

An Oracle White Paper
June 2012

Creating an Oracle BI Presentation Layer from Imported Oracle OLAP Cubes

Introduction

Oracle Business Intelligence Enterprise Edition version 11.1.1.5 and later has the ability to import metadata from Oracle OLAP 11.2 cubes into the Physical layer of the Oracle BI repository. This Physical layer provides the foundation for creating a Business Model and one or more Presentation layers in the repository.

Depending on the contents of the Oracle OLAP dimensions and cubes and the requirements of the application, there are many different ways that a Business Model and Presentation Layer might be configured. This document provides a guide for creating a typical Business Model and Presentation layer that will satisfy the requirements of most business users and will minimize the amount of work that must be done by the Oracle BI application developer.

It is important to note that there are two methods of automatically creating subject area in an Oracle BI repository.

One method, which may be used with Oracle BI versions 10.2 and 11.1, uses the OBIEE plug-in for Analytic Workspace Manager. This method pushes metadata from OLAP dimensions and cubes using the Oracle BI UDML language, producing the Physical, Business Model and Presentation layers. It creates only a tabular representation of the cube and queries the hierarchy and cube views using SQL.

The other method is the Import Metadata feature introduced in Oracle BI Administration Tool version 11.1.1.5. This method produces both a tabular and “OLAP” representation with hierarchical columns and queries the cube using SQL and the OLAP_TABLE table function.

While it is not the goal of this document to describe the differences between each of the resulting subject areas, it is important to understand that this document only applies to the subject area created using the Import Metadata feature of the Oracle BI Administration Tool.

Getting Started

The process of creating the Presentation layer begins with the design of the OLAP dimensions and cubes. By understanding the requirements of the applications that will query the cube you can exclude certain features of dimensions, some of which will simplify the process of creating the Oracle BI Business Model and Presentation layers. The following table lists features in the OLAP dimension that you can disable when using the OBIEE Import Metadata feature.

TABLE 1. SIMPLIFY THE OLAP DIMENSION PRIOR TO IMPORTING INTO ORACLE BI

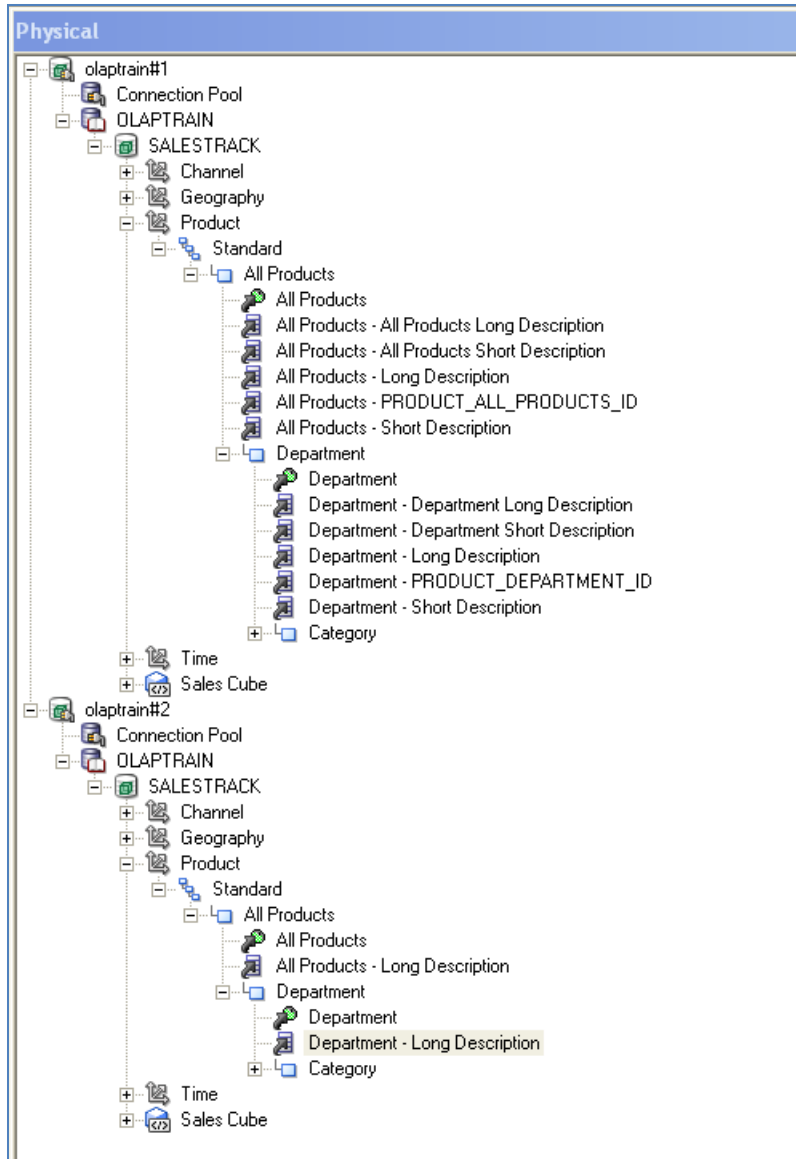
IF ...	THEN DO THIS ...
OBIEE 11.1.1.5 (and later) and Microsoft Excel are the only tools that will be used to query the cube and you will use the Import Metadata feature in the Oracle BI Administrator.	1. Turn off “Create level attribute columns in views” for all attributes in every OLAP dimension. This will eliminate the “spread attributes ¹ ” columns in the hierarchy views of the OLAP dimension. Note: “Create level attribute columns in views” is turned on by default. 2) Turn off: <ul style="list-style-type: none"> a) “Enable Query Rewrite” on the cube. b) “Enable Materialized View Refresh of the Cube” on the each cube. c) “Enabled Materialized View Refresh of the Dimension” on each dimension. Note: These materialized view features are turned off by default.
Business users do not require separate Long and Short Description attributes.	Delete either the Short or Long Description attribute. (Microsoft Excel uses the Long Description attribute, so if you are using Excel delete the Short Description attribute.)

