

# ODI 12c - File to Table

## Before You Begin

### Purpose

This tutorial walks you through the steps that are needed to use **Oracle Data Integrator Studio** (ODI Studio) graphical tools to export an ASCII flat file of columnar data to an Oracle Database 12c relational table.

### Time to Complete

Approximately 1 hour.

### Introduction

There are many ways to get a flat file of data into an Oracle Database. For example, SQL\*Loader is one of the proprietary tools included in many versions of Oracle Database. The problem is that other versions of other databases (such as IBM DB2, or Microsoft SQL Server) use different proprietary tools. It would be attractive to have one standard way of importing files into many different databases. ODI can do just that. Furthermore, SQL\*Loader is designed to take the data in pretty much unmodified, whereas ODI is designed to transform the data while it is being loaded. (Ignore for the moment exactly where and when the transformation is taking place.)

This is the second in a series of four OBEs in the Oracle Data Integrator 12c: Getting Started ([https://apex.oracle.com/pls/apex/f?p=44785:24:15503233161396:::P24\\_CONTENT\\_ID,P24\\_PREV\\_PAGE:7952,24](https://apex.oracle.com/pls/apex/f?p=44785:24:15503233161396:::P24_CONTENT_ID,P24_PREV_PAGE:7952,24)) series.

### Scenario

Linda works as a database administrator for Example Enterprise Corp (<http://www.example.com>). In Example Enterprise, Linda is responsible for performing database management and integration tasks on the various resources within the organization. In particular, Linda is responsible for data loading, transformation, and validation.

To begin working on her projects (such as exporting a flat file to a relational table), she needs to create the new Master repository and Work repository. This was already done in a previous OBE. Following that, she will make a flat file to relational table project (this OBE). Following that, she will make an agent. These other tasks are covered in later OBEs.

### Hardware and Software Requirements

The following is a list of software requirements:

- A supported operating system (the OS shown here is 64-bit Oracle Enterprise Linux release 6.7 (Santiago), other versions such as Windows are supported as well)
- Oracle Database (the version shown here is Enterprise Edition 12.1.0.2.0, other versions are supported as well)
- Oracle Data Integrator 12c (the version shown here is 12.2.1)

The following is a list of *OPTIONAL* software:

- Oracle WebLogic Server (WLS) 12c (other versions are supported as well)
- Oracle SQL Developer 12c (included with Database 12c install)

If WLS is present, it must be in the same Middleware `HOME` directories as ODI.

## Prerequisites

Before starting this tutorial, you should have:

- Started Oracle Database services and components such as the Listener.
- A Database user with DBA role privileges such as the `SYS` or `SYSTEM` account.
- Installed the `HR` schema included in the Oracle Database. The Sample Schemas Installation Guide is part of the Oracle Database documentation set, and is also available online at: <http://otn.oracle.com> (<http://otn.oracle.com>).
- Already installed and configured ODI Master and Work repositories (instructions on how to do this are contained in the first ([https://apex.oracle.com/pls/apex/f?p=44785:24:15503233161396::NO:24:P24\\_CONTENT\\_ID,P24\\_PREV\\_PAGE:7946,16](https://apex.oracle.com/pls/apex/f?p=44785:24:15503233161396::NO:24:P24_CONTENT_ID,P24_PREV_PAGE:7946,16)) of four OBEs in the Getting Started series).

## 1. Preparing the Source Flat File

The source and target definitions are very similar. You will use the ODI Studio GUI to create the models to represent these objects.

Many of the tasks for the GUI have directional terms, such as, "On the left..." or "On the right..." This presumes the default panel layout. You have the option to undock and relocate the panels, and also have the option to return the panels to their default layout by clicking **Main Menu Bar > Window > Reset Windows to Factory Settings**.

To prepare a flat file to be the source for an export to a relational table target, perform the following steps:

### 1.1 Starting ODI Studio

In a terminal session, navigate to `$ODI_HOME/studio`. In this example, that is `/u01/app/oracle/Middleware/Oracle_Home/odi/studio`. Start ODI Studio by running script `odi.sh`.

## OS Prompt

```
[myuser@myhost ~]$ cd $ODI_HOME/studio
[myuser@myhost studio]$ pwd
/u01/app/oracle/Middleware/Oracle_Home/odi/studio
[myuser@myhost studio]$ ./odi.sh

Oracle Fusion Data Integrator Studio 12c
Copyright (c) 1997, 2015, Oracle and/or its affiliates. All rights reserved.
```

This console window must remain open (but can be minimized) while Studio is running. The splash screen and progress bar displays.

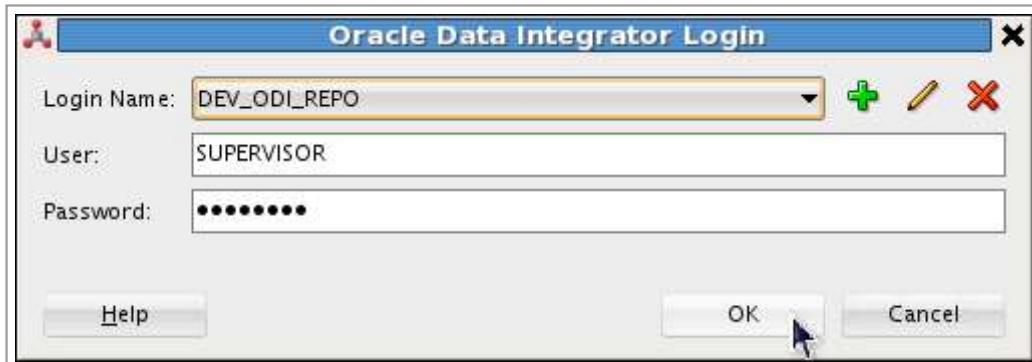
On the left panel, on either the **Designer** or **Topology** tab, click **Connect To Repository...**



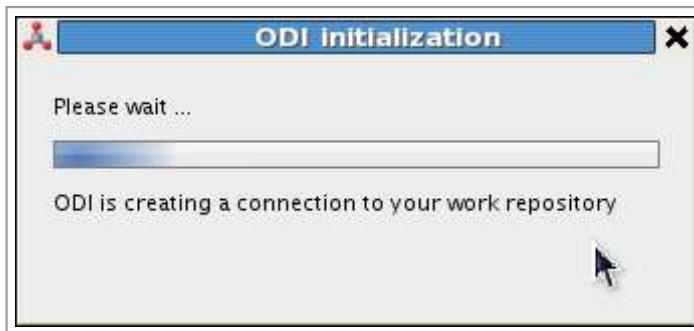
Enter the **Wallet Password**, and click **OK** to continue. (It may be pre-stored.)



Enter the **SUPERVISOR Password** and click **OK** to continue. (It may be pre-stored.)



A progress bar is displayed.



When ODI Studio is finished connecting to the repositories, the navigation panel on the left is populated.

In summary, you have successfully started ODI Studio.

## 1.2 Create Source Text File

Create a text file named `continents.txt` that will be the source for the import. You can do this task using a GUI tool such as `gedit`, or you can do it using a CLI tool such as `vi`. Create a column-based ASCII text file with column headings. Use spaces (not tabs) as the delimiters. A sample can be found here ([files/continents.txt](#)) for cut-n-paste.

```
oracle@edvmrlp0:~$ gedit continents.txt
*continents.txt (~) - gedit
File Edit View Search Tools Documents Help
New Open Save Print... Undo Redo Cut
*continents.txt
1 ABBR NAME LAND_KM
2 AA Antarctica 13209000
3 NA North America 24256000
4 SA South America 17819000
5 EU Europe 9938000
6 AS Asia 44579000
7 AU Australia 7687000
8 AF Africa 30065000|
```

Ln 8, Col 30      INS

Save it or move it to `$ODI_HOME/demo/file/` (along with some other sample files included as part of the install).

**Note:** The `$ODI_HOME/demo/file/` may exist from a previous install, or may be downloaded separately from OTN, or may not exist at all. It is just sample and demo files, and is not needed for this OBE. If it does not exist, just make the subdirectories `demo/file/` for convenience.

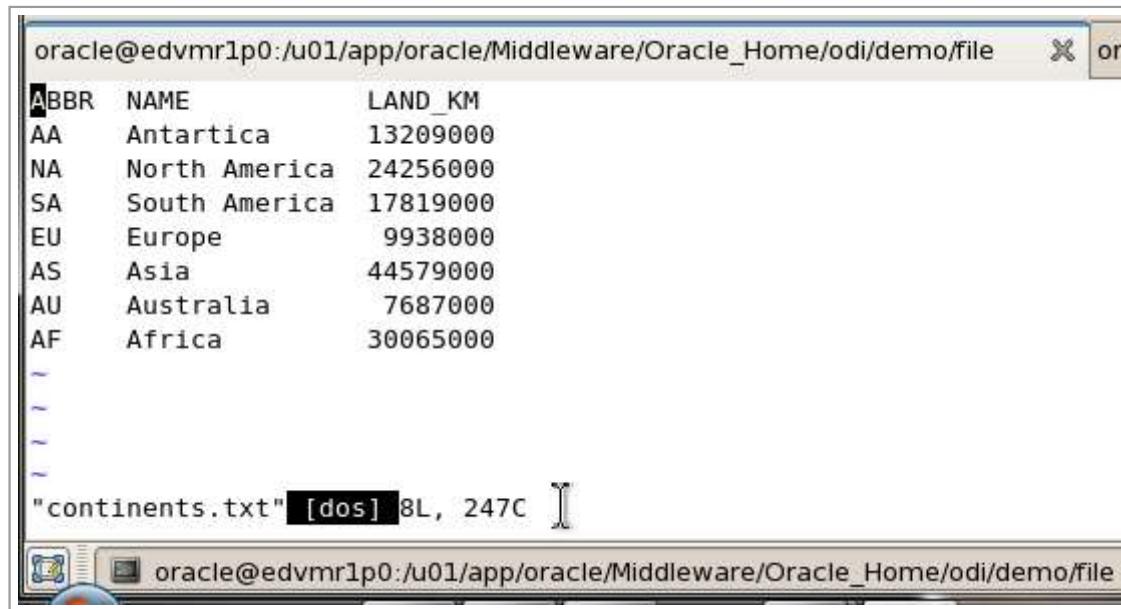
Run the utility `unix2dos` on the file to make sure that the line endings are DOS-style (carriage-return + line-feed) as opposed to UNIX-style (just carriage-return).

### OS Prompt

```
[myuser@myhost ~]$ cd /u01/app/oracle/Middleware/Oracle_Home/odi/demo/file/
[myuser@myhost file]$ cp ~/continents.txt .
[myuser@myhost file]$ ll
total 16
-rw-r--r-- 1 oracle oinstall 247 Jul 29 11:00 continents.txt
-rw-r----- 1 oracle oinstall 153 Jul 11 04:37 SRC_AGE_GROUP.txt
-rw-r----- 1 oracle oinstall 2128 Jul 11 14:37 SRC_SALES_PERSON.txt
[myuser@myhost file]$ unix2dos continents.txt
unix2dos: converting file continents.txt to DOS format ...
[myuser@myhost file]$
```

If you don't do this, the symptom is that the file will only import one line (the first row) and then stop.

You can prove to yourself that it worked by looking at the file with `vi`. Note the [dos] type at the bottom. You may (or may not) also see control-M characters.



```
oracle@edvmr1p0:/u01/app/oracle/Middleware/Oracle_Home/odi/demo/file
ABBR NAME          LAND_KM
AA   Antarctica   13209000
NA   North America 24256000
SA   South America 17819000
EU   Europe        9938000
AS   Asia          44579000
AU   Australia    7687000
AF   Africa        30065000
-
-
-
-
"continents.txt" [dos] 8L, 247C ]
```

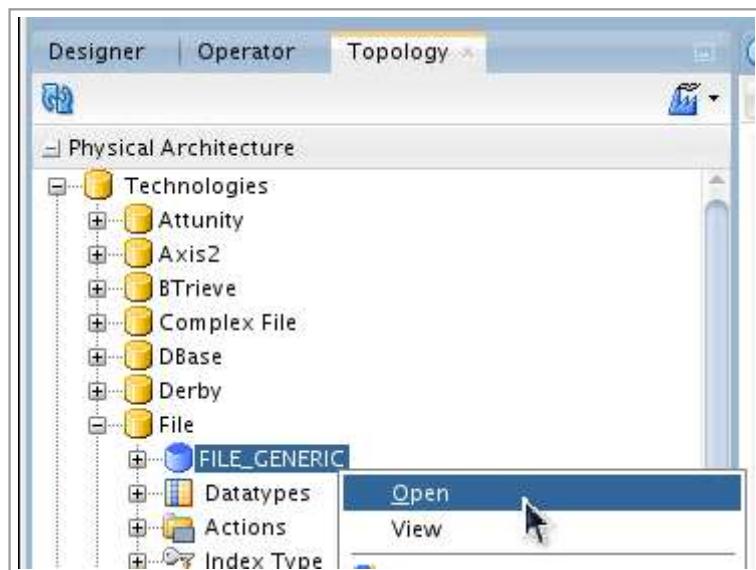
The screenshot shows a terminal window with the command `vi continents.txt` running. The file contains data about continents with their abbreviations (ABBR), names (NAME), and land areas (LAND\_KM). The status bar at the bottom indicates "continents.txt" is in [dos] mode, has 8 lines, and 247 characters. The window title is "oracle@edvmr1p0:/u01/app/oracle/Middleware/Oracle\_Home/odi/demo/file".

There is nothing you need to do with `vi` other than see the " [dos]" status at the bottom, close the file.

In summary, you created the source text file.

### 1.3 Defining Source Topology Physical Architecture

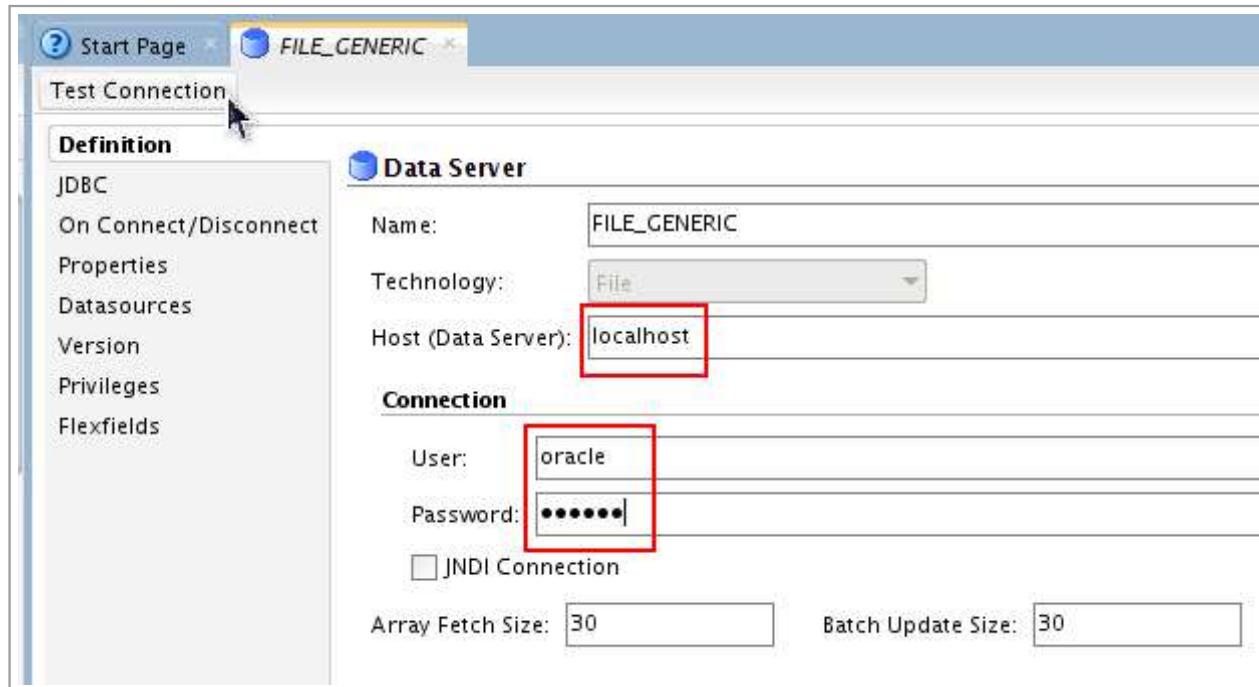
In ODI Studio, on the left, click the **Topology** tab, expand **Physical Architecture**, then expand **Technologies > File**. Right-click **FILE\_GENERIC** and select **Open**.



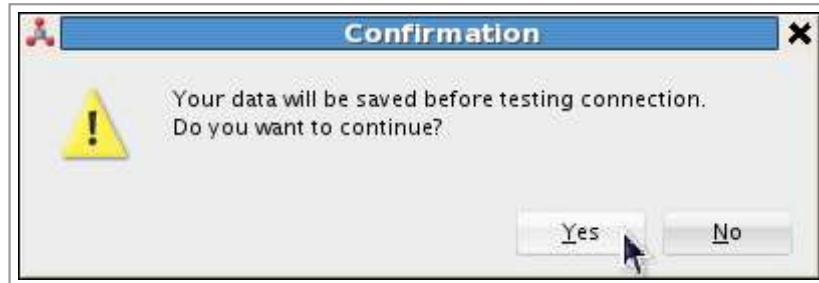
The **File\_GENERIC** panel opens on the right.

On the **Definition** tab, enter the **Host** name such as `localhost`.

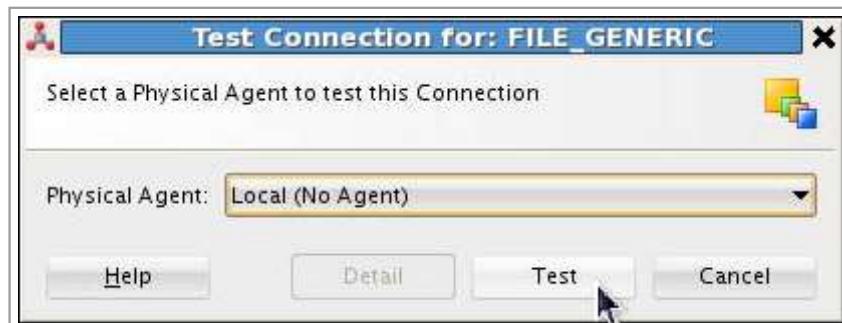
The **Connection OS User** such as `oracle` or `myuser`, the **OS Password** such as `Welcome1` (which will not display) are not required, nor does their presence affect anything if Technology=File. You can put them in to be self-documenting, or not, it is optional. (Obviously User and Password are required for communications access such as JDBC.)



Click **Test Connection** to verify connectivity. If you did not save your work already by clicking , you have to save it now.



Click **Yes** to save and continue. You will be asked to pick a **Physical Agent**.

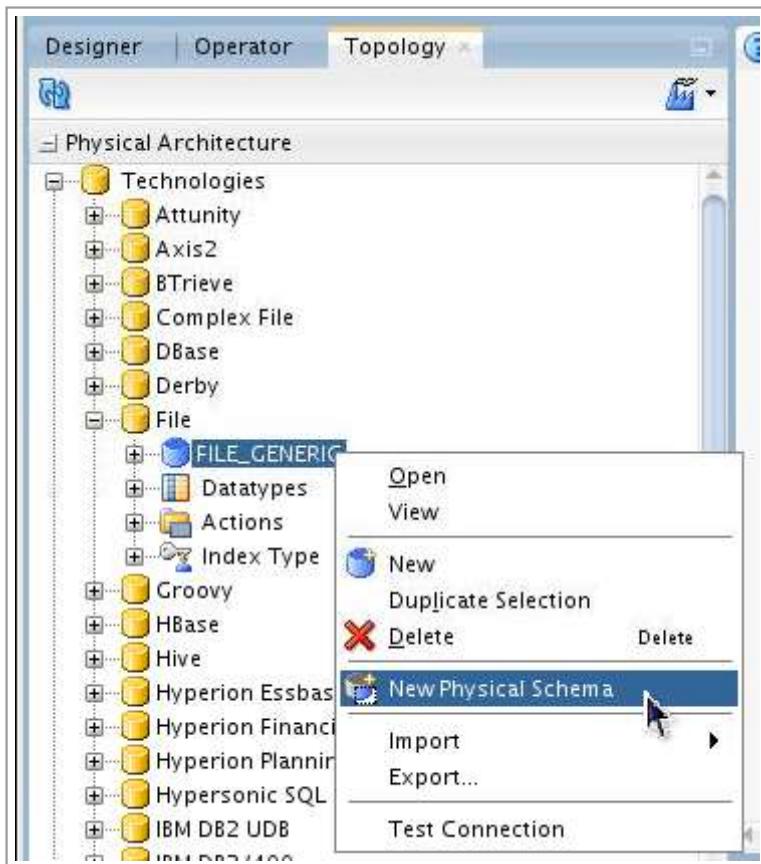


There are no agents defined yet, so the default of **Local (no Agent)** is fine. (Agents will be defined in a later OBE ([https://apex.oracle.com/pls/apex/f?p=44785:24:15503233161396::NO:24:P24\\_CONTENT\\_ID,P24\\_PREV\\_PAGE:7949,16](https://apex.oracle.com/pls/apex/f?p=44785:24:15503233161396::NO:24:P24_CONTENT_ID,P24_PREV_PAGE:7949,16)).) Click **Test** to test and continue.



If the test was successful, you will get a message saying, "Successful." Click **OK** to continue.

On the **Topology** tab, right-click on **FILE\_GENERIC** and click **New Physical Schema**.

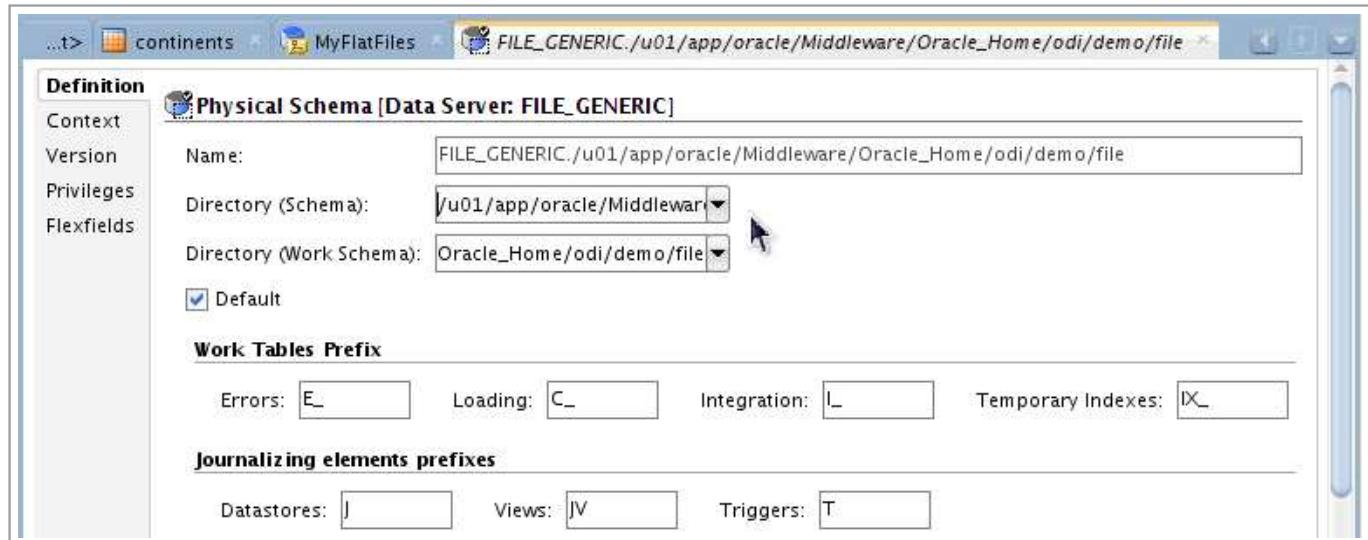


The **Physical Schema [Data Server]** panel opens on the right.

