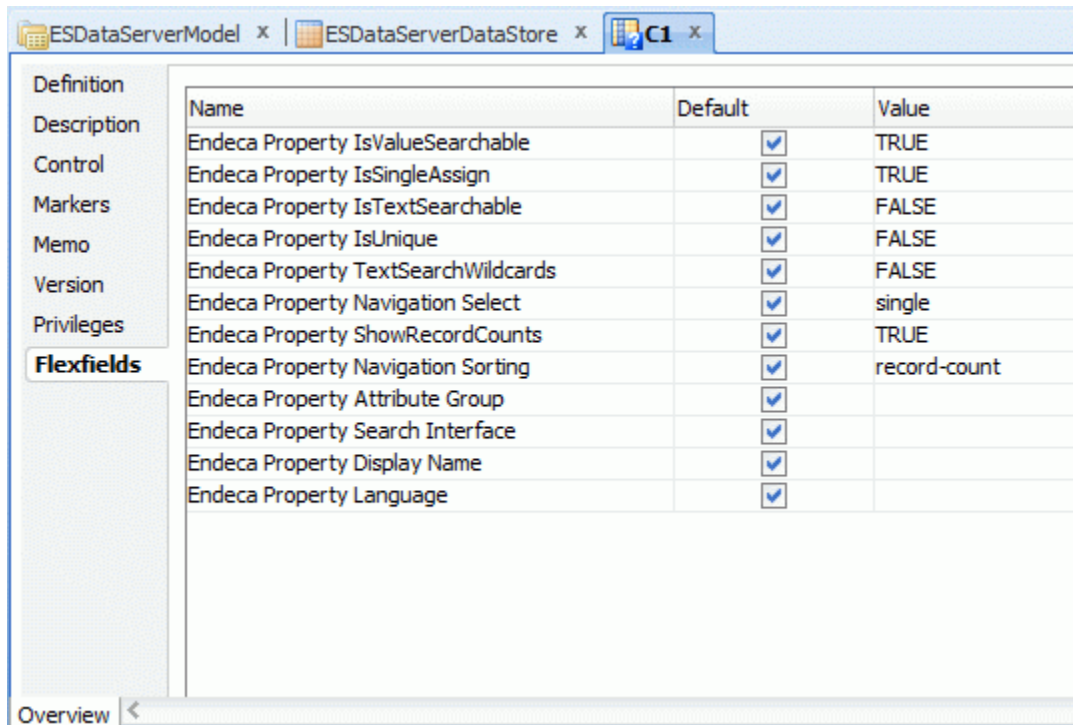
Modifying column settings in a datastore

Once you have created a datastore and its columns, you can assign Endeca properties to each column in the datastore. The Endeca flexfield properties configure the column values with Endeca search settings. For example, the Endeca properties define whether the value of a column is text searchable, value searchable, single assign or multi-assign, and so on. The definition of each Endeca property is documented in the "Property Description Record (PDR)" topic of the *Oracle Endeca Server Developer's Guide*.

To modify column settings in a datastore:

1. In ODI Studio, select the **Designer** tab.
2. Expand **Models**.
3. Expand the datastore that you created earlier.
4. Expand **Columns**.
5. Double-click a column name.
 - (a) Select the **Flexfields** pane.
 - (b) In the **Default** field, enable or disable each Endeca flexfield as appropriate for that column.

For example:



6. Repeat step 5 as necessary to configure each column in your datastore with Endeca flexfield properties.

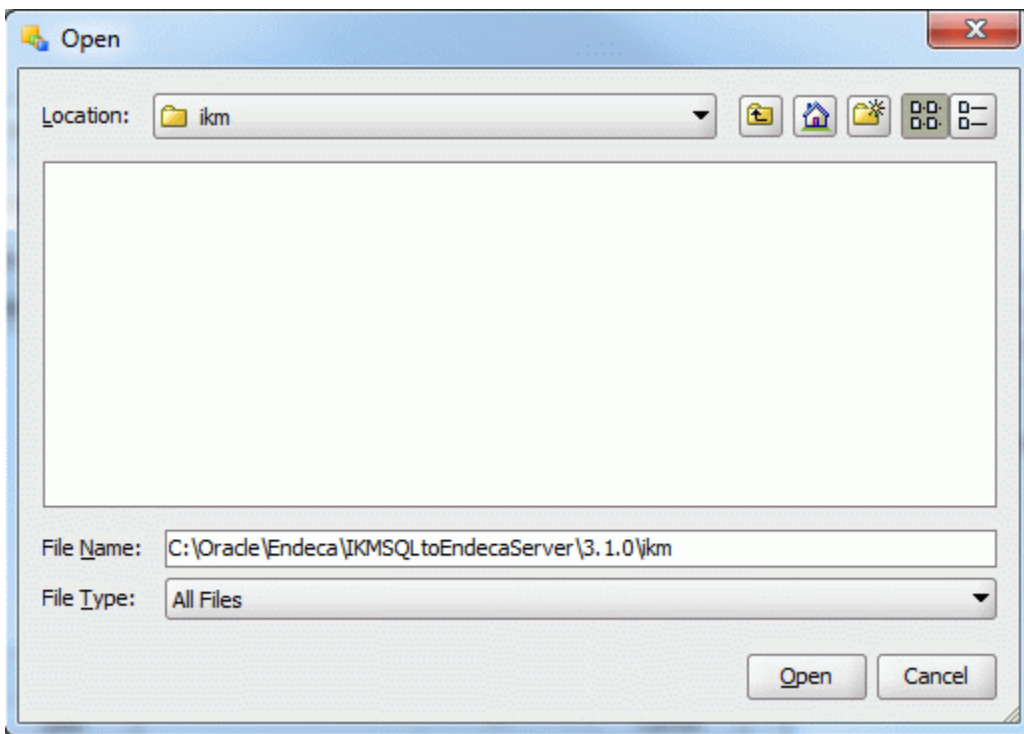
Importing the knowledge module

In this procedure, you import the knowledge module contained in the `<install path>\ikm_sql_to_endeca_server\3.1.0\ikm` directory. This module is project dependent and cannot be imported until you first create and configure a project.