Notes:

- Turning off “Create level attribute columns in views” will disable the ability to create a subject area using the OBIEE Plug-in for Analytic Workspace Manager. If you wish to try both methods leave “Create level attribute columns in views” enabled and follow instructions later in this document for deleting unneeded attribute columns in the Business Model layer.
- These recommendations will create dimensions that are compatible with the use of Microsoft Excel Pivot tables via the MDX Provider for Oracle OLAP. (Excel does not require level attribute columns in views. It uses the Long Description attribute to represent dimension members to Excel users; it does not require the Short Description attribute.)

The following illustration includes two Physical Schemas, both imported from the OLAPTRAIN sample analytic workspace.

¹ “Spread attributes” are used to represent attributes as different levels of aggregation as separate columns in views over the OLAP hierarchy views, making these views resemble a typical star style dimension table or view. These columns are



Physical schema olaptrain#2 has been simplified by eliminating features in the OLAP dimensions that are not needed with OLAP cubes that have been imported into Oracle BI.

Physical Schema olaptrain#1 was imported from the analytic workspace with “Create level attribute columns in views” enabled for both the Long Description and Short Description attributes and materialized view refresh of the dimension is enabled. Note that this Physical Schema contains many more columns as compared to olaptrain#2.

Physical Schema olaptrain#2 was imported from the analytic workspace with “Create level attribute columns in views” disabled, the dimensions included only the Long Description attribute and materialized view refresh of the dimension is disabled. For the typical implementation, this is a better starting point because columns that are not usually needed in OBIEE have been eliminated by the design of the OLAP dimensions.

Importing Metadata into Oracle BI

The Physical Schema is created in the Oracle BI repository using the Import Metadata feature in the Oracle BI Administration tool. Either open an existing repository and choose the File/Import Metadata command or create a new repository. In either case, choose the connection type Oracle OLAP.

Create New Repository - Select Data Source

1 Repository Information
2 **Select Data Source**
3 Select Metadata Types
4 Select Metadata Objects
5 Map to Logical Model
6 Publish to Warehouse

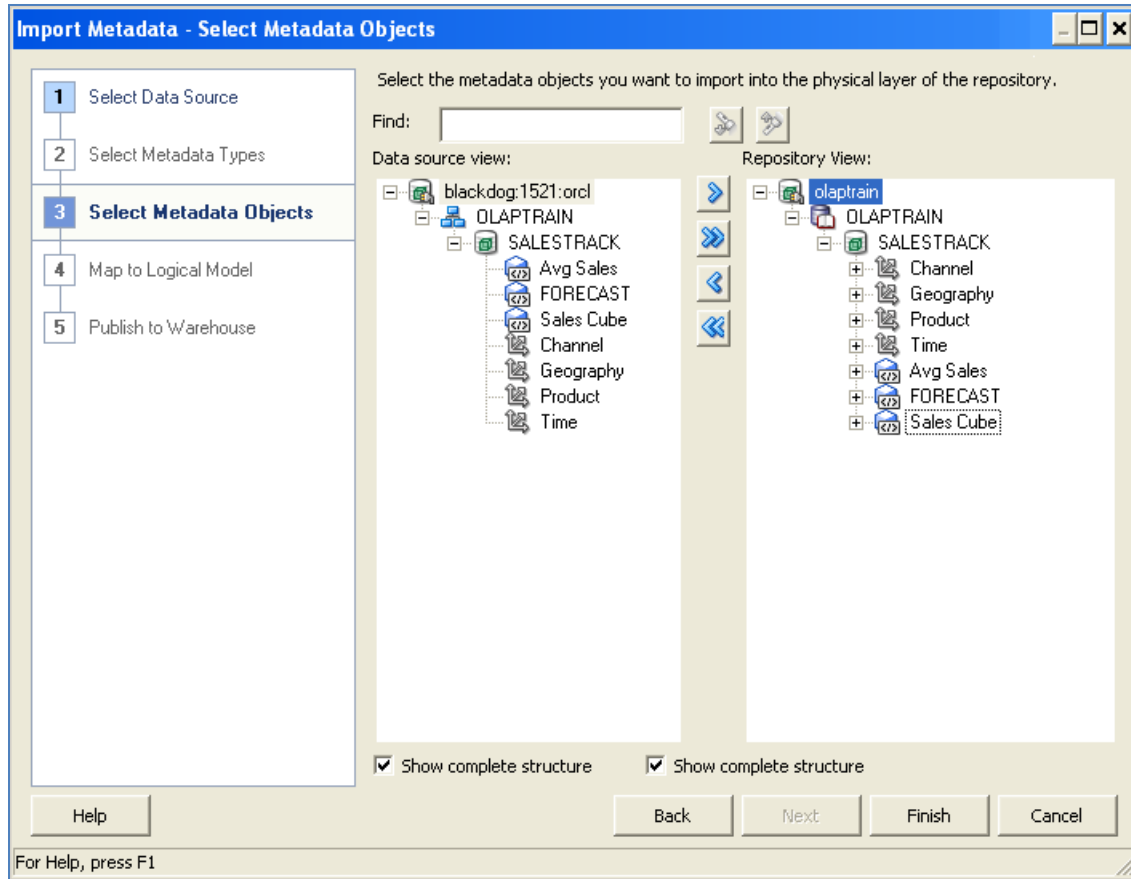
Import Type: Local Machine
Connection Type: Oracle OLAP
Data Source Name: blackdog:1521:orcl
User Name: olaptrain
Password: ●●●●●●●●
Target Database: New: olaptrain Existing: Browse...

Help Back Next Finish Cancel

For Help, press F1

Making the database connection when importing from Oracle OLAP.

Oracle BI will discover all dimensions and cubes in the analytic workspace. It is usually best to import all objects from the analytic workspace rather than to select individual objects. If the analytic workspace includes more than one cube, import all objects and remove unneeded objects from the Physical or Business Model layers. Be sure not to remove any dimensions that are used by cubes that will be used in a Business Model or Presentation layer.