On the **Physical Schema** panel, in **Directory (Schema)**, enter

`/u01/app/oracle/Middleware/Oracle_Home/odi/demo/file` and then enter the same thing again in **Directory (Work Schema)**. This is equivalent to `$ODI_HOME/demo/file/`, but you should not use OS

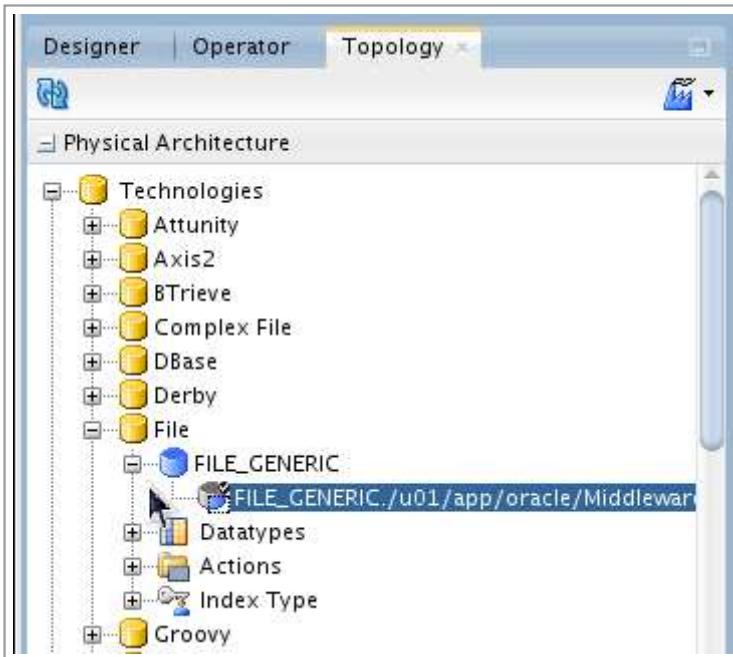
environment variables in configuring ODI; use the full path names. Notice that you must type it twice, you cannot browse to it.



Unsaved work is indicated by the *italics* on the tab name. Save your work by clicking . When you save it, the **Name** suffix automatically changes to the directory you just typed.  
You will get a informational reminder that eventually you will need to make a context in the **Designer** tab.



Click to continue. Back on the **Topology** tab, expand **FILE\_GENERIC** and confirm that the **FILE\_GENERIC./u01/app/oracle/Middleware/Oracle\_Home/odi/demo/file** physical schema has been added.

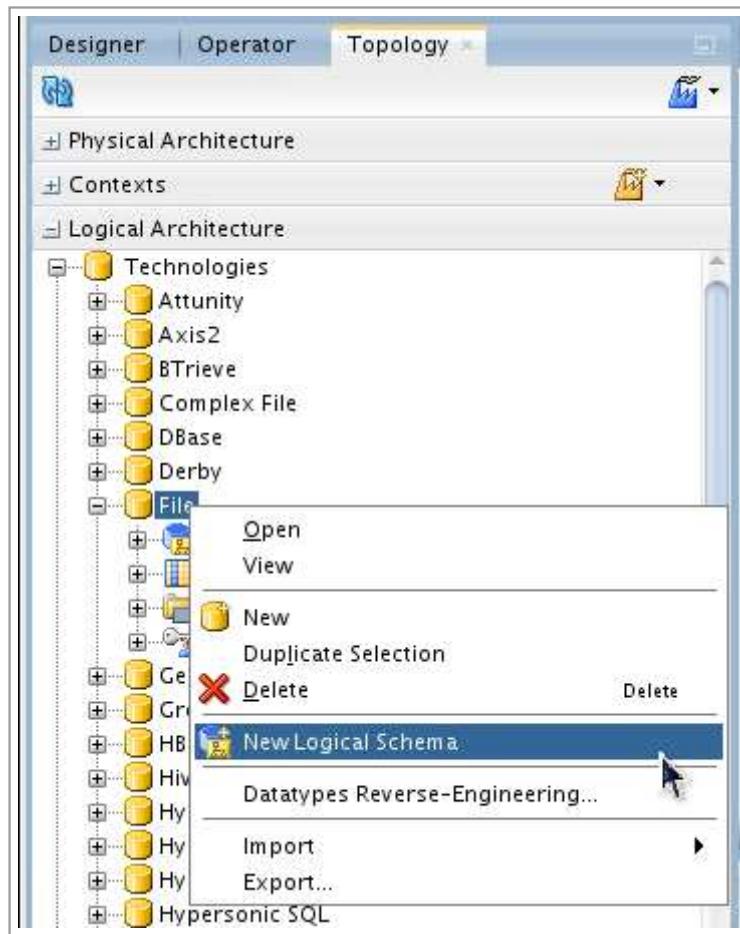


You can close the **FILE\_GENERIC** Physical Schema tab on the right by clicking the small red X at the end of the tab name.

In summary, you have configured an existing Physical Architecture File Technology to point to your host. You can have multiple physical hosts.

## 1.4 Defining Source Topology Logical Architecture

On the left, click the **Topology** tab, expand **Logical Architecture**, then expand **Technologies > File**. Right-click **File** and select **New Logical Schema**.

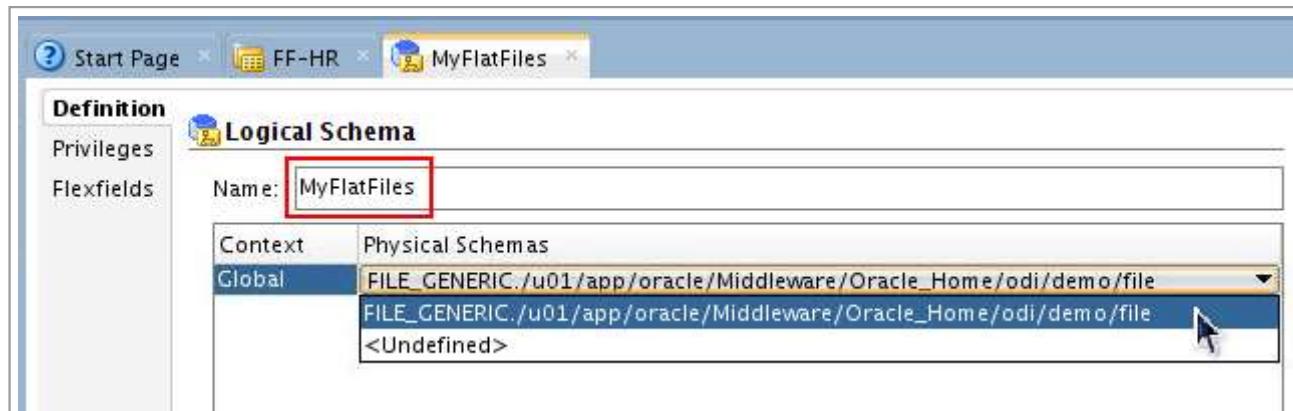


If the **Topology** panel is getting crowded, you may optionally collapse **Physical Architecture**, you will not need it for a while.

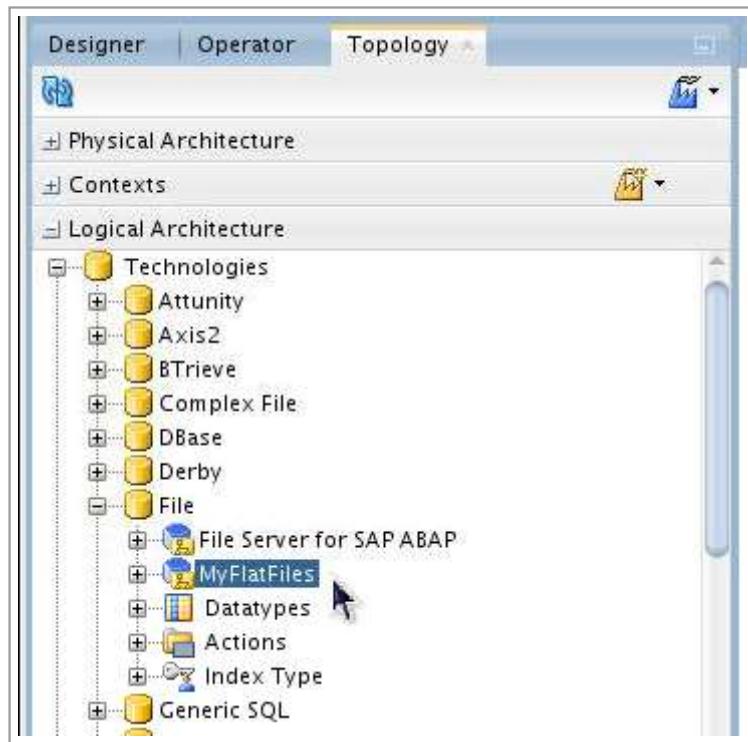
On the **Logical Schema** panel, on the **Definition** tab, enter a **Name** of `MyFlatFiles`.

Under **Physical Schemas**, use the pull-down and select

`FILE_GENERIC./u01/app/oracle/Middleware/Oracle_Home/odi/demo/file`.



Save your work by clicking . You should now see **MyFlatFiles** on the left under **File**.

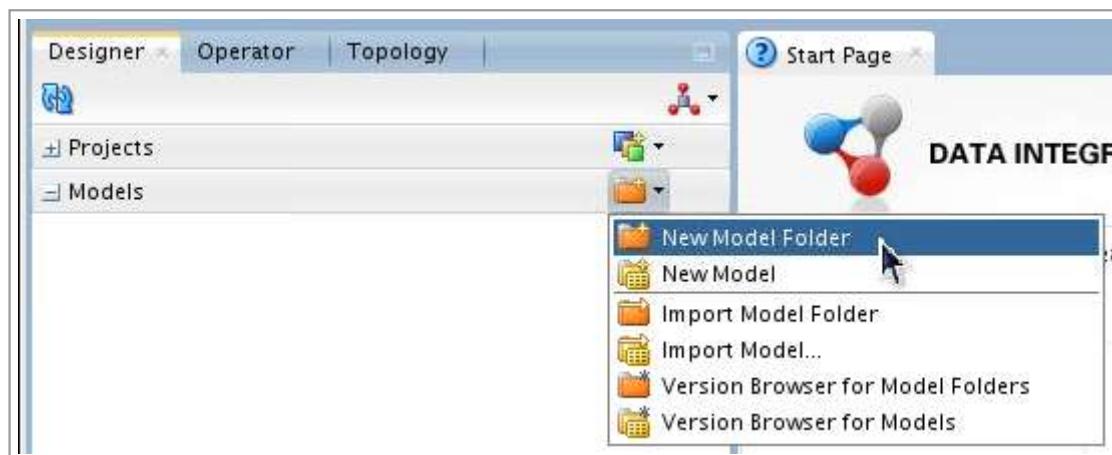


If the **Topology** panel is getting crowded, you may optionally collapse **Logical Architecture**, you will not need it any more.

In summary, you have created a logical schema in the context of a physical schema.

## 1.5 Defining Source Designer Model

On the **Designer** tab, expand **Models** (it may be empty). Under the folder icon pull-down, click **New Model Folder**.



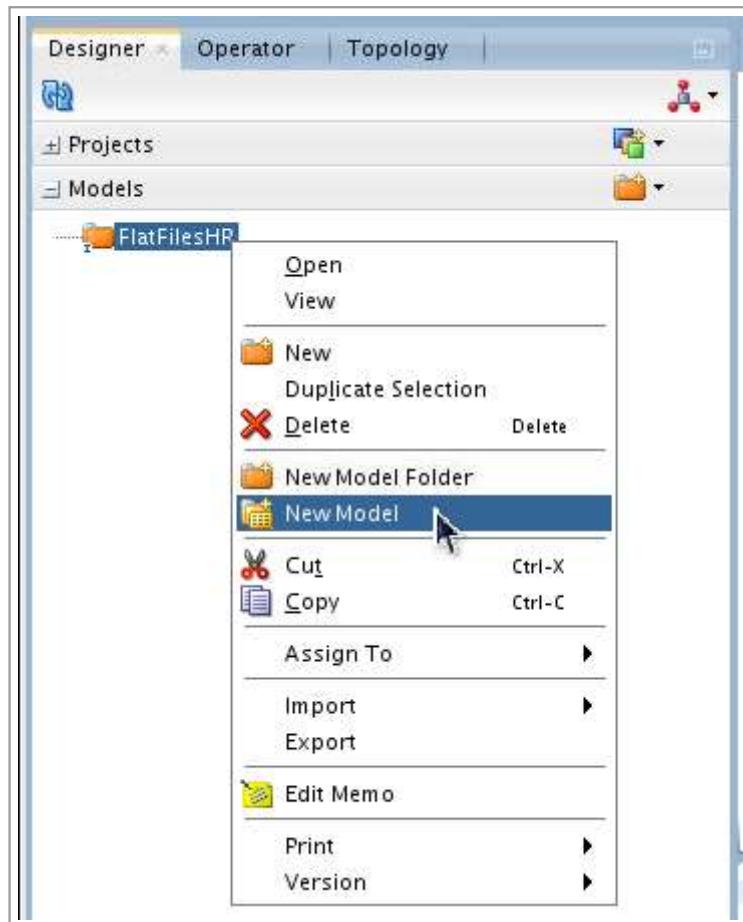
A new **Model Folder** panel opens on the right. Model folders don't actually do anything, they simply contain models. As such, the folder is optional, but helpful for organizational purposes.

On the **Definition** tab, enter a **Name** of `FlatFilesHR` and an optional **Description**.



Save your work by clicking . **FlatFilesHR** should now be under **Models** on the left. You can close the **FlatFilesHR** tab on the right.

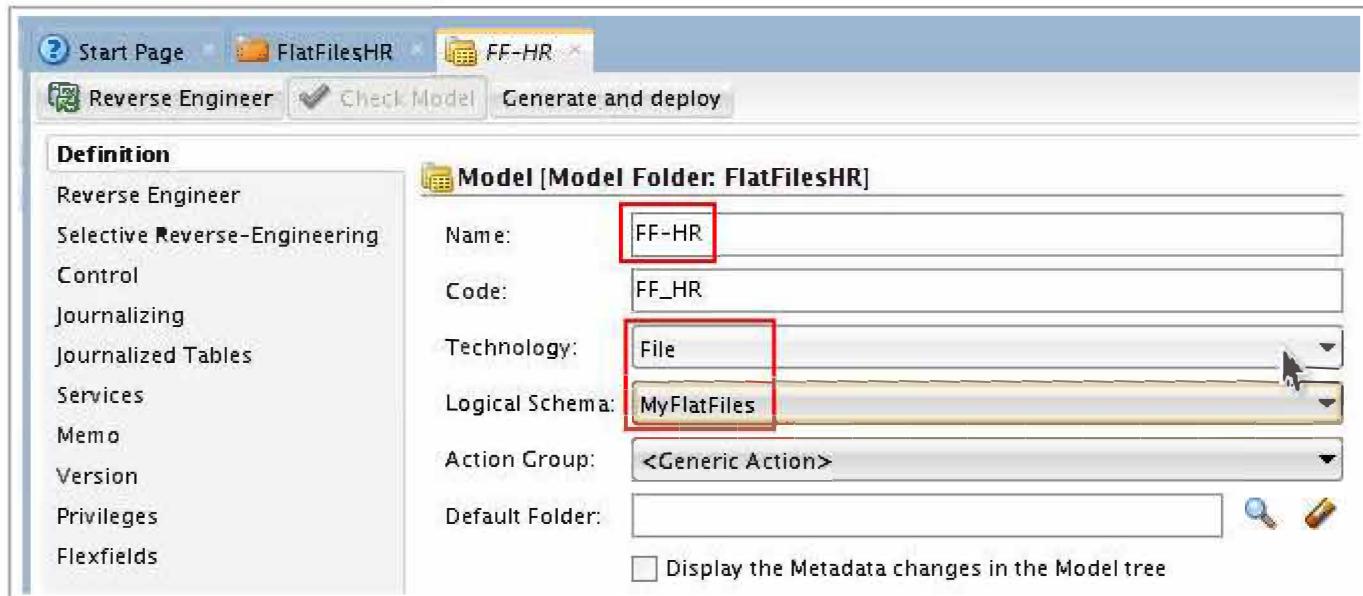
Right-click on **FlatFilesHR** and click **New Model**.



The **Model** panel opens on the right.

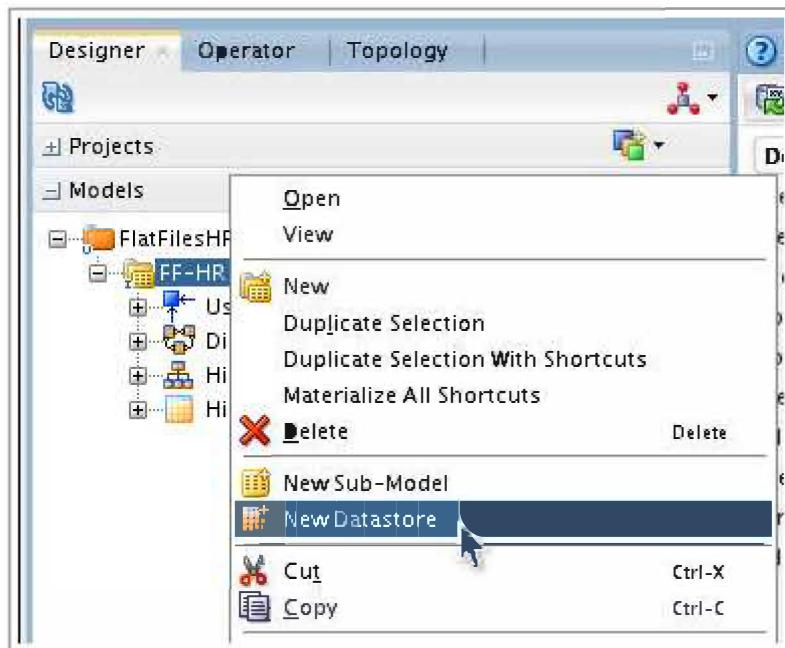
On the **Model** panel, on the **Definition** tab, enter the following information:

- In the **Name** field, enter **FF-HR** . As you enter that name, the system automatically copies the name below to the **Code** field, except it replaces the hyphen with an underscore.
- In the **Technology** pull-down, select **File**.
- In the **Logical Schema** pull-down, select **MyFlatFiles**.



Save your work by clicking . There should now be a **FF-HR** model under the model folder. You may have to expand the model folder to see the new FF-HR model.

In **Designer > Models > FlatFilesHR**, right-click on **FF-HR** and select **New Datastore**.



The **Datastore** panel opens on the right.

On the **Datastore** panel, on the **Definition** tab, in **Name**, enter `continents`. As you do that, the tab automatically changes to say **continents**, the **Resource Name** changes as well (though you will have to elaborate that), and the **Alias** becomes the first three letters of the name, that is, **CON**.

**Datastore [Model: FF-HR > Sub-Model: Global]**

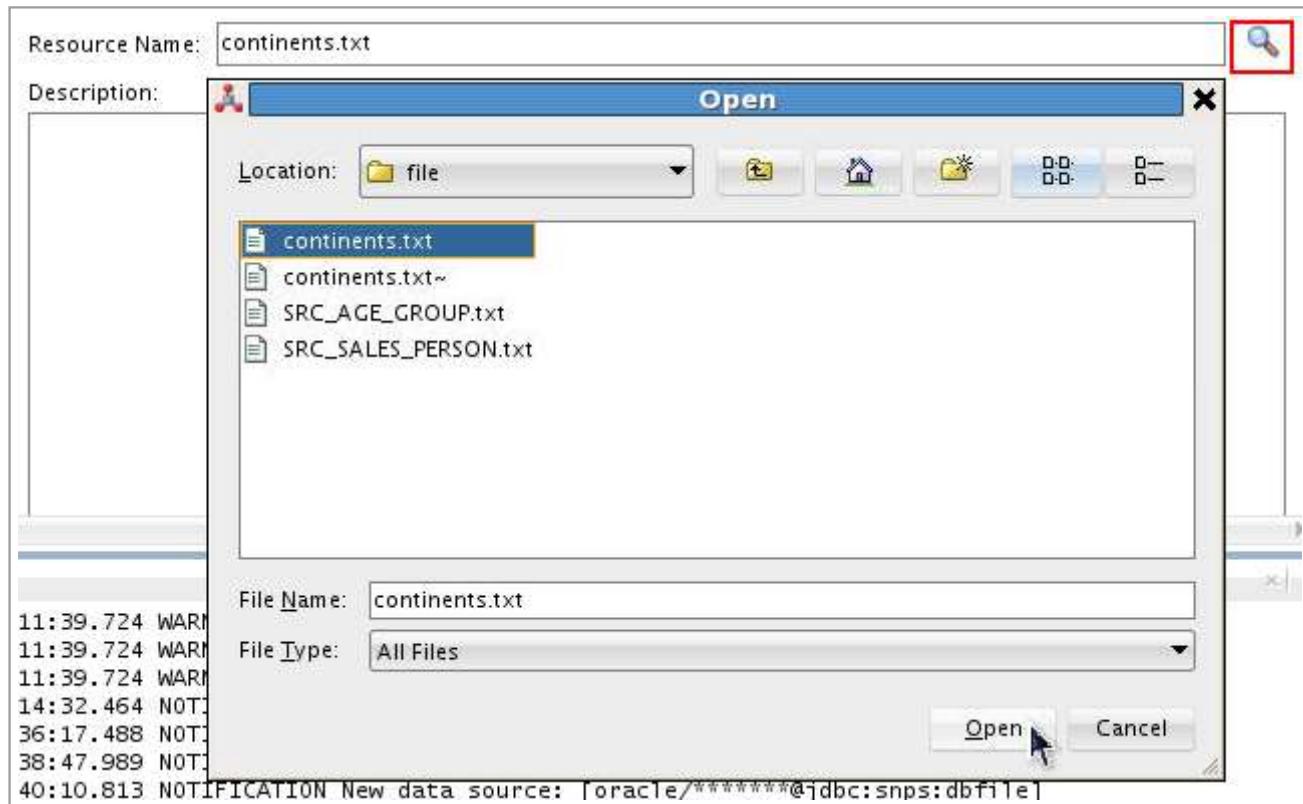
Name: **continents** Alias: CON

Datastore Type: Table OLAP Type: <Undefined>

Resource Name: continents

Description:

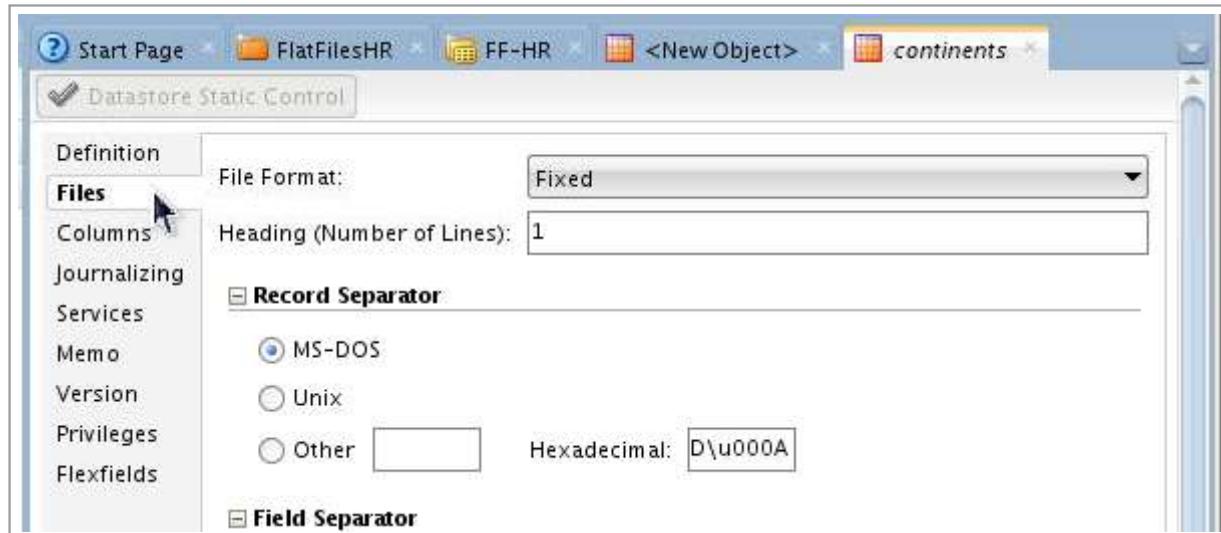
On the **Resource Name** line, click the magnifying glass to browse for the proper file name. Navigate to the `$ODI_HOME/demo/file` directory and select **continents.txt** to populate the **File Name** field.