To import the knowledge module into ODI:

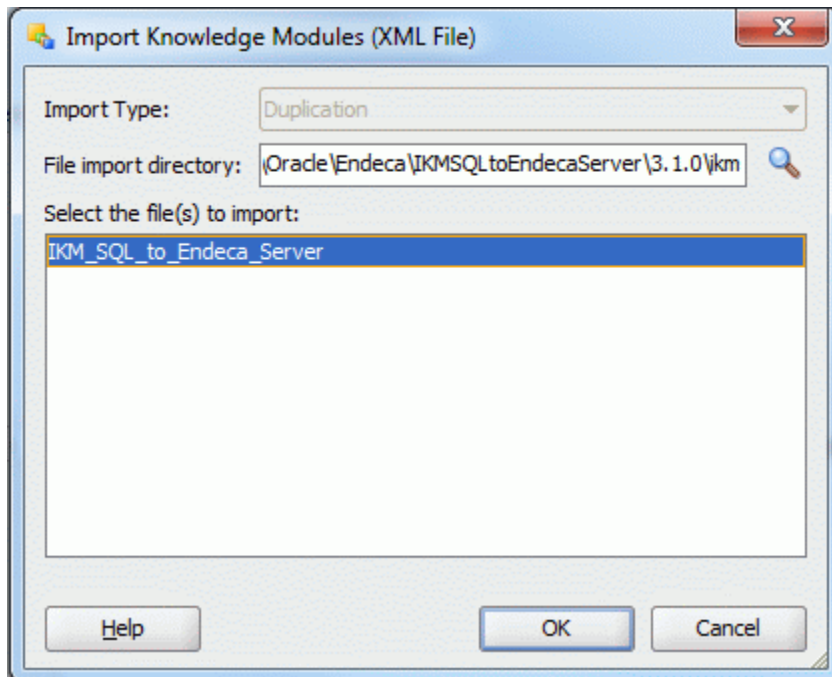
1. In ODI Studio, select the **Designer** tab.
2. Expand **Projects**.
3. Select your project and right click **Knowledge Modules**.
4. Select **Import Knowledge Modules...**
5. In the **Import Technology (XML File)** dialog, browse to `<install path>\IKMSQLtoEndecaServer\ikm` and click **Open**.

For example:



6. Select **IKM_SQL_Endeca_Server** and click **OK**.

For example:



The **Import Report** dialog displays with a row for IKM SQL to Endeca Server.

7. Click **Close**.

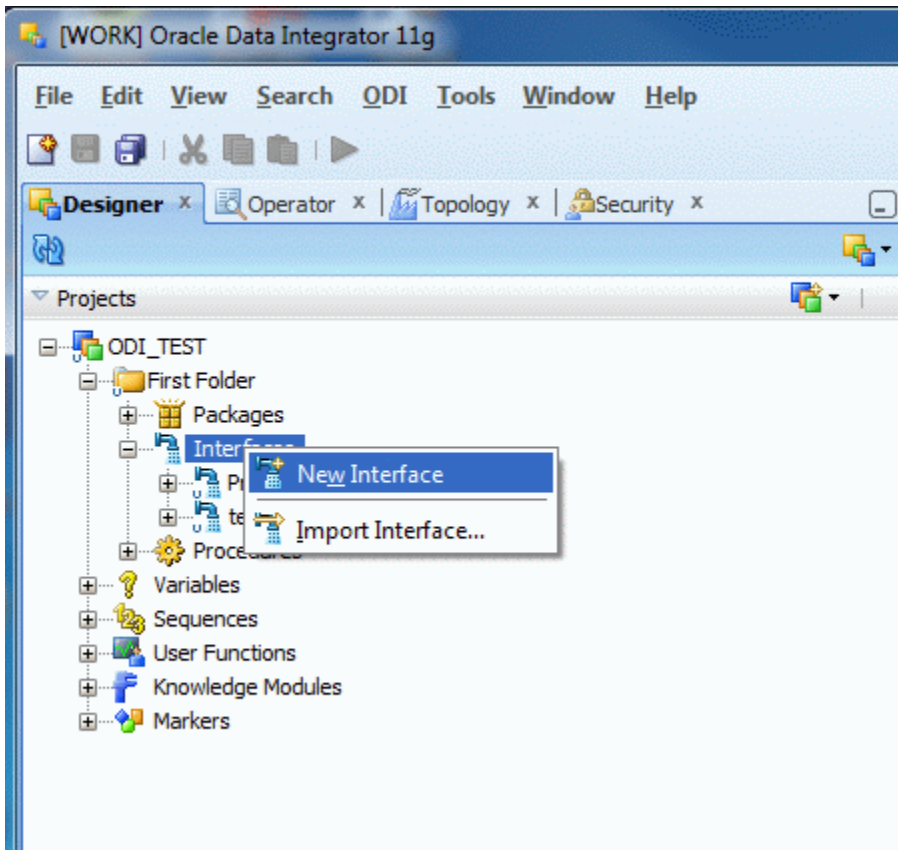
Creating an interface

Next, you create an interface.

To create an interface:

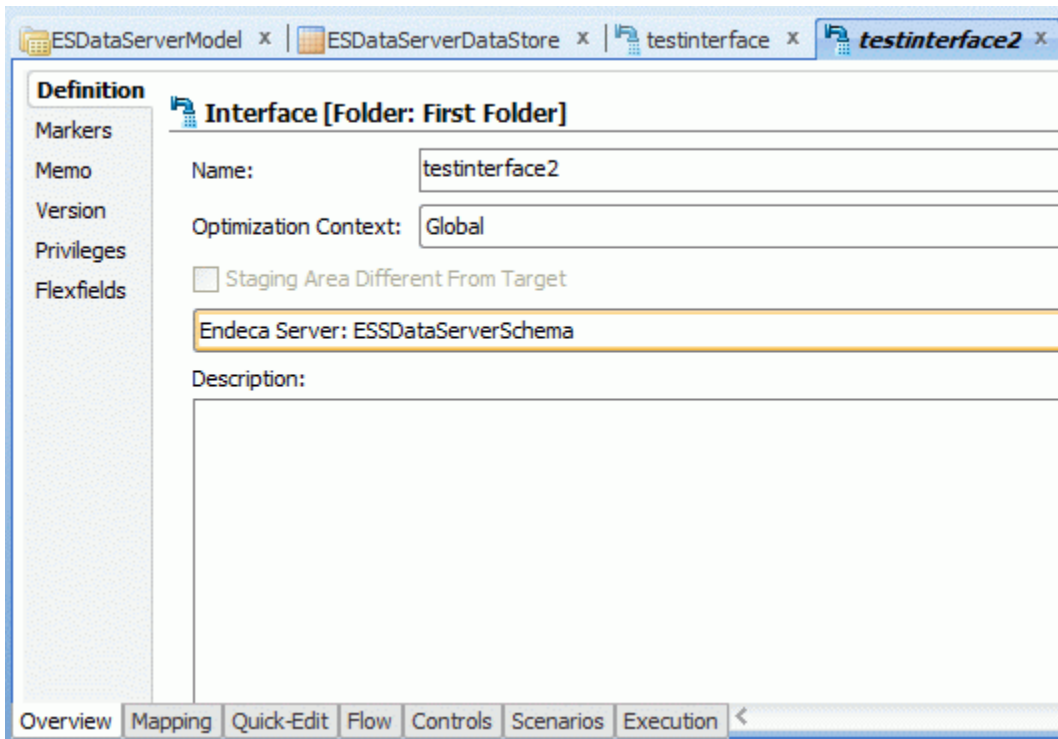
1. In ODI Studio, select the **Designer** tab.
2. Expand **Projects** and go to your project folder.
3. Expand **<project name> Folder**.

4. Right click **Interfaces** and select **New Interface**.
For example:



5. Select the **Definition** pane and specify a unique value for **Name**.

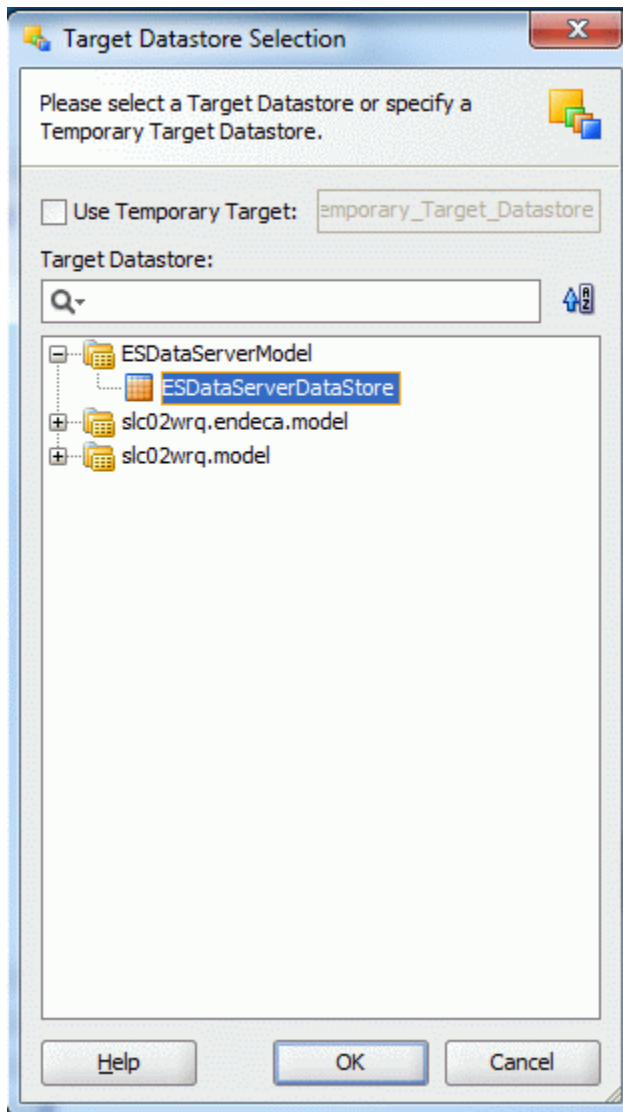
- From the drop-down list, select **Endeca Server: <name of logical schema>** that you created earlier. For example:



- Select the **Quick Edit** tab.
- Click the **Target Datastore Selection** icon.
- Uncheck **Use Temporary Target**.

- Expand the Endeca Server model and select a datastore.

For example:



- Click **OK**.
- Select the **Mapping** tab and drag the data store of the source data into the **Mapping** tab.
- Map the source columns to the target datastore columns.
- Select the **Overview>Definition**, then check **Staging Area Different From Target**.
- From the drop-down list, select a staging database.
The staging database must support SQL.
- Select the **Flow** tab.
- Click the **Target** component.
The **Target Area - Property Inspector** dialog displays.

18. From the **IKM Selector** drop-down list, select **IKM SQL to Endeca Server**.

19. Under **Options**, modify target options as necessary.

You must specify an Endeca data domain name for `DATADOMAIN`. Other options are optional. For details about the options, see [Configuration reference for the IKM SQL to Endeca Server on page 27](#).

20. Click **Save**.

You can now run the ODI Interface according to your scheduling requirements.

Configuration reference for the IKM SQL to Endeca Server

When you open the **Target** component for the IKM SQL to Endeca Server, you find configuration options in the **Options** area of the **Target Area - Property Inspector** dialog. These options are described below.

Setting Name	Setting Requirement	Setting Description
APPLY_SCHEMA	Required.	Specifies whether to create a schema before integrating data into an Endeca Server target. A value of <code>True</code> creates a schema, <code>False</code> does not. The default is <code>True</code> .
AUTO_GENERATE_RECORDSPEC	Required.	Specifies whether to automatically generate the record spec for the data collection. A value of <code>True</code> generates the record spec, <code>False</code> does not. If you do not set the PK column for the modules, Oracle recommends you set this option to <code>True</code> . The default is <code>False</code> .
BUFFER_SIZE	Required.	Specifies the amount of buffer space the module can use, measured in bytes. The module buffers records up to this size and then writes them to the Endeca Server as a batch. The default is 50000000 (approximately 50 MB).
BULK_LOAD_TIMEOUT	Required.	Specifies the number of milliseconds before a bulk load operation times out. The default is 90000.
CERT_KEYSTORE_FILE	Optional.	Specifies a path to the keystore file of Endeca Server. For example, <code>C:\Oracle\Middleware\user_projects\domains\endecaserver\config\ssl\endecaServerClientCert.ks</code> .