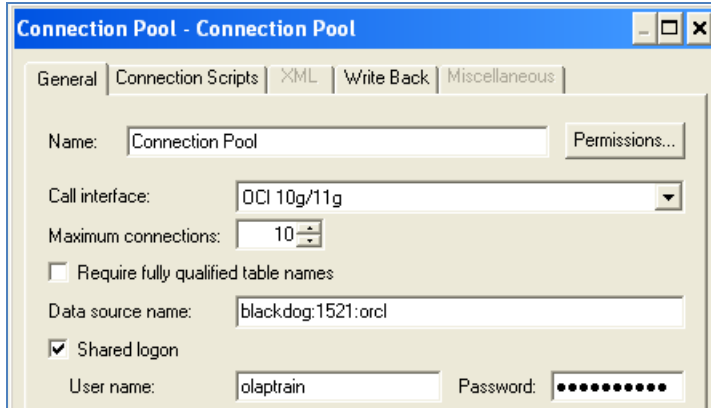
Importing cubes and dimensions in the OLAPTRAIN schema.

Updating the Physical Schema

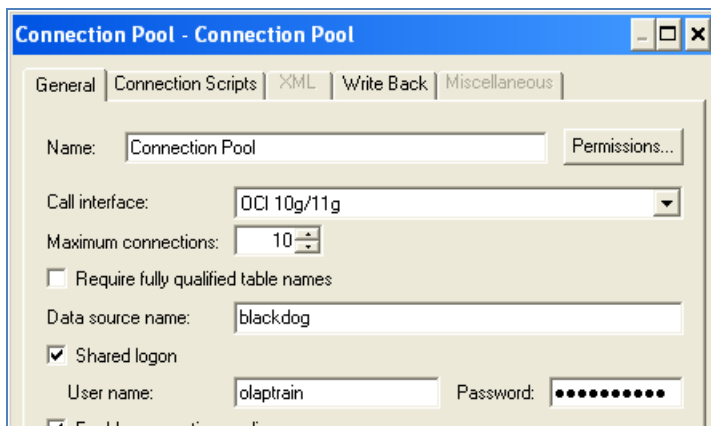
This document will continue based on the assumption that the OLAP dimensions resemble the Physical Schema olap#1 as in the previous example. If your dimensions more resemble olap#2 you will learn in more detail why this is a better starting point, but more importantly you will have less work to do!

Updating the Connection Pool

When specifying the connection in the Metadata Import, you might have provided it in the form *host:port:sid* as shown in the example below. Change this connection to a database alias in your TNSNAMES.ORA file. This will allow the Oracle BI server to connect to the database.



Data source name as created by the Import Metadata tool.



Data source name has been changed to use a database alias in the TNSNAMES.ORA file.

Eliminating Unnecessary Columns

If certain features of the OLAP dimensions are enabled there will be columns imported into the Oracle BI Physical layer that are not required for the typical application. They include columns representing:

- “Spread attributes”, the attributes created when the “Create level attribute columns in views” option is enabled.
- The Short Description attribute if this attribute is not required by business users. (Long and Short Description attributes are often mapped to the same values. If this is the case, keeping both in Oracle BI would be redundant.)
- The original keys from the dimension tables if the “Materialized view refresh of the dimension” feature is enabled.

All of these columns may be removed from the Physical Schema (or the Business Model) for all dimensions and all levels. They can easily be identified by the column names described in the following table.

Do not delete columns representing the dimension key and the Long Description attribute.²

² If both the Long Description and Short Description attributes contain the same value, the Long Description attribute is kept because that attribute is used by Microsoft Excel Pivot tables.

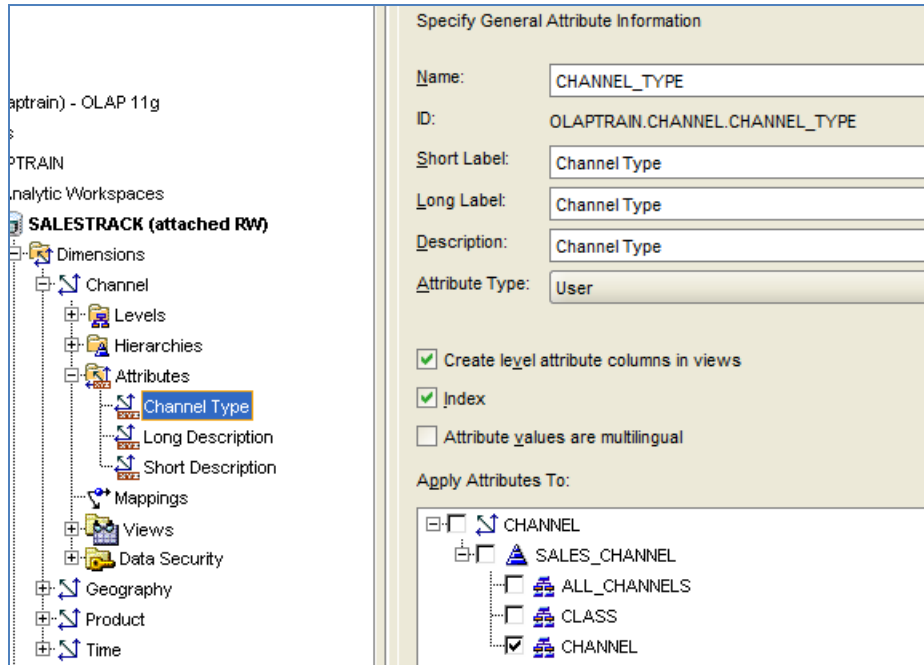
TABLE 2. COLUMNS THAT MAY BE DELETED FROM THE PHYSICAL LAYER