Click **Open** to continue.

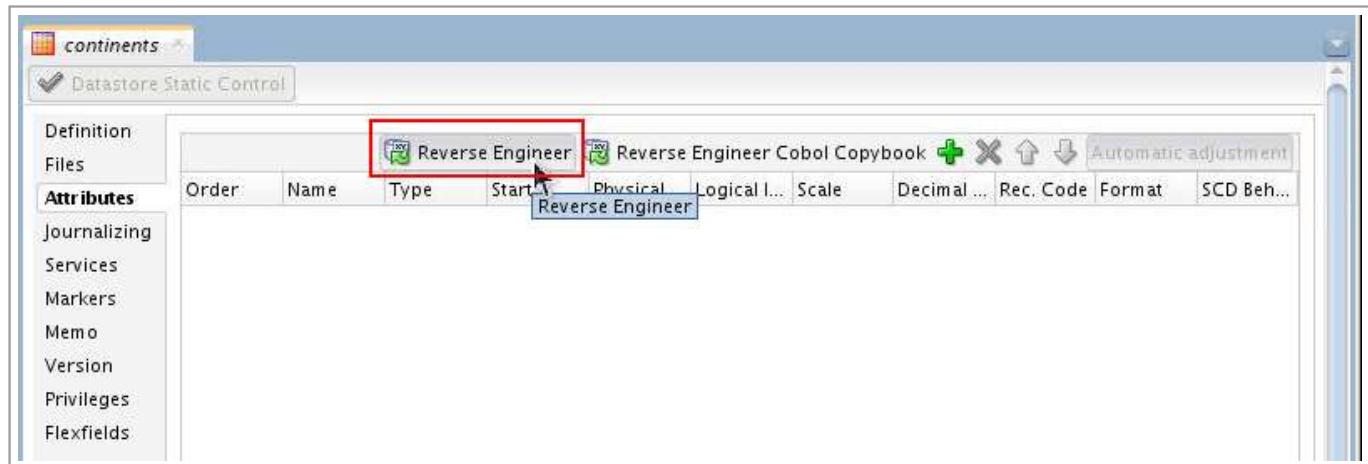
On the **Files** tab, on the **File Format** pull-down, select **Fixed** (as opposed to Delimited).

On the **Heading** field, enter **1** so that it knows that the very first row in the text file is column names, not data.

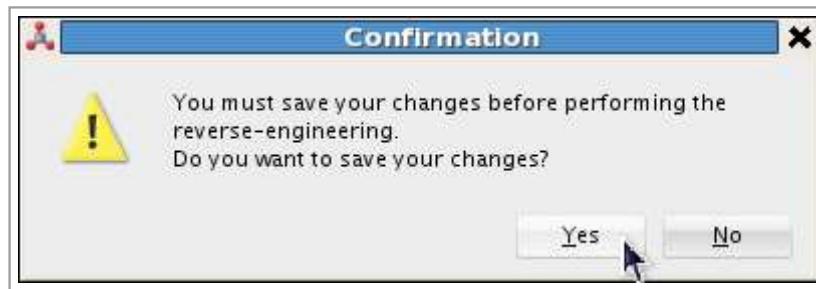


Don't change anything else, but notice the **Record Separator** option defaults to **MS-DOS**. Earlier you ran `unix2dos` on the `continents.txt` file. Had you not done that, you could have accommodated the UNIX-style line endings here by selecting **Unix**.

On the **Attributes** tab, click **Reverse Engineer**.



If you did not save your work already by clicking , you have to save it now by clicking to continue.



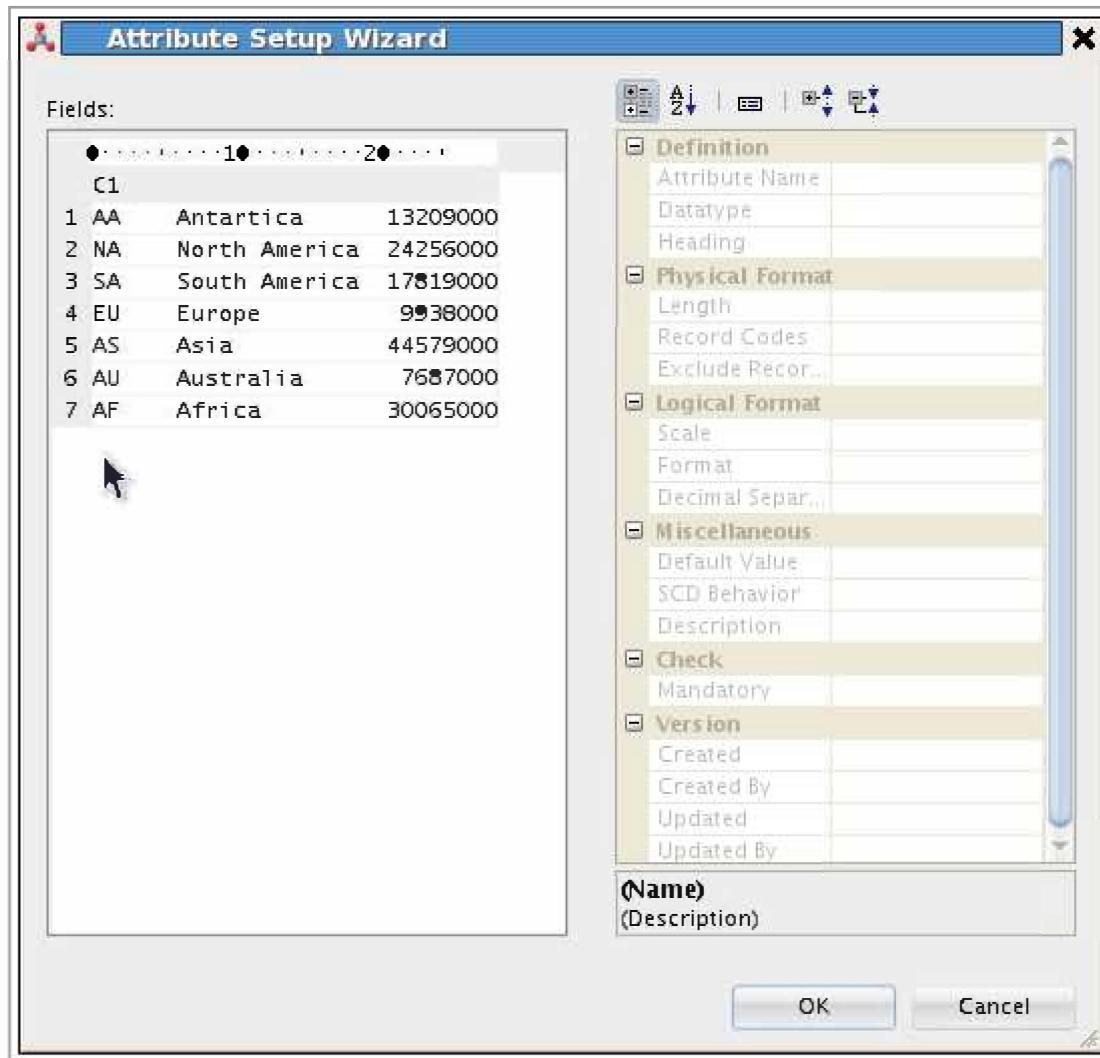
The **Locking Object** dialog box shows up all the time. In a multi-user ODI environment (multiple people using the *same* schema at the same time), you *do* want to always lock the objects during editing (that is, "Yes"). In a single-user ODI environment, locking is an extra step that is unnecessary (that is, "No"). Make

your selection in either case and optionally select **Don't show this window next time** to remember that decision.

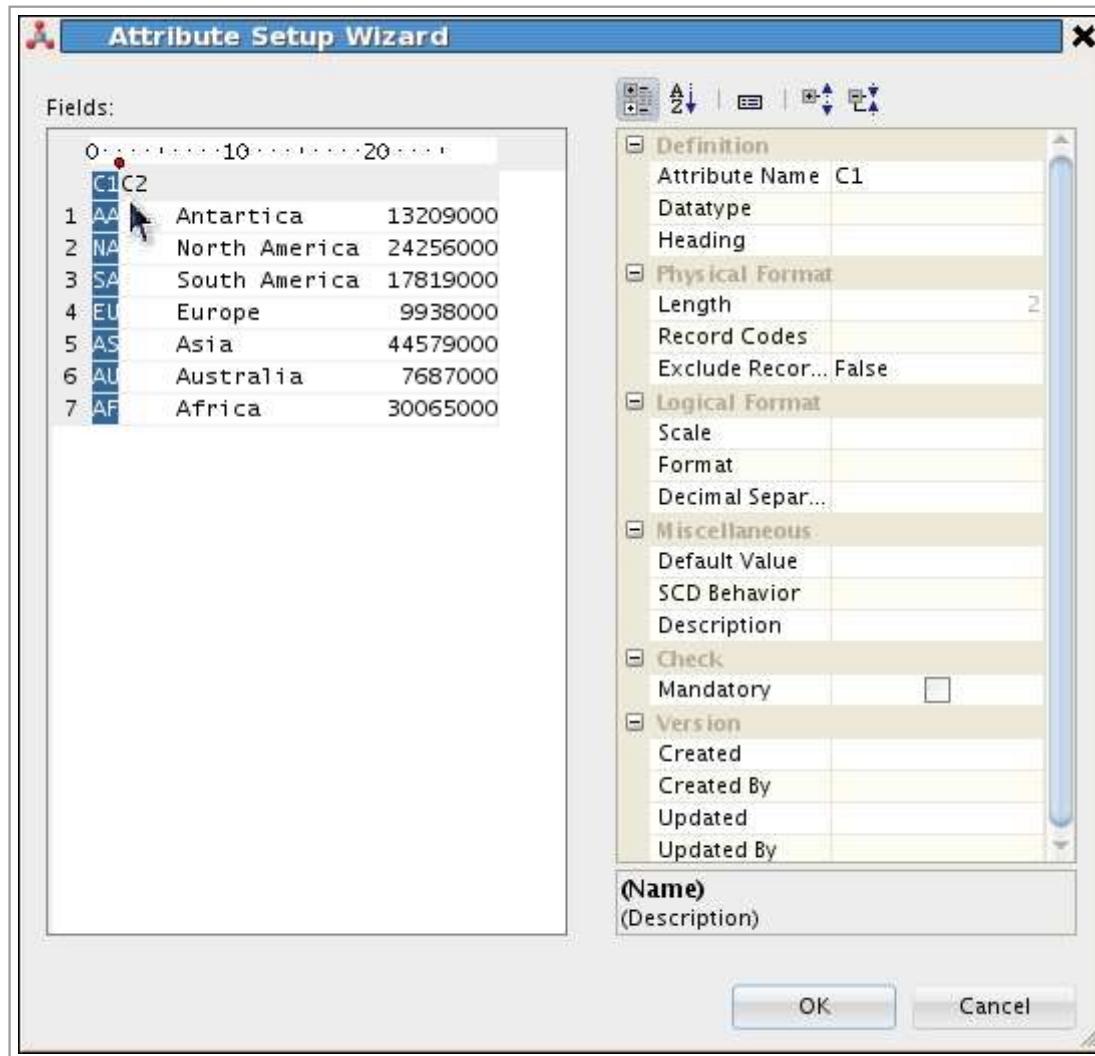


Based on your environment, click **Yes** or **No** to continue.

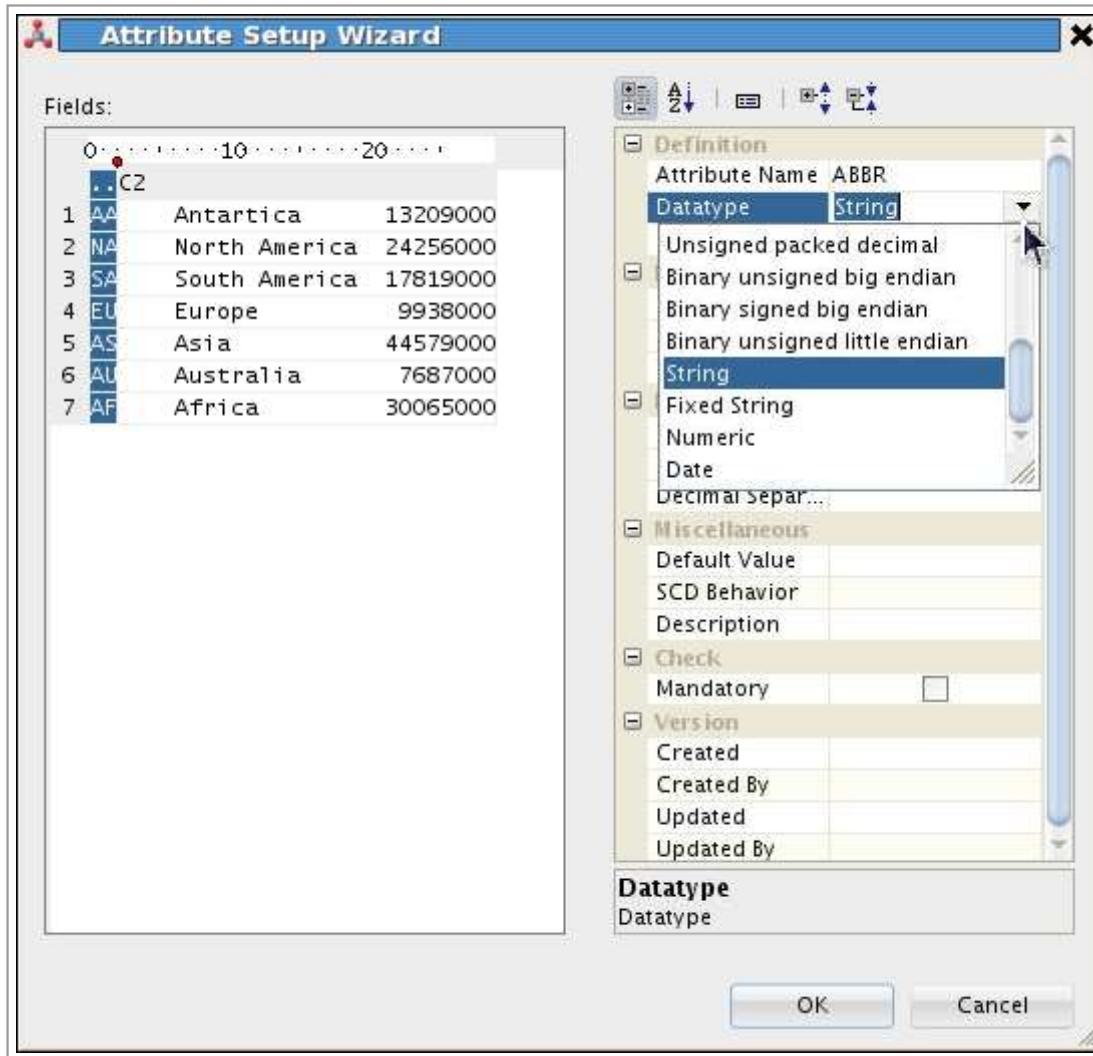
The **Attribute Setup Wizard** (previously known as the Column Setup Wizard) opens and displays the first couple of rows of the text file table. Notice that the column headings in the file do not count as the first row.



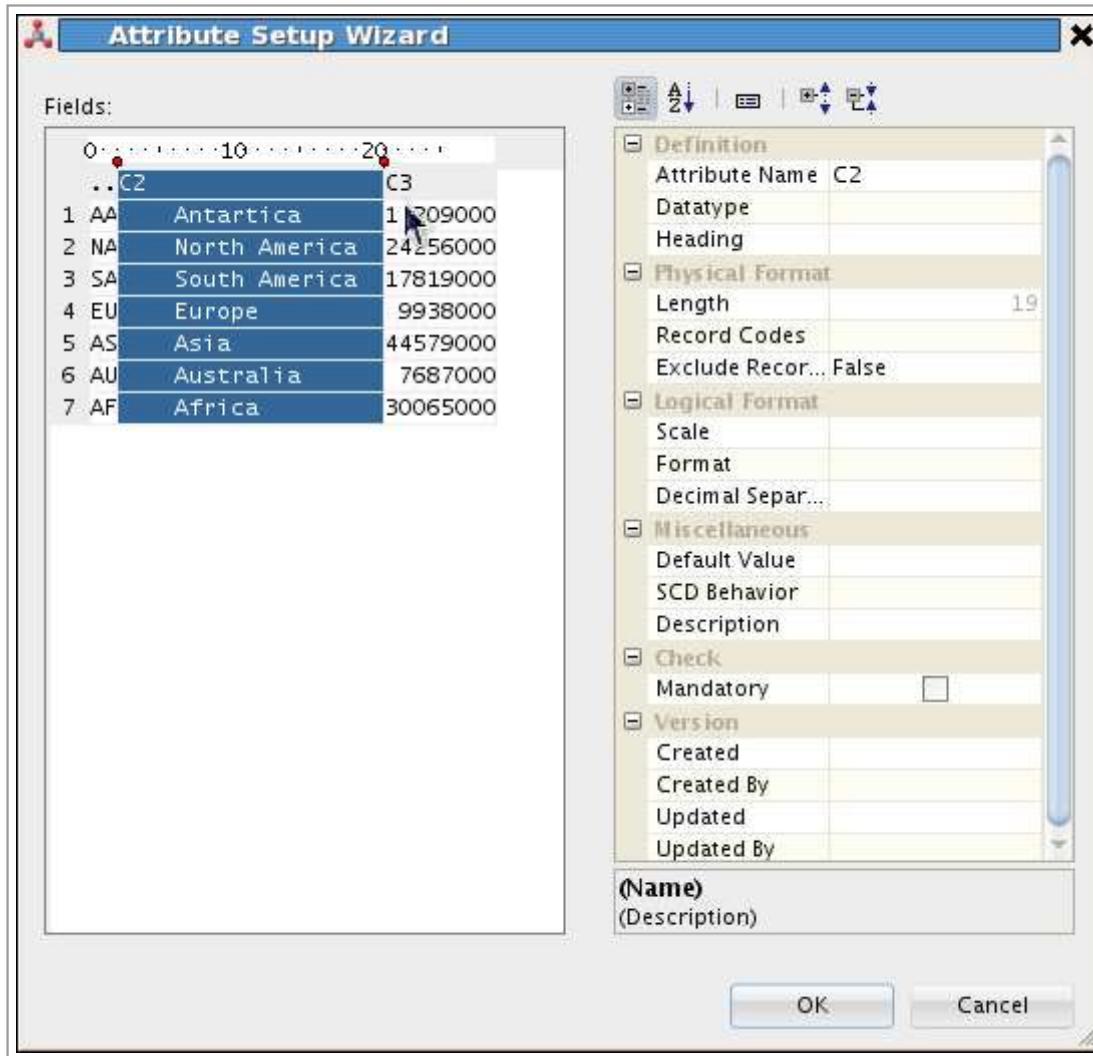
On the ruler, click at the end of the first column of data. A red dot indicates where you clicked. For the primary key column, that would be position 2. Actually, in this example, you could click anywhere on 2 through 6 and later optionally `TRIM` the field to remove trailing blanks. The column displays as blue background to show the width to verify that you did not accidentally truncate any data.



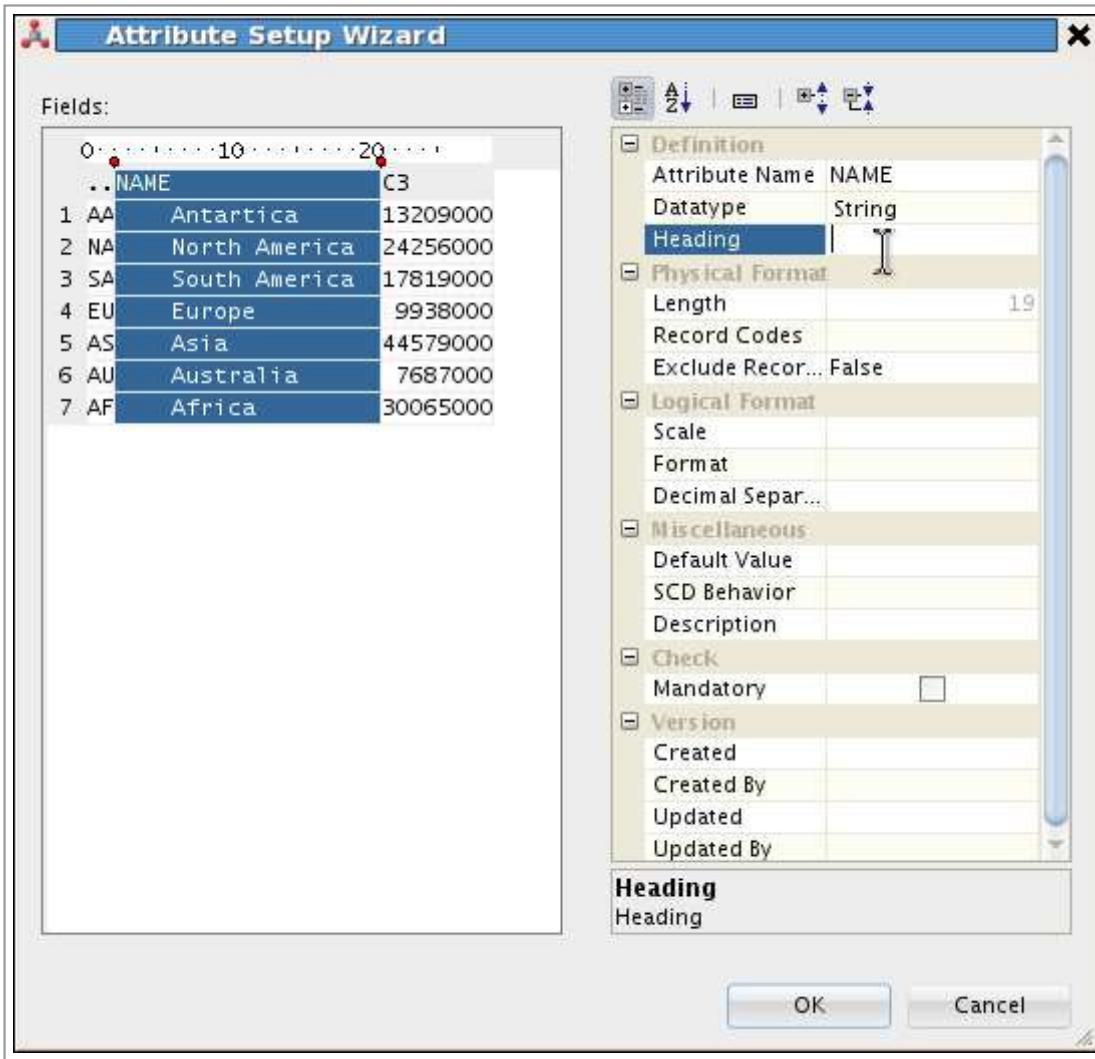
On the right side, under **Definition**, enter an **Attribute Name** (in this case a Column name) of `ABBR`, and in the **Datatype** pull-down, select **String** (or **Fixed String**). You could optionally override the **Length**, but the Wizard will calculate it for you.