Setting Name	Setting Requirement	Setting Description
CERT_KEYSTORE_PASSWORD	Required if CERT_KEYSTORE_FILE is specified, otherwise optional.	Specifies the corresponding password value for CERT_KEYSTORE_FILE. This option is necessary only if SSL client certification is required by Endeca Server.
COLLECTION_NAME	Required if LOAD_DATA is set to True, otherwise optional.	Specifies the name of the data collection in Endeca Server that the knowledge module writes data to.
DATADOMAIN	Required.	Specifies the domain name of the Endeca Server running in WebLogic Server. An empty value causes an error.
DATADOMAIN_PROFILE	Optional.	Optional. Specifies the name of the template used to create the Endeca data domain. A data domain profile is a named template that provides configuration settings used to create Endeca data domains.
LOAD_DATA	Required.	Specifies whether to load data into Endeca Server. A value of True loads data, False does not. This option is useful in cases where you want to create a schema without loading data into a data collection. This setting can be used in conjunction with APPLY_SCHEMA. The default is True.
LOG_FILE	Optional.	Specifies an absolute path to the log file for the knowledge module. For example, D:\logfolder\odi.log. If unspecified, the module does not write logging information.
LOG_LEVEL	Optional.	Sets the logging level for the knowledge module. This can be set to DEBUG, INFO, WARN, or ERROR. The default is WARN.
MAX_ERROR	Required.	Specifies the maximum number of errors that can occur when loading data to Endeca Server. If this value is exceeded, the loading process fails. Errors include record processing errors. Connection errors are not part of this value. The default is 0.

Setting Name	Setting Requirement	Setting Description
MULTIASSIGN_VALUE_DELIMITER	Optional.	Specifies the delimiter character for multi-assign values in a record. The default delimiter is (the pipe sign).
RESET_DOMAIN	Required.	Specifies whether to delete the data and the schema in the domain when running this interface. If the data or schema has changed, or if you are loading a new schema into an old domain, Oracle recommends setting this value to <code>True</code> . A value of <code>True</code> deletes the data, <code>False</code> does not. The default is <code>False</code> .
TRANSACTION	Required.	Specifies whether to load data using batched transactions. A value of <code>True</code> loads all data as a single transaction, or if a failure occurs, none of the records are loaded. The default is <code>False</code> . The <code>TRANSACTION</code> option is typically used in conjunction with <code>MAX_ERROR</code> . You can specify the maximum number of errors that can occur before rolling back a transaction.
TRUNCATE_COLLECTION	Required.	Deletes all data in the data collection before writing new data. A value of <code>True</code> deletes data. The default is <code>False</code> . Also, <code>LOAD_DATA</code> must be set to <code>True</code> if you set <code>TRUNCATE_DATACOLLECTION</code> to <code>True</code> .
TRUST_KEYSTORE_FILE	Optional.	Specifies a path to the truststore file of Endeca Server. For example, <code>C:\Oracle\Middleware\user_projects\domains\endecaserver\config\ssl\endecaServerTrustStore.ks</code> . This option is necessary only if SSL client certification is required by Endeca Server.

Setting Name	Setting Requirement	Setting Description
TRUST_KEYSTORE_PASSWORD	Required if TRUST_KEYSTORE_FILE is specified, otherwise optional.	Specifies the corresponding password value for TRUST_KEYSTORE_FILE. This option is necessary only if SSL client certification is required by Endeca Server.
WRITER_PROTOCOL	Required.	Specifies the protocol that interface uses to load the data into Endeca Server. This can be set to <code>Webservice</code> or <code>bulkload</code> . Both options group records into 50 MB batches and send them to Endeca Server as a batch. If you specify <code>bulkload</code> , the Endeca Server maintains a record queue to commit the records. The default is <code>bulkload</code> .

Mapping source data types to Endeca Server data types

When the knowledge module writes source data to an Endeca Server target, it is possible to encounter errors if data types do not map correctly between the database source and the Endeca Server target.

For example, an Endeca Server data type of `mdex:time` has no direct equivalent in an Oracle database. If you do not map `mdex:time` to an equivalent Oracle data type, the ODI interface may fail at runtime. In these cases, you can map Endeca Server data types in the target to equivalent data types in the database source.

The following table provides a mapping of Endeca Server data types to Oracle and MySQL data type equivalents. This mapping information may also be useful for any other database that supports SQL. For additional details about Endeca Server types, see *Oracle Endeca Server Developer's Guide*.

Endeca Server Type	Oracle Type	MySQL Type	Notes
<code>mdex:boolean</code>	NUMBER	TINYINT	The <code>mdex:boolean</code> type accepts both of "true"/"false" and 1/0 as Boolean values. So string SQL types such as <code>varchar</code> and <code>int</code> SQL types can be mapped to <code>mdex:boolean</code> .