COLUMN TYPE	NAME FORMAT	EXAMPLES
Spread Attribute – Long and Short Description attributes	<i>Level Name – Level Name Attribute</i>	All Channels – All Channels Long Description
	<i>Name</i>	All Channels – All Channels Short Description
		Class – Class Long Description
		Class – Class Short Description
		Channel – Channel Long Description
Spread Attribute – End Date and Time Span attributes	<i>Level Name – Level Name Attribute</i>	Calendar Year – Calendar Year Time Span
	<i>Name</i>	Calendar Quarter – Calendar Quarter Time Span
		Month – Month Time Span
		Calendar Year – Calendar Year End Date
		Calendar Quarter – Calendar Quarter End Date
Spread Attributes – User attributes	<i>Level Name – Level Name</i>	Channel - Channel
	See below for more information.	
Short Description Attributes	<i>Level Name – Short Description</i>	All Channels – Short Description
		Class – Short Description
		Channel – Short Description
Original keys from table	<i>Level Name – DIMENSION</i>	All Channels – CHANNEL_ALL_CHANNELS_ID
	<i>NAME_TABLE COLUMN NAME</i>	Class – CHANNEL_CLASS_ID
		Channel – CHANNEL_CHANNEL_ID

User Attributes

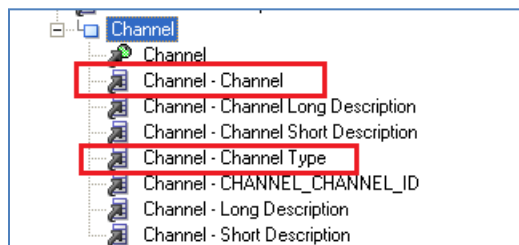
The column names created from User attributes are different from the Long and Short Description attributes. This requires careful examination of physical columns representing these attributes.

Continuing with the example of the Channel dimension in OLAPTRAIN, note that there is a Channel Type attribute that is applied only at the Channel level. Also note that the “Create level attribute columns in view” option is enabled as shown in the following illustration.



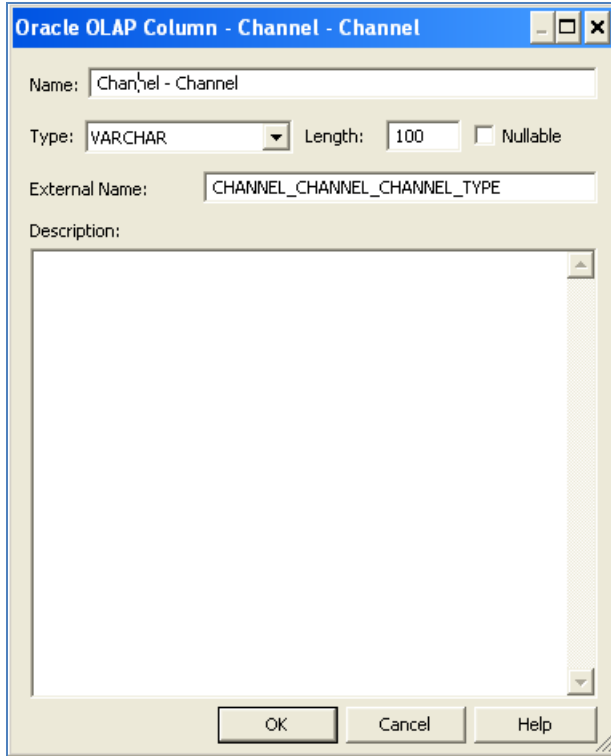
The Channel Type attribute as defined in Analytic Workspace Manager. Note that the attribute type is User and that this attribute is applied to only the Channel level.

In this case the Channel Type attribute will be represented by two columns, one for the attribute and another for the spread attribute. The naming of the spread attribute column, however, is different from the columns representing the Long and Short Description attributes. Instead of being named as *Level Name – Level Name Attribute Name* it will simply be *Level Name – Level Name* as shown in the following illustration.



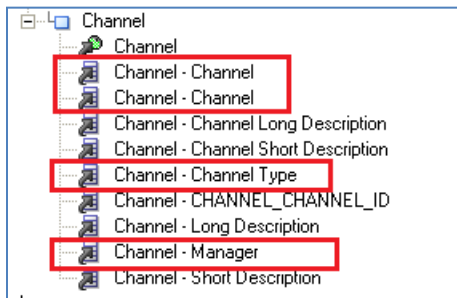
Columns representing the Channel Type attribute.

An examination of the properties of the Channel – Channel column reveals that the External Name identifies the role of this column as being the Channel Type attribute.



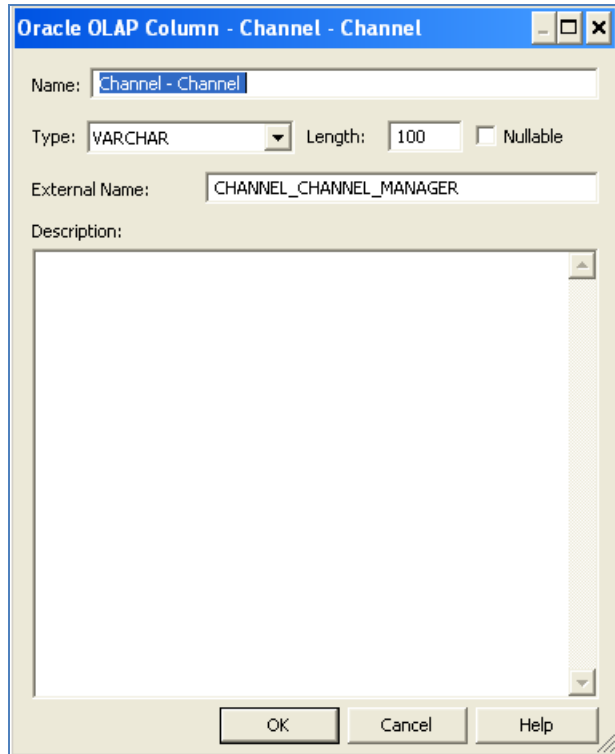
Properties of the Channel – Channel column.