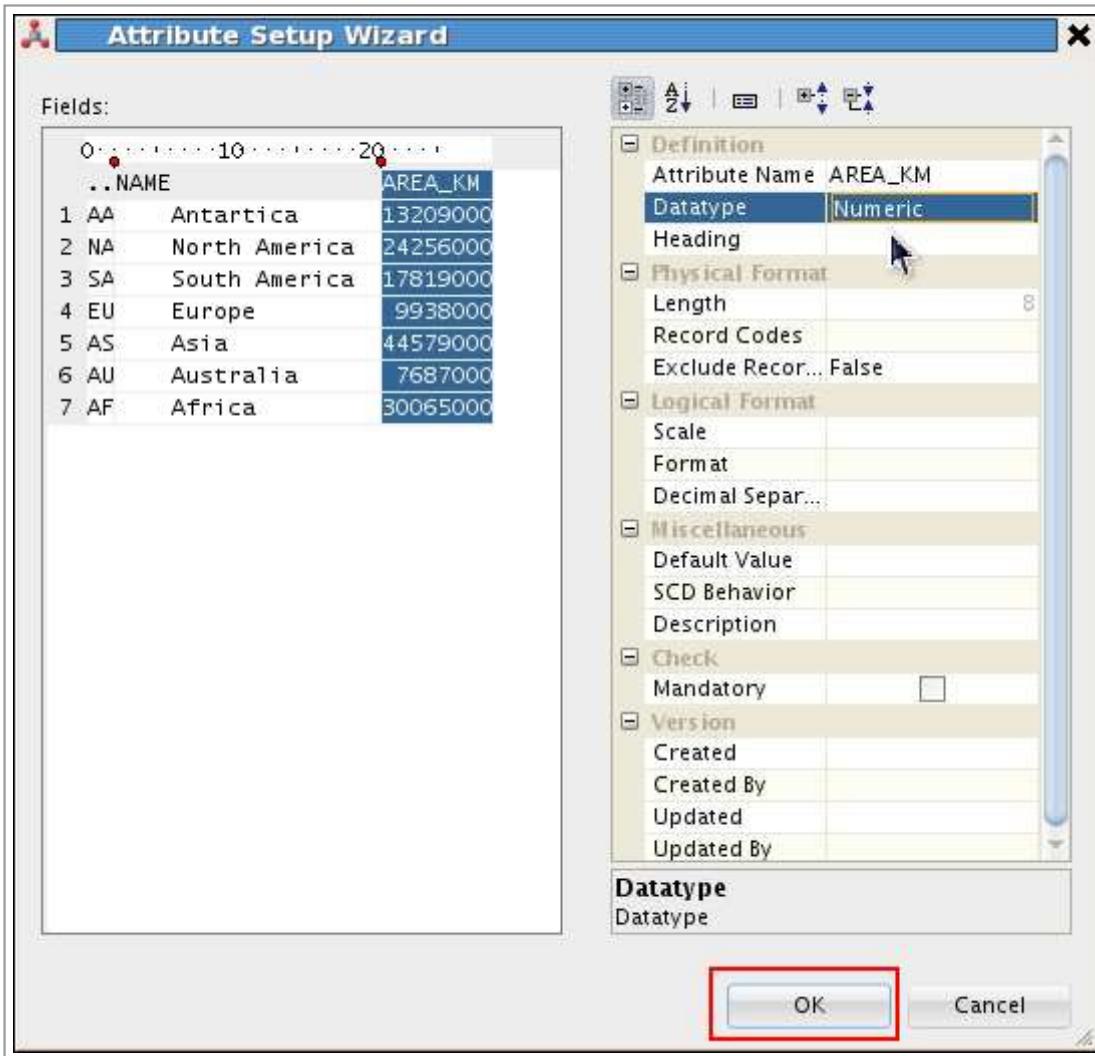
Repeat the process for the remaining columns. Click at the end of the second column (either position 19, 20, or 21). A second red dot shows the end of that second column. Again, you could later optionally `TRIM` the field to remove leading and trailing blanks.



On the right side, under **Definition**, enter an **Attribute Name** of **NAME**, and in the **Datatype** pull-down, select **String**. The **Heading** is similar to a SQL table alias, so if you had a column name that was cryptic, for example, **TAB4COL2**, you could give it a more readable name as well.



The last column is whatever data remains, so the red dot click on the ruler is not needed. Click the column heading on the left to get the Definitions on the right. Enter an **Attribute Name** of `AREA_KM`, and in the **Datatype** pull-down, select **Numeric**. Had this been a money amount, you would have to also select **Logical Format** and **Scale** and **Format** to indicate fractional digits, thousands and decimal separators.



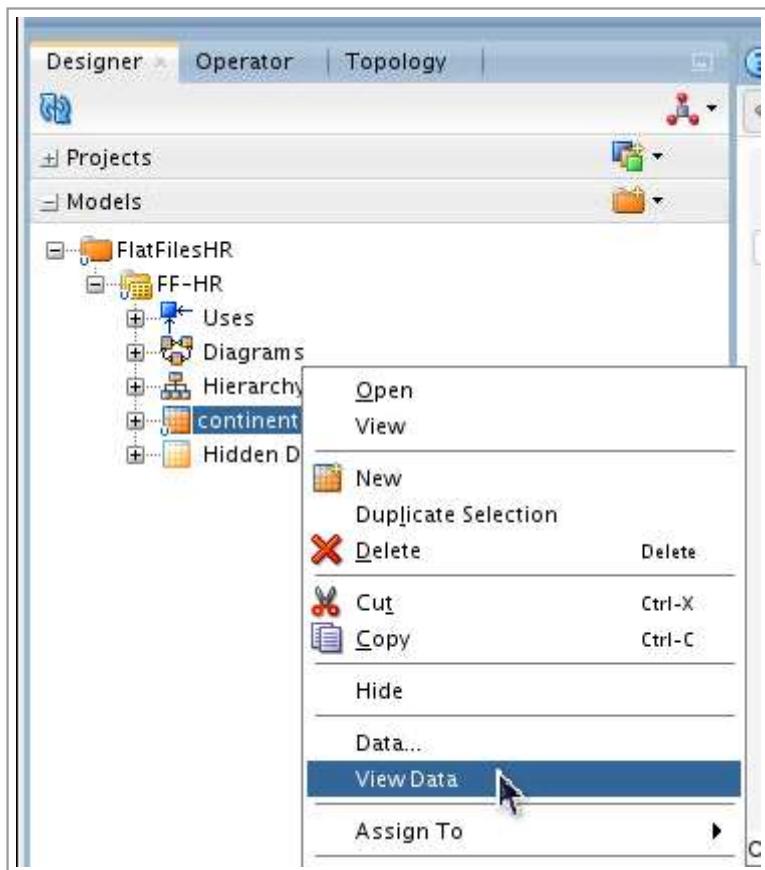
Click **OK** to continue.

The screenshot shows the 'Datastore Static Control' window for the 'continents' datastore. The 'Attributes' tab is selected. The table displays the following attributes:

Order	Name	Type	Start	Physical ...	Logical I...	Scale	Decimal S...	Rec. Code	Format	SCD Behavior
1	ABBR	String		1	2	2				<Undefined>
2	NAME	String		3	19	19				<Undefined>
3	AREA_KM	Numeric		22	8	8				<Undefined>

The **Attributes** tab displays the quantitative results of the Wizard. You could fine-tune it at this point (or you could have entered the whole thing this way). Save your work by clicking . You can close the **continents** tab now on the right.

Test whether the model is functional. On the left, in **Designer > Models > FlatFilesHR > FF-HR**, right-click on **continents**, and select **View Data**.



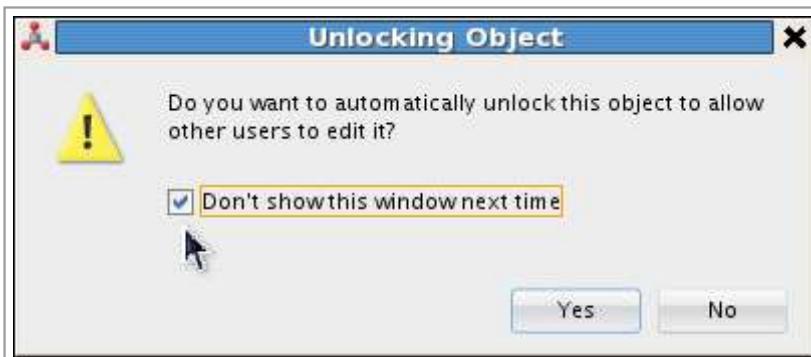
The data appears on the right.

The screenshot shows the 'Data: continents' tab in the ODI Data View window. The table displays the following data:

	NAME	AREA_KM
1	AA Antarctica	13209000
2	NA North America	24256000
3	SA South America	17819000
4	EU Europe	9938000
5	AS Asia	44579000
6	AU Australia	7687000
7	AF Africa	30065000

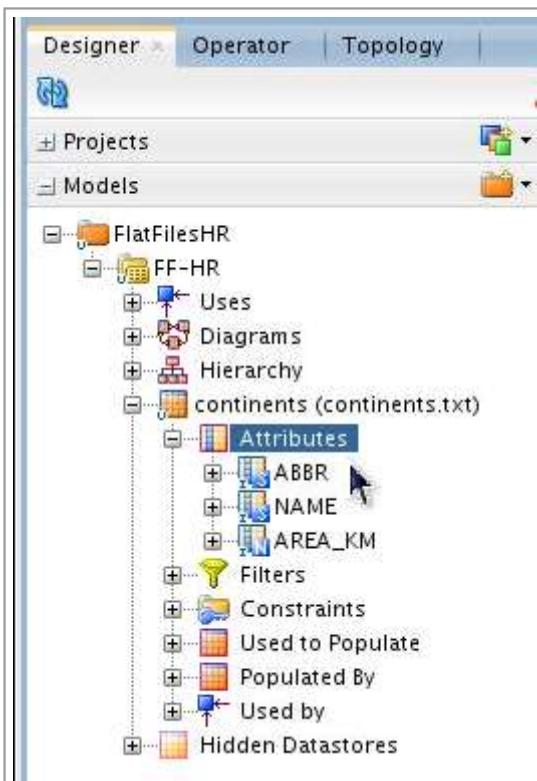
You can resize columns, go to top/bottom/next/previous rows, refresh, as well as add/edit/delete data. Close the **continents** tab after you have viewed the data.

Just as you had the **Locking Objects** in previous task 8, here is the corresponding **Unlocking Objects**. Assuming that the data displayed correctly, you would want to unlock the object now. If there was an error and you needed to go back and re-edit the model, you might leave it locked at this point.



Click **Yes** to continue.

Just to confirm that all worked, you can display the individual column definitions from **Designer > Models > FlatFilesHR > FF-HR > continents > Attributes**.



There is no need to do anything at this point, that panel was just another way to confirm the previous work.

In summary, you created a Designer Model to represent the source text file.

## 2. Preparing the Target Table

This is very similar to the source definitions you just did.

To prepare a relational table in an Oracle Database to be the target of an export, perform the following steps:

## 2.1 Creating the SQL Table Definition

Does your database have the sample schemas installed? Is the `HR` schema unlocked? If so, *SKIP THE REST OF THIS STEP AND GO TO STEP 2.1.2*. If not, then create the `HR` sample schema by signing on to `sqlplus` with DBA privileges using an account such as `sys` or `system`.

### DBA Prompt

```
[myuser@myhost ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.1.0.2.0 Production on Mon Jul 29 13:27:03 2015
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Last Successful login time: Fri Jul 26 2015 15:02:23 -04:00
Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options

SQL> CREATE USER hr IDENTIFIED BY hr ACCOUNT UNLOCK;
User created.

SQL> GRANT CONNECT, RESOURCE, CREATE TRIGGER, CREATE VIEW TO hr;
Granted.

SQL> exit

[myuser@myhost ~]$
```

When you are done creating the sample user `HR`, exit out of the DBA account (just to be safe).

Sign on to `sqlplus` as user `HR` and create the empty `continents` table.

### SQL Prompt

```
[myuser@myhost ~]$ sqlplus hr/hr
```

```
SQL*Plus: Release 12.1.0.2.0 Production on Mon Jul 29 13:27:03 2015
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Last Successful login time: Fri Jul 26 2015 15:02:23 -04:00
Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options
```

```
SQL> DROP TABLE hr_continents;
Table dropped.
```

```
SQL> CREATE TABLE hr_continents
  ( abbr      VARCHAR2(4)
  , name      VARCHAR2(25)
  , land_km  NUMBER(12)
  , CONSTRAINT tab_contin_pk PRIMARY KEY(abbr)
  );
```

```
SQL> DESC hr_continents;
Name          Null?    Type
-----  -----
ABBR          NOT NULL VARCHAR2(4)
NAME           VARCHAR2(25)
LAND_KM        NUMBER(12)
```

```
SQL> SELECT * FROM hr_continents;
```

no rows selected

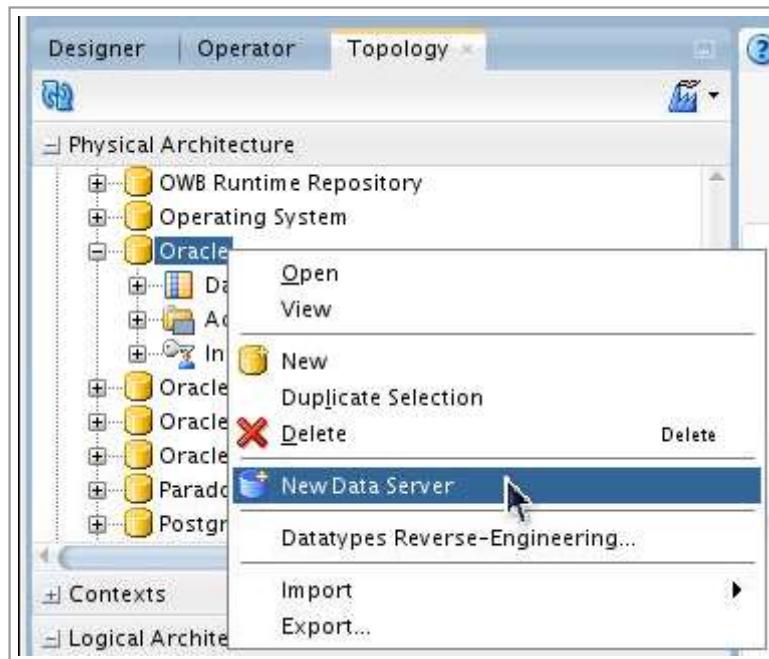
```
SQL>
```

If a table does not exist at the moment, attempting to drop a non-existent table will give an error. You can safely ignore the error.

In summary, you now have a sample user `HR` and an empty sample target relational table `hr_continents`.

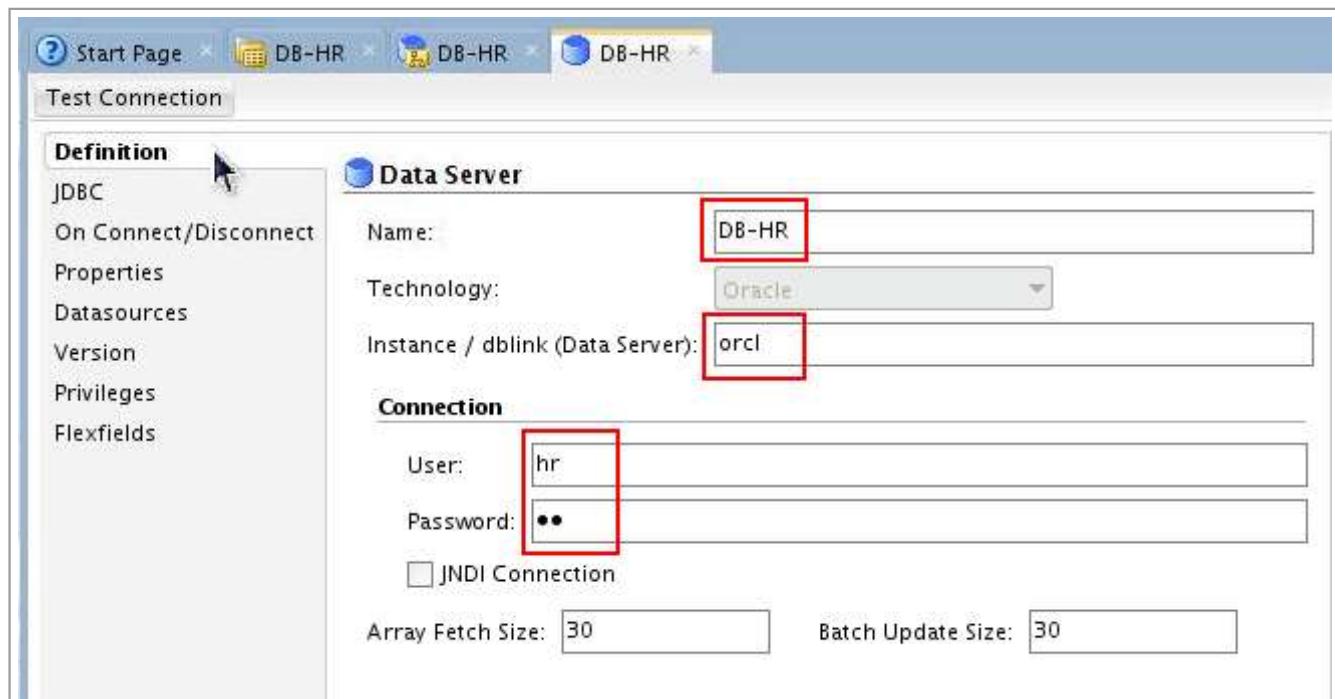
## 2.2 Defining Target Topology Physical Architecture

On the left, click the **Topology** tab, expand **Physical Architecture**, then expand **Oracle**. Right-click **Oracle** and select **New Data Server**.



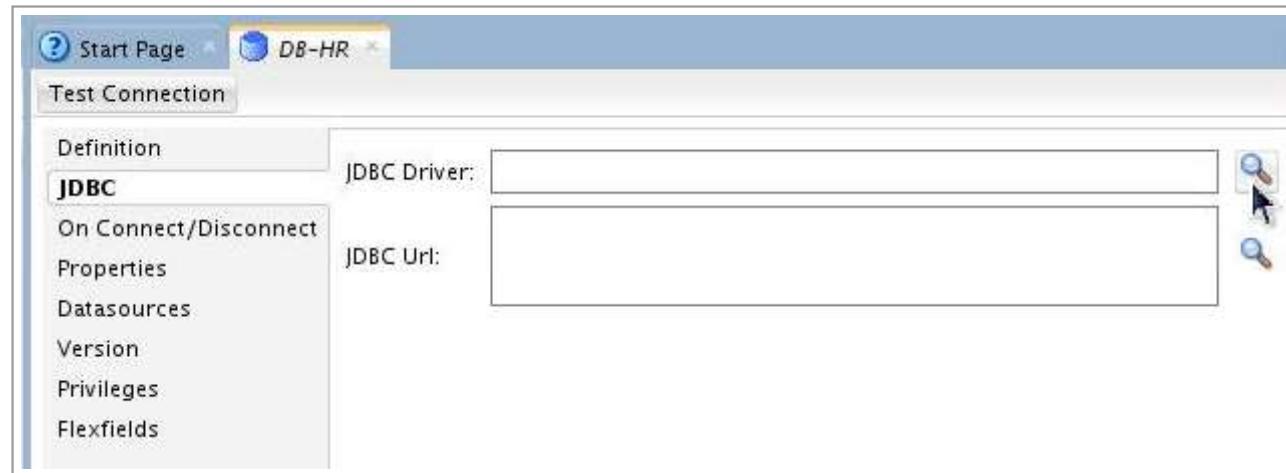
On the right, the **Data Server** panel opens on the **Definition** tab.

- In **Name**, enter `DB-HR`.
- In **Instance**, enter `orcl` (note that this is not the longer service name).
- In the **Connection** section, in **User**, enter the schema user `hr`. This is either the schema owner or someone else with privileges to read/write in that schema, for example, a DBA.
- In **Password**, enter the password for the user.

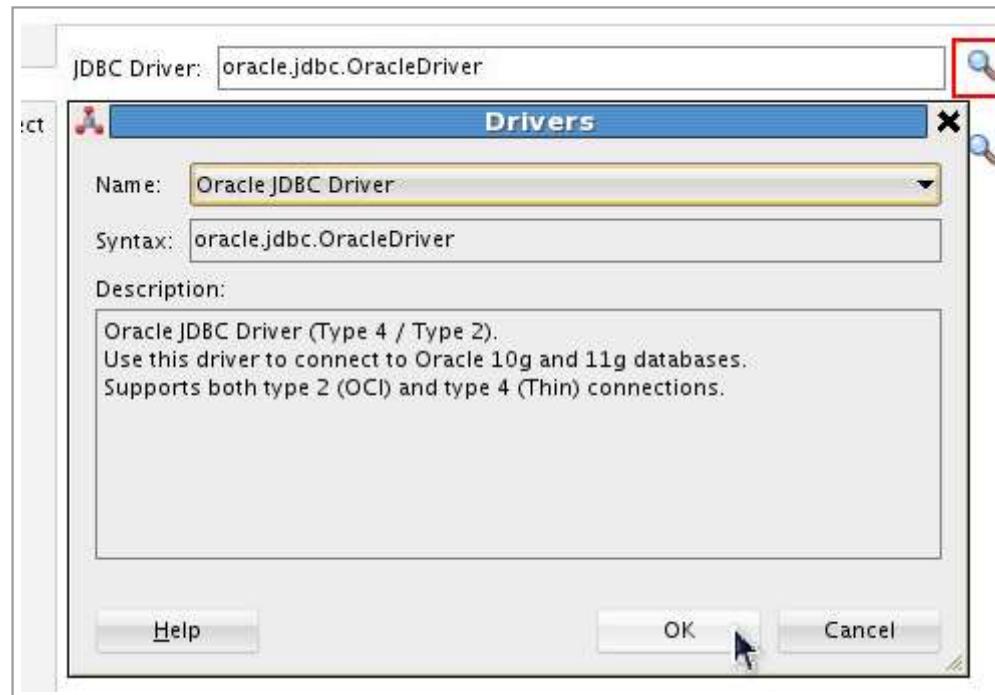


By default, the RCU owner **DEV\_ODI\_REPO** does not have global read/write privileges, and therefore could not be the user here.

On the **JDBC** tab, on the **JDBC Driver** line, click the magnifying glass to select the driver.