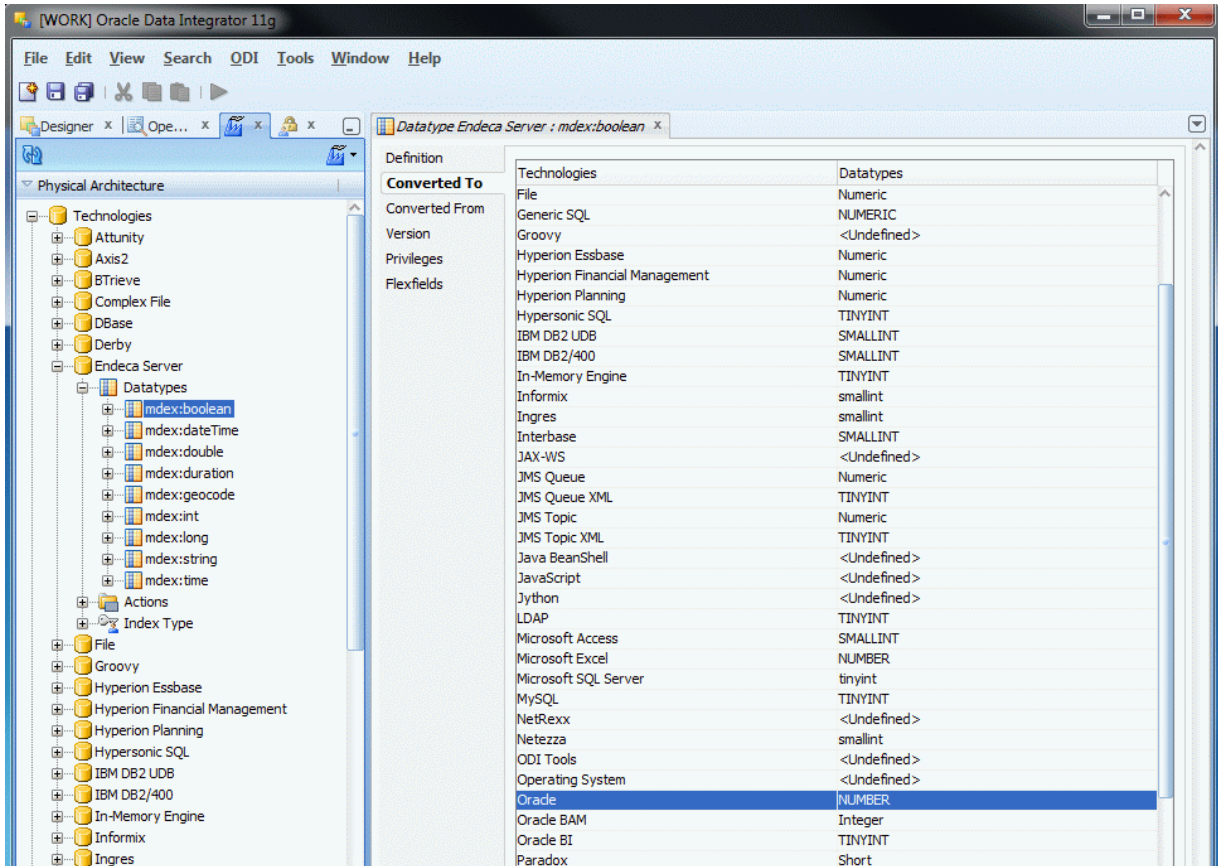
Endeca Server Type	Oracle Type	MySQL Type	Notes
mdex:dateTime	TIMESTAMP	DATETIME	<p>The SQL types TIMESTAMP and DATETIME can map to mdex:dateTime.</p> <p>If there is no datetime type in the input technology (the technology of the staging area), use a string type instead. The string values that you map to mdex:dateTime must match the pattern: yyyy '-' mm '-' dd 'T' hh ':' mm ':' ss {'.' s+} Z.</p> <p>For example: 2011-11-18T17:00:00Z</p>
mdex:double	NUMBER	DOUBLE	<p>Numeric types can map to mdex:double.</p> <p>String types can map to mdex:double as well, but the string values must be valid numeric values.</p>
mdex:duration	VARCHAR2	VARCHAR	<p>In SQL types, there is no duration type. Use a string type for duration values. The string values that you map to mdex:duration must match the format: 'P' {d 'D'} 'T' {h 'H'} {m 'M'} {s {'.' s+} 'S'}.</p> <p>For example: P429DT1H2M3S</p>

Endeca Server Type	Oracle Type	MySQL Type	Notes
mdex:geocode	VARCHAR2	VARCHAR	In SQL types, there is no geocode type. Use a string type for geocode values. The string values that you map to mdex:geocode must match the format: latvalue lonvalue. For example: 42.365615 -71.075647
mdex:int	NUMBER	INT	Integer types can map to mdex:int. String types can map to mdex:int as well, but the string values must be valid integer values.
mdex:long	NUMBER	BIGINT	Long integer types can map to mdex:long. String types can map to mdex:long as well, but the string values must be valid long integer values.
mdex:string	VARCHAR2	VARCHAR	String SQL types such as VARCHAR, VARCHAR2, TEXT can map to mdex:string.
mdex:time	VARCHAR2	TIME	The mdex:time type represents a specific time that recurs every day. MySQL supports time value with the TIME type. There is no TIME type in Oracle databases. Use a string type for mdex:time, but the string values must match the format: hh ':' mm ':' ss {'.' s+} Z. For example: 18:30:00Z

To map source data types to Endeca Server data types:

1. In ODI Studio, select the **Topology** tab.

2. Expand **Physical Architecture>Technologies>Endeca Server>Datatypes**.
3. Double click an Endeca Server data type.
For example, to map Endeca Server Boolean types to Oracle NUMBER, double click `mdex:boolean`.
4. In the right pane, select **Converted to**.
5. Scroll down the rows until you locate name of the database source.
For example, select Oracle:



6. In the row, select the value in the **Datatypes** column and choose the desired data type. This changes the source data type for the Endeca Server type you selected in step 3.
For example, select NUMBER.

The example steps map the Oracle data type NUMBER to the Endeca Server type `mdex:boolean`. This is necessary because Oracle uses 1 and 0 to represent Boolean values rather than True and False. You may also want to map Endeca server data types such as time, dateTime, and geocode to different source data types.



Chapter 3

Using the ODI Procedure to Write to Endeca Server

This section describes how to configure and run the ODI procedure provided with the IKM SQL to Endeca Server.

[Configuring and running the ODI procedure](#)

[Configuration reference for the ODI procedure](#)

Configuring and running the ODI procedure

The procedure is installed in `<install path>\IKMSQLtoEndecaServer\3.1.0\procedure`. It allows you to directly invoke a Web service, using the SOAP protocol, to post a request to write data to Endeca Server and get a response file.