If there is more than one User type attribute, column names might be repeated. In this example, a Manager attribute was added to the Channel dimension. The Channel – Channel column is repeated in the Physical layer.



Levels for the Channel level after adding the Manager attribute. Not the repetition of the Channel – Channel column.

An examination of the column properties will reveal the role of each column.



Column properties for the second Channel – Channel column reveals its role as the spread attribute column of the Manager attribute.

Since the spread attributes can be deleted from the Physical Schema, both of the Channel – Channel columns can be deleted.

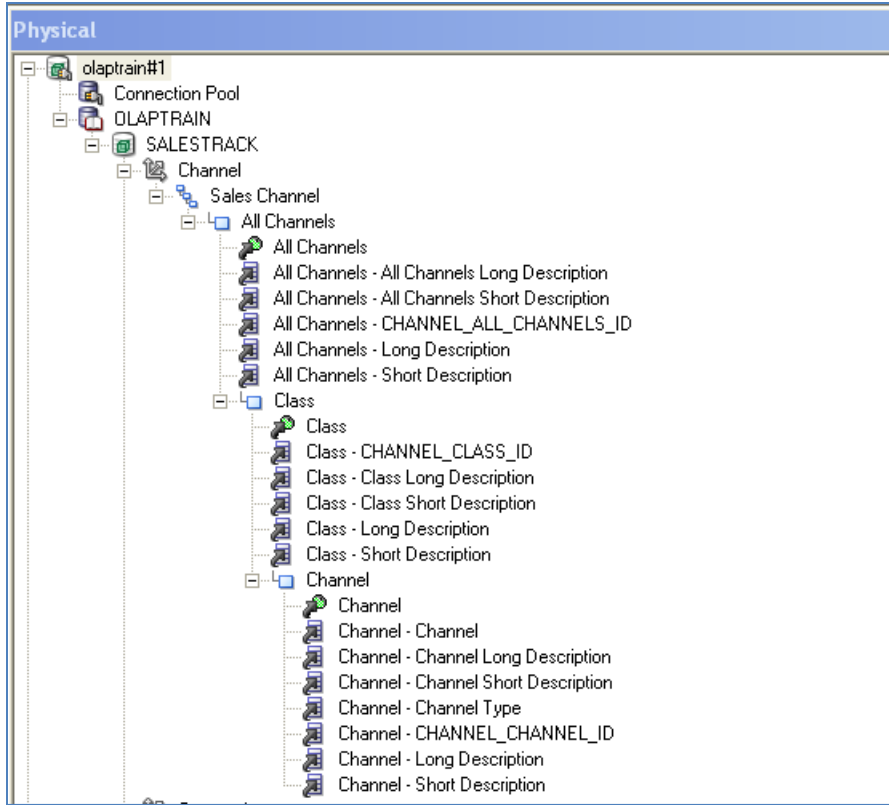
End Date and Time Span Attributes

The End Date and Time Span attributes follows the same naming conventions in the metadata import as the Long and Short Description attributes, making it easy to identify the columns representing the spread attributes. The columns representing the spread attributes for these attributes can be removed from the Physical layer.

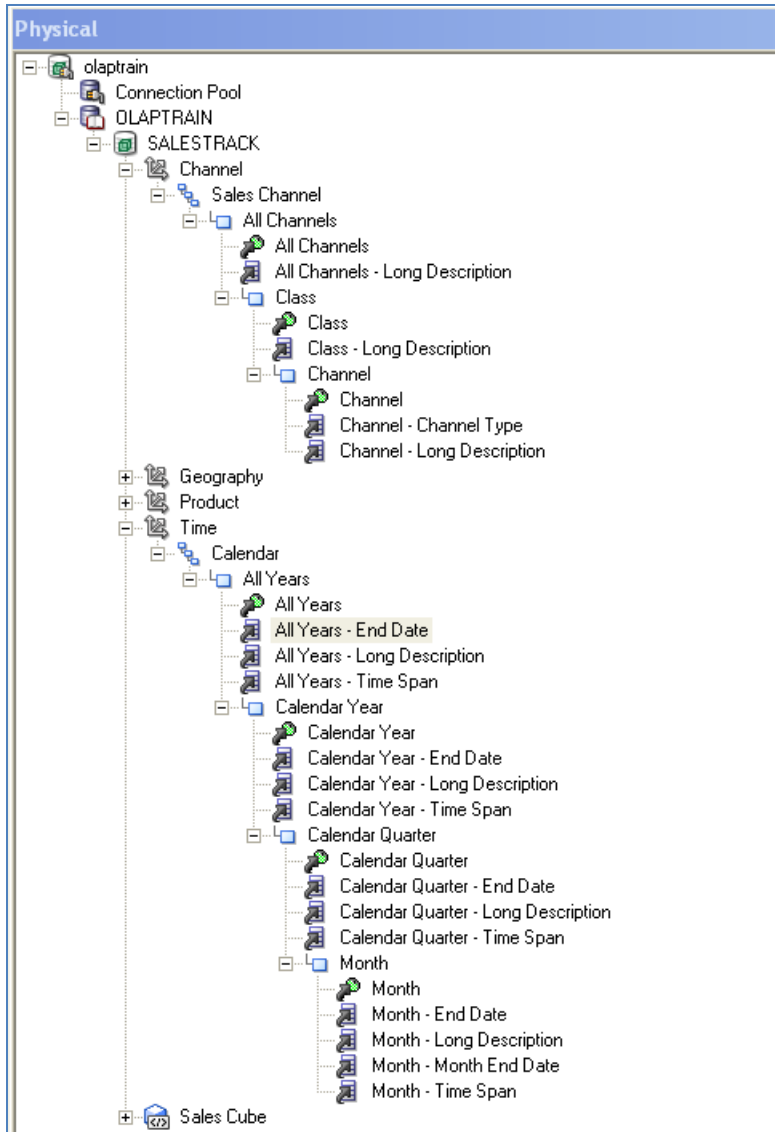
The End Date attribute is used for sorting members of the Time dimension and should not be deleted. Whether you keep the Time Span attribute is up to you. It's not used in the Business Model, but you might find it useful at some point in the future for calculation purposes.