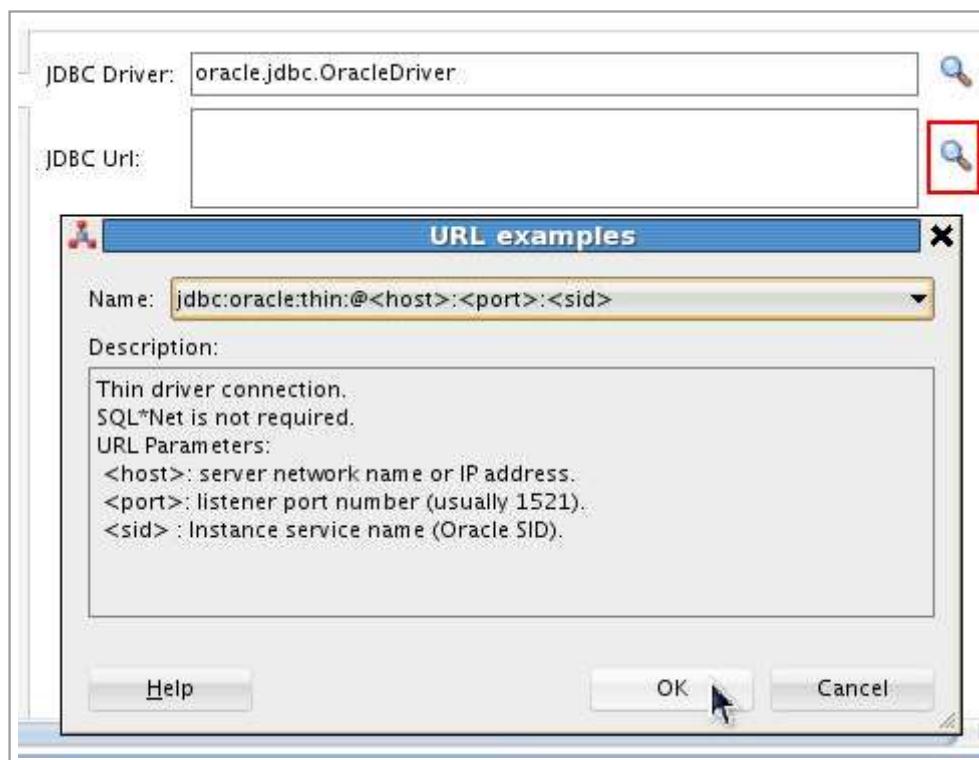
In the **Name** pull-down, select **Oracle JDBC Driver**.



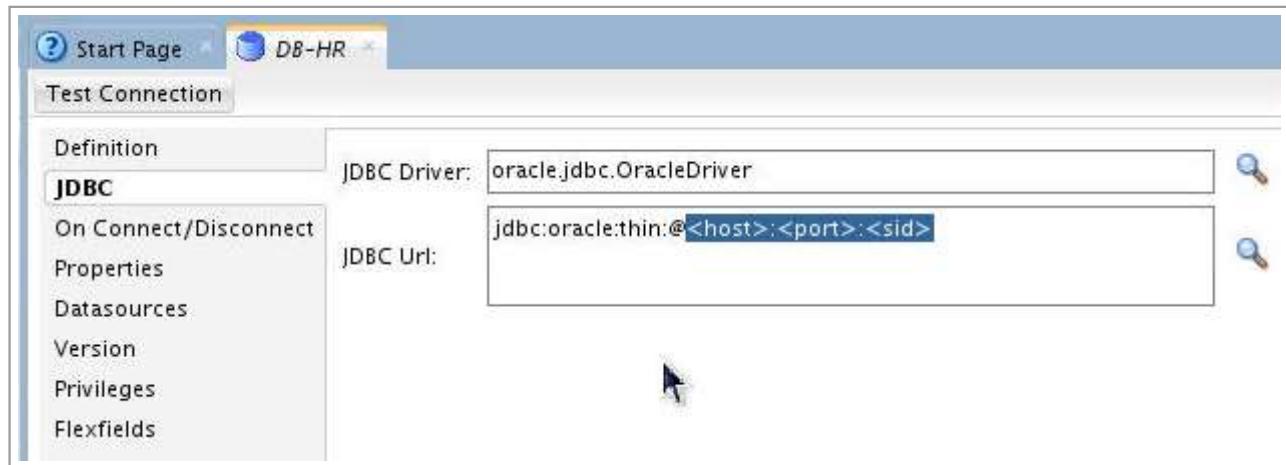
Click **OK** to continue.

On the **JDBC Url** line, click the magnifying glass to select the URL. The **URL Examples** dialog box displays.

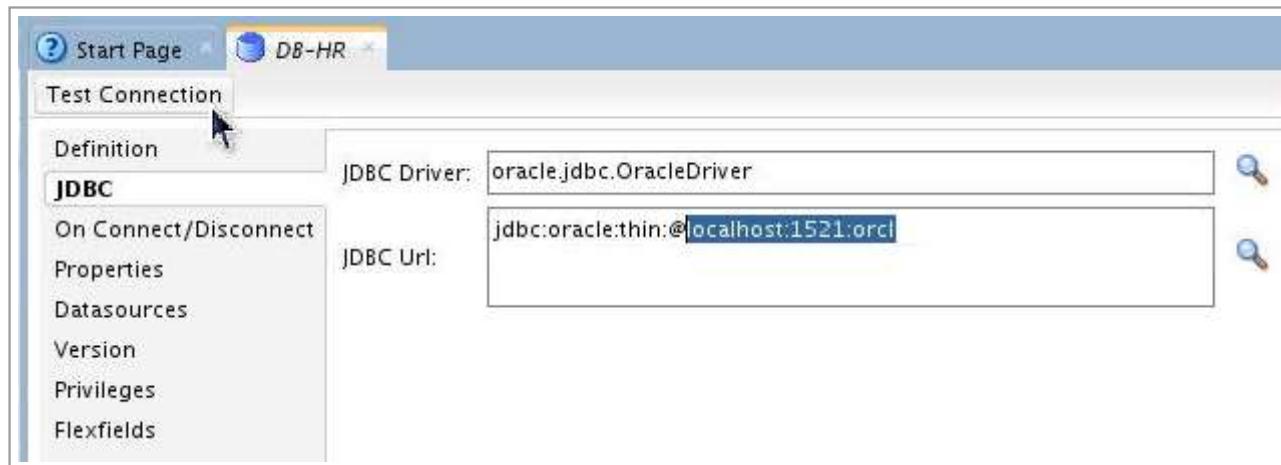
Select **jdbc:oracle:thin@<host>:<port>:<sid>**.



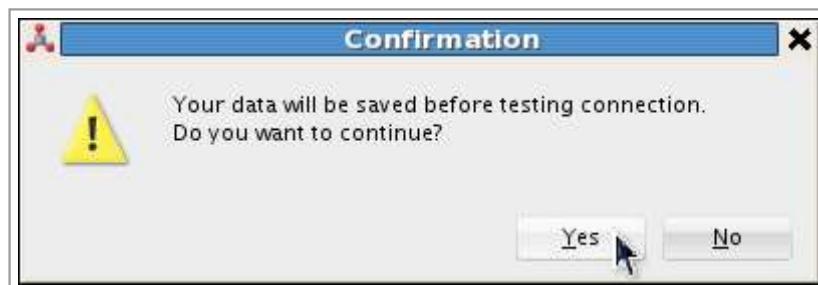
Click **OK** to continue. You will need to replace the variables with actual values.



In **JDBC Url**, replace **<host>:<port>:<sid>** with **localhost:1521:orcl**, or whatever is appropriate for your environment.

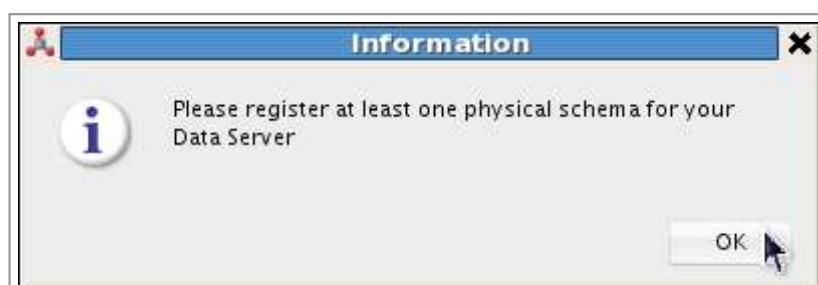


Click **Test Connection** to verify connectivity. If you did not save your work already by clicking , you have to save it now.



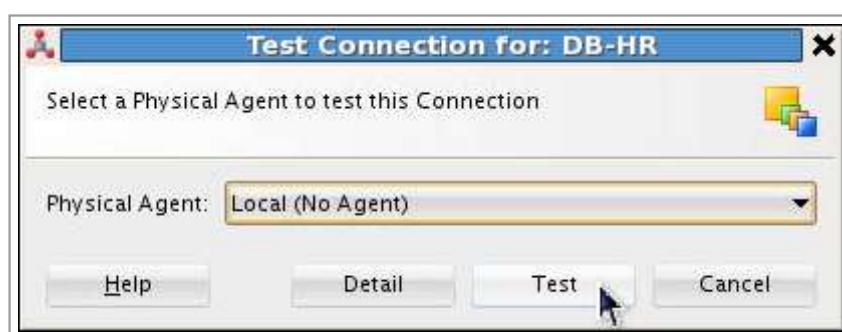
Click **Yes** to save and continue.

You will get a informational reminder that eventually you will need to make a physical (and logical) schema.



Click **OK** to continue.

On the **Test Connection** dialog box, the test will run using the default local agent (that is, no agent).

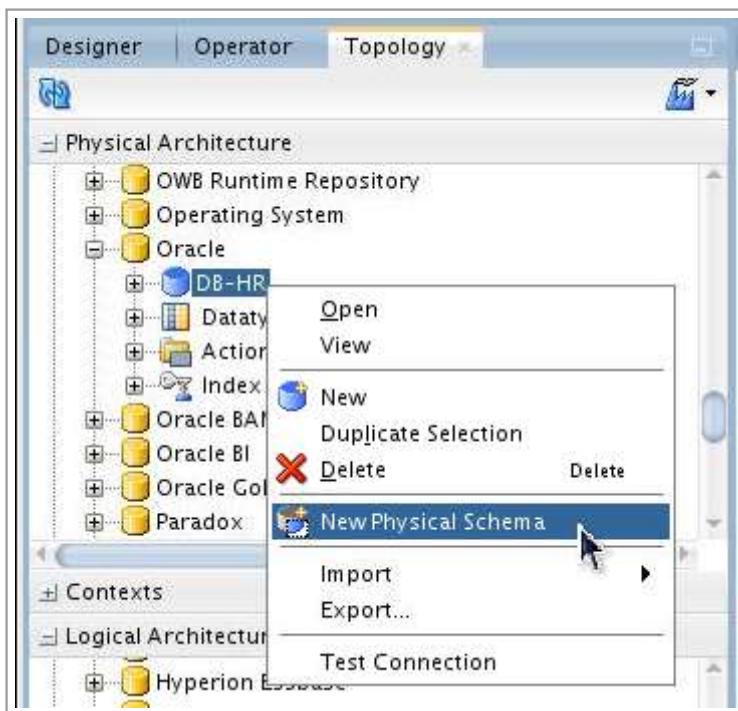


Click **Test** to run the test and continue. If connectivity is successful, you will receive a message to that effect.



Click **OK** to continue.

On the left, right-click on **DB-HR** and select **New Physical Schema**.



On the right, on **Physical Schema**, on the **Definition** tab, on the **Schema (Schema)** pull-down, select **HR**. On the **Schema (Work Schema)** pull-down, also select **HR**. Note that ODI is reading this information dynamically from the Database via JDBC. Note the **Name** suffix automatically changes to the schema you just selected.

Save your work by clicking . You will get a informational reminder that eventually you will need to make a context in the **Designer** tab.

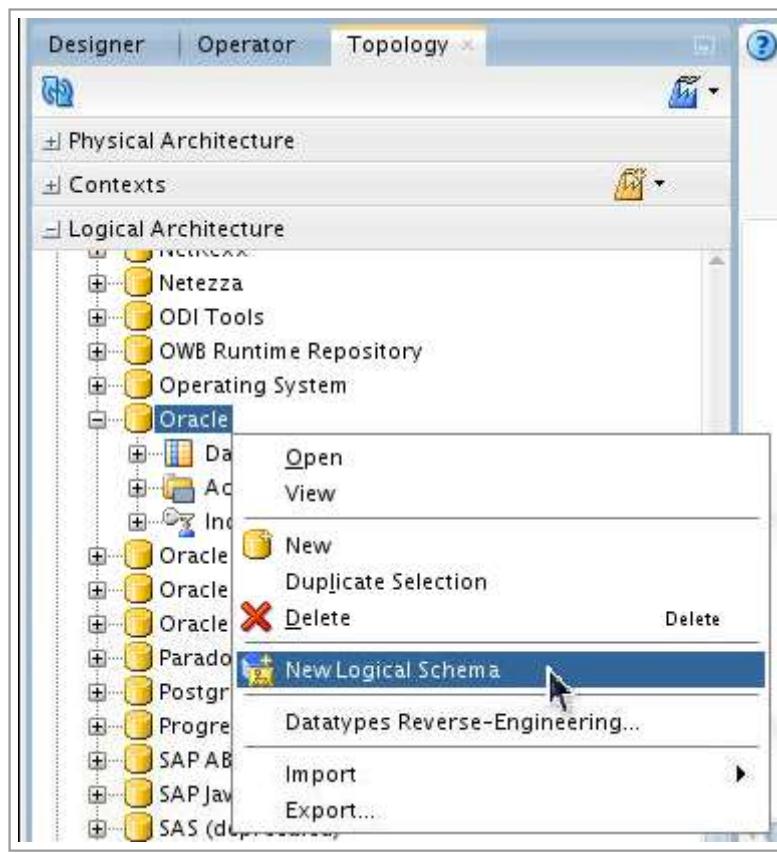


Click to continue. You can close some of the tabs (continents, DB-HR) on the right if you wish.

In summary, you created an ODI physical schema associated with the Oracle Database schema `HR`.

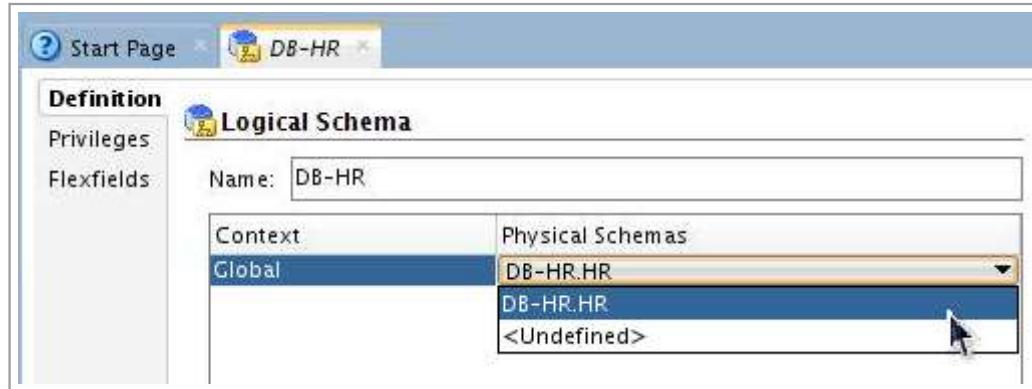
## 2.3 Defining Target Topology Logical Architecture

On the left, click the **Topology** tab, expand **Logical Architecture**, then expand **Oracle**. Right-click **Oracle** and select **New Logical Schema**.



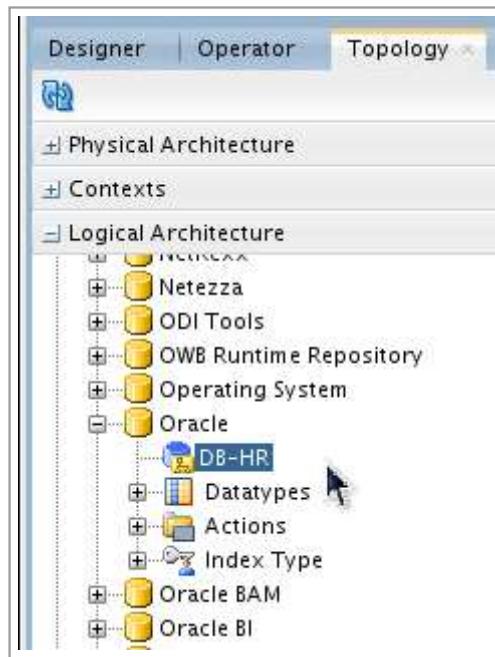
If the **Topology** panel is getting crowded, you may optionally collapse **Physical Architecture**, you will not need it for a while.

On the right **Logical Schema** panel, on the **Definition** tab, in the **Physical Schemas** pull-down, select **DB-HR.HR**.



Save your work by clicking . Close the **DB-HR** tab on the right.

You do not need to do anything, just confirm that **DB-HR** now displays under **Oracle** on the left.



In summary, you created an ODI logical schema to match the ODI physical schema which corresponds to the actual Oracle Database `HR` schema.

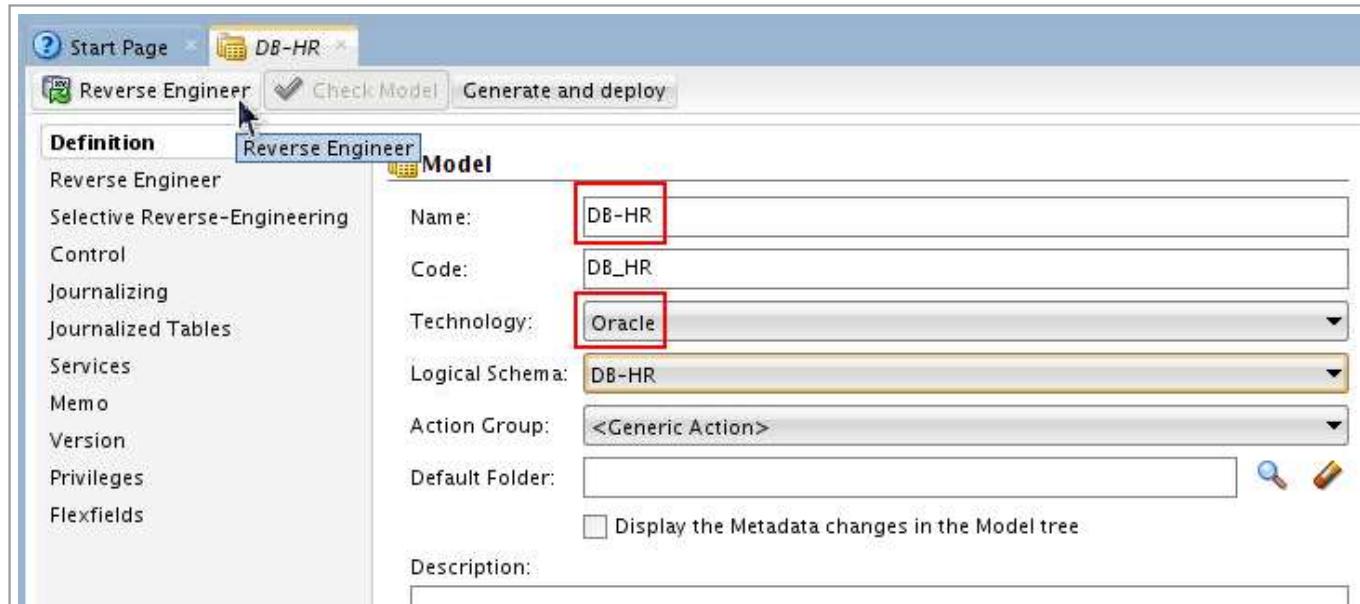
## 2.4 Defining Target Designer Model

On the left, click the **Designer** tab, expand **Models**, then under the folder icon pull-down, click **New Model**.



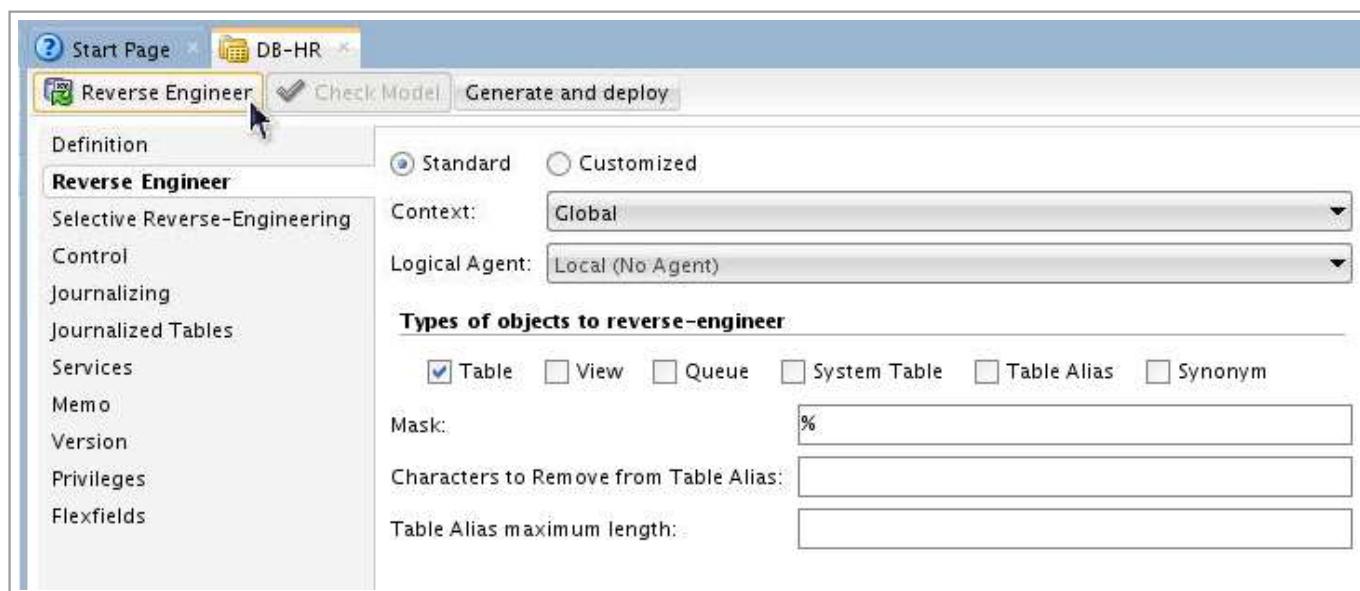
On the right **Model** panel, on the **Definition** tab, enter the following information:

- In the **Name** field, enter `DB-HR`. As you enter that name, the system automatically copies the name below to the **Code** field, except it replaces the hyphen with an underscore.
- In the **Technology** pull-down, select **Oracle**.
- In the **Logical Schema** pull-down, select **DB-HR**.

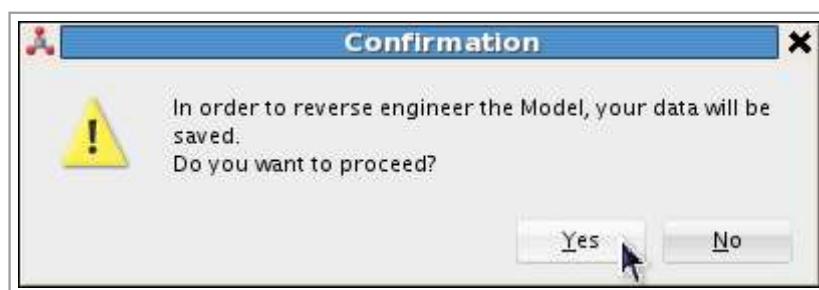


Do not click **Reverse Engineer** button yet.

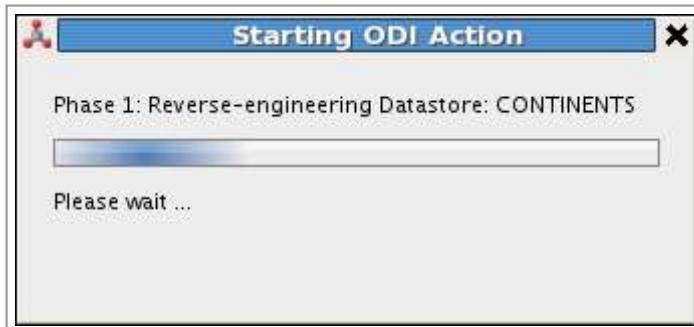
Click the **Reverse Engineer** tab, and now click **Reverse Engineer** to continue.



