
Appendix A

Practices and Solutions

Practices Overview

The goal of the course practices is to progressively build an OLAP data model to support the following requirements:

- Summary management for existing SQL table queries by designing OLAP cubes and enabling query rewrite against the cubes
- Enhanced analytic content, including a wide range of calculated measures that enable easy end-user access to rich analytic information
- Easy access to OLAP data by using simple SQL:
 - Creating and executing OLAP queries in SQL Developer
 - Using OLAP queries in Oracle Application Express (APEX)
 - Using OLAP queries in Oracle BI Enterprise Edition (BI EE)
- Cube security
- Efficient cube design for performance and scalability

Practices for Lesson 3

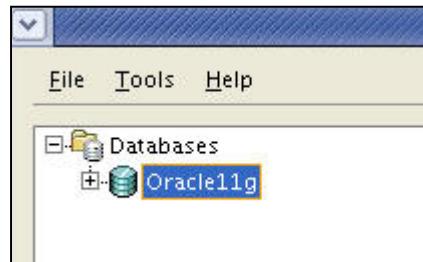
In the practices for this lesson, you start Analytic Workspace Manager (AWM), create a new AW, and build an OLAP cube. You will perform the following tasks:

- Create a database connection for your OLAP environment and log in.
- Create an Analytic Workspace.
- Create Dimensions for your OLAP data model, including:
 - Levels
 - Hierarchies
 - Attributes
 - Dimension mappings
- Load dimension data.
- View dimension data using:
 - OLAP Data Viewer
 - Dimension and hierarchy views
- Create a cube for Sales data.
 - Add the Sales measure.
 - Map the cube to relational sources.
- Load data for the cube.
- View the cube build logs:
 - In AWM
 - At the SQL level
- View cube data using:
 - OLAP Data Viewer
 - Cube views

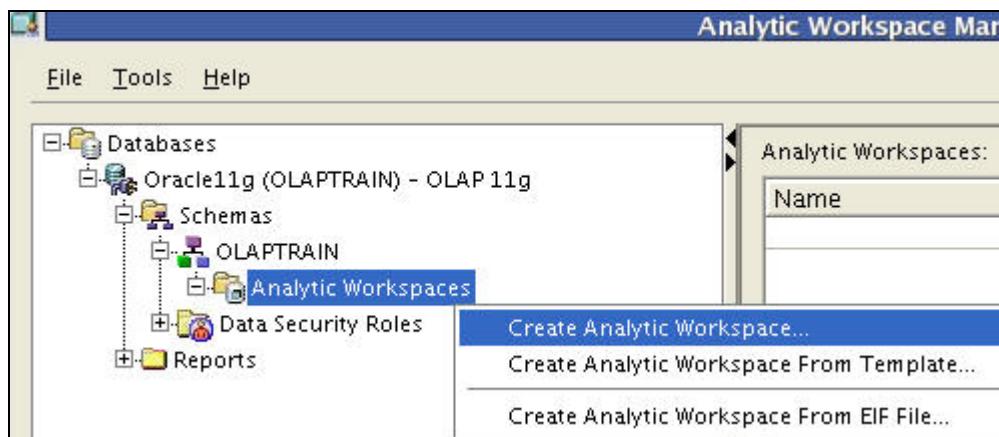
Practice 3-1a: Create the Analytic Workspace

In this practice section, you create an analytic workspace using an existing database connection to the classroom schema. To perform this task:

1. Launch AWM from the desktop.
2. Select the **Databases** node.
3. **Result:** The database connection node named “Oracle11g” appears.



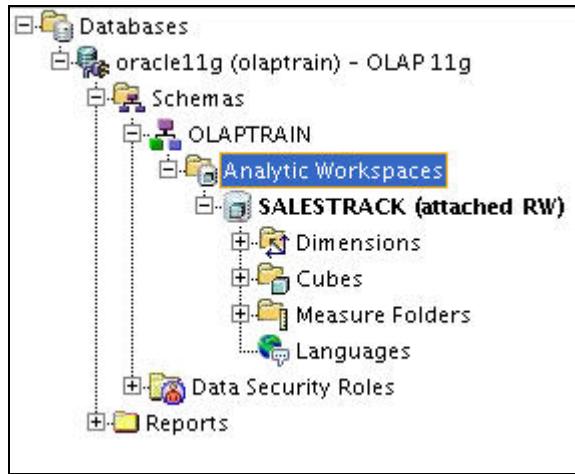
4. Click the plus sign (+) next to the Oracle11g database node.
5. In the Connect to database dialog box, enter and perform the following:
 - a. Username: **olaptrain**
 - b. Password: **oracle**
 - c. Click **OK**.
- Result:** Two folders appear below the database connection – Schemas and Reports.
6. Select **Schemas > OLAPTRAIN**. Then, right-click Analytic Workspaces and select **Create Analytic Workspace** from the menu.



7. In the Create Analytic Workspace dialog box, enter or select the following:
 - a. Name: **SALESTRACK**
 - b. Tablespace: **<default>**
 - c. Click **Create**.

Result: An AW named SALESTRACK appears in the navigator.

8. Select the SALESTRACK node to display the folders that organize OLAP data objects.



Practice 3-1b: Create the Dimensions

In this practice section, you create the four dimensions of what will organize your first OLAP cube: Channel, Time, Product, and Geography. You also create the levels and a hierarchy for each dimension. Finally, you will map the dimensions to their relational sources, and then load the data for one of the dimensions.

1. Right-click the Dimensions node and select **Create Dimension** from the menu.
2. In the General tab of the Create Dimension window, enter or select the following:
 - a. Name: **CHANNEL**
 - b. Dimension Type: **User Dimension**

Note: Caps Lock is auto-enabled for object names. Label and Description fields are auto-filled.

3. In the Levels tab, create three levels for the Channel dimension:

- ALL_CHANNELS
- CLASS
- CHANNEL

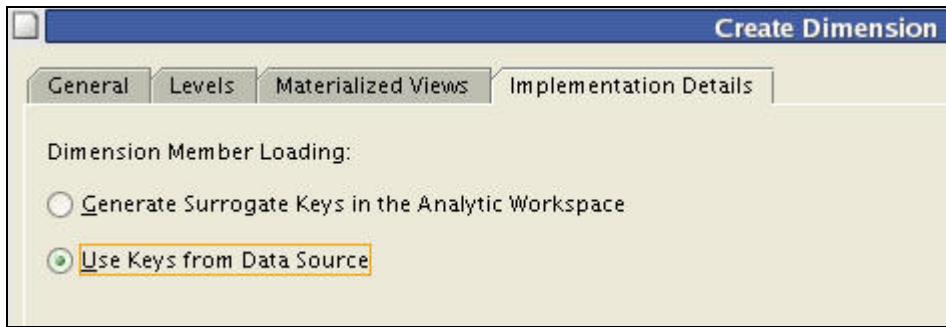
Note: Enter each level name (as listed previously) in the Name field. As before, Label and Description fields are auto-filled. A new level is automatically added when you place the cursor in the Name field.

When you are done, the Levels tab should look like this:

The screenshot shows the 'Create Dimension' dialog box with the 'Levels' tab selected. The table below lists three levels for the 'CHANNEL' dimension.

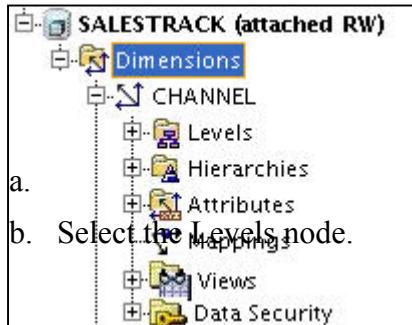
Name	Long Label	Short Label	Description
ALL_CHANNELS	All Channels	All Channels	All Channels
CLASS	Class	Class	Class
CHANNEL	Channel	Channel	Channel

4. In the Implementation Details tab of the Create Dimension dialog box, select **Use Keys from Data Source**.

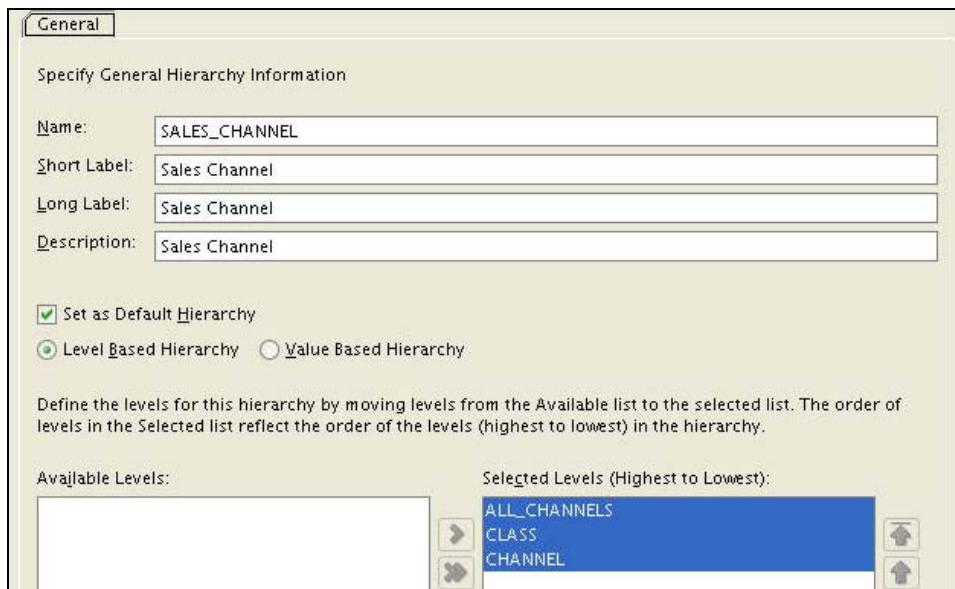


5. Click **Create**.

6. Select **Dimensions > CHANNEL**. The following should appear in the navigator:



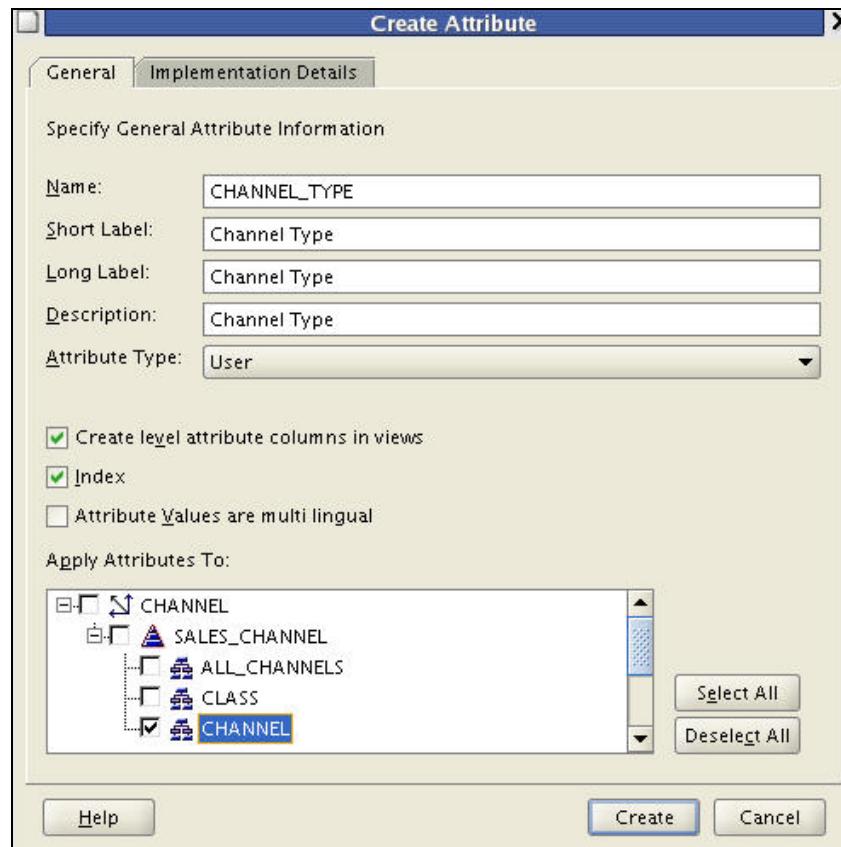
7. Create a Hierarchy for the Channel dimension by performing the following:
- Right-click **Hierarchies** and select **Create Hierarchy**.
 - In the Create Hierarchy dialog box, enter **SALES_CHANNEL** as the name.
 - Click the Add All tool . The General tabbed page should look like this:



- Click **Create**.
- Create a User Attribute for Channel Type by doing the following:
 - Right-click the **Attributes** node below the Channel dimension, and select **Create Attribute**.
 - In the Create Attribute dialog box, select or enter the following:
 - Name = **CHANNEL_TYPE**
 - Attribute Type = **User**
 - In the “Apply Attributes To” box:
 - Click the plus sign next to the Channel dimension.
 - Deselect the Channel dimension check box.
 - Select the **CHANNEL** level check box (the lowest level).

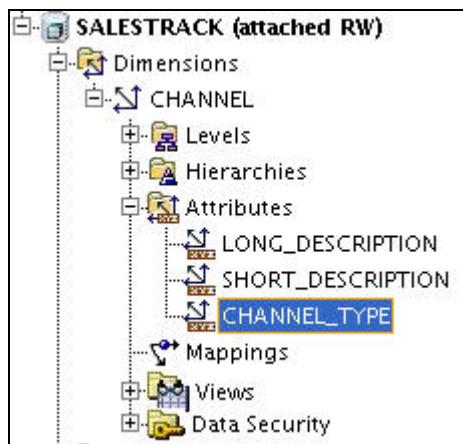
Note: The Channel Type attribute only applies to the lowest level in the Sales Channel hierarchy.

The dialog box should look like this:



- Click **Create**.

The Channel Type attribute is added to the dimension.



- Map the Channel dimension to its relational sources by performing the following:

- Click the **Mappings** node below the Channel dimension.

Result: The Schemas and Mapping panes appear to the right.

- In the Schemas pane, select **OLAPTRAIN > Tables** to display the source tables for the OLAPTRAIN schema. Then, select **CHANNELS** to display its columns.
- Ensure that **Star Schema** is selected for the Type of Dimension Table, and that **Table Mapping View** is selected.
- Drag the following source columns from the Schema pane to the Mapping pane for the CHANNEL and CLASS levels. In addition, enter the following constants for the ALL_CHANNELS level:
 - Member = '**ALL_CHANNELS**'
 - Description attributes = '**All Channels**'

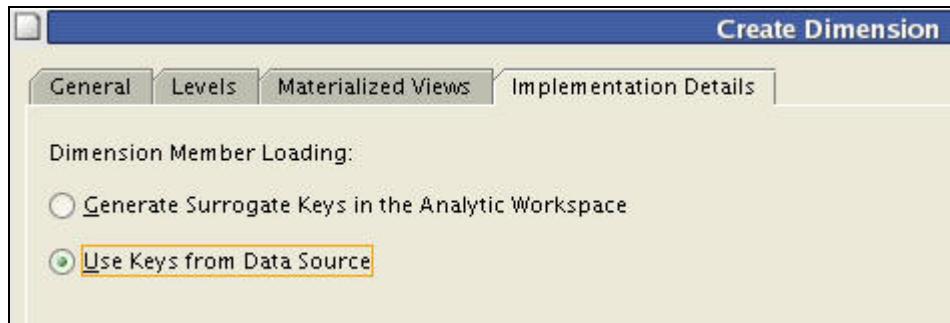
Source Column	
CHANNEL	
HIERARCHIES	
SALES_CHANNEL	
ALL_CHANNELS	
Member	'ALL_CHANNELS'
LONG_DESCRIPTION	'All Channels'
SHORT_DESCRIPTION	'All Channels'
CLASS	
Member	OLAPTRAIN.CHANNELS.CLASS_KEY
LONG_DESCRIPTION	OLAPTRAIN.CHANNELS.CLASS_NAME
SHORT_DESCRIPTION	OLAPTRAIN.CHANNELS.CLASS_NAME
CHANNEL	
Member	OLAPTRAIN.CHANNELS.CHANNEL_KEY
LONG_DESCRIPTION	OLAPTRAIN.CHANNELS.CHANNEL_NAME
SHORT_DESCRIPTION	OLAPTRAIN.CHANNELS.CHANNEL_NAME
CHANNEL_TYPE	OLAPTRAIN.CHANNELS.CHANNEL_TYPE

- Click **Apply**.

Result: The Channel dimension is ready for data loading.

- To create another dimension, right-click the Dimensions node and select **Create Dimension** from the menu.
- In the General tab of the Create Dimension dialog box, enter or select the following:
 - Name: **TIME**
 - Dimension Type: **Time Dimension**
- Using the same techniques that you applied for the Channel dimension, add the following levels to the Time dimension:
 - ALL_YEARS
 - CALENDAR_YEAR
 - CALENDAR_QUARTER
 - MONTH

13. On the Implementation Details tabbed page, select **Use Keys from Data Source**.



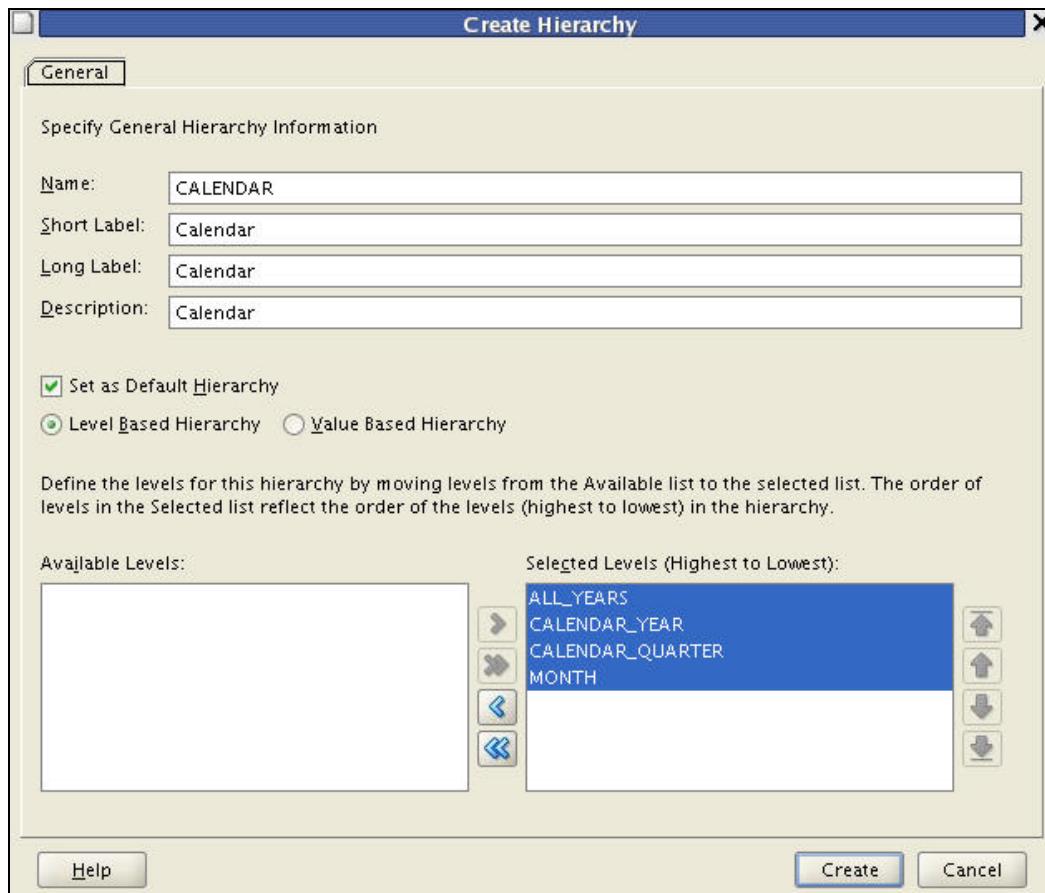
14. Click **Create**.

Result: TIME appears under the Dimensions node.

15. Select **TIME** to display its associated nodes.

16. Using the same techniques that you applied for the Channel dimension, create a Hierarchy named CALENDAR for the Time dimension.

The completed Create Hierarchy dialog box should look like this:



17. Click **Create**.

18. Using the TIMES table in the OLAPTRAIN schema, map the Time dimension as shown in the following image.

Note: For the ALL_YEARS level, use the following values in the Source Column fields (including single quotes):

- Member: **'ALL_YEARS'**
- LONG DESCRIPTION: **'All Years'**
- SHORT DESCRIPTION: **'All Years'**
- END_DATE: **TO_DATE('2008-DEC-31', 'YYYY-MON-DD')**
- TIME_SPAN: **3000**

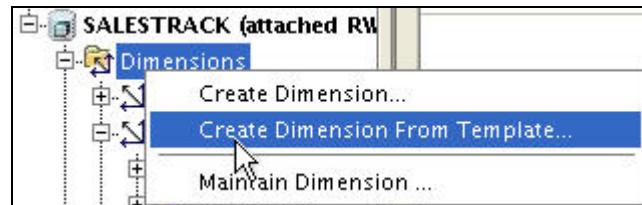
Type of Dimension Table(s):		Star Schema
<input checked="" type="checkbox"/> IX		
<input checked="" type="checkbox"/> MDDATA		
<input checked="" type="checkbox"/> MDSYS		
<input checked="" type="checkbox"/> MGMT_VIEW		
<input checked="" type="checkbox"/> OE		
<input checked="" type="checkbox"/> OLAPSYS		
<input checked="" type="checkbox"/> OLAPTRAIN		
Tables		
<input checked="" type="checkbox"/> CHANNELS		
<input checked="" type="checkbox"/> CUBE_BUILD_LOG		
<input checked="" type="checkbox"/> CUSTOMERS		
<input checked="" type="checkbox"/> DEBUG		
<input checked="" type="checkbox"/> DELTA\$_TABLE10		
<input checked="" type="checkbox"/> DELTA\$_TABLE14		
<input checked="" type="checkbox"/> DELTA\$_TABLE18		
<input checked="" type="checkbox"/> DELTA\$_TABLE55		
<input checked="" type="checkbox"/> DIMENSION_EXCEPTIONS		
<input checked="" type="checkbox"/> MONTHLY_COST_SUMMARY		
<input checked="" type="checkbox"/> MV_CAPABILITIES_TABLE		
<input checked="" type="checkbox"/> PRODUCTS		
<input checked="" type="checkbox"/> REWRITE_TABLE		
<input checked="" type="checkbox"/> SALES_FACT		
<input checked="" type="checkbox"/> TIMES		
<input checked="" type="checkbox"/> VERSION		
TIME		Source Column
<input checked="" type="checkbox"/> HIERARCHIES		
<input checked="" type="checkbox"/> CALENDAR		
<input checked="" type="checkbox"/> ALL_YEARS		
Member		'ALL_YEARS'
LONG_DESCRIP...		'All Years'
SHORT_DESCR...		'All Years'
END_DATE		TO_DATE('2008-DEC-31', 'YYYY-MON-DD')
TIME_SPAN		3000
<input checked="" type="checkbox"/> CALENDAR_YEAR		
Member		OLAPTRAIN.TIMES.CALENDAR_YEAR_ID
LONG_DESCRIP...		OLAPTRAIN.TIMES.CALENDAR_YEAR_NAME
SHORT_DESCR...		OLAPTRAIN.TIMES.CALENDAR_YEAR_NAME
END_DATE		OLAPTRAIN.TIMES.CALENDAR_YEAR_END_DATE
TIME_SPAN		OLAPTRAIN.TIMES.CALENDAR_YEAR_TIME_SPAN
<input checked="" type="checkbox"/> CALENDAR_QUAR...		
Member		OLAPTRAIN.TIMES.CALENDAR_QUARTER_ID
LONG_DESCRIP...		OLAPTRAIN.TIMES.CALENDAR_QUARTER_NAME
SHORT_DESCR...		OLAPTRAIN.TIMES.CALENDAR_QUARTER_NAME
END_DATE		OLAPTRAIN.TIMES.CALENDAR_QUARTER_END_DATE
TIME_SPAN		OLAPTRAIN.TIMES.CALENDAR_QUARTER_TIME_SPAN
<input checked="" type="checkbox"/> MONTH		
Member		OLAPTRAIN.TIMES.MONTH_ID
LONG_DESCRIP...		OLAPTRAIN.TIMES.MONTH_NAME
SHORT_DESCR...		OLAPTRAIN.TIMES.MONTH_NAME
END_DATE		OLAPTRAIN.TIMES.MONTH_END_DATE
TIME_SPAN		OLAPTRAIN.TIMES.MONTH_TIME_SPAN

19. Click Apply.

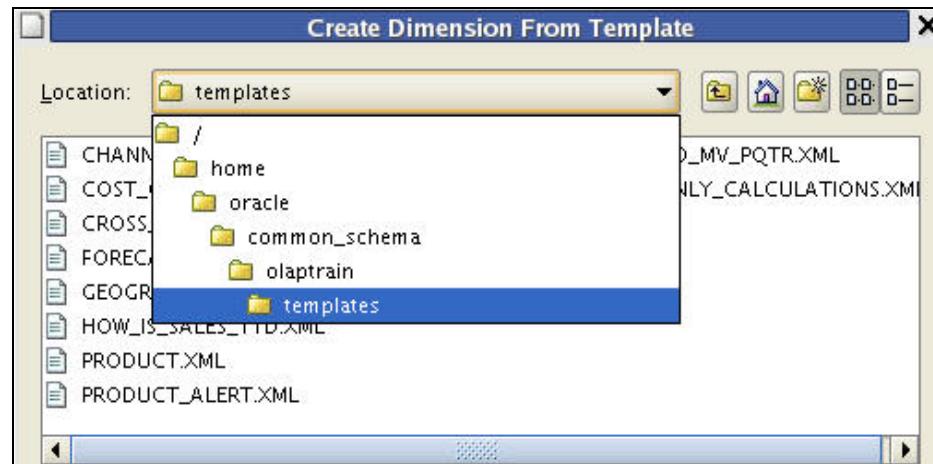
Result: The Time dimension is ready for data loading.

20. Create the Geography dimension from an XML template file.

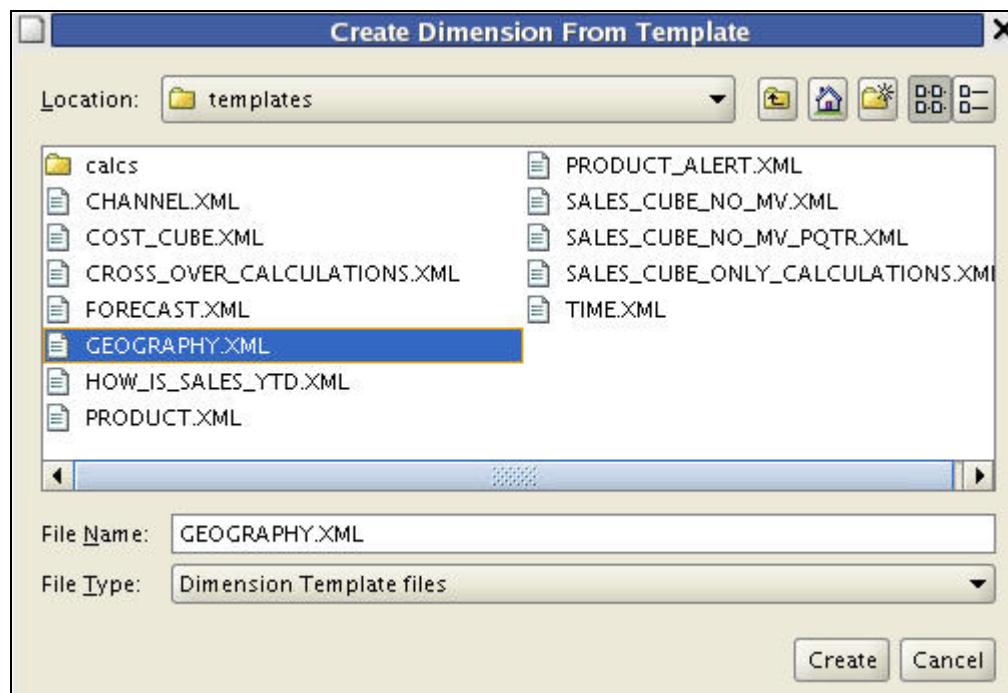
- a. In the navigator, right-click Dimensions and select **Create Dimension From Template**.



- b. In the Create Dimension From Template dialog box, navigate to the home/oracle/common_schema/olaptrain/templates folder.



- c. Select **GEOGRAPHY.XML** and click **Create**.



Result: The Geography dimension is created.

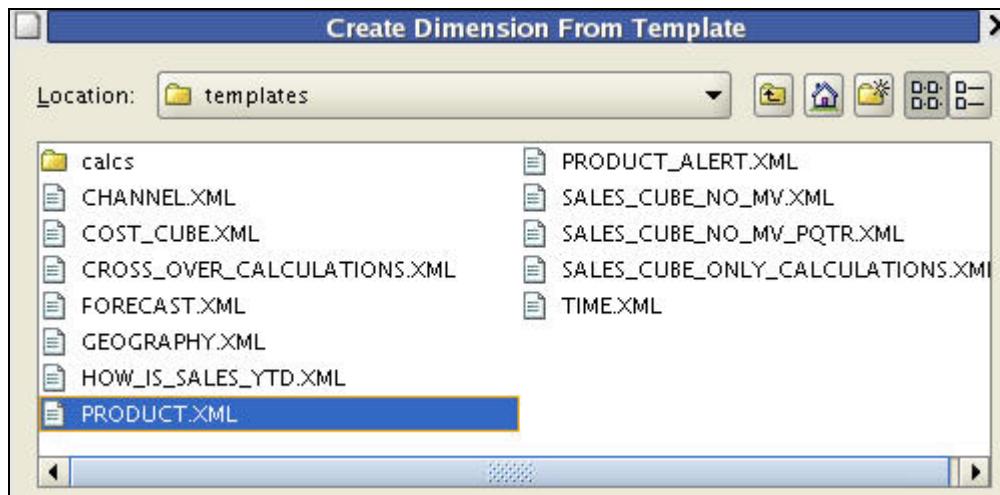
21. Under the Dimensions node in the navigator, select **GEOGRAPHY**, and then the **Mappings** node.

The completed mapping for the Product dimension appears.

GEOGRAPHY	Source Column
HIERARCHIES	
REGIONAL	
ALL_REGIONS	
Member	'ALL_REGIONS'
LONG_DESCRIP...	'All Regions'
SHORT_DESCRIP...	'All Regions'
REGION	
Member	OLAPTRAIN.CUSTOMERS.REGION_KEY
LONG_DESCRIP...	OLAPTRAIN.CUSTOMERS.REGION_NAME
SHORT_DESCRIP...	OLAPTRAIN.CUSTOMERS.REGION_NAME
COUNTRY	
Member	OLAPTRAIN.CUSTOMERS.COUNTRY_KEY
LONG_DESCRIP...	OLAPTRAIN.CUSTOMERS.COUNTRY_NAME
SHORT_DESCRIP...	OLAPTRAIN.CUSTOMERS.COUNTRY_NAME
STATE_PROVINCE	
Member	OLAPTRAIN.CUSTOMERS.STATE_PROVINCE_KEY
LONG_DESCRIP...	OLAPTRAIN.CUSTOMERS.STATE_PROVINCE_NAME
SHORT_DESCRIP...	OLAPTRAIN.CUSTOMERS.STATE_PROVINCE_NAME

22. Create the Product dimension from an XML template file.

- In the navigator, right-click Dimensions and select **Create Dimension From Template**.
- Select **PRODUCT.XML** and click **Create**.



23. Under the Dimensions node in the navigator, select PRODUCT and then the **Mappings** node.

The completed mapping for the Product dimension appears.

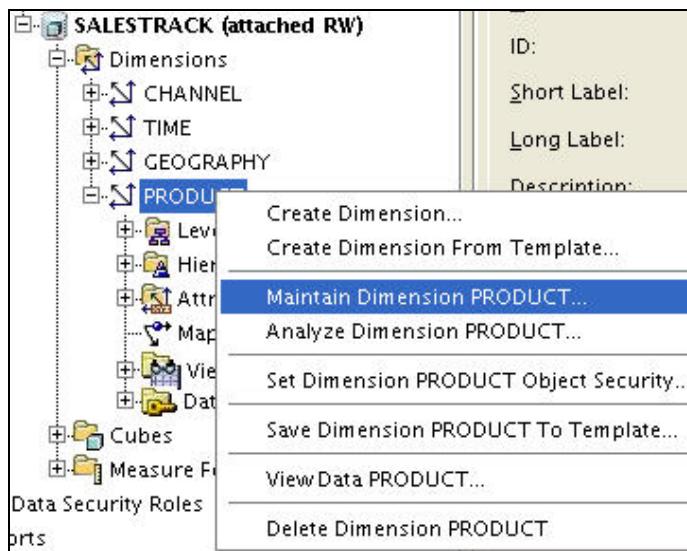
PRODUCT	Source Column
HIERARCHIES	
STANDARD	
ALL_PRODUCTS	
Member	'ALL_PRODUCTS'
LONG_DESCRIP...	'All Products'
SHORT_DESCRIP...	'All Products'
DEPARTMENT	
Member	OLAPTRAIN.PRODUCTS.DEPARTMENT_KEY
LONG_DESCRIP...	OLAPTRAIN.PRODUCTS.DEPARTMENT_NAME
SHORT_DESCRIP...	OLAPTRAIN.PRODUCTS.DEPARTMENT_SHORT_DESC
CATEGORY	
Member	OLAPTRAIN.PRODUCTS.CATEGORY_KEY
LONG_DESCRIP...	OLAPTRAIN.PRODUCTS.CATEGORY_NAME
SHORT_DESCRIP...	OLAPTRAIN.PRODUCTS.CATEGORY_SHORT_DESC
TYPE	
Member	OLAPTRAIN.PRODUCTS.TYPE_KEY
LONG_DESCRIP...	OLAPTRAIN.PRODUCTS.TYPE_NAME
SHORT_DESCRIP...	OLAPTRAIN.PRODUCTS.TYPE_SHORT_DESC
SUBTYPE	
Member	OLAPTRAIN.PRODUCTS.SUBTYPE_KEY
LONG_DESCRIP...	OLAPTRAIN.PRODUCTS.SUBTYPE_NAME
SHORT_DESCRIP...	OLAPTRAIN.PRODUCTS.SUBTYPE_SHORT_DESC
ITEM	
Member	OLAPTRAIN.PRODUCTS.ITEM_KEY
LONG_DESCRIP...	OLAPTRAIN.PRODUCTS.ITEM_NAME
SHORT_DESCRIP...	OLAPTRAIN.PRODUCTS.ITEM_SHORT_DESC

Practice 3-1c: Load and View Dimension Data

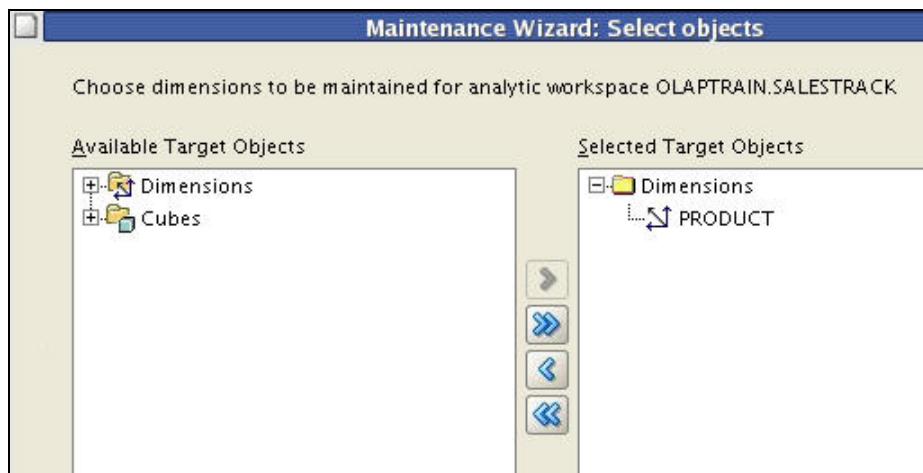
In this practice section, you load data for the Product dimension and view the results.

1. Load the data for the Product dimension by performing the following steps:

- a. In the navigator, right-click PRODUCT and select **Maintain Dimension PRODUCT** from the menu.



Result: The Maintenance Wizard appears.

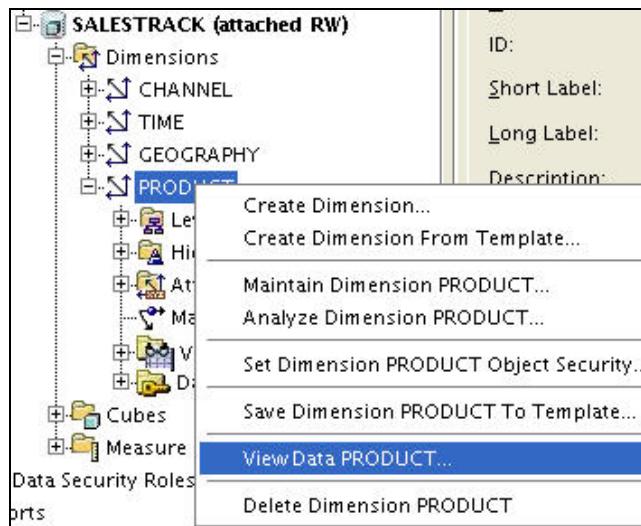


- b. Accept the defaults and click **Finish**.

Result: Data for the Product dimension is loaded and the AWM Build Log appears when the data load is finished.

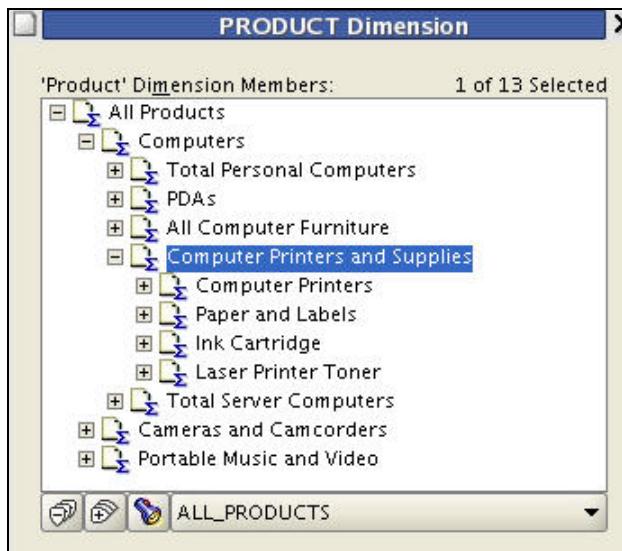
Build Log						
Fetched 7 rows						
BUILD_ID	STATUS	ELAPSED	START_TIME	END_TIME	BUILD_OBJECT	
4	COMPLETED	0 H 0M 02...	2008-8-4 1...	2008-8-4 1...		
4	COMPLETED	0 H 0M 00...	2008-8-4 1...	2008-8-4 1...		
4	COMPLETED	0 H 0M 00...	2008-8-4 1...	2008-8-4 1...	PRODUCT	
4	COMPLETED	0 H 0M 01...	2008-8-4 1...	2008-8-4 1...	PRODUCT	
4	COMPLETED	0 H 0M 00...	2008-8-4 1...	2008-8-4 1...	PRODUCT	
4	COMPLETED	0 H 0M 00...	2008-8-4 1...	2008-8-4 1...	PRODUCT	
4	COMPLETED	0 H 0M 00...	2008-8-4 1...	2008-8-4 1...	PRODUCT	

2. Click **Close**.
3. In the navigator, right-click PRODUCT and select **View Data PRODUCT** from the menu.



Result: The Product dimension appears in the AWM Data Viewer.

- a. You can review the hierarchical structure of a dimension using the Data Viewer. For example, select **All Products > Computers > Computer Printers and Supplies**. You should see the following display:



- b. Feel free to navigate through the dimension. After you are finished, click **Close**.
4. Next, view data from the relational Hierarchy View that was created for the Product dimension.

Note: Remember that OLAP automatically creates a Dimension view, and at least one Hierarchy view for each OLAP dimension that you create. These views are used for SQL access to the OLAP data.

- a. In the navigator, select **Views** node under the PRODUCT dimension, and then select **PRODUCT_STANDARD_VIEW**.



Result: The structure of the Hierarchy view appears in the General tab, displayed in the right pane.

- b. Click the **Data** tab.

Result: The following appears in the right pane:

General Data

Dimension: PRODUCT View: PRODUCT_STANDARD_VIEW Hierarchy: STANDARD

Columns... Sort... Filter: Enter Where Clause

	DIM_KEY	LEVEL_NAME	LONG_DESCRIPTION	SHORT_DESCRIPTION
1	ALL_PRODUCTS	ALL_PRODUCTS	All Products	All Products
2	-520	DEPARTMENT	Portable Music and Video	Portable Music and Video
3	-519	DEPARTMENT	Cameras and Camcorders	Cameras and Camcorders
4	-518	DEPARTMENT	Computers	Computers
5	-536	CATEGORY	Total Server Computers	Total Server Computers
6	-535	CATEGORY	Computer Printers and Supplies	Computer Printers and Supplies
7	-534	CATEGORY	Camcorders and Accessories	Camcorders and Accessories
8	-533	CATEGORY	All Computer Furniture	All Computer Furniture
9	-532	CATEGORY	Cameras and Accessories	Cameras and Accessories
10	-531	CATEGORY	PDAs	PDAs
11	-530	CATEGORY	Total iPlayer Family	Total iPlayer Family
12	-529	CATEGORY	Total Personal Computers	Total Personal Computers
13	-600	TYPE	PC Sound	PC Sound
14	-599	TYPE	Laser Printer Toner	Laser Printer Toner
15	-598	TYPE	Camera Accessories	Camera Accessories
16	-597	TYPE	Ink Cartridge	Ink Cartridge
17	-596	TYPE	Input Devices	Input Devices
18	-595	TYPE	Servers Systems	Servers Systems

- c. Enter the following WHERE clause in the Filter box:
LEVEL_NAME = 'CATEGORY'

- d. Press **Enter**.

Result: The following set of data should appear:

General Data

Dimension: PRODUCT View: PRODUCT_STANDARD_VIEW Hierarchy: STANDARD

Columns... Sort... Filter: LEVEL_NAME = 'CATEGORY'

	DIM_KEY	LEVEL_NAME	LONG_DESCRIPTION	SHORT_DESCRIPTION
1	-536	CATEGORY	Total Server Computers	Total Server Computers
2	-535	CATEGORY	Computer Printers and Supplies	Computer Printers and Supplies
3	-534	CATEGORY	Camcorders and Accessories	Camcorders and Accessories
4	-533	CATEGORY	All Computer Furniture	All Computer Furniture
5	-532	CATEGORY	Cameras and Accessories	Cameras and Accessories
6	-531	CATEGORY	PDAs	PDAs
7	-530	CATEGORY	Total iPlayer Family	Total iPlayer Family
8	-529	CATEGORY	Total Personal Computers	Total Personal Computers

Note: Later in the course, you will learn how to use these columns in SQL queries to return the desired OLAP data.

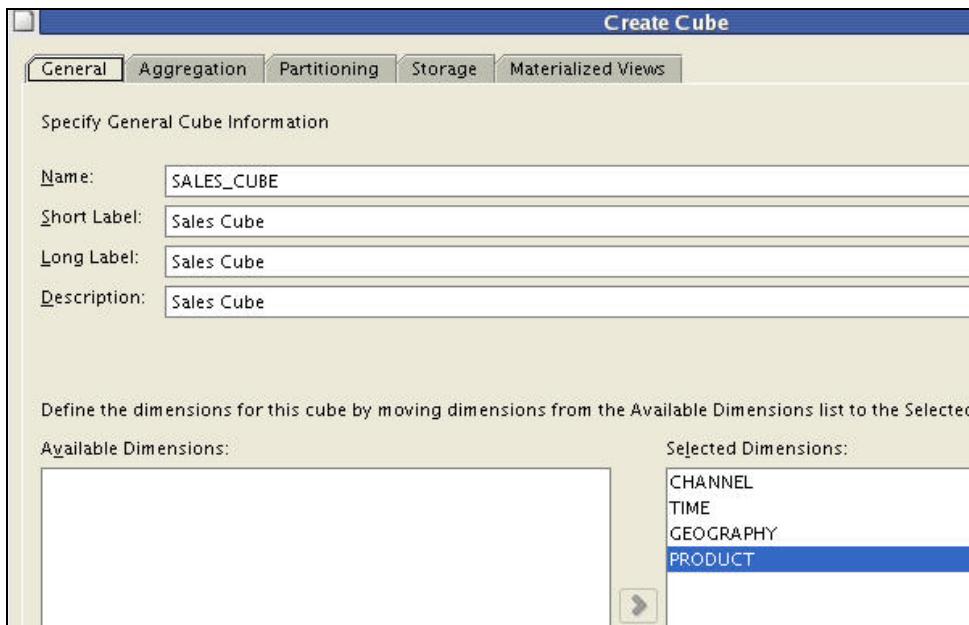
- 5. Collapse the PRODUCT dimension in the navigator.

Practice 3-2a: Create the Sales Cube

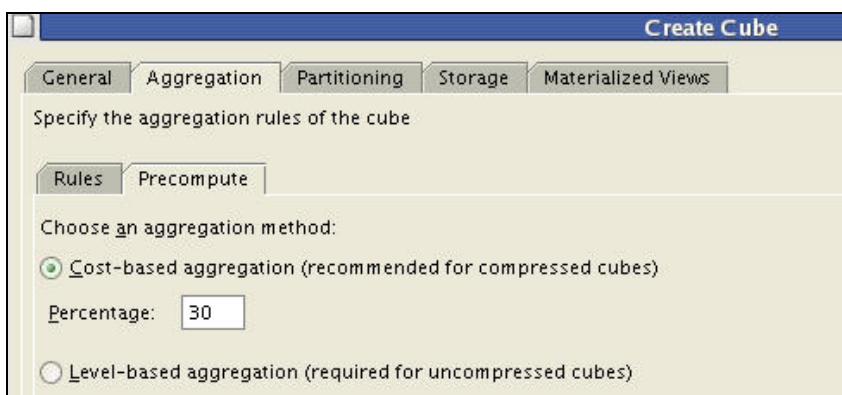
In the practices in this section, you create a cube and add Sales and Quantity measures to it. You map the data, and then enable the cube for query rewrite. Finally, you load the data and view the results.

1. Right-click the Cubes node and select **Create Cube** from the menu.
2. In the General tab of the Create Cube dialog box, specify the following:
 - a. Name: **SALES_CUBE**
 - b. Selected dimension order:
 - CHANNEL
 - TIME
 - GEOGRAPHY
 - PRODUCT

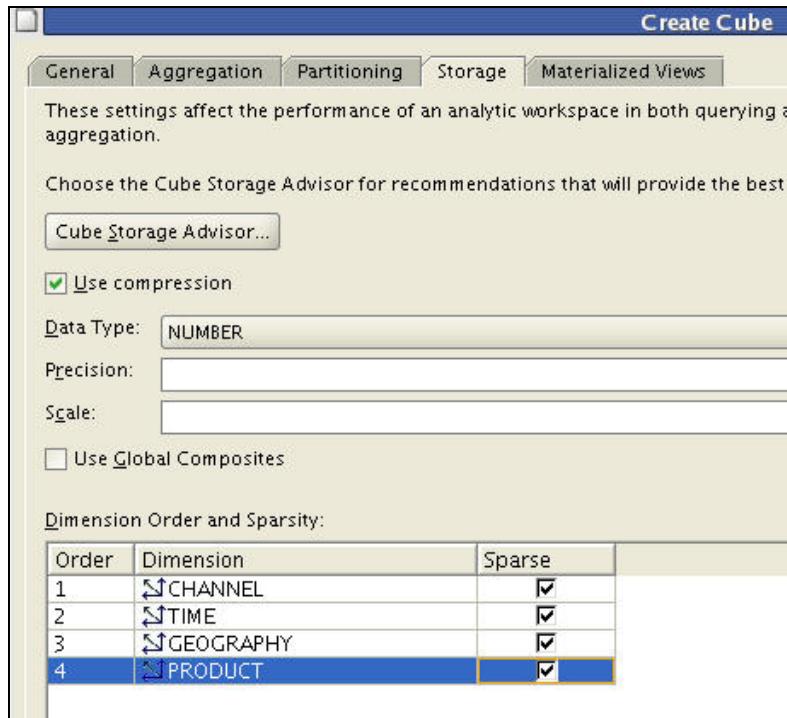
Result: The Create Cube dialog box should look like this:



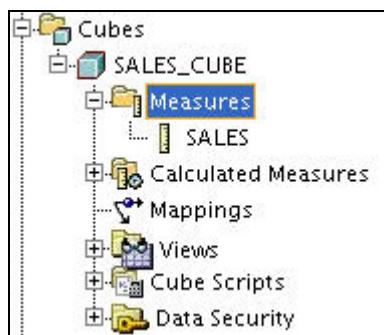
3. Select the Aggregation tab. Then, in the Precompute subtab, specify a value of 30 for **Cost-based aggregation**.



- Select the Storage tab. Select the **Sparse** option for all dimensions.



- Click **Create**.
- In the navigator, select **SALES_CUBE**.
- Right-click Measures and select **Create Measure**.
- In the Create Measure dialog box, enter **SALES** as the name and click **Create**.
- In the navigator, select **Measures**. You should see the following:



- Using the same techniques described in steps 7 and 8, create a second measure named **QUANTITY**.
- Click the **Mappings** node under **SALES_CUBE**, and ensure that the **Table Mapping View tool** (shown below) is selected.



- Under the Schemas node, select **OLAPTRAIN > Tables**.

13. Using the SALES_FACT, CHANNELS, TIMES, CUSTOMERS, and PRODUCTS tables, map the SALES_CUBE as shown in the following table.

Note: When mapping the Join Condition:

- First, drag the foreign key column from the fact table to the Source Column field.
- Then, drag the primary key column from the dimension table to the Source Column field. The equal sign (“=”) is automatically inserted after you drag the second column into the Source Column field.

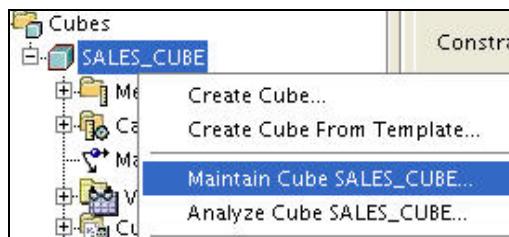
SALES_CUBE	Source Column
MEASURES	
SALES	OLAPTRAIN.SALES_FACT.SALES
QUANTITY	OLAPTRAIN.SALES_FACT.QUANTITY
DIMENSIONS	
CHANNEL	
ALL_CHANNELS	
CLASS	
CHANNEL	OLAPTRAIN.CHANNELS.CHANNEL_KEY
Join Condition	OLAPTRAIN.SALES_FACT.CHANNEL=OLAPTRAIN.CHANNELS.CHANNEL_KEY
TIME	
ALL_YEARS	
CALENDAR_YEAR	
CALENDAR_QUARTER	
MONTH	OLAPTRAIN.TIMES.MONTH_ID
Join Condition	OLAPTRAIN.SALES_FACT.DAY_KEY=OLAPTRAIN.TIMES.DAY_KEY
GEOGRAPHY	
ALL_REGIONS	
REGION	
COUNTRY	
STATE_PROVINCE	OLAPTRAIN.CUSTOMERS.STATE_PROVINCE_KEY
Join Condition	OLAPTRAIN.SALES_FACT.CUSTOMER=OLAPTRAIN.CUSTOMERS.CUSTOMER_KEY
PRODUCT	
ALL_PRODUCTS	
DEPARTMENT	
CATEGORY	
TYPE	
SUBTYPE	
ITEM	OLAPTRAIN.PRODUCTS.ITEM_KEY
Join Condition	OLAPTRAIN.SALES_FACT.PRODUCT=OLAPTRAIN.PRODUCTS.ITEM_KEY

14. Click **Apply**.

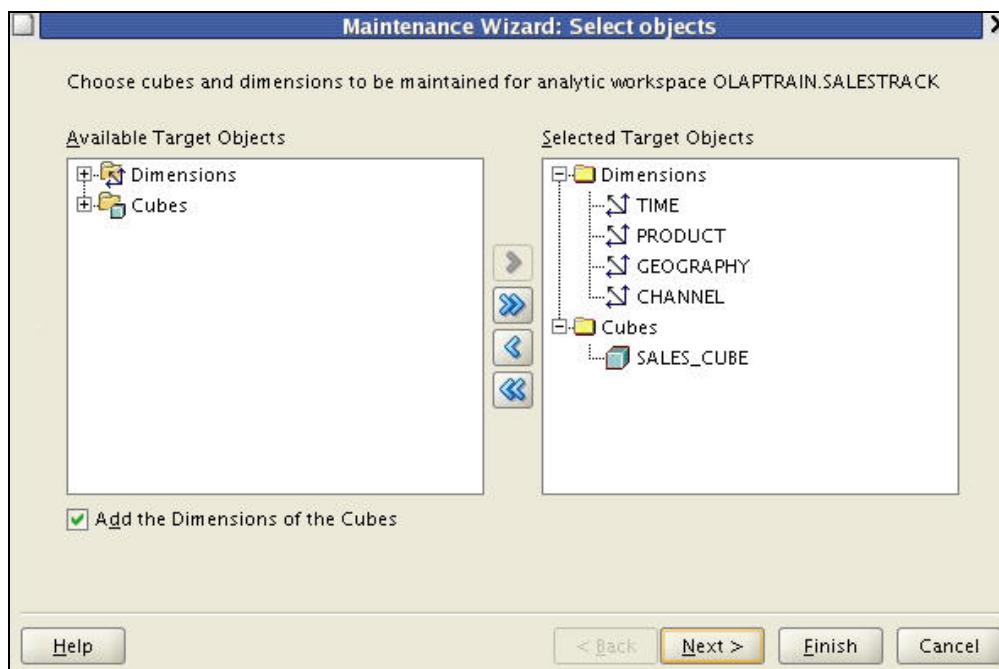
Practice 3-2b: Load and View Data for the Sales Cube

In the practices in this section, you load the data to the Sales cube, and you view cube data using both the AWM Data Viewer and the cube views.

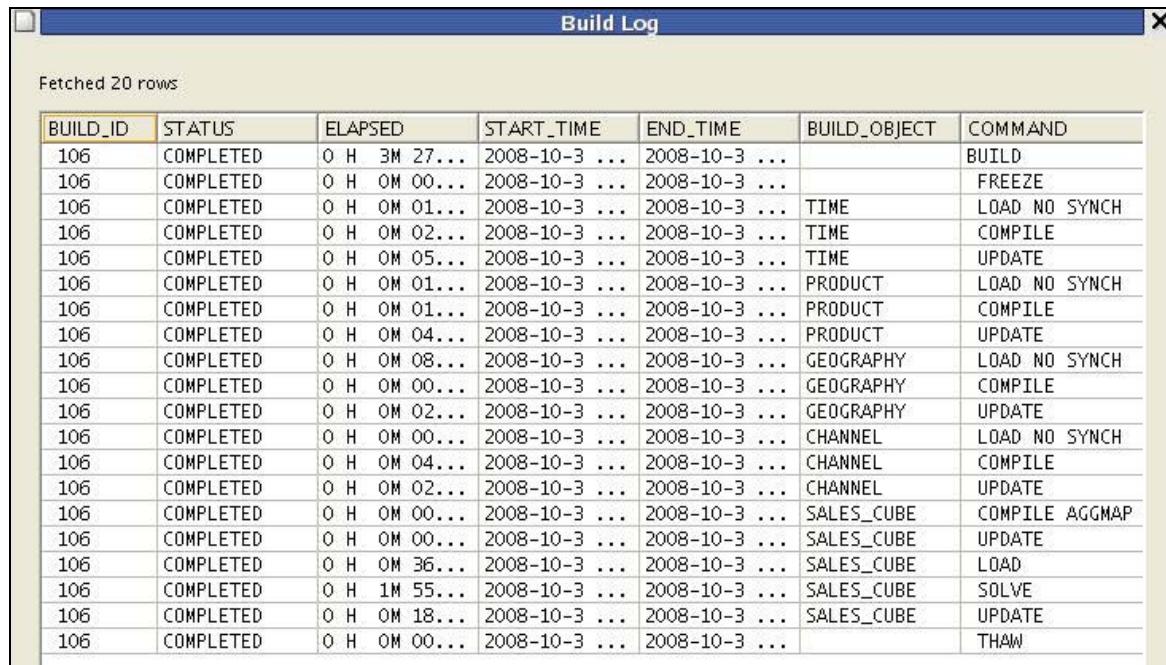
1. In the navigator, right-click SALES_CUBE and select **Maintain Cube SALES_CUBE** from the menu.



2. In the Maintenance Wizard, accept all defaults and click **Finish** to begin the load process.



When the build and load process is complete, the AWM Build Log appears:



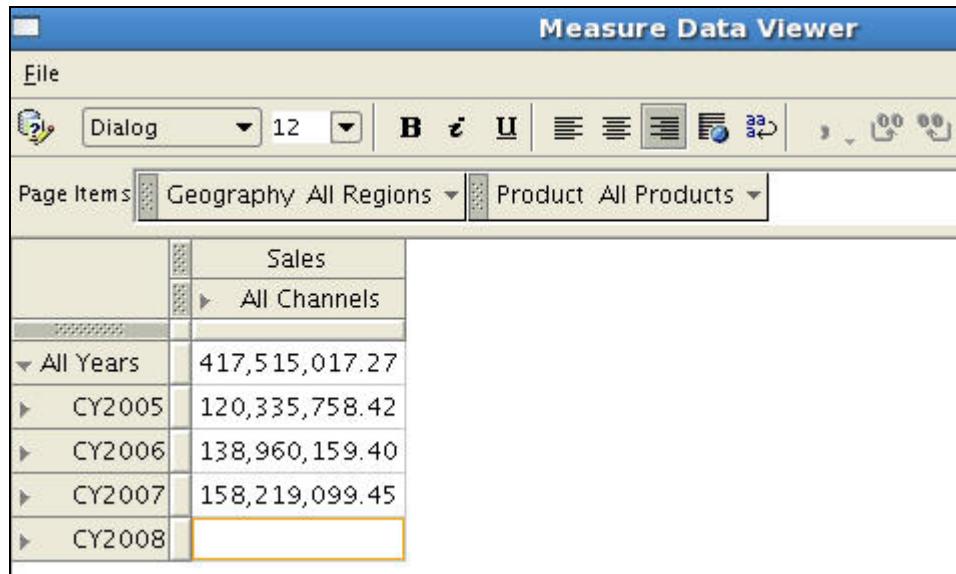
The screenshot shows a window titled "Build Log" with a message "Fetched 20 rows" at the top. Below is a table with columns: BUILD_ID, STATUS, ELAPSED, START_TIME, END_TIME, BUILD_OBJECT, and COMMAND. The data consists of 20 rows of build log entries.

BUILD_ID	STATUS	ELAPSED	START_TIME	END_TIME	BUILD_OBJECT	COMMAND
106	COMPLETED	0 H 3M 27...	2008-10-3 ...	2008-10-3 ...		BUILD
106	COMPLETED	0 H 0M 00...	2008-10-3 ...	2008-10-3 ...		FREEZE
106	COMPLETED	0 H 0M 01...	2008-10-3 ...	2008-10-3 ...	TIME	LOAD NO SYNC
106	COMPLETED	0 H 0M 02...	2008-10-3 ...	2008-10-3 ...	TIME	COMPILE
106	COMPLETED	0 H 0M 05...	2008-10-3 ...	2008-10-3 ...	TIME	UPDATE
106	COMPLETED	0 H 0M 01...	2008-10-3 ...	2008-10-3 ...	PRODUCT	LOAD NO SYNC
106	COMPLETED	0 H 0M 01...	2008-10-3 ...	2008-10-3 ...	PRODUCT	COMPILE
106	COMPLETED	0 H 0M 04...	2008-10-3 ...	2008-10-3 ...	PRODUCT	UPDATE
106	COMPLETED	0 H 0M 08...	2008-10-3 ...	2008-10-3 ...	GEOGRAPHY	LOAD NO SYNC
106	COMPLETED	0 H 0M 00...	2008-10-3 ...	2008-10-3 ...	GEOGRAPHY	COMPILE
106	COMPLETED	0 H 0M 02...	2008-10-3 ...	2008-10-3 ...	GEOGRAPHY	UPDATE
106	COMPLETED	0 H 0M 00...	2008-10-3 ...	2008-10-3 ...	CHANNEL	LOAD NO SYNC
106	COMPLETED	0 H 0M 04...	2008-10-3 ...	2008-10-3 ...	CHANNEL	COMPILE
106	COMPLETED	0 H 0M 02...	2008-10-3 ...	2008-10-3 ...	CHANNEL	UPDATE
106	COMPLETED	0 H 0M 00...	2008-10-3 ...	2008-10-3 ...	SALES_CUBE	COMPILE AGGMAP
106	COMPLETED	0 H 0M 00...	2008-10-3 ...	2008-10-3 ...	SALES_CUBE	UPDATE
106	COMPLETED	0 H 0M 36...	2008-10-3 ...	2008-10-3 ...	SALES_CUBE	LOAD
106	COMPLETED	0 H 1M 55...	2008-10-3 ...	2008-10-3 ...	SALES_CUBE	SOLVE
106	COMPLETED	0 H 0M 18...	2008-10-3 ...	2008-10-3 ...	SALES_CUBE	UPDATE
106	COMPLETED	0 H 0M 00...	2008-10-3 ...	2008-10-3 ...		THAW

3. After examining the Build Log, click **Close**.
4. Right-click the SALES measure and select **View Data SALES** from the menu.

Result: Sales data is displayed for a default set of dimension members.

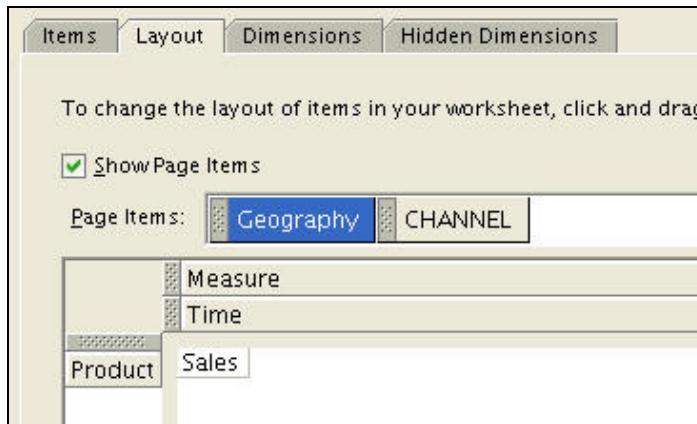
5. In the Data Viewer, select **All Years**.



6. Click the **Query Builder** tool  , under the File menu.
7. Click the **Layout** tab, and then perform the following:
 - a. Move **Channel** to the Page Items region.
 - b. Move **Time** underneath the Measure in the column axis

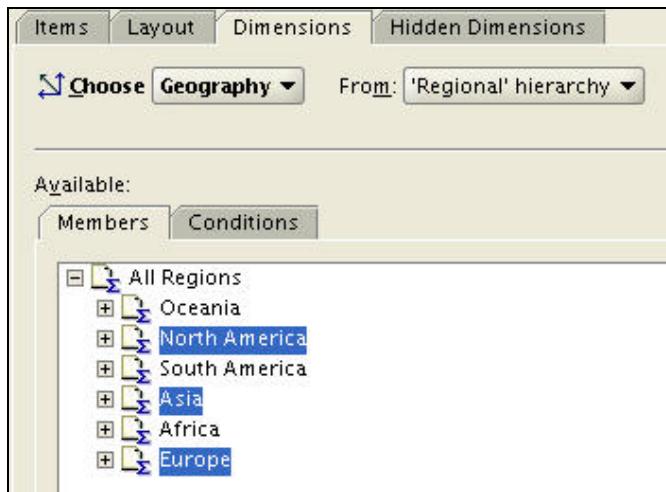
- c. Move **Product** to the row axis.

The layout tab should look like this:

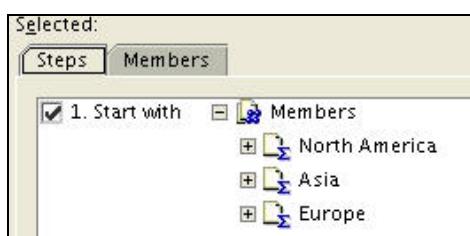


8. Click the **Dimensions** tab, and then do the following:

- Select **Geography** from the Choose list.
- Click the **Remove All Items** tool to clear the Selected list.
- In the Members tab of the Available list, select All Regions and select **North America, Asia, and Europe**.

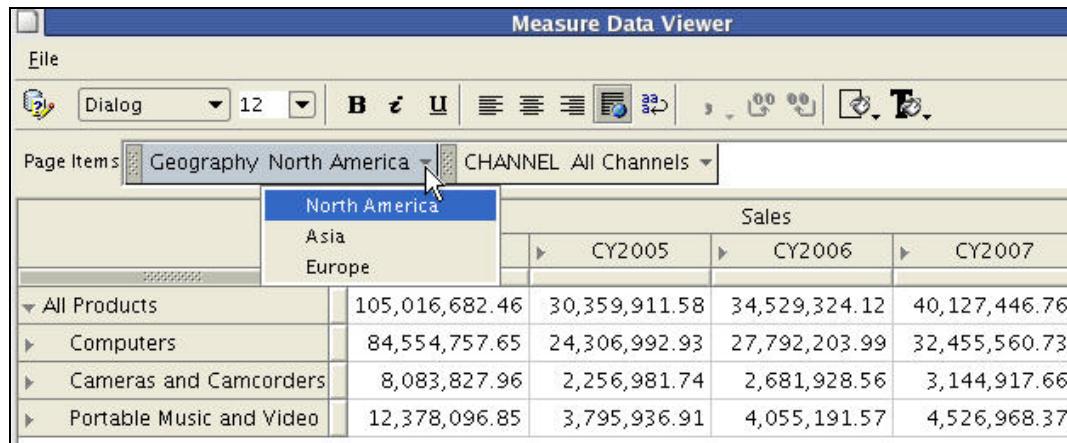


- Click the **Add Selected Items** tool to move the chosen dimension members to the Selected list, like this:



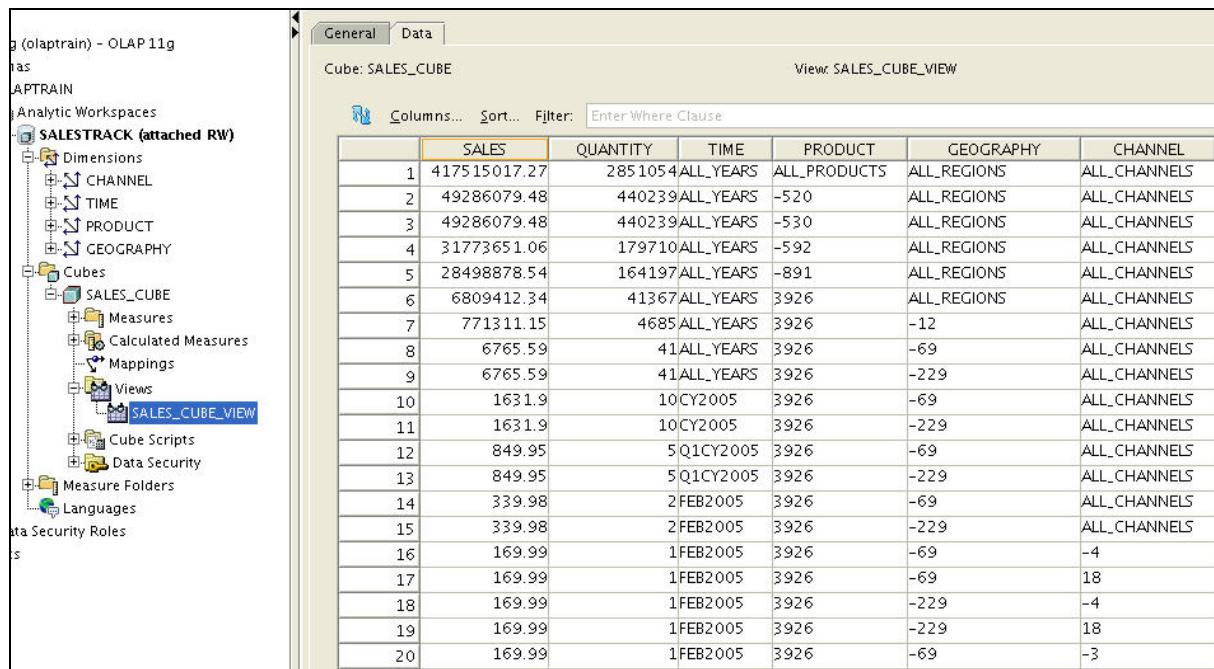
- Click **OK** to refresh the data with your new selections.

- In the Data Viewer, select **All Products**. Also, notice that you can select from a list of members for the Geography dimension.



The screenshot shows the Measure Data Viewer interface. At the top, there are buttons for File, Dialog, and various toolbar icons. Below the toolbar, the page items are set to "Geography North America" and "CHANNEL All Channels". The main area displays a table titled "Sales" with columns for Product, Sales, Quantity, Time, and Geography. The "North America" dimension is selected, highlighted in blue. The table data includes rows for All Products, Computers, Cameras and Camcorders, and Portable Music and Video, with corresponding sales figures for CY2005, CY2006, and CY2007.

- When you are done experimenting with the report, close the Data Viewer.
- Select the **Views** node under the **SALES_CUBE**.
- Select **SALES_CUBE_VIEW** from the navigator, and click the Data tab in the right pane.



The screenshot shows the Oracle Advanced Multidimensional (AWM) interface. On the left, the Navigator pane displays the database structure, including the SALESTRACK workspace, dimensions (CHANNEL, TIME, PRODUCT, GEOGRAPHY), cubes (SALES_CUBE), and views (SALES_CUBE_VIEW). The SALES_CUBE_VIEW node is currently selected. The main pane shows the "Data" tab for the SALES_CUBE cube, with the View: SALES_CUBE_VIEW selected. The data grid displays 20 rows of sales data with columns for Sales, Quantity, Time, Product, Geography, and Channel. The data shows various sales figures across different years and channels.

Note: AWM provides a view of the relational cube view representation of the data from within the administrative tool. Later, you learn how to query the OLAP data directly by using SQL against cube views.

- Collapse the SALES_CUBE node by clicking its minus sign.

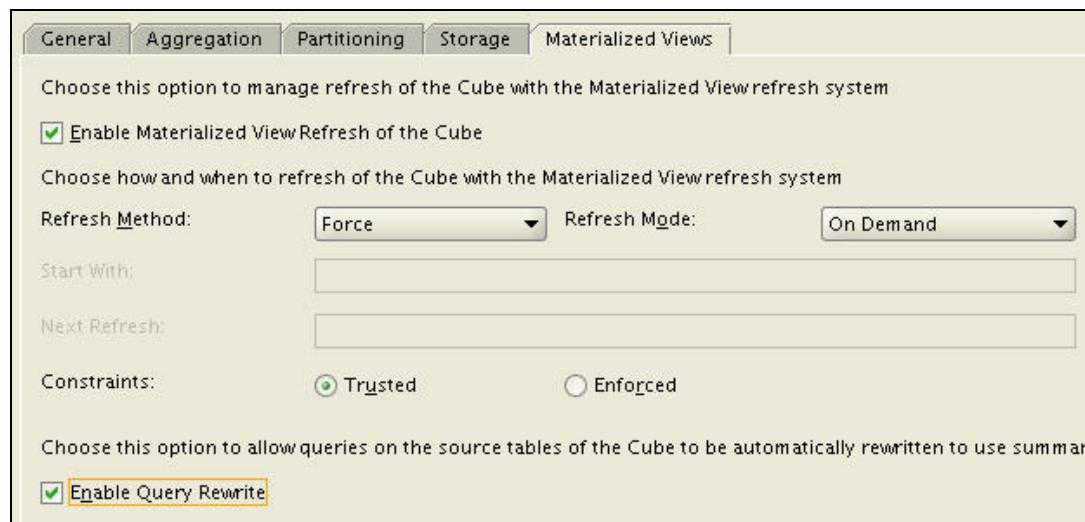
Practices for Lesson 4

In the practices for this lesson, you enable Query Rewrite and MV Refresh for the Sales Cube, and then perform a build to create the Cube MVs. Then, you run some queries that do not rewrite correctly to the Cube MVs. Using the `mv_cube_advice` package, you determine why the queries are not rewriting. You then modify the queries so that they rewrite to the cube MVs.

Practice 4-1: Enabling and Building Cube MVs

In this practice section, you enable Cube MVs in AWM and run a build.

1. In the AWM navigator, click **SALES_CUBE**.
2. In the right pane, click the **Materialized Views** tab and select:
 - a. **Enable Materialized View Refresh of the Cube**
 - b. **Enable Query Rewrite**



- c. Accept the default settings for all other options, and click **Apply**.

Result: The following information box appears:



3. When the processing is complete, go to the navigation pane and select **Dimensions > Geography**.

4. Select the **Mappings** node under Geography, and notice that OLAP has added a new Unique Key attribute to each level in the dimension, and automapped the attributes:

GEOGRAPHY	Source Column
HIERARCHIES	
REGIONS	
ALL_REGIONS	
Member	'ALL_REGIONS'
LONG_DESCRIPTION	'All Regions'
SHORT_DESCRIPTION	'All Regions'
GEOGRAPHY_ALL_REGIONS_ID	'ALL_REGIONS'
REGION	
Member	OLAPTRAIN.CUSTOMERS.REGION_KEY
LONG_DESCRIPTION	OLAPTRAIN.CUSTOMERS.REGION_NAME
SHORT_DESCRIPTION	OLAPTRAIN.CUSTOMERS.REGION_NAME
GEOGRAPHY_REGION_ID	OLAPTRAIN.CUSTOMERS.REGION_KEY
COUNTRY	
Member	OLAPTRAIN.CUSTOMERS.COUNTRY_KEY
LONG_DESCRIPTION	OLAPTRAIN.CUSTOMERS.COUNTRY_NAME
SHORT_DESCRIPTION	OLAPTRAIN.CUSTOMERS.ISO_CODE
GEOGRAPHY_COUNTRY_ID	OLAPTRAIN.CUSTOMERS.COUNTRY_KEY
STATE_PROVINCE	
Member	OLAPTRAIN.CUSTOMERS.STATE_PROVINCE_KEY
LONG_DESCRIPTION	OLAPTRAIN.CUSTOMERS.STATE_PROVINCE_NAME
SHORT_DESCRIPTION	OLAPTRAIN.CUSTOMERS.STATE_PROVINCE_NAME
GEOGRAPHY_STATE_PROVINCE_ID	OLAPTRAIN.CUSTOMERS.STATE_PROVINCE_KEY

These Unique Key attributes are required by the Materialized View subsystem for the query rewrite. Each level of each dimension hierarchy in the cube contains the correct additional attributes.

- In the navigator, right-click SALES_CUBE and select **Maintain Cube SALES_CUBE** from the menu.
- In the Maintenance Wizard, accept all defaults and click **Finish** to begin the load process.
- When the build completes, close the Build Log window.
- Exit AWM.

Practice 4-2: Running Summary Queries with and Without Query Rewrite

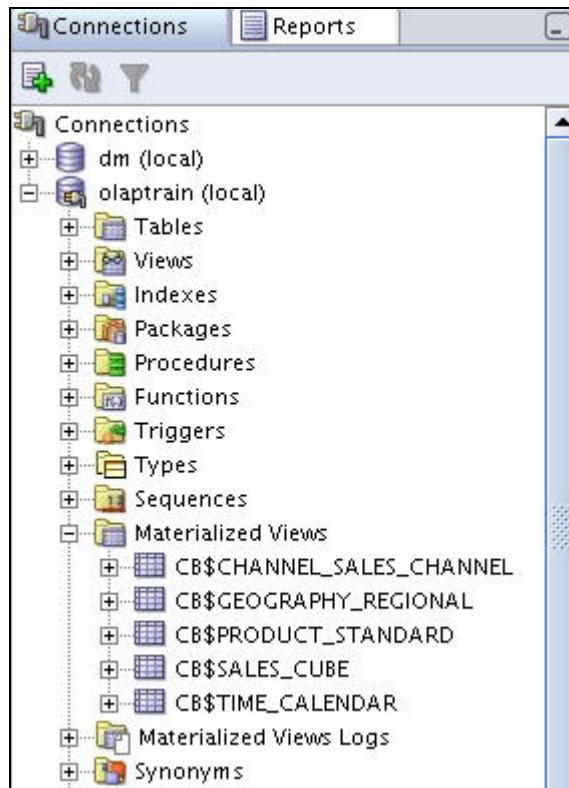
In this practice section, you run a set of summary queries against the OLAPTRAIN schema. First, you turn query rewrite off to see how the summary queries perform against the fact table. Then, you turn on rewrite and run the queries again.

- Observe how the database automatically rewrites the summary queries to the OLAP Cube MVs.
- Compare the performance of the SQL summary queries to the OLAP Cube MVs queries.

1. Launch **SQL Developer** from the desktop.
2. In the Connections tab of the navigator, click the **olaptrain** connection node.

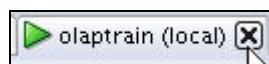
Result: A connection to the olaptrain schema is made, and schema object types appear in the navigator.

3. In the navigator, select the **Materialized Views** node.



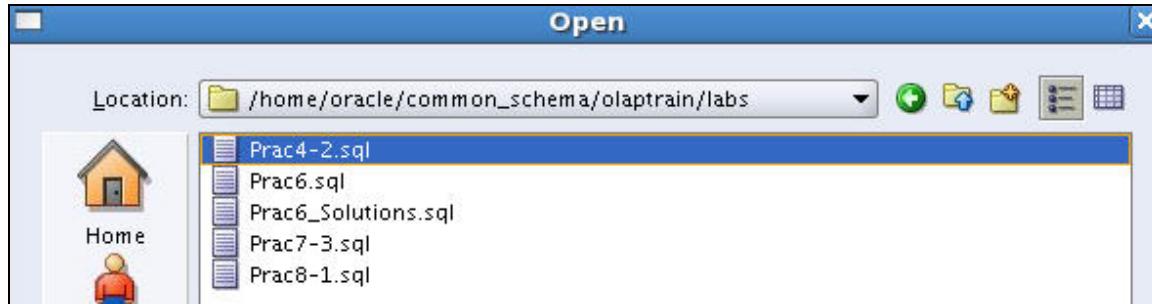
The CB\$ tables – one each for the cube and the associated dimension hierarchies – were automatically created by the OLAP build process.

4. Close the **olaptrain (local)** tab by clicking the X.

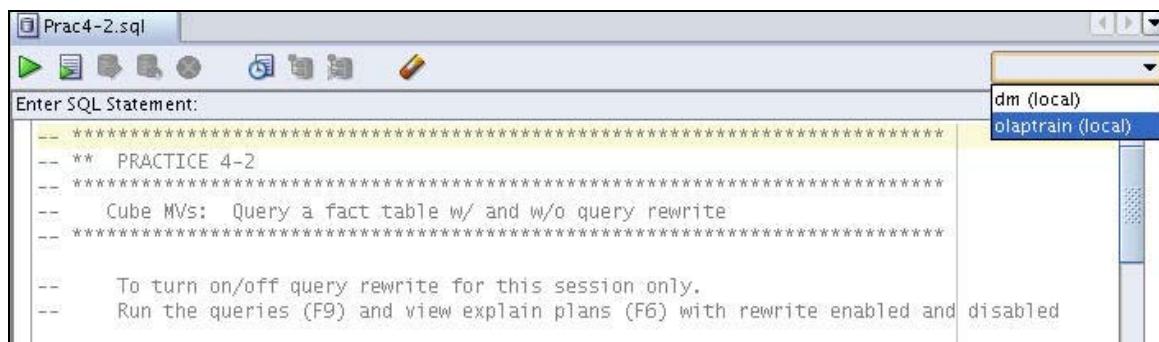


Next, you will see how these Cube MVs may be used for rewrite when summary queries are executed against the OLAPTRAIN fact and dimension tables.

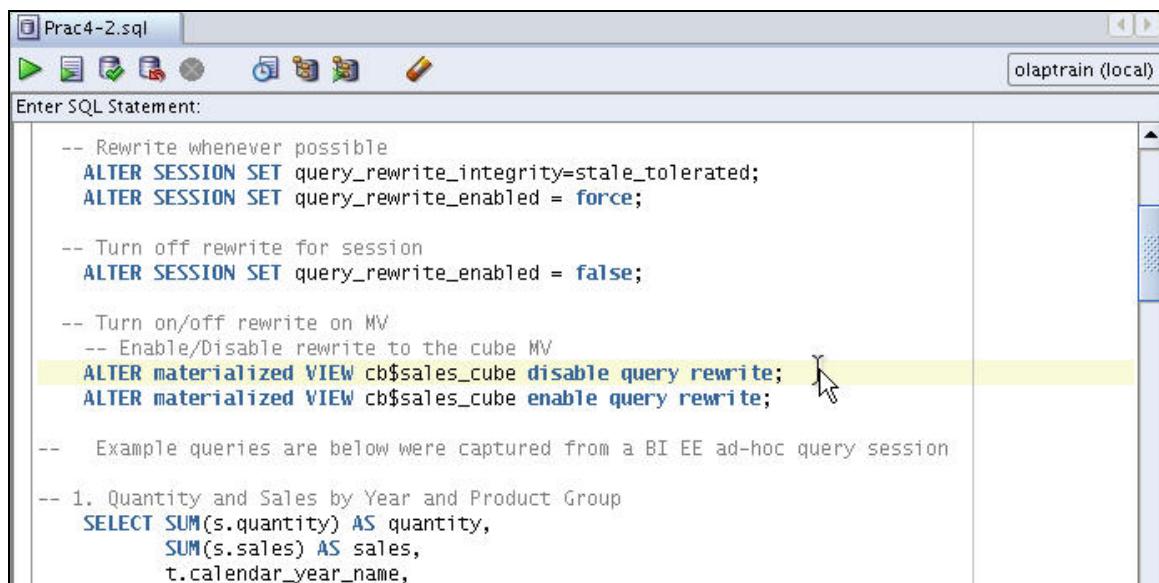
5. From the main menu, select **File > Open**.
 - a. Navigate to the **/home/oracle/common_schema/olaptrain/labs** folder.
 - b. Select the **Prac4-2.sql** file.



- c. Click **Open**.
6. Vertically resize the SQL statement pane so that it takes up at least half of the space in the SQL Developer window.
7. At the top-right corner of the Enter SQL Statement pane, select **olaptrain** from the list.



8. Scroll down until you reach the **ALTER materialized VIEW** statements. Click the statement that disables query rewrite for cb\$sales_cube.



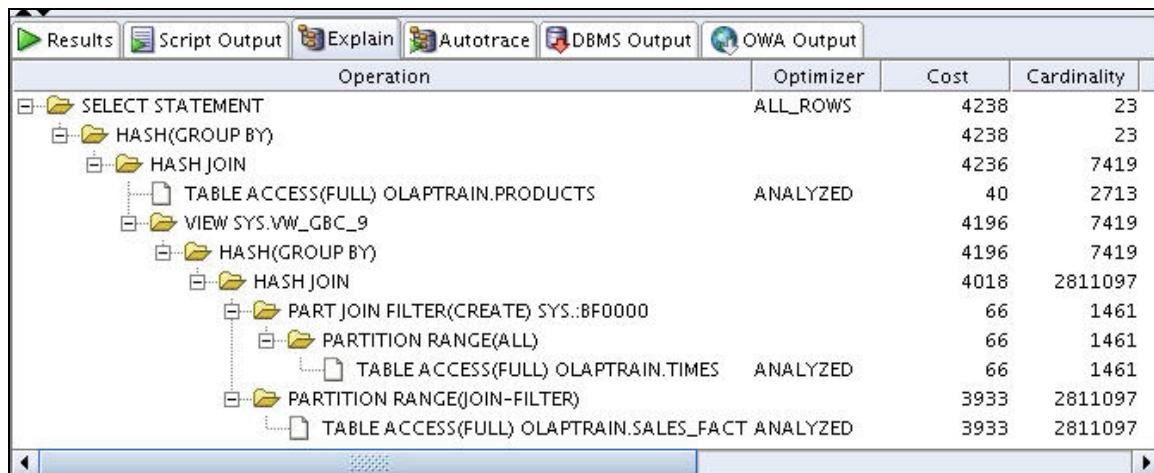
9. Press **F9** or click the  tool to execute the statement.

Result: Query rewrite to the Sales Cube MV is now disabled.

10. Place your cursor anywhere inside the first query, Quantity and Sales by Year and Product Group, which is just below the statements that turn query rewrite on and off.

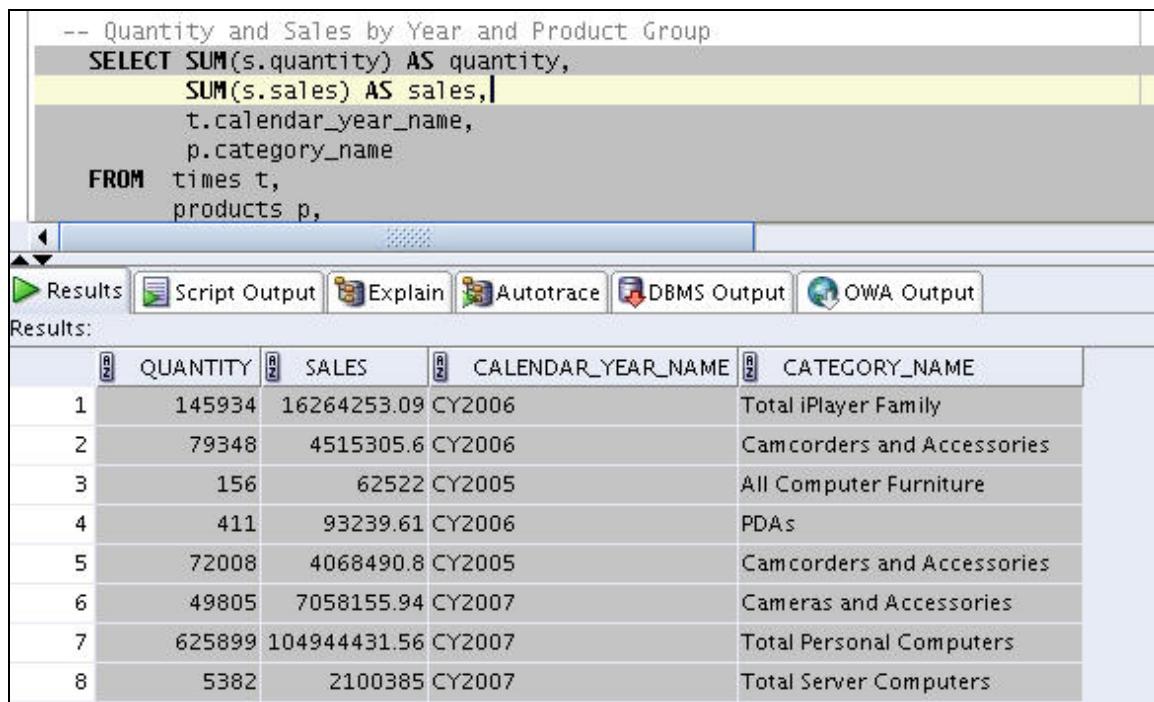
11. Press **F6** to display the Explain Plan for the query.

Result: When run, the query joins the fact table to associated dimension tables and performs a full table scan of the fact table in order to return data the requested data.



12. Press **F9** to execute the query.

Result: A report similar to the following example appears in the Results tab.



13. Record the time it took to run the query, up to two decimal places.

Note: The value appears at the top of the SQL pane, as shown in the following screenshot. In the example, the query time would be rounded to **2.54 seconds**. The value you see will probably be different from the example, although it should be close.



Query 1 performance: **seconds**

- Click inside the second query and press **F9** to execute the query, as the following screenshot shows. Again, record your query time.

```

sales_fact s
WHERE p.item_key = s.product
AND s.day_key = t.day_key
GROUP BY p.category_name, t.calendar_year_name;

-- 2. Quantity and Sales by Year, Department, Class and Country
SELECT SUM(s.quantity) AS quantity,
       SUM(s.sales) AS sales,
       t.calendar_year_name,
       p.department_name,
       c.class_name,
       cu.country_name
  FROM times t,
       products p,
       channels c,
       customers cu,
       sales_fact s
 WHERE p.item_key = s.product

```

	QUANTITY	SALES	CALENDAR_YEAR_NAME	DEPARTMENT_NAME	CLASS_NAME	COUNTRY_NAME
1	451	66510.46	CY2005	Computers	Indirect	El Salvador
2	190	18595.48	CY2005	Portable Music and Video	Direct	Congo
3	11874	971079.92	CY2005	Cameras and Camcorders	Direct	United States
4	9674	762122.88	CY2005	Cameras and Camcorders	Indirect	India
5	145	17334.66	CY2005	Portable Music and Video	Direct	Austria
6	1981	331317.06	CY2005	Computers	Indirect	Thailand
7	1282	106553.33	CY2005	Cameras and Camcorders	Indirect	Japan
8	1600	251525.77	CY2005	Cameras and Camcorders	Direct	Colombia

Query 2 performance: **seconds**

- Using the same technique as in steps 10-13, execute the third and fourth queries. Record each of the query times.

Query 3 performance: **seconds**

Query 4 performance: **seconds**

16. Scroll back up in the SQL file. Enable query rewrite to **cb\$sales_cube** by clicking the enable query rewrite statement, and then pressing **F9**.

```
Enter SQL Statement:
-- Turn on/off rewrite on MV
-- Enable/Disable rewrite to the cube MV
ALTER materialized VIEW cb$sales_cube disable query rewrite;
ALTER materialized VIEW cb$sales_cube enable query rewrite;
```

17. To confirm that the queries will rewrite, click within the first query and press **F6** to display the Explain Plan, as shown in the following screenshot:

```
-- 1. Quantity and Sales by Year and Product Group
SELECT SUM(s.quantity) AS quantity,
       SUM(s.sales) AS sales,
       t.calendar_year_name,
       p.category_name
  FROM times t,
       nproducts n
```

Operation	Optimizer	Cost	Cardinality	Byte
SELECT STATEMENT	ALL_ROWS	12	2859	21
HASH(GROUP BY)		12	2859	21
MAT_VIEW REWRITE CUBE ACCESS OLAPTRAIN.CB\$SALES_CUBE		11	2859	21

18. Press **F9** to execute the query. Record your performance to two decimal places.

```
Prac4-2.sql
0.18976898 seconds
olaptrain (local)

Enter SQL Statement:
ALTER SESSION SET query_rewrite_enabled = force;
-- Turn off rewrite for session
ALTER SESSION SET query_rewrite_enabled = false;

-- Turn on/off rewrite on MV
-- Enable/Disable rewrite to the cube MV
ALTER materialized VIEW cb$sales_cube disable query rewrite;
ALTER materialized VIEW cb$sales_cube enable query rewrite;

-- Example queries are below were captured from a BI EE ad-hoc query session

-- 1. Quantity and Sales by Year and Product Group
SELECT SUM(s.quantity) AS quantity,
       SUM(s.sales) AS sales,
       t.calendar_year_name,
       p.category_name
  FROM times t,
       nproducts n
```

QUANTITY	SALES	CALENDAR_YEAR_NAME	CATEGORY_NAME
145934	16264253.09	CY2006	Total iPlayer Family
79348	4515305.6	CY2006	Camcorders and Accessories
156	62522	CY2005	All Computer Furniture
411	93239.61	CY2006	PDAs
72008	4068490.8	CY2005	Camcorders and Accessories
49805	7058155.94	CY2007	Cameras and Accessories

Query 1 – Cube MV performance: **seconds**

19. Using the same technique as in step 18, execute the second through fourth queries. Record each of the query times.

Query 2 – Cube MV performance: **seconds**

Query 3 – Cube MV performance: **seconds**

Query 4 – Cube MV performance: **seconds**

Note: Query times for a Cube MV rewrite commonly are 10 to 50 times faster than summary queries against relational fact tables.

20. **Close** the Prac4_2.sql file.

21. **Exit** SQL Developer.

Practices for Lesson 5

In the practice for this lesson, you create a range of calculations, including:

- Calculated measures using the Calculation Builder
- Calculated measures using XML templates
- Custom calculations using expression syntax

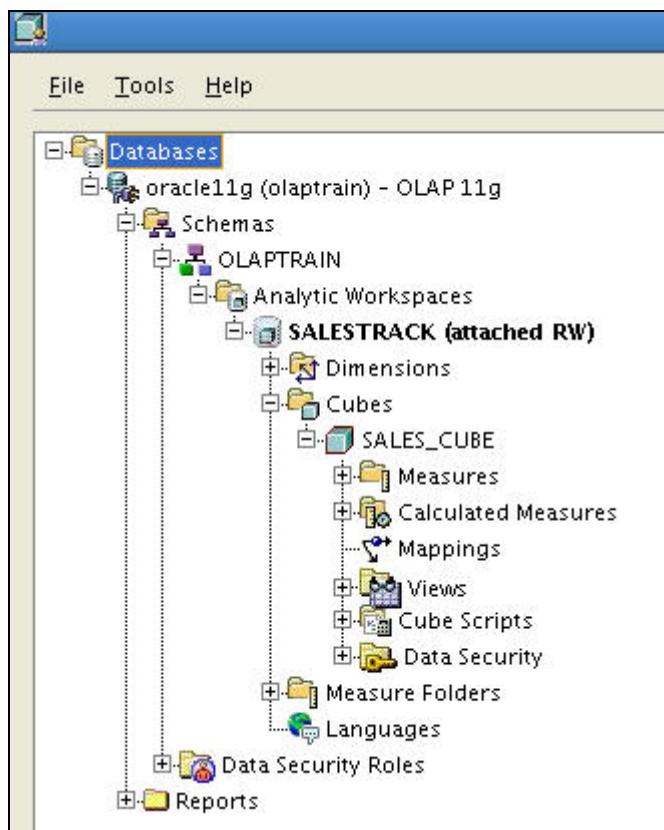
Practice 5-1: Creating Calculations Using the Calculation Builder

In this practice section, you create several calculated measures using the AWM Calculation Builder. These calculations include:

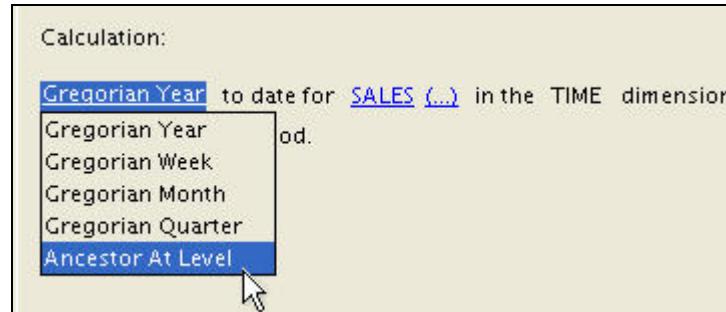
- Sales Year-To-Date (YTD)
- Sales YTD Prior Year
- Sales YTD Prior Year Percent Change
- Sales Prior Year

Follow these steps to create the YTD calculations:

1. Open AWM, connect to the database using the id/password combination **olaptrain/oracle**, and then open your analytic workspace by clicking the plus sign (+) next to **SALESTRACK**.
2. In the AWM navigator, select **Cubes > SALES_CUBE** as the following screenshot shows.



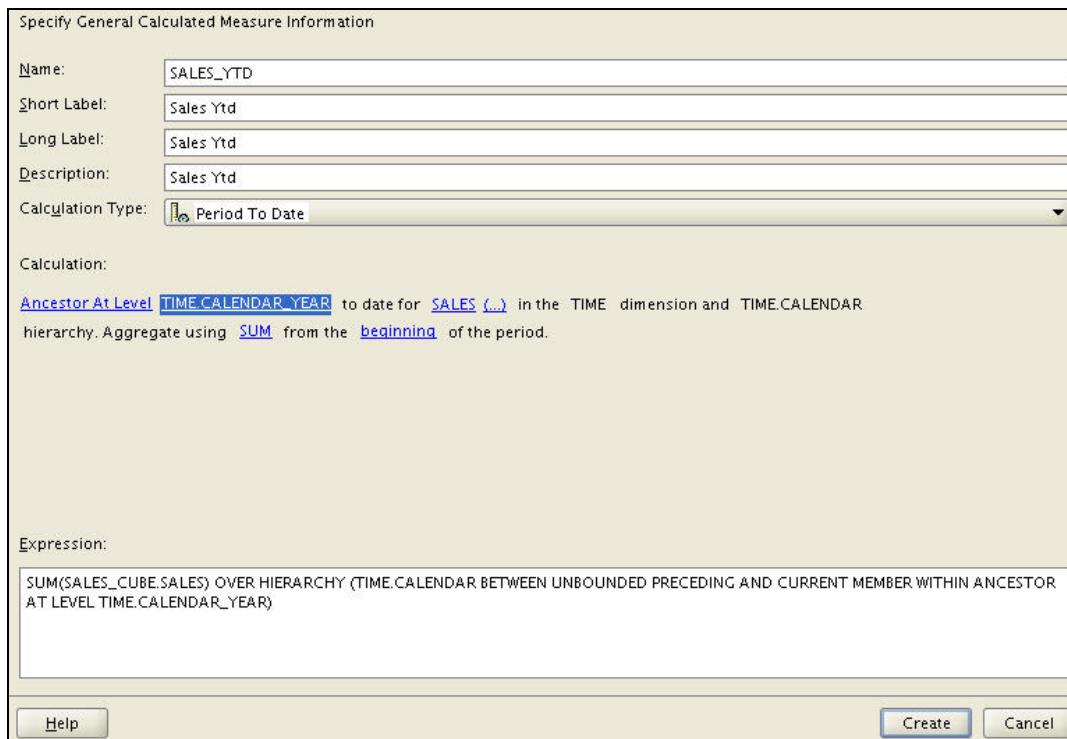
3. Right-click the Calculated Measures node and select **Create Calculated Measure** from the menu.
4. In the Create Calculated Measure dialog box, enter or select the following:
 - a. Name = **SALES_YTD** (the Name field is automatically all caps, and the Label and Description fields are autofilled)
 - b. Calculation Type = **Period to Date**
 - c. In the Calculation inputs section, select the following:
 - i) First hyperlink = **Ancestor At Level**



Result: A new hyperlink appears next to the Ancestor At Level hyperlink.

- ii) Second hyperlink = **TIME.CALENDAR_YEAR**

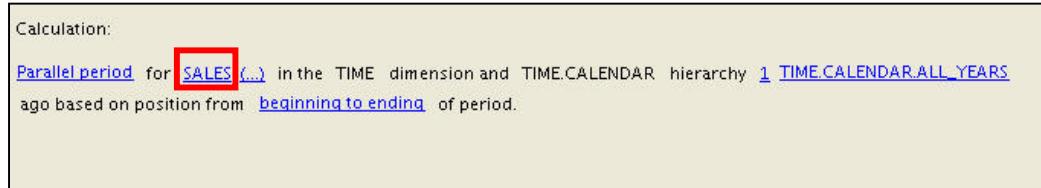
Result: The Create Calculated Measure dialog box should now look like this:



- d. Click **Create**.
5. Select the **Calculated Measures** node.

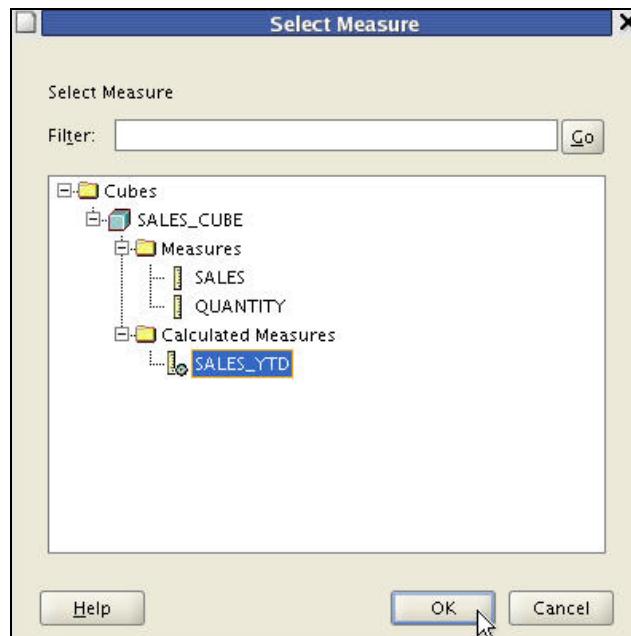
Result: The SALES_YTD calculation appears.

6. Create a second YTD calculation:
 - a. Right-click Calculated Measures and select **Create Calculated Measure** from the menu.
 - b. In the Create Calculated Measure dialog box, enter or select the following:
 - i) Name = **SALES_YTD_PY**
 - ii) All Label and Description boxes = **Sales Ytd Pr Year**
 - iii) Calculation Type = **Parallel Period**
 - iv) In the Calculation inputs section, click the SALES hyperlink.



Result: The Select Measure dialog box appears.

- v) Select **SALES_YTD**, and then click **OK**.



Result: The Calculation updates with the selected measure.

- vi) Click the **TIME.CALENDAR.ALL_YEARS** hyperlink and select **TIME.CALENDAR.CALENDAR_YEAR** from the list.



- c. Click **Create**.

Result: The SALES_YTD_PY calculation appears below the Calculated Measures node in the Navigator.

7. Create a third Year-to-Date calculation.
 - a. Right-click Calculated Measures and select **Create Calculated Measure** from the menu.
 - b. In the Create Calculated Measure dialog box, enter or select the following:
 - i) Name = **SALES_YTD_PY_PCT_CHG**
 - ii) All Label and Description boxes = **Sales Ytd Pr Yr Pct Chg**
 - iii) Calculation Type = **Percent Difference From Parallel Period**
 - iv) In the Calculation inputs section, click the SALES hyperlink.
 - v) In the Select Measure window, select **SALES_YTD** and click **OK**.
 - vi) Click the TIME.CALENDAR.ALL_YEARS hyperlink and select **TIME.CALENDAR.CALENDAR_YEAR** from the list.

The calculation should now look like this:

Name:	SALES_YTD_PY_PCT_CHG
Short Label:	Sales YTD Pr Yr Pct Chg
Long Label:	Sales YTD Pr Yr Pct Chg
Description:	Sales YTD Pr Yr Pct Chg
Calculation Type:	Percent Difference From Parallel Period
Calculation:	
Percent difference from <u>Parallel period</u> for <u>SALES_YTD</u> in the TIME dimension and TIME.CALENDAR hierarchy <u>1 TIME.CALENDAR.CALENDAR_YEAR</u> ago based on position from <u>beginning to ending</u> of period.	

- ii) In the Expression field, add the following syntax to the beginning of the expression: **100 ***

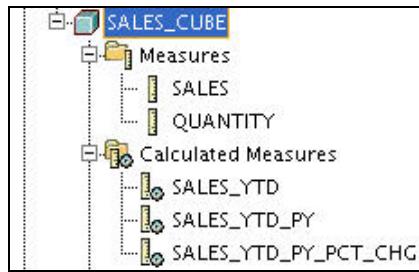
Note: This additional syntax causes the calculation to display percentage figures in whole numbers.

The expression should now look like this:

Expression:	100 * LAG_VARIANCE_PERCENT(SALES_CUBE.SALES_YTD,1) OVER HIERARCHY (TIME.CALENDAR BY ANCESTOR AT LEVEL TIME.CALENDAR.CALENDAR_YEAR POSITION FROM BEGINNING)
-------------	--

- c. Click **Create**.

The Sales Cube now contains the following measures:



Practice 5-2: Creating Calculations Using XML Templates

In this practice section, you create a number of calculated measures using XML templates and a few more using the Calculation Wizard, including the following:

From XML Templates

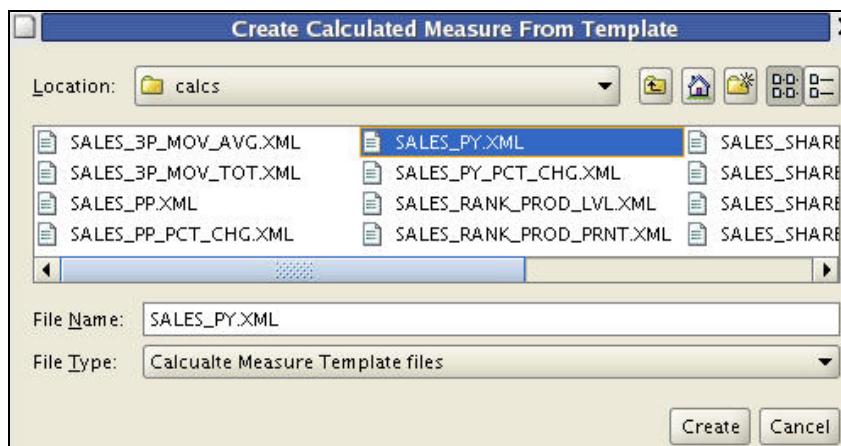
- Sales Prior Year
- Sales Prior Year Percent Change
- Sales Prior Period
- Sales Prior Period Percent Change
- Sales Share of Parent – Channel
- Sales Share of Total – Product
- Sales Share of Parent – Product
- Sales Rank By Product Parent
- Sales Share of Total – Channel
- Sales Moving Total – 3 Periods

Using the Calculation Wizard

- Sales Share of Total – Channel
- Sales Rank By Product Level
- Sales Moving Average – 3 Periods

Use the following steps:

1. Right-click Calculated Measures and select **Create Calculated Measure from Template**.
2. Navigate to the ...olaptrain/templates/calcs directory and select **SALES_PY.XML**.



3. Click **Create**.

Result: The calculation appears in the navigator.

4. Select the new calculation.

Result: The calculation definition appears in the right pane.

Name:	SALES_PY
ID:	OLAPTRAIN.SALES_CUBE.SALES_PY
Short Label:	Sales Pr Year
Long Label:	Sales Pr Year
Description:	Sales Pr Year
Calculation Type:	Parallel Period
Calculation:	<p>Parallel period for SALES (...) in the TIME dimension and TIME.CALENDAR hierarchy 1 TIME.CALENDAR.CALENDAR_YEAR ago based on position from beginning to ending of period.</p>
Expression:	LAG(SALES_CUBE.SALES,1) OVER HIERARCHY (TIME.CALENDAR BY ANCESTOR AT LEVEL TIME.CALENDAR.CALENDAR_YEAR POSITION FROM BEGINNING)

5. Repeat steps 1-3 to create three more calculations, using the following XML files:
[SALES_PY_PCT_CHG.XML](#), [SALES_PP.XML](#), and
[SALES_PP_PCT_CHG.XML](#).
6. Select the **Calculated Measures** node in the navigator to display the list of calculated measures in the cube.

Calculated Measures:	
Name	Long Description
SALES_YTD	Sales Ytd
SALES_YTD_PY	Sales Ytd Pr Year
SALES_YTD_PY_PCT_CHG	Sales Ytd Pr Yr Pct Chg
SALES_PY	Sales Pr Year
SALES_PY_PCT_CHG	Sales Pr Year Pct Chg
SALES_PP	Sales Pr Period
SALES_PP_PCT_CHG	Sales Pr Period Pct Chg

7. Create a Share measure for the Channel dimension. You may either use the Calculation Builder or an XML template.

Note: If using the Calculation Builder, follow steps 7a – 7b. If using an XML template, select the SALES_SHARE_TOT_CHAN.XML file.

- a. Right-click Calculated Measures and select **Create Calculated Measure** from the menu.
- b. In the Create Calculated Measure dialog box, enter or select the following:

- i) Name = **SALES_SHARE_TOT_