Before performing this procedure, you must have already done the following:

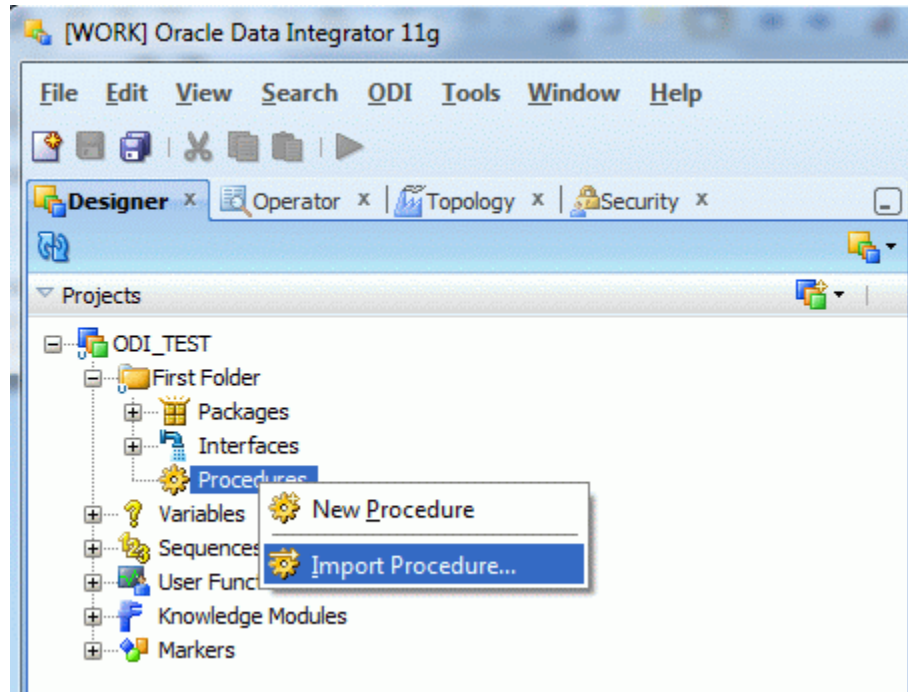
- Started ODI Studio and connected to your Work repository.
- Installed the knowledge module according to [Installing the IKM SQL to Endeca Server on page 7](#).
- Imported the Endeca Server technology according to [Importing the Endeca Server technology into ODI on page 10](#).

To configure and run the ODI procedure:

1. Import the procedure into ODI:
 - (a) In ODI Studio, select the **Designer** tab.
 - (b) Expand **Projects><Project Name>>First Folder**.

- (c) Right click **Procedures** and select **Import Procedure....**

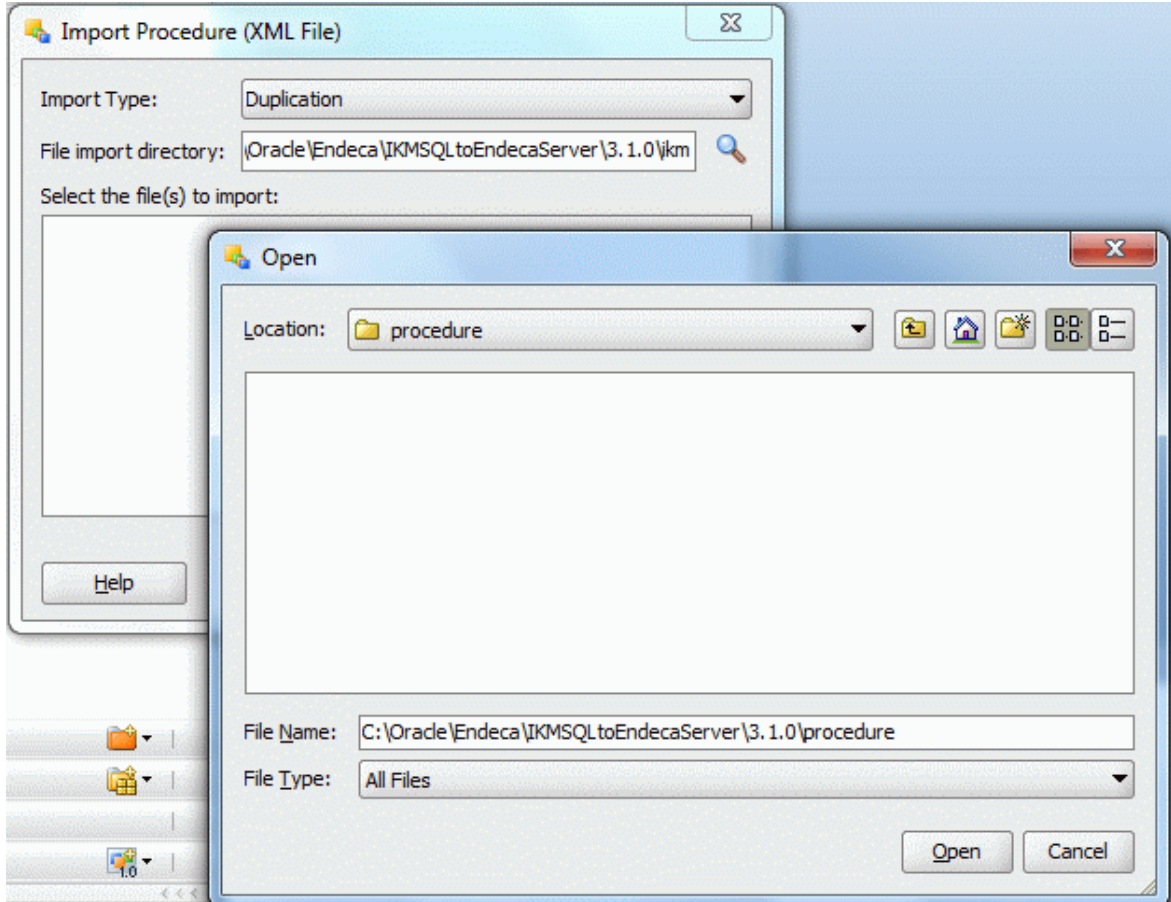
For example:



The **Import Procedure (XML File)** dialog displays.

- (d) In the **File import directory** field, browse to <install path>\IKMSQLtoEndecaServer\3.1.0\procedure and click **Open**.