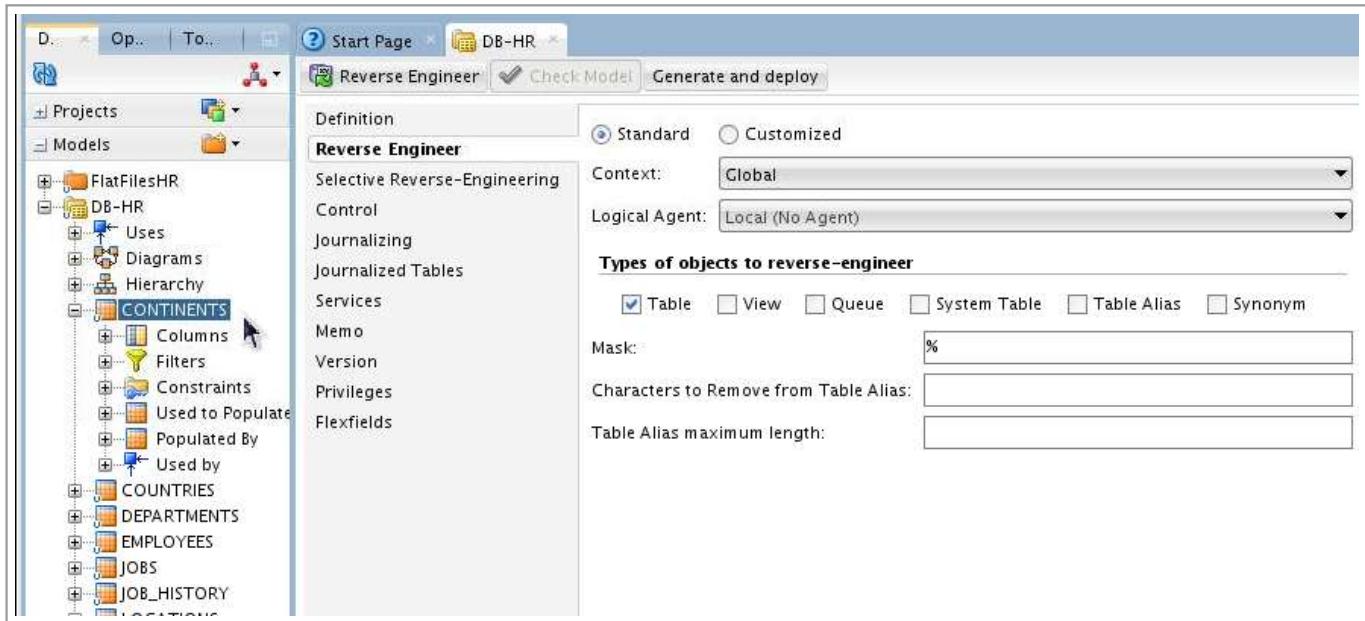
If you did not save your work already by clicking , you have to save it now.



Click **Yes** to save and continue. A progress bar displays.

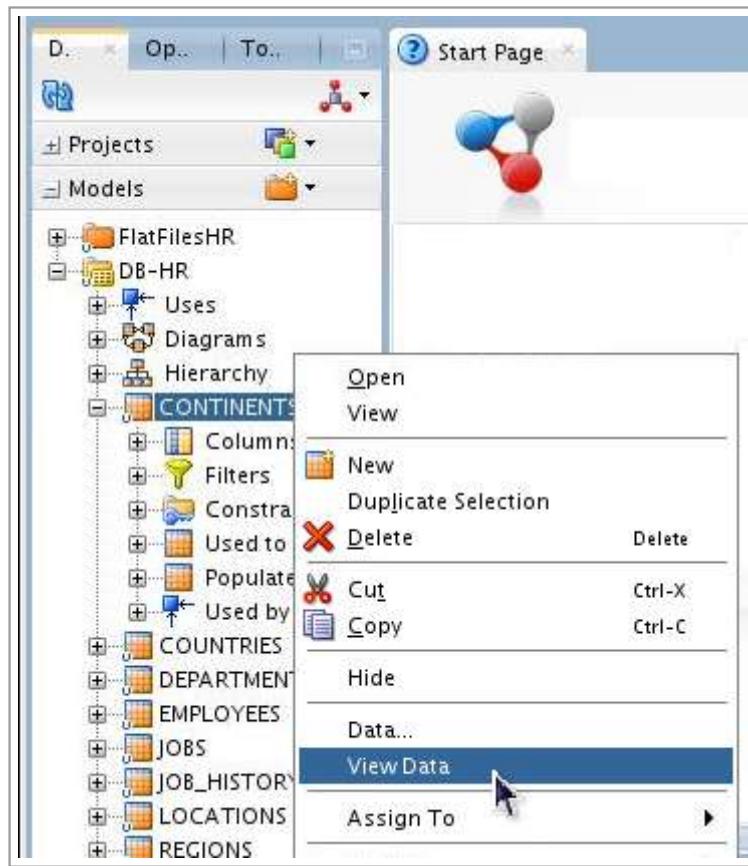


Notice that it not only gets the `CONTINENTS` table, but everything in the `HR` schema: the `EMPLOYEES`, `REGIONS`, `DEPARTMENTS` tables, and so on.

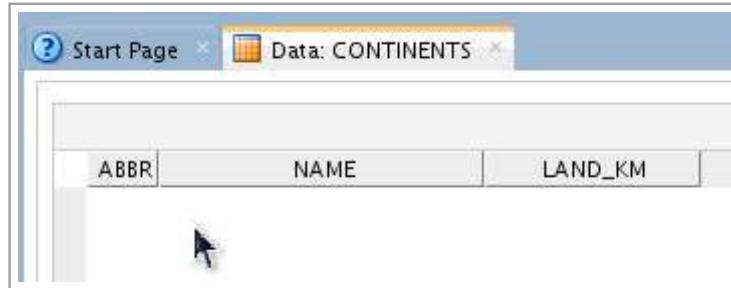


Verify on the left that the **CONTINENTS** table information is present under **Models**.

Test whether the model is functional. On the left, in **Designer > Models > DB-HR**, right-click on **CONTINENTS**, and select **View Data**.



At this point, the query should work but the table should be empty. No rows are displayed.



The next step is to do the export from the flat file source into the relational table target via a mapping.

In summary, you created an ODI Designer Model to represent the target Oracle Database relational table.

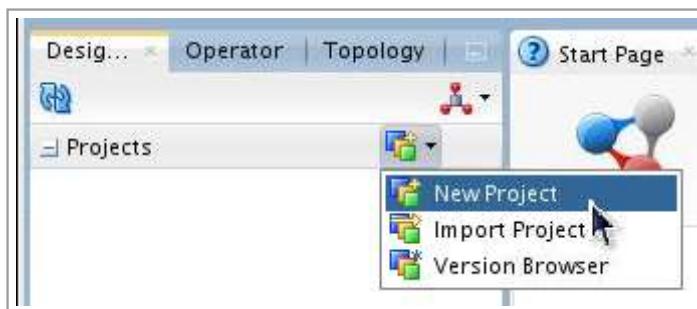
### 3. Preparing the Mapping

In previous versions of ODI, mappings were called interfaces. The Map Editor is a GUI drag-n-drop tool.

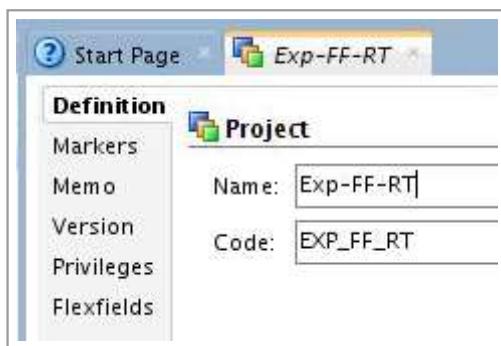
To map the flat file source to the relational table target, perform the following steps:

#### 3.1 Creating the Designer Project

On the left, click the **Designer** tab, expand **Projects**, then under the 3-box icon pull-down, click **New Project**.



On the right, on the **Project** panel, on the **Definition** tab, in **Name** enter `Exp-FF-RT` ("Export from Flat File to Relational Table"). As you type the name, the system automatically copies the name below to the **Code** field, except it replaces the hyphens with underscores.



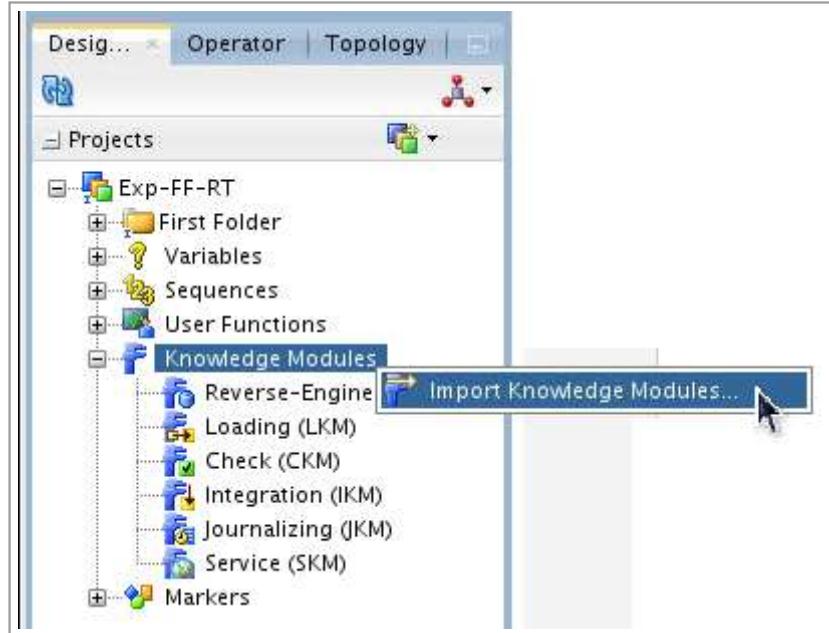
Save your work by clicking . You should now see **Exp-FF-RT** on the left as a project.

In summary, you created a project to hold the knowledge modules and maps.

### 3.2 Importing the Knowledge Modules (OPTIONAL)

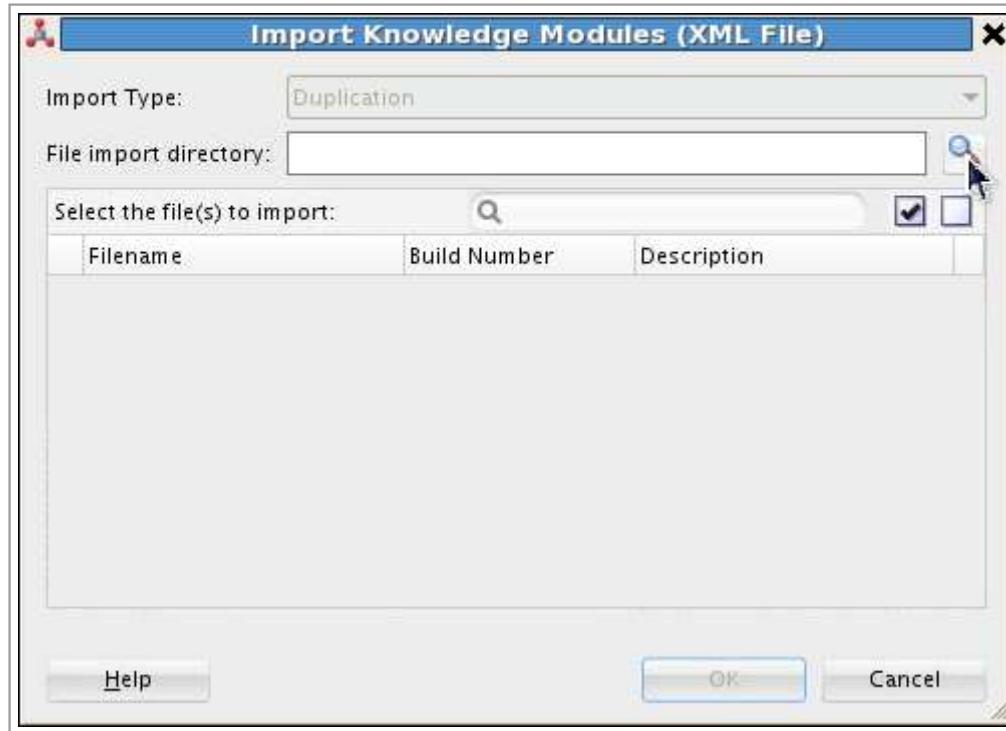
There are global knowledge modules and local knowledge modules (KM, not to be confused with land area kilometers in the tables). Import the appropriate local KMs into this project. Some KMs are pre-seeded and do not need to be imported. This particular OBE does not need any additional KMs, but three are imported optionally to show you how it is done. "Extra" (unused) KMs take up disk space but don't cause any harm.

Expand **Exp-FF-RT**, expand **Knowledge Modules**, right-click **Knowledge Modules** and select **Import Knowledge Modules...**

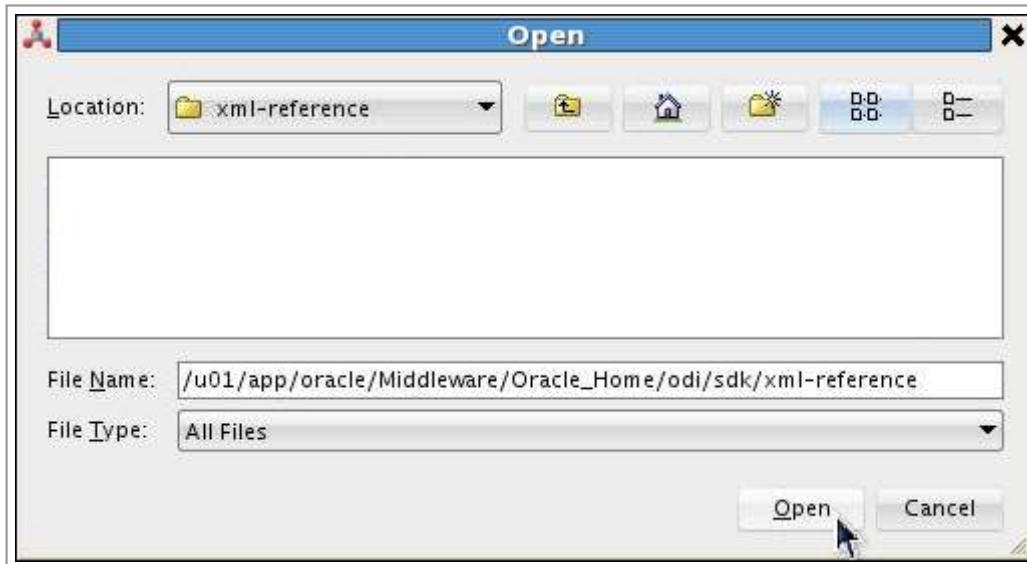


If you care to look, there are no KMs under any of the sub-KM headings yet.

On the **Import Knowledge Modules** dialog box, click the magnifying glass to browse for the proper directory.

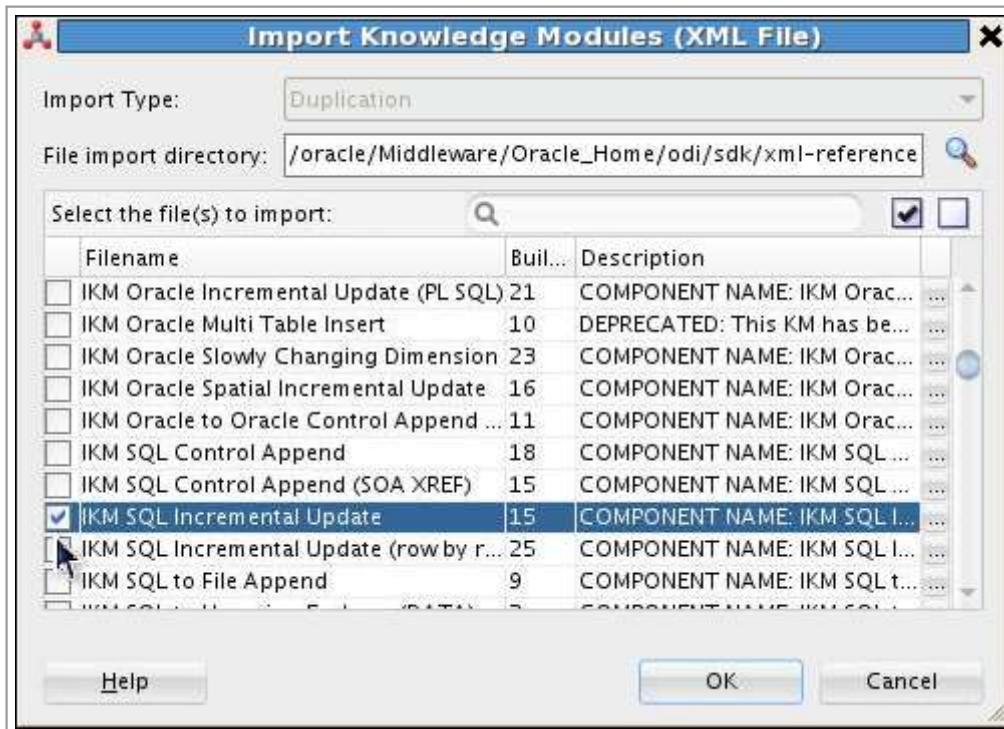


In the **Open** dialog box, navigate to the `$ODI_HOME/sdk/xml-reference/` directory.

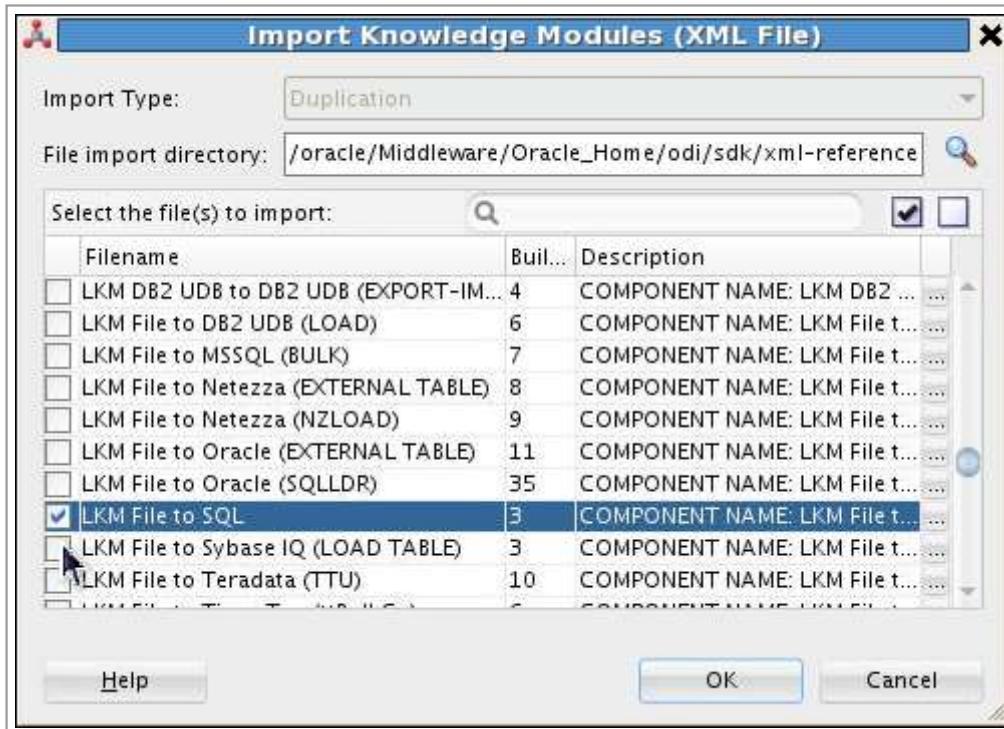


Click **Open**. It takes a few seconds to read them all.

Select two filenames: **IKM SQL Incremental Update** (not the row-by-row one), and **LKM File to SQL**.



Scroll down to find the second KM to import.



Click **OK** to continue importing. A progress bar is displayed.



An **Import Report** is displayed with the Knowledge Modules that you imported. Note that you automatically get the Check KM Hive.

**Import Report**

Duplication

**Imported Objects**

Object Type	Object Name	Imported As	Original ID	New ID After Import	Global ID	Parent Type After Import	Parent Name After Import
Knowledge Module	CKM Hive	CKM Hive	5846	35	032718a5-afad-44c0-833a-65f414aedd59	Project	Exp-FF-RT
Knowledge Module	IKM SQL Incremental Update	IKM SQL Incremental Update	473	36	4ecdbb0e-cbdf-404d-939f-3b30123ae485	Project	Exp-FF-RT
Knowledge Module	LKM File to SQL	LKM File to SQL	6661	37	c063b3ad-d1d2-4da6-bfdc-a02964349d05	Project	Exp-FF-RT

**Deleted Objects**

Object Type	Object Name	Original ID	Global ID
-------------	-------------	-------------	-----------

**Not Imported Objects**

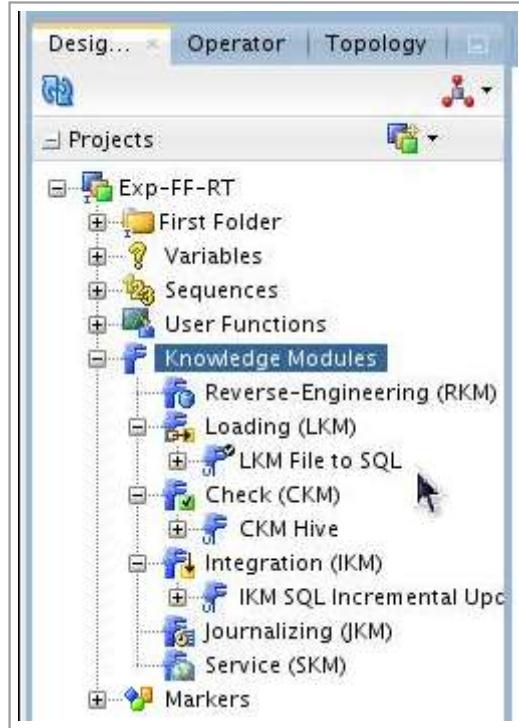
Object Type	Object Name	Original ID	Global ID
-------------	-------------	-------------	-----------

**Created Missing References**

[Help](#) [Save...](#) [Close](#)

There is nothing to do on the report, click **Close** to continue.

To verify that it imported the KMs correctly, you can optionally expand  **Loading (LKM) and Integration (IKM)** to see the added KMs.

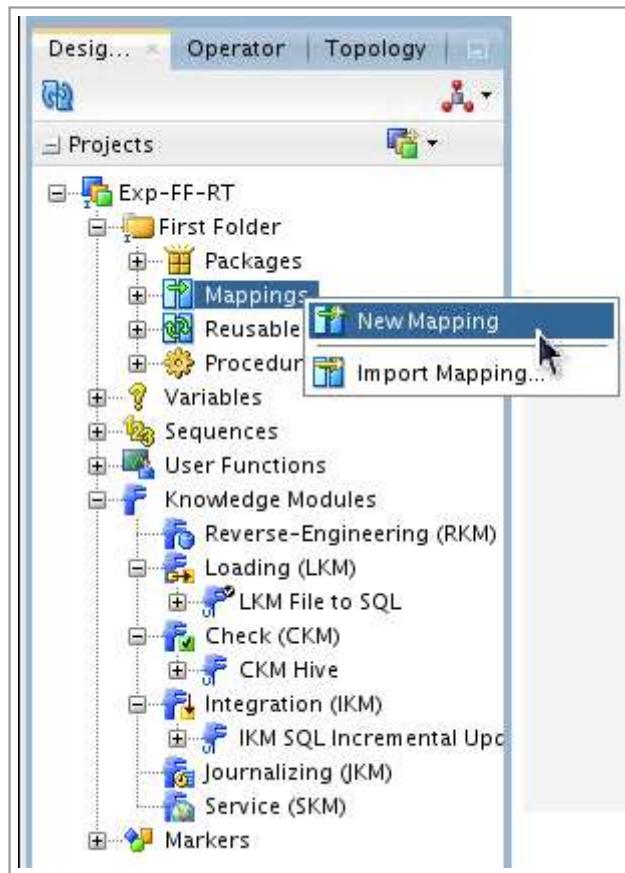


You can optionally collapse **Knowledge Modules**, you will not need it anymore.