Physical Schema Before and After Cleanup

The following illustrations show Physical Schema as it was before and after removing unnecessary columns.



The Channel dimension in the Physical layer as created by the metadata import.



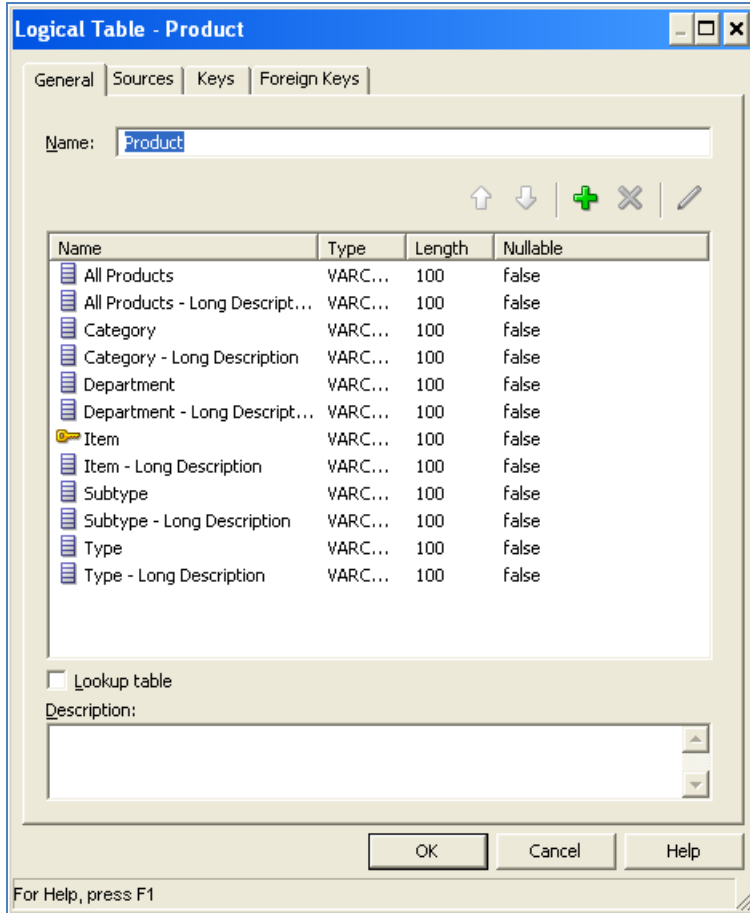
Channel and Time dimensions in the Physical layer after removing unneeded columns.

Creating and Updating the Business Model

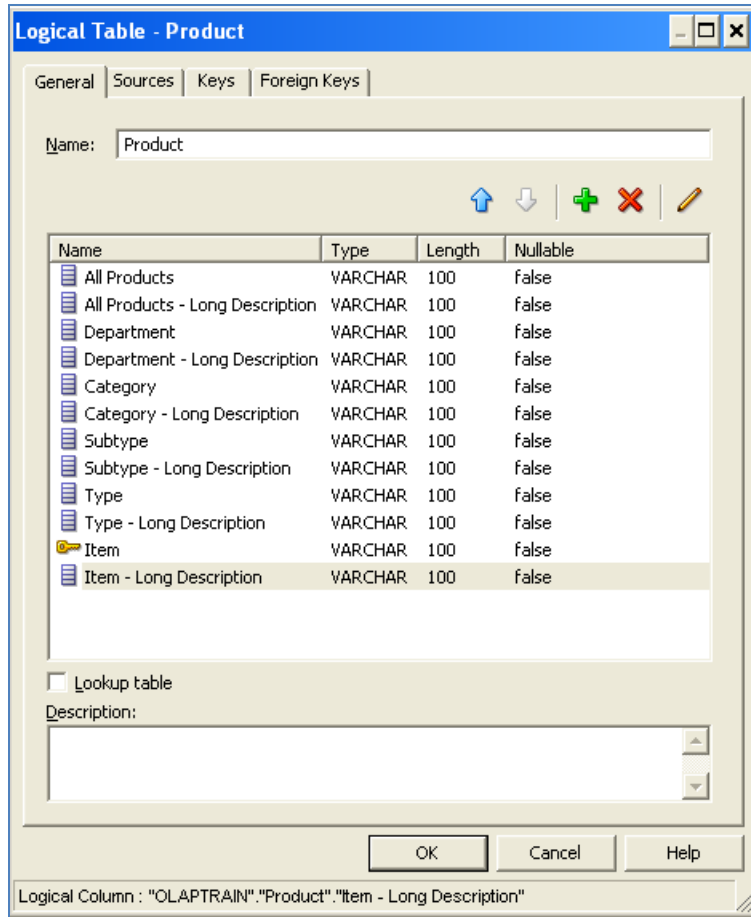
To create the Business Model, select the schema in the Physical layer and drag it to the Business Model and Mapping panel.

Ordering Columns

After the metadata import columns will be ordered alphabetically. Business users would mostly likely prefer columns to be ordered hierarchically. This can be done in the Presentation layer or the Business Model. To reduce the amount of work needed in the case where more than one Presentation layer is created from the Business Model, this paper will order columns in the Business Model.



Columns in the Product dimension before reordering. Note that columns are sorted alphabetically.



Columns in the Product dimension reordered hierarchically.

Creating and Updating the Presentation Layer

To create the Presentation Layer, select the Business Model and drag it to the Presentation layer.

For the most part, the updates made to the Presentation Layer will depend on what columns the business users are interested in using and making sure that the correct display columns are used for hierarchical columns. Business users might also prefer different column names in the Presentation Layer.