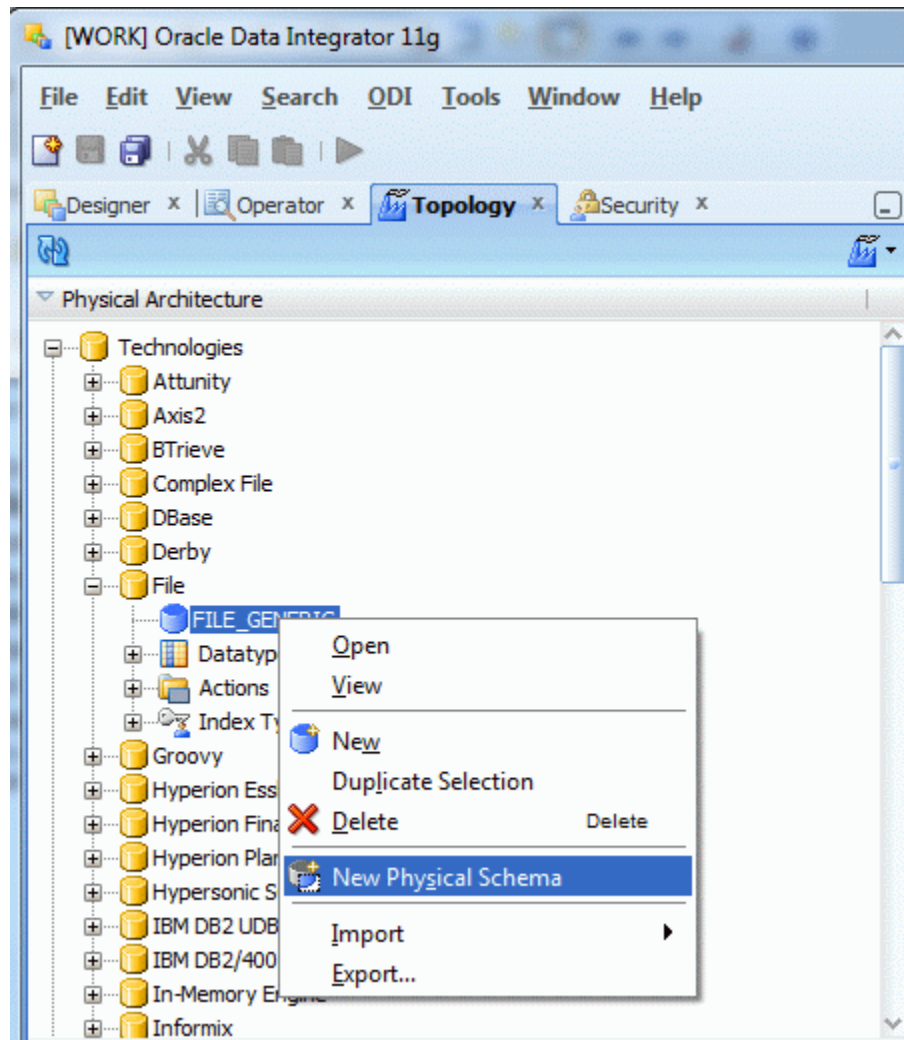
For example:



- (e) Select **Endeca_Web_Service_Procedure** and click **OK**.
- (f) Click **Close** to dismiss the Import Report.
2. Create a physical schema and logical schema to associate with the Web Service request:
- Select the **Topology** tab.
 - Expand **Physical Architecture>Technologies>File**.

- (c) Right click **File Generic** and select **New Physical Schema**.

For example:



- (d) Select the **Context** tab.
- (e) Click the plus (+) icon.
- (f) From the **Context** list, select an existing context.
- (g) From the **Logical Schema** list, select an existing logical schema or specify a new name to define a new logical schema.
- If you specify a logical schema that does not exist yet, ODI creates it automatically when you save the physical schema.
- (h) Click **Save All**.
- (i) Select the **Definition** tab.

- (j) For both the **Directory (Schema)** list and **Directory (Work Schema)** list, specify a path to a directory where the request and response files are stored.

Note that on Windows systems, you need to escape paths containing the \ character by adding a second \ character. For example, C:\\test.

3. Double-click the imported procedure and select the **Details** tab.
4. Double-click **Set Endeca Server Logical Schema** and from the **Command on Source** list, select the Endeca Server logical schema.
5. Return to the **Details** tab.
6. Double-click **Set File Logical Schema** and from the **Command on Source** list, select the File logical schema.
7. Now configure the ODI procedure itself:
 - (a) Under **Options**, specify values for ENDECA_WS_RELATIVE_PATH, REQUEST_FILE_NAME, and RESPONSE_FILE_NAME.
 - (b) Other options are optional. For details about the options, see [Configuration reference for the ODI procedure on page 38](#).
8. Click **Save**.

You can now run the ODI procedure according to your scheduling requirements.

Configuration reference for the ODI procedure

When you open the **Options** tab of the ODI procedure, you find configuration options that are described below.

Setting Name	Setting Requirement	Setting Description
CERT_KEYSTORE_FILE	Optional.	Specifies a path to the keystore file of Endeca Server. For example, C:\Oracle\Middleware\user_projects\domains\endecasever\config\ssl\endecaServerClientCert.ks.
CERT_KEYSTORE_PASSWORD	Required if CERT_KEYSTORE_FILE is specified, otherwise optional.	Specifies the corresponding password value for CERT_KEYSTORE_FILE. This option is necessary only if SSL client certification is required by Endeca Server.

Setting Name	Setting Requirement	Setting Description
ENDECA_WS_RELATIVE_PATH	Required.	Specifies a relative path to the Endeca Web Service. The path is relative to the Endeca Server domain running on WebLogic Server. For example, if there is a data domain called 'domain1' running on Endeca Server, at the URL <code>https://localhost:7002/endeca-server/ws/sconfig/domain1</code> , and its sconfig is going to be invoked, the ENDECA_WS_RELATIVE_PATH is <code>/ws/sconfig/domain1</code> .
LOG_FILE	Optional.	Specifies an absolute path to the log file for the knowledge module. For example, <code>D:\logfolder\odi.log</code> . If unspecified, the module does not write logging information.
LOG_LEVEL	Optional.	Sets the logging level for the knowledge module. This can be set to <code>DEBUG</code> , <code>INFO</code> , <code>WARN</code> , or <code>ERROR</code> . The default is <code>WARN</code> .
REQUEST_FILE_NAME	Required.	Specifies a relative path to the XML request file that contains the SOAP request for Endeca Server. The path is relative to the logical schema (of type File) that you created.
RESPONSE_FILE_NAME	Optional.	Specifies a relative path to the XML response file that contains the response message after the request completes. The path is relative to the logical schema (of type File) that you created. If this is unspecified, the procedure does not save the response file returned by Endeca Server.
TRUST_KEYSTORE_FILE	Optional.	Specifies a path to the truststore file of Endeca Server. For example, <code>C:\Oracle\Middleware\user_projects\domains\endecaserver\config\ssl\endecaServerTrustStore.ks</code> . This option is necessary only if SSL client certification is required by Endeca Server.

Setting Name	Setting Requirement	Setting Description
TRUST_KEYSTORE_PASSWORD	Required if TRUST_KEYSTORE_FILE is specified, otherwise optional.	Specifies the corresponding password value for TRUST_KEYSTORE_FILE. This option is necessary only if SSL client certification is required by Endeca Server.



This section describes how to uninstall the IKM SQL to Endeca Server from Oracle Data Integrator.

Uninstalling the IKM SQL to Endeca Server

Uninstalling the IKM SQL to Endeca Server

To uninstall the module, you delete all the objects you created, in the opposite order that you first created them. For example, you created an interface last, so you delete that first. Then you remove the IKM data integration code, then delete the datastore, then the model, and so on until all aspects of the IKM SQL to Endeca Server have been removed.

To uninstall the IKM SQL to Endeca Server:

1. Delete the interface associated with IKM SQL to Endeca Server:
 - (a) In ODI Studio, select the **Designer** tab.
 - (b) Expand **Projects** and go to your project folder.
 - (c) Expand **<project name> Folder**.
 - (d) Right click **Interfaces** and select **Delete**.
 - (e) Click **Yes**.
2. Remove the knowledge module:
 - (a) In ODI Studio, select the **Designer** tab.
 - (b) Expand **Projects**.
 - (c) Expand **<project name> Folder**.
 - (d) Expand **Knowledge Modules>Integration (IKM)**.
 - (e) Right click the module and select **Delete**.
 - (f) Click **Yes**.
 - (g) Click **Save All**.
3. Delete the datastore:
 - (a) In ODI Studio, select the **Designer** tab.
 - (b) Expand **Models** and expand your project model.
 - (c) Right click the datastore and select **Delete**.
 - (d) Click **Yes**.
 - (e) Click **Save All**.
4. Delete the model:
 - (a) In ODI Studio, select the **Designer** tab.
 - (b) Expand **Models**.

- (c) Right click the model and select **Delete**.
 - (d) Click **Yes**.
 - (e) Click **Save All**.
5. Delete the logical schema:
 - (a) In ODI Studio, select the **Topology** tab.
 - (b) Expand **Logical Architecture>Technologies>Endeca Server**.
 - (c) Right click the logical schema and select **Delete**.
 - (d) Click **Yes**.
 - (e) Click **Save All**.
6. Delete the physical schema:
 - (a) In ODI Studio, select the **Topology** tab.
 - (b) Expand **Physical Architecture>Technologies>Endeca Server**.
 - (c) Right click the physical schema and select **Delete**.
 - (d) Click **Yes**.
 - (e) Click **Save All**.
7. Delete the Endeca flexfields:
 - (a) In ODI Studio, select the **Security** tab.
 - (b) Expand **SECURITY ADMIN**.
 - (c) Double-click **Column**.
 - (d) In the **Column** editor, select all rows that begin with "Endeca Property", and click the delete icon in the right-top corner.
 - (e) Click **Save All**.
8. Delete the Endeca Server technology:
 - (a) In ODI Studio, select the **Topology** tab.
 - (b) Expand **Physical Architecture>Technologies>Endeca Server**.
 - (c) Right click Endeca Server and select **Delete**.
 - (d) Click **Yes**.
 - (e) Click **Save All**.
9. Delete the JAR files for the ODI agent as a standalone application.
 - (a) From a command prompt, stop the ODI agent.
 - (b) On the file system, browse to <install path>\Oracle\Middleware\Oracle_ODI1\oracledi\agent\drivers (or the equivalent path on UNIX).
 - (c) Delete the following JAR files: `endeca-server-driver.jar`, `endeca_bulk_load-7.6.jar`, `guava-14.0.jar`, `httpclient-4.0.3.jar`, `httpcore-4.0.1.jar`, `log4j-1.2.15.jar`, `slf4j-api-1.7.2.jar`, and `slf4j-log4j12-1.7.2.jar`.
10. Delete the JAR files for the ODI agent as a Java application running in WebLogic Server.
 - (a) Stop the WebLogic Server that is running the ODI agent.
 - (b) On the file system, browse to <install path>\Oracle\Middleware\Oracle_ODI1\oracledi\agent\drivers (or the equivalent path on UNIX).

- (c) Delete the following JAR files: `endeca-server-driver.jar`, `endeca_bulk_load-7.6.jar`, `guava-14.0.jar`, `httpclient-4.0.3.jar`, `httpcore-4.0.1.jar`, `log4j-1.2.15.jar`, `slf4j-api-1.7.2.jar`, and `slf4j-log4j12-1.7.2.jar`.
- (d) Navigate to `<install path>\Oracle\Middleware\user_projects\domains\<domain name>\bin` and open `setDomainEnv.cmd` (on Windows) or `setDomainEnv.sh` (on UNIX) in a text editor.
- (e) On Windows, delete the following line: `" set JAVA_OPTIONS=-DUseSunHttpHandler=true "`.
- (f) On UNIX, delete the two lines containing `JAVA_OPTIONS=-DUseSunHttpHandler=true` and `export JAVA_OPTIONS`.
- (g) Save and close `setDomainEnv`.