In summary, you have imported the Loading and Integration KMs that could be used for some more sophisticated project. This particular project uses the pre-seeded KMs, therefore this whole step was optional. You can see one of the built-in (pre-seeded) LKMs mentioned on the screenshot for 4.2.2.

### 3.3 Creating the Mapping

Expand **Designer > Projects > Exp-FF-RT**, expand **First Folder**, right-click **Mappings**, and select **New Mapping**.



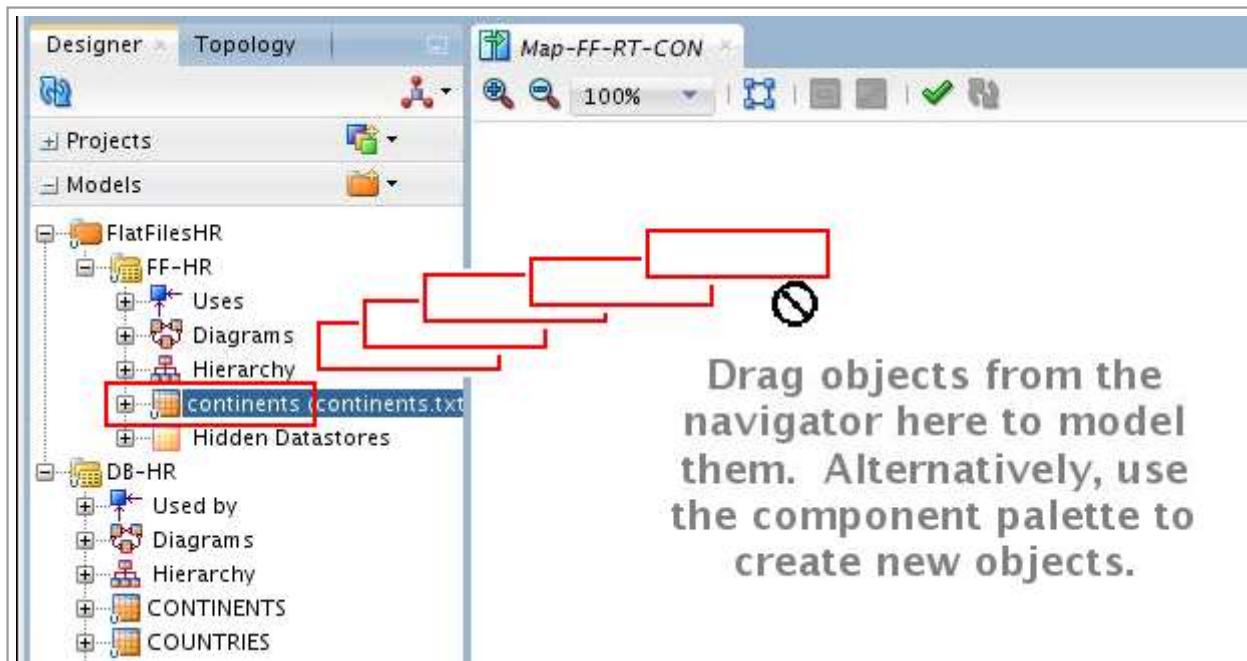
If you care to look under **Mappings**, there are no mappings yet.

In the **New Mapping** dialog box, in **Name**, enter `Map-FF-RT-CON` and an optional description. Deselect **Create Empty Dataset**. (You would only need the Empty Dataset if you are making more than one source or target.)

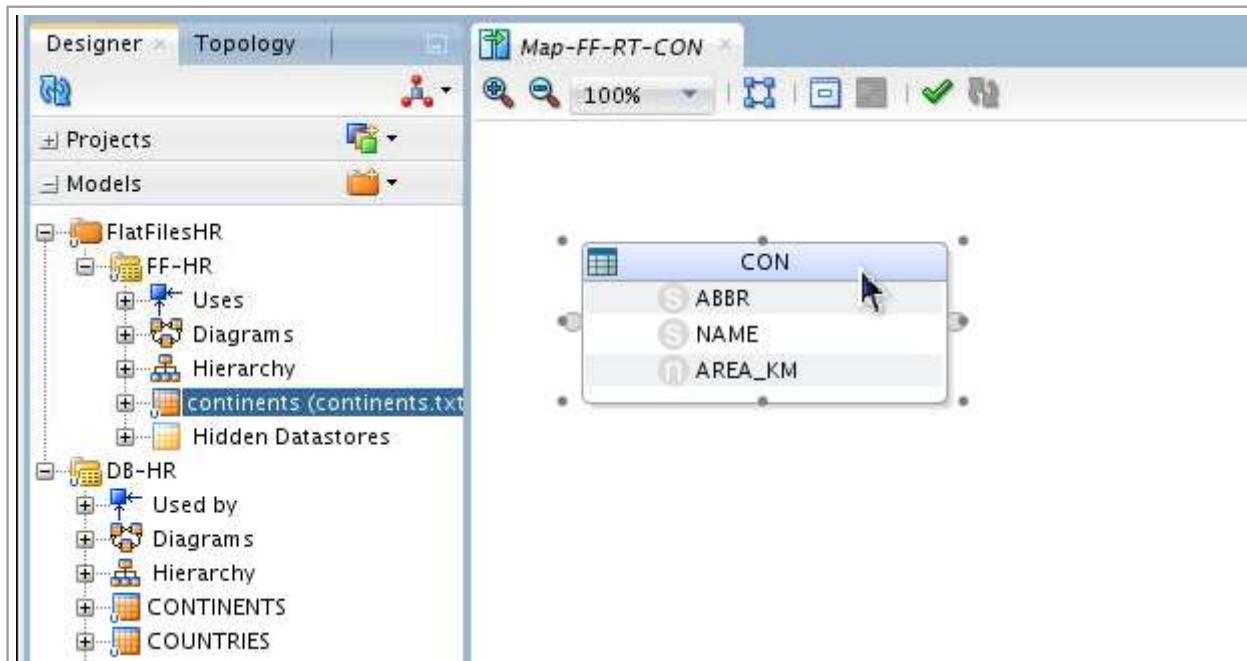


Click **OK** to continue.

When the Editor opens on the right, expand **Designer > Models > FlatFilesHR > FF-HR** and **Models > DB-HR** to expose both **continents** objects. Drag **continents** from **FF-HR** on the left to the empty pane on the right. Drop it in the area that says, "Drag objects from the navigator here..."

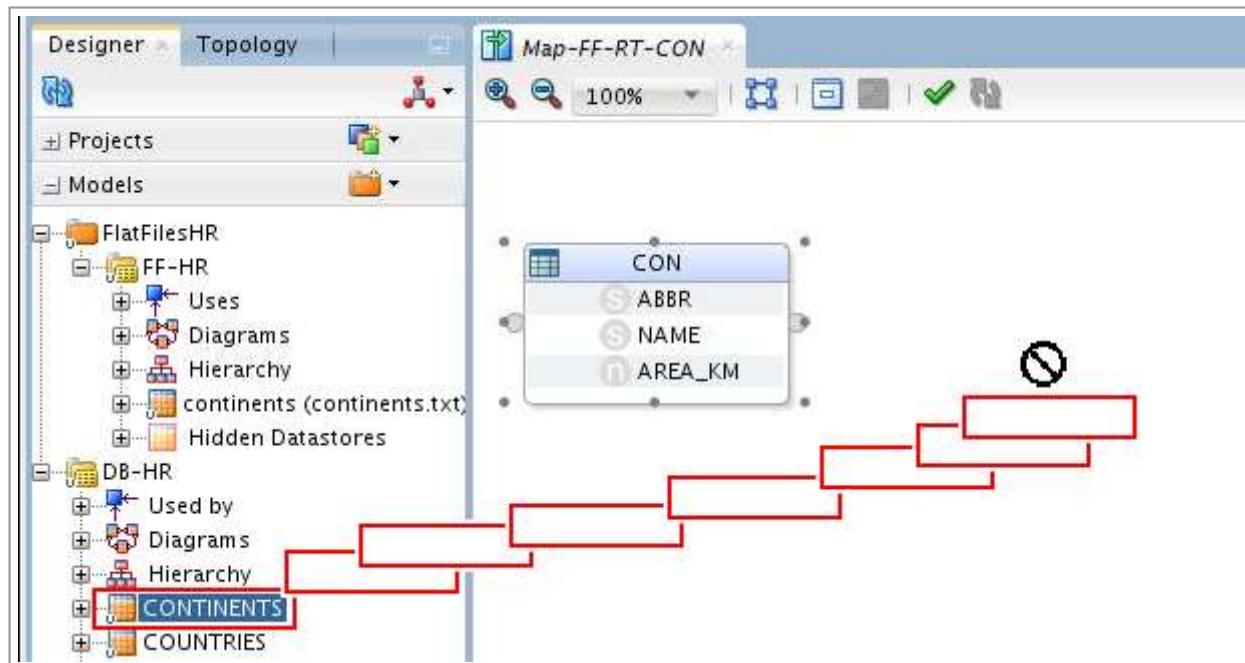


When you drop it, the **CON** table shows its column names.

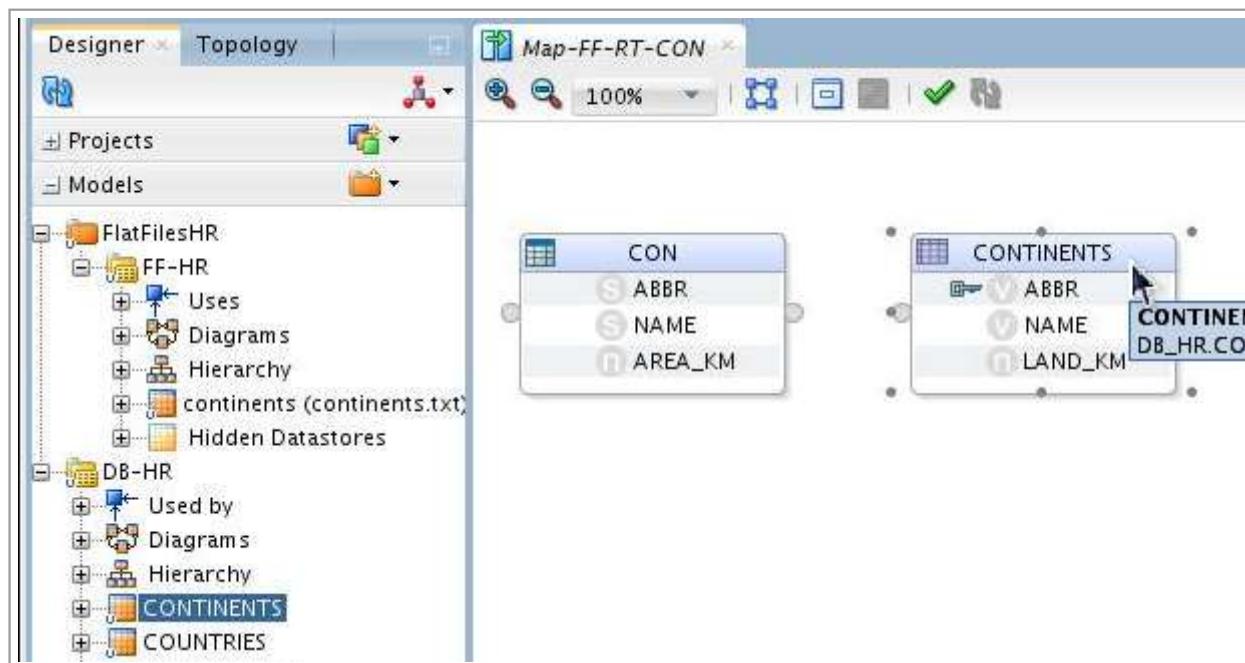


Don't be too concerned about exact placement in the box, there is a Prettify option later.

Similarly, drag **CONTINENTS** from **DB-HR** on the left to *past* the **CON** box on the right. Drop it to the right of the box.

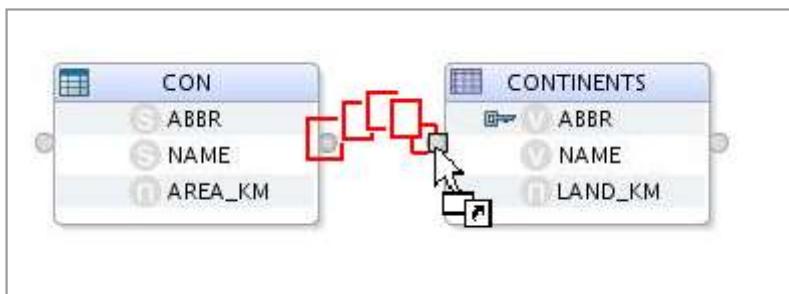


When you drop it, the **CONTINENTS** table shows its column names.

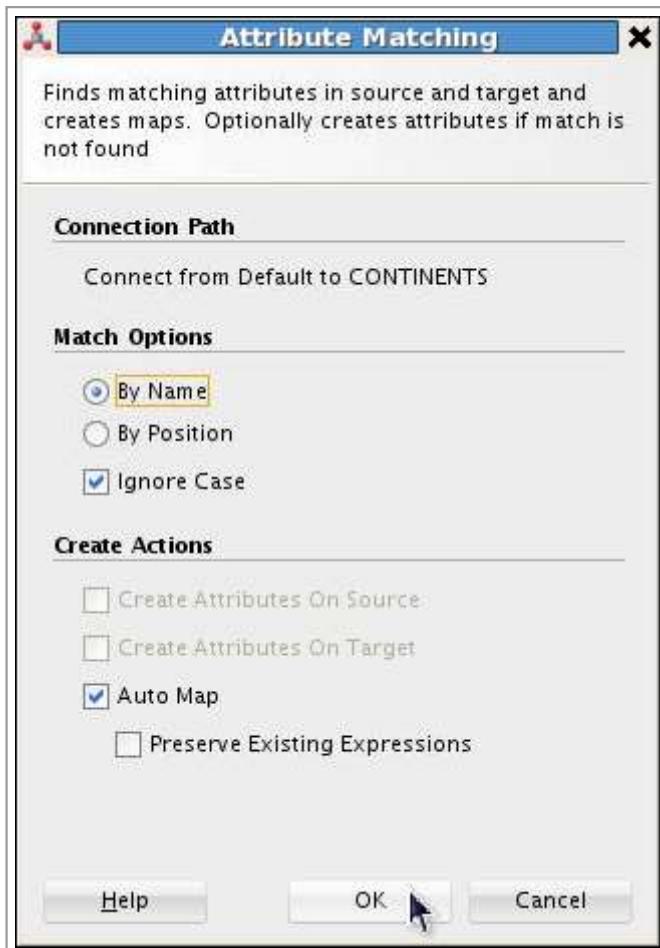


Notice next to the column names are datatype icons: (S) for String, (V) for Varchar, (N) for numeric, and the key next to the primary key (implies unique and not null).

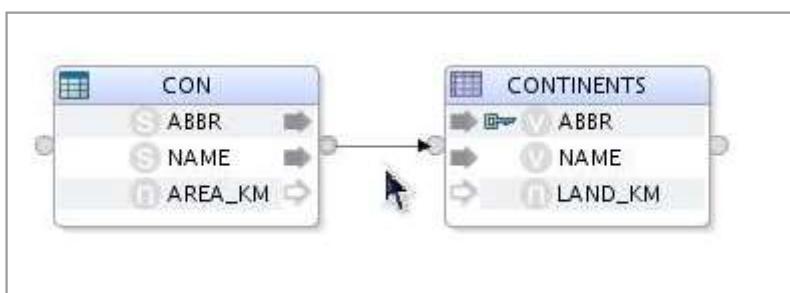
There are connectors (small grey round circles) on the sides of the boxes. Drag the connector from the right side of the **CON** box (flat file model, source) to the left side of the **CONTINENTS** box (relational table model, target).



Notice that some of the column names *could* be assumed to map source-to-target where the names are identical, but maybe that is not what you want. After all, "NAME" is rather generic. The **Attribute Matching** dialog box asks if/how you want to do that.

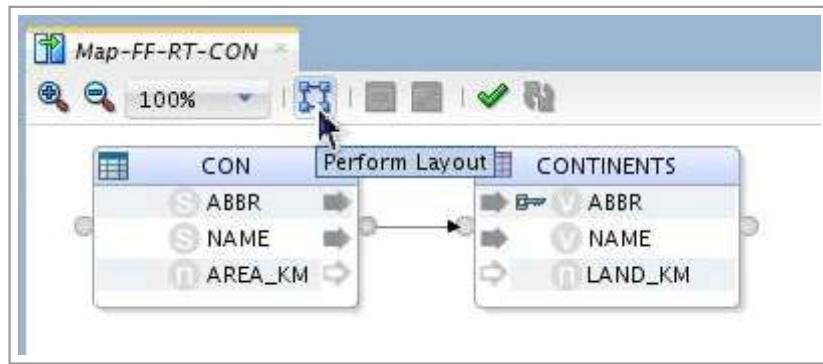


Click **OK** to continue. An arrow shows the direction of the map flow.



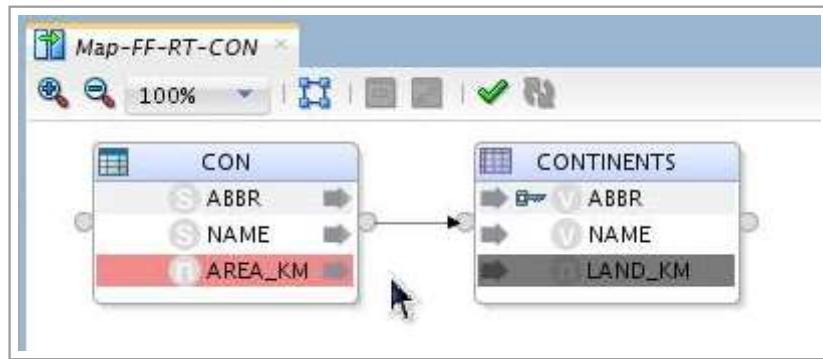
Columns **ABBR** and **NAME** got auto-mapped to each other since they have the exact same column names. You can tell by the solid arrow heads. The fact that some of the arrows next to each column are still hollow light grey indicates that the automatic map did not happen for all fields, especially where there is a name-mismatch. That will be addressed in a later step.

Optionally, to prettify the alignment, change the size of the default box edges and click **Perform Layout** (the four blue squares in a grid icon).



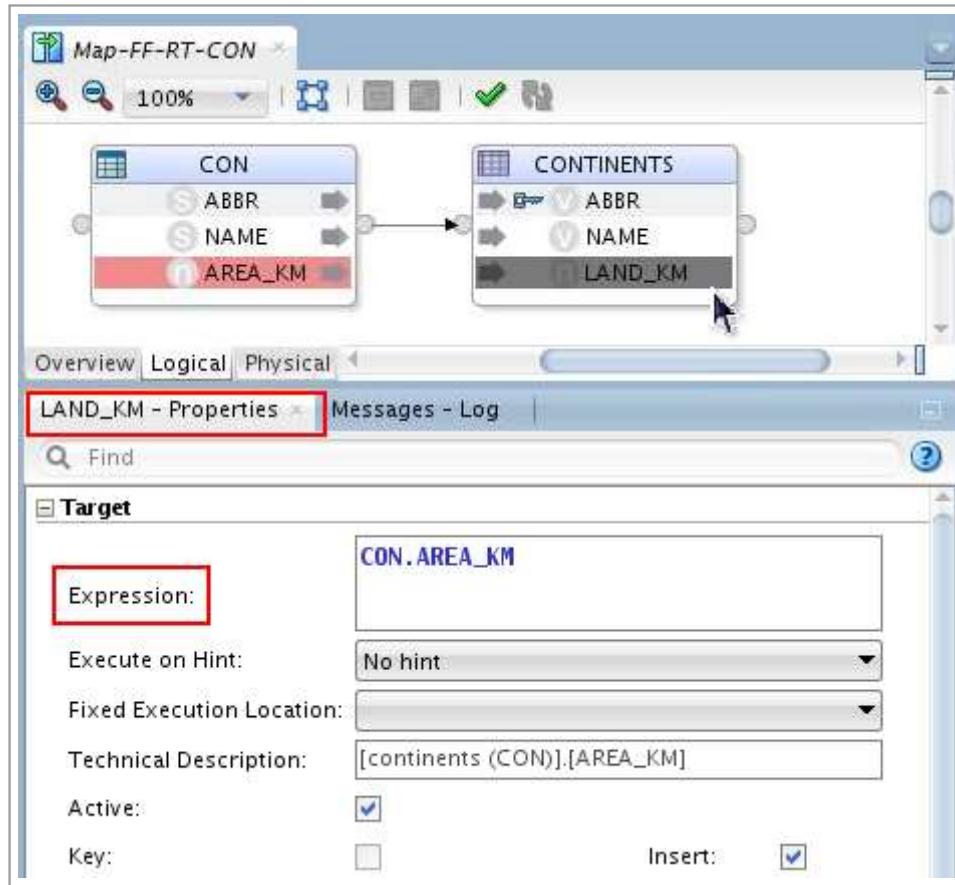
Lovely.

To map the individual columns that did not get auto-mapped, you will drag each from source to target. Drag **AREA\_KM** from **CON** and drop it on **LAND\_KM** in **CONTINENTS**.



When you are done, the arrows on the column line turn solid dark grey. All six column arrows should now be solid dark grey.

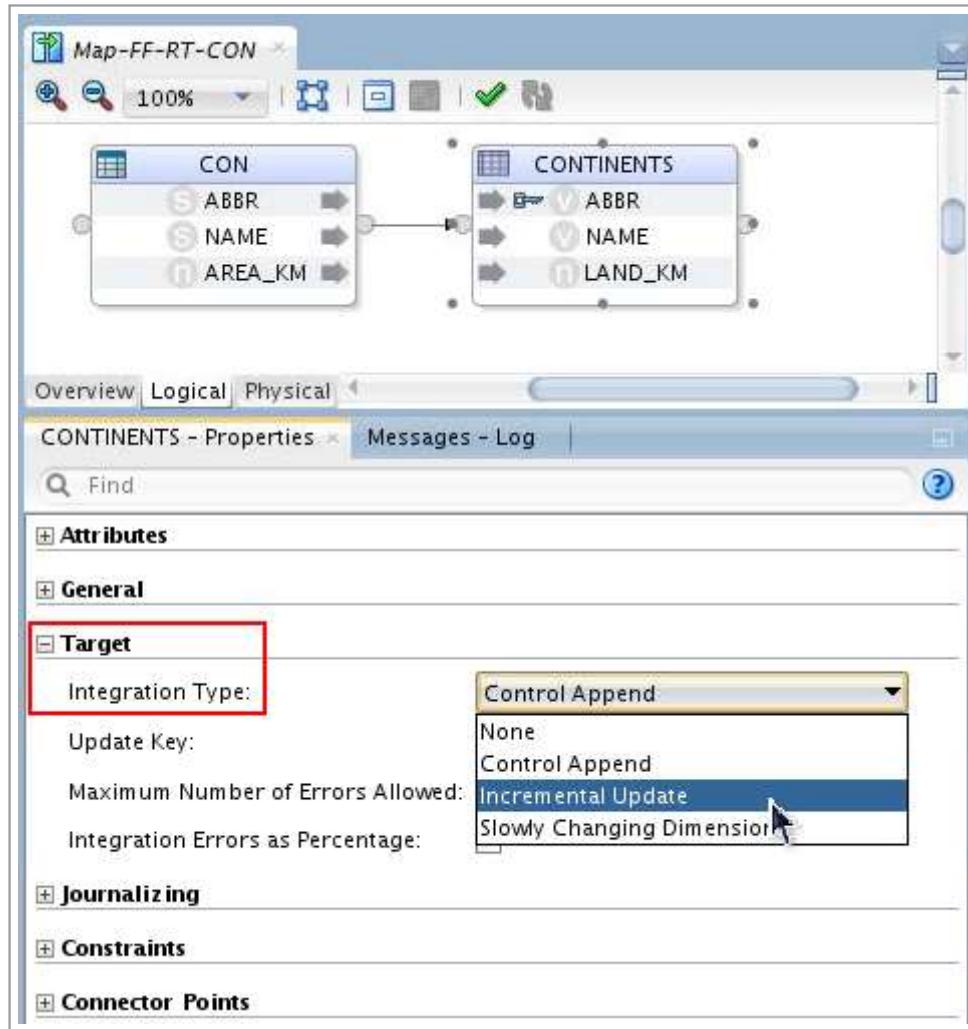
**OPTIONAL:** If you want to see what the GUI tool did for mapping, click a target column, for example, **LAND\_KM**. Look below the drag-n-drop area.