Dimension Member Columns

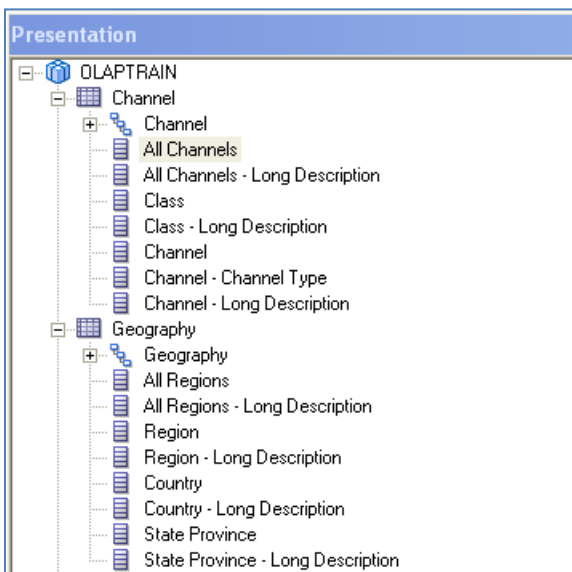
Business users typically view data using the Long or Short Descriptions rather than dimension members (keys). If this is the case, the columns representing the dimension members can be removed from the presentation layer. These columns are easily identified by name since they take the name of the level they represent.

The following report from Oracle BI Answers illustrates the difference between the columns representing dimension members and the Long Description attribute in the OLAPTRAIN sample.

Calendar Year - Long Description	Region	Region - Long Description	Sales
CY2007	-10	Oceania	88,552
	-11	North America	30,359,912
	-12	South America	13,213,791
	-7	Asia	53,199,450
	-8	Africa	5,239,815
	-9	Europe	18,234,240
CY2008	-10	Oceania	111,101
	-11	North America	34,529,324
	-12	South America	15,061,507
	-7	Asia	61,931,241
	-8	Africa	6,347,085
	-9	Europe	20,979,901
CY2009	-10	Oceania	96,192
	-11	North America	40,127,447
	-12	South America	17,428,290
	-7	Asia	69,856,269
	-8	Africa	6,938,766
	-9	Europe	23,772,136

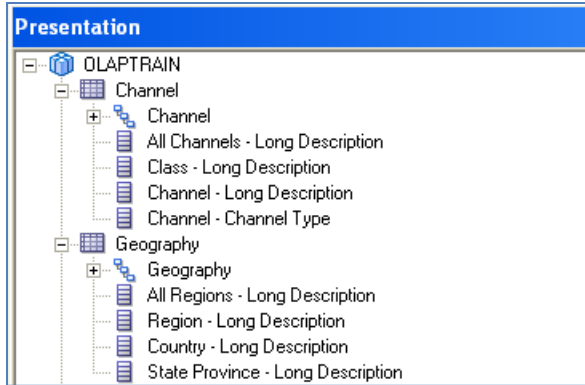
Reporting in Oracle BI Answers with both the member and description columns for Region.

In the following Presentation layer columns for both the dimension members and the Long Description attribute are included.



Channel and Geography dimensions with dimension member and description columns.

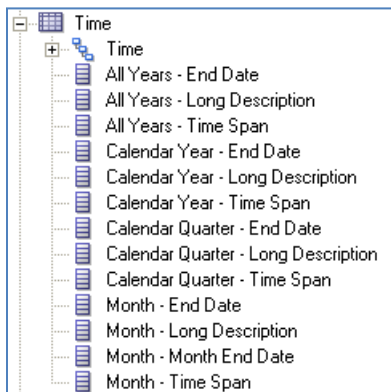
In the next example, the columns representing the dimension members have been removed.



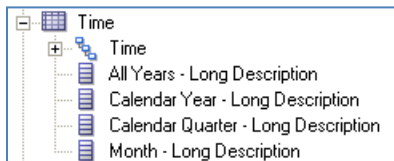
Channel and Geography dimensions with dimension member columns removed.

End Date and Time Span Columns

It is unlikely the business uses will want to include the End Date and Time Span attributes in reports. If this is the case, these columns can be removed from the presentation layer.



Time dimension with End Date and Time Span Columns.

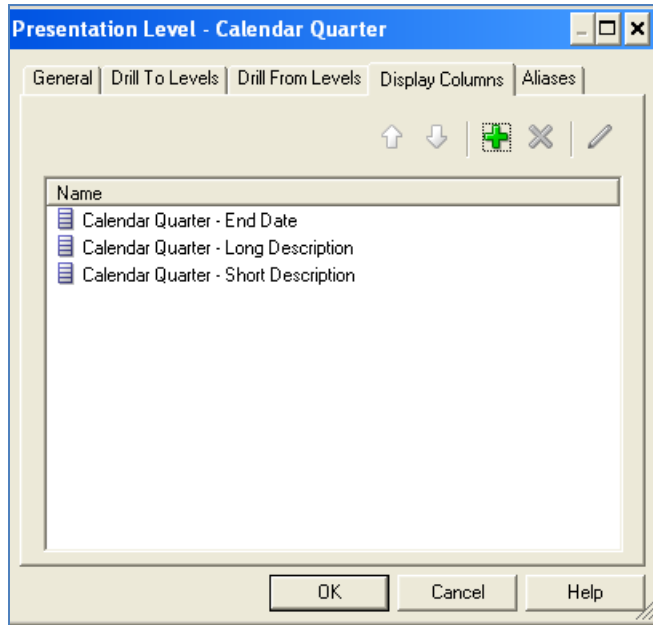


Time dimension with End Date and Time Span columns removed.

Display Columns

Check the display columns for hierarchical columns in the presentation layer. The display column should typically be either the Long or Short Description column. Depending on your model, other columns might be included as display columns.

The following example shows the display columns for the Calendar Quarter level as they are set by the metadata import. (In this example, the End Date and Short Description attributes have not yet been removed from the presentation layer.)



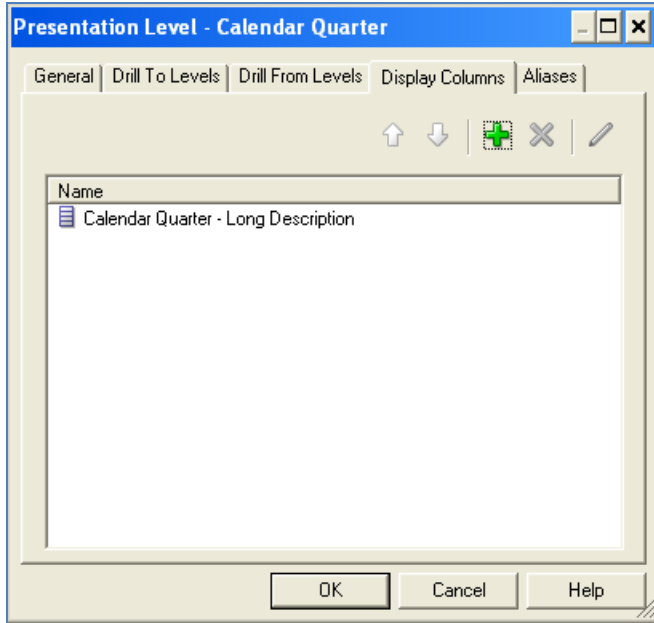
Display columns from the metadata import, before cleanup.

If the above display columns are used a report in Oracle BI Answers will display the values of the End Date attribute instead of descriptions for time dimension members as shown in the following example.

Time	Product	Sales
[-] Total	[+] Total	417,515,017
[-] 12/31/2010 12:00:00 AM	[+] Total	417,515,017
[-] 12/31/2007 12:00:00 AM	[+] Total	120,335,758
[-] 3/31/2007 12:00:00 AM	[+] Total	31,216,351
[-] 6/30/2007 12:00:00 AM	[+] Total	26,084,945
[-] 9/30/2007 12:00:00 AM	[+] Total	29,030,884
[-] 12/31/2007 12:00:00 AM	[+] Total	34,003,578
[-] 12/31/2008 12:00:00 AM	[+] Total	138,960,159
[-] 12/31/2009 12:00:00 AM	[+] Total	158,219,099

Report in Oracle BI Answers displaying End Date attributes as description for time members.

Display columns should only include the column for the Long or Short Description as shown in the example below.



Display column for Calendar Quarter level, after cleanup.

This case, the report displays as expected as seen below.

		Sales
Time	Product	
<input type="checkbox"/> Total	<input type="checkbox"/> Total	417,515,017
<input type="checkbox"/> All Years	<input type="checkbox"/> Total	417,515,017
<input type="checkbox"/> CY2007	<input type="checkbox"/> Total	120,335,758
<input type="checkbox"/> Q1-CY2007	<input type="checkbox"/> Total	31,216,351
<input type="checkbox"/> Q2-CY2007	<input type="checkbox"/> Total	26,084,945
<input type="checkbox"/> Q3-CY2007	<input type="checkbox"/> Total	29,030,884
<input type="checkbox"/> Q4-CY2007	<input type="checkbox"/> Total	34,003,578
<input type="checkbox"/> CY2008	<input type="checkbox"/> Total	138,960,159
<input type="checkbox"/> CY2009	<input type="checkbox"/> Total	158,219,099

Grand Total Levels

The OBIEE Import Metadata feature automatically adds a Grand Total level to every hierarchy. In the following example, the Total members are Grand Total levels in the OBIEE hierarchy and duplicate the All Years and All Products members in the OLAP dimension.

		Sales
Time	Product	
<input type="checkbox"/> Total	<input type="checkbox"/> Total	417,515,017
	<input type="checkbox"/> All Products	417,515,017
	<input type="checkbox"/> Cameras and Camcorders	31,820,248
	<input type="checkbox"/> Computers	336,408,689
	<input type="checkbox"/> Portable Music and Video	49,286,079
<input type="checkbox"/> All Years	<input type="checkbox"/> Total	417,515,017

Report with both OBIEE Grand Total levels and "All" levels in the OLAP hierarchies.

The presentation of the data can be simplified by removing the Total levels from the hierarchical columns in the presentation layer as seen in the following example.

		Sales
Time	Product	
<input checked="" type="checkbox"/> All Years	<input checked="" type="checkbox"/> All Products	417,515,017
	<input checked="" type="checkbox"/> Cameras and Camcorders	31,820,248
	<input checked="" type="checkbox"/> Computers	336,408,689
	<input checked="" type="checkbox"/> Portable Music and Video	49,286,079

Report with OBIEE Grand Total levels removed from the Presentation Layer.

Summary

The Import Metadata feature in the Oracle BI Administration Tool will quickly create a Physical layer from Oracle OLAP cubes and dimensions. The following checklist summarizes the modifications to the Business Model and Presentation layers that will result in an easy to use application for business users.

Preparing the OLAP dimensions:

- Turn off “spread attributes”.
- Consider deleting Short description attributes.
- Turn off dimension and cube materialized views.

Updating the Physical layer:

- Replace host:port:sid in the Data source name of the Connection Pool with a database alias (from the TNSNAMES.ORA file).
- Delete columns representing “spread attributes” for all attributes.
- Consider keeping only the Long or Short Description attributes.
- Delete columns representing the source table keys if cube materialized views are enabled.

Updating the Business Model layer:

- Order columns in the logical tables hierarchically.

Updating the Presentation layer:

- Consider removing columns representing the dimension members (keys).
- Remove End Data and Time Span columns.
- Check Display Columns, keeping either the Long or Short Description attributes.
- Remove Grand Total levels from hierarchical columns.



Creating an Oracle BI Presentation Layer from Imported Oracle OLAP Cubes

June 2012

Author: William Endress

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com



Copyright © 2012, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group. 0612

Hardware and Software, Engineered to Work Together