In **LAND\_KM - Properties**, the **Target Expression** indicates a simple map: **CON.AREA\_KM**. You can imagine how this could be a more complex formula expression taking multiple columns and doing arithmetic on them (perhaps converting square kilometers to acres or miles).

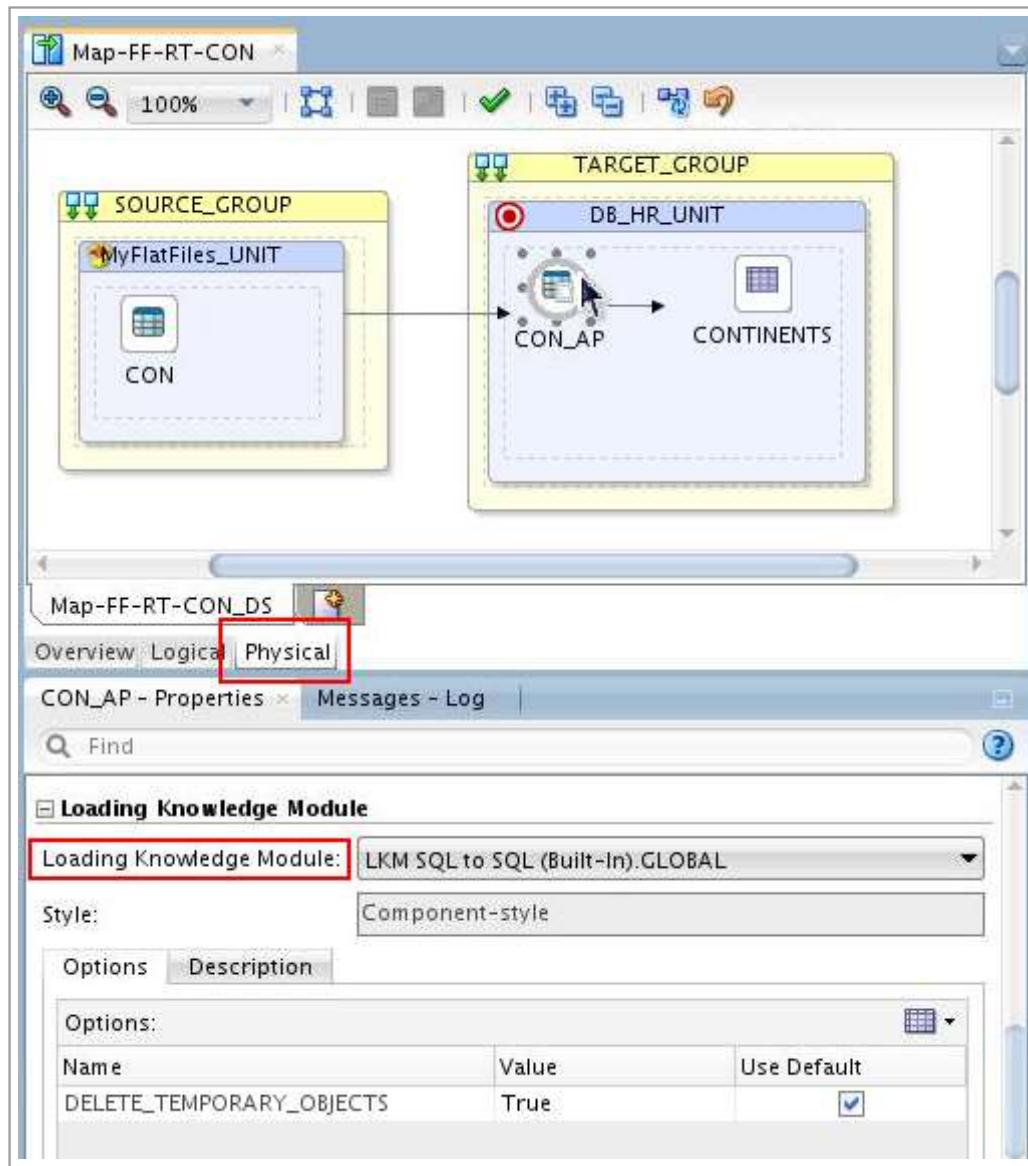
If you did not save your work already by clicking , you have to save it now.

**OPTIONAL:** The defaults assume that the `hr.continents` table is initially empty. You can control that assumption by clicking the **CONTINENTS** table and in the **Target > Integration Type**, you can change it from the default of **Append** to **Update**. This way if you need to run the mapping a second time, it will update existing rows rather than error out with Duplicate Key violations.



Nothing to change or do here, just for looking right now.

OPTIONAL: If you want to see what the physical flow through the components looks like, click the **Physical** tab under the diagram.



If you click the **CON\_AP** (Continents\_Access\_Point), you can see the built-in Loading Knowledge Module (LKM) used in the mapping.

Nothing to change or do here, just for looking right now. You can close the Map-FF-RT-CON tab.

In summary, you mapped the source model to the target model and provided a column-by-column mapping, some of it automatically, and some of it manually.

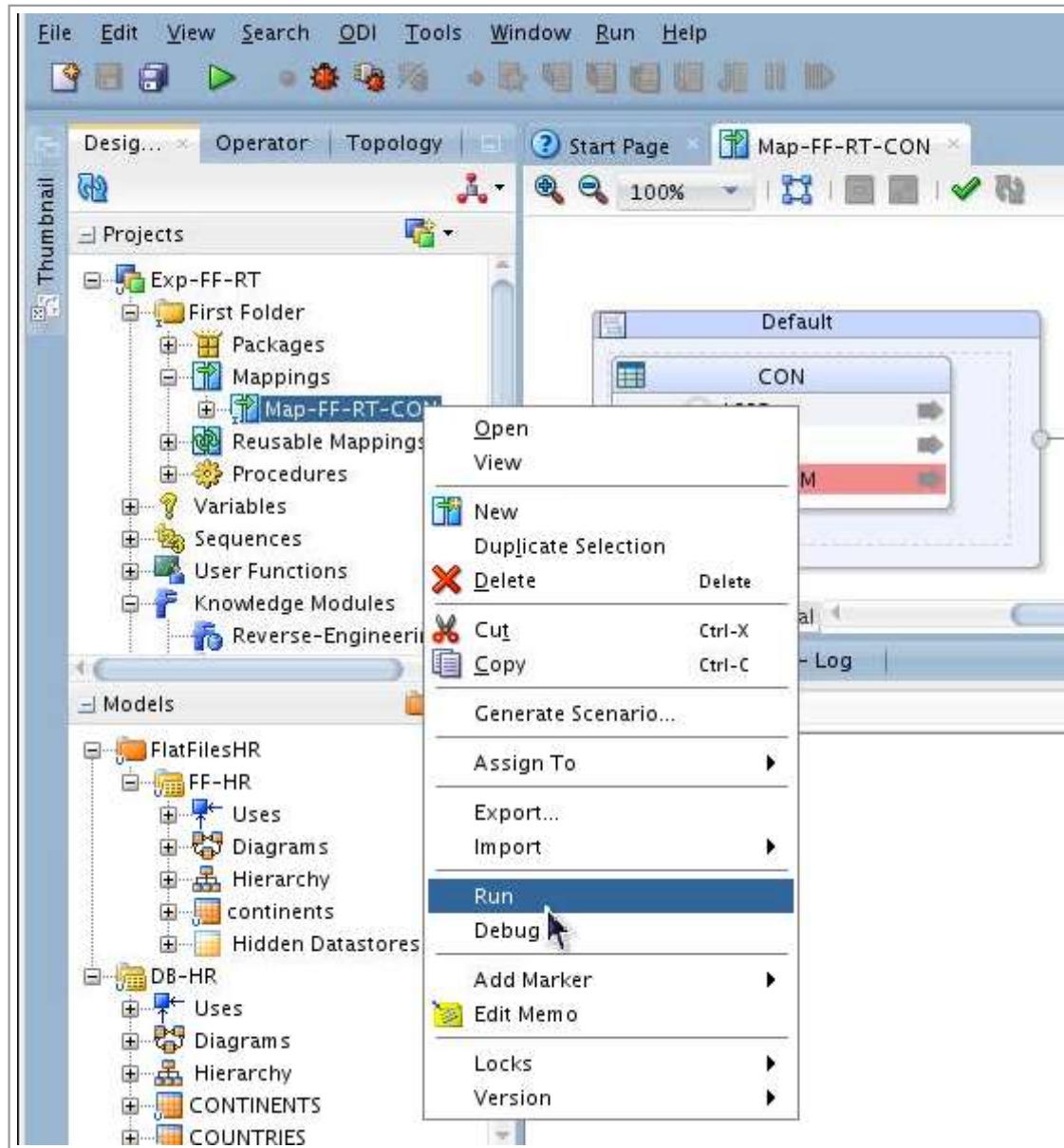
## 4. Executing the Mapping

Mappings can be scheduled to run based on the calendar/clock, or can be run on demand.

To run the mapping now, perform the following steps:

### 4.1 Executing the Mapping Itself

Like all good GUIs, there are several ways to do the same thing. One way to run the mapping is to right-click on it in the **Designer > Projects** section. Expand **Exp-FF-RT > First Folder > Mappings** and right-click on **Map-FF-RT-CON**. Select **Run**.



Alternatively (do not do both), you could have clicked the green arrow , or you could have clicked **Run > Run** on the main menu bar. All three methods do the same thing.



Any one of those three methods displays the **Run** dialog box.



Keep the defaults and click **OK** to continue.



After it says, "Session started," click **OK** to continue.

In summary, you ran a mapping using one of three methods.

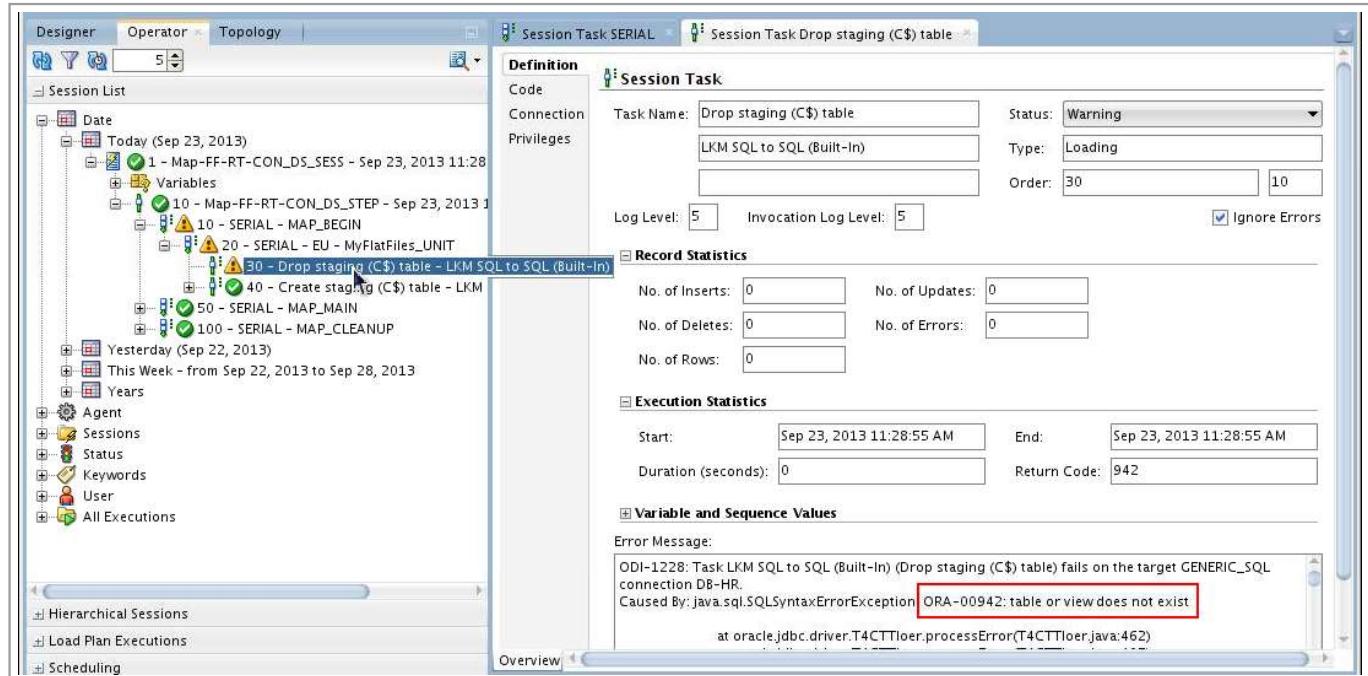
## 4.2 Checking the Operator Tab

To check the status of your mapping run, click the **Operator** tab on the left. There are several ways to see the last run, one way is to expand **Date > Today** and then keep expanding all the items under that. One of them will have today's most recent date as a suffix.



The green circle checkmarks are good. The yellow triangle exclamation points are warnings (perhaps that can be ignored).

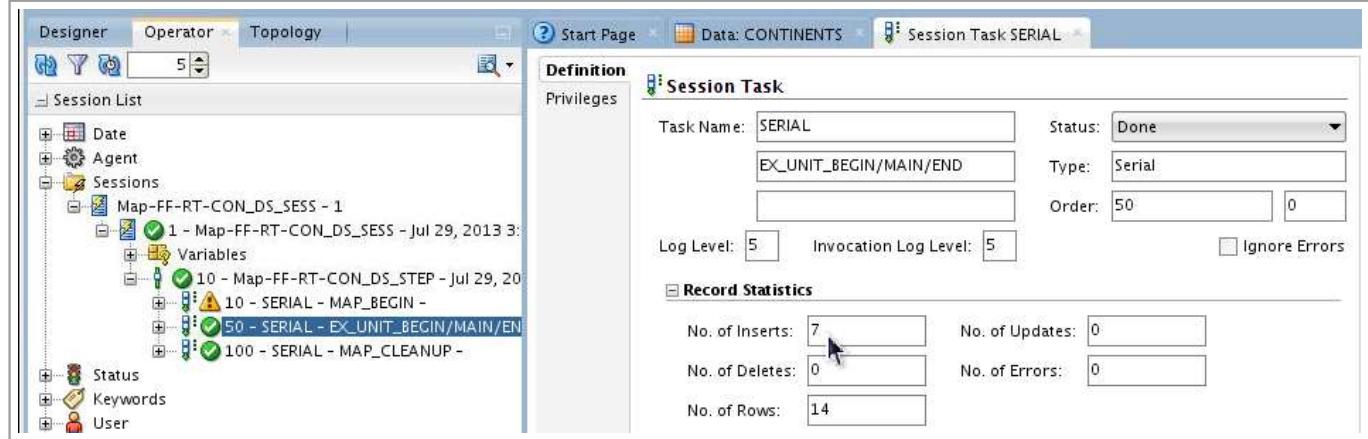
To find out what the warning was, click the line with the yellow triangle, **10 - SERIAL - MAP\_BEGIN**, then keep expanding it ("drill down") until you cannot expand it any more. Double-click the lowest-level error (line 30 - Drop staging).



In the **Error Message**, the warning is that it could not drop the staging table, because it does not exist yet. If you truncated `hr.continents`, ran this mapping a second time, the staging table would be created, then you would not get the error, because the existing staging table can now be dropped. This is a

standard Oracle "feature," so you can ignore it.

To find out statistics on the number of rows imported, click the line **50 - SERIAL EX\_UNIT\_BEGIN/MAIN/END**.

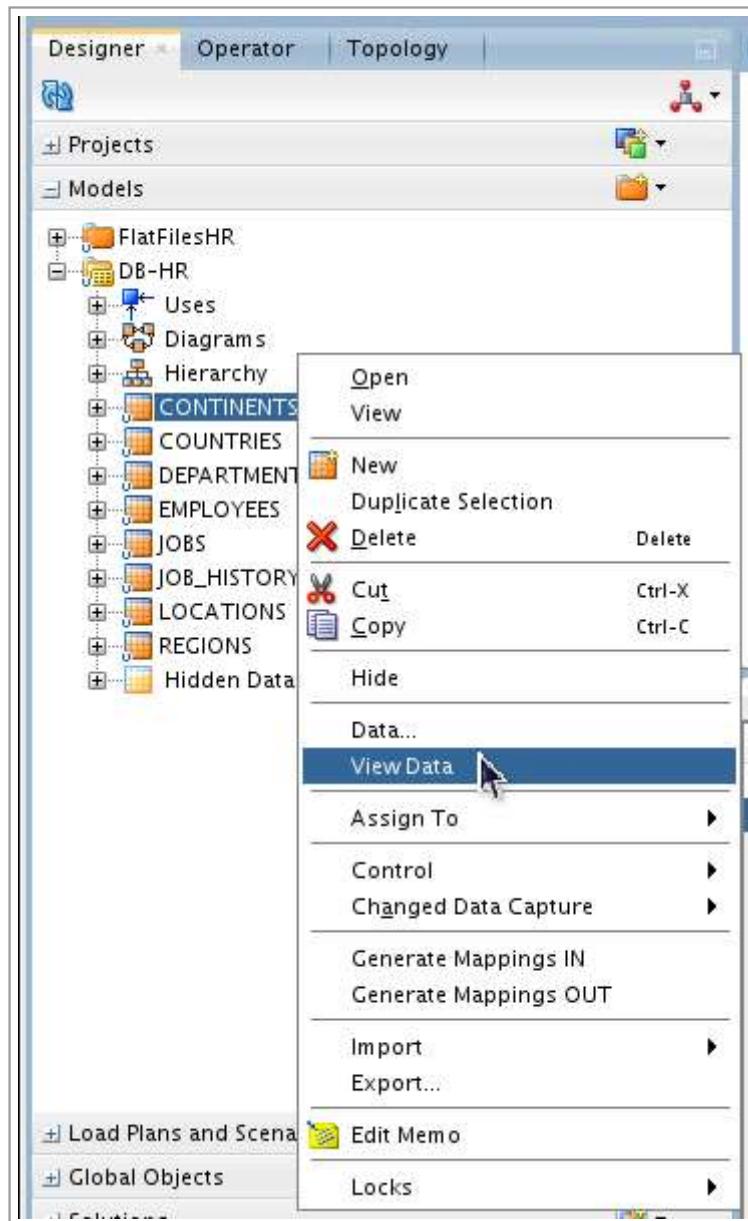


On the **Definition** tab, it shows that **No. of inserts** is 7, which is correct based on the source flat file.

In summary, the mapping ran successfully with one warning that you can ignore.

#### 4.3 Checking the Target Table

Verify whether the data was exported correctly from the source flat file to the target relational table. Expand **Designer > Models > DB-HR** to expose **CONTINENTS**. Right-click on **CONTINENTS** and select **View Data**.



There are other ways to do the same thing.

The newly imported data from the table displays on the right.

	ABBR	NAME	LAND_KM
1	AA	Antartica	13209000
2	NA	North America	24256000
3	SA	South America	17819000
4	EU	Europe	9938000
5	AS	Asia	44579000
6	AU	Australia	7687000
7	AF	Africa	30065000

It is hard to tell from the screen, but there is still an issue with leading (and trailing) blanks stored in the data around the `NAME` column. That can be addressed later with `TRIM`.

Instead of viewing the data from inside of ODI, view it using SQL\*Plus or SQL Developer.

### SQL Prompt

```

SQL> 1
 1* SELECT * FROM hr.continents
SQL> r
 1* SELECT * FROM hr.continents

ABBR NAME          LAND_KM
---- -----
AA      Antarctica    13209000
NA      North America   24256000
SA      South America   17819000
EU      Europe           9938000
AS      Asia             44579000
AU      Australia        7687000
AF      Africa           30065000

7 rows selected.

SQL>

```

The window may have been left open from earlier task 2.1.2. Recall that the first time you did this it was blank.

In summary, you have successfully imported data from a flat file to an Oracle Database relational table and displayed that table both inside and outside of ODI Studio.

## Want to Learn More?

In this tutorial, you should have learned how to:

- Prepare flat files to be the source of an export
- Prepare relational tables to be the target of an export
- Use ODI Studio Designer and Topology to make Physical and Logical Schemas, Models, and Projects
- Map source objects to target objects using the drag-and-drop tool
- Create and execute projects, and evaluate their status using the Operator tool
- View data in datastore objects using ODI tools and external tools such as SQL\*Plus

## Resources

The following are conceptual or procedural Help topics relevant to the topic of this tutorial:

- Online documentation, viewlets, samples, and OLN URLs on OTN:
  - Current version: here (<http://docs.oracle.com/middleware/1221/odi/index.html>)
  - Older versions: here (<http://www.oracle.com/technetwork/middleware/data-integrator/documentation/index.html>)
- The following specific courses:
  - [D82167GC10](http://education.oracle.com/pls/web_prod-plq-dad/db_pages.getpage?page_id=609&get_params=dc:D82167GC10,p_preview:N) ([http://education.oracle.com/pls/web\\_prod-plq-dad/db\\_pages.getpage?page\\_id=609&get\\_params=dc:D82167GC10,p\\_preview:N](http://education.oracle.com/pls/web_prod-plq-dad/db_pages.getpage?page_id=609&get_params=dc:D82167GC10,p_preview:N)) Oracle Data Integrator 12c: Integration and Administration  
(replaces D64974GC20 Oracle Data Integrator 11g: Integration and Administration Ed2)
  - [D82171GC10](http://education.oracle.com/pls/web_prod-plq-dad/db_pages.getpage?page_id=609&get_params=dc:D82171GC10&p_org_id=44003&lang=US) ([http://education.oracle.com/pls/web\\_prod-plq-dad/db\\_pages.getpage?page\\_id=609&get\\_params=dc:D82171GC10&p\\_org\\_id=44003&lang=US](http://education.oracle.com/pls/web_prod-plq-dad/db_pages.getpage?page_id=609&get_params=dc:D82171GC10&p_org_id=44003&lang=US)) Oracle Data Integrator 12c: New Features
- External Web sites for related information:
  - [www.oracle.com/goto/odi](http://www.oracle.com/goto/odi) (<http://www.oracle.com/goto/odi>)
  - OTN Community Forums  
([https://forums.oracle.com/community/developer/english/business\\_intelligence/system\\_management\\_and\\_in](https://forums.oracle.com/community/developer/english/business_intelligence/system_management_and_in))
- To learn more about Oracle Data Integrator 12c, refer to additional OBEs in the Oracle Learning Library (<http://www.oracle.com/goto/oll>), or on the ODI Studio Start Page.

## Credits

- Lead Curriculum Developer: Elio Bonazzi
- Other Contributors: Viktor Tchemodanov, Alex Kotopoulos, Julien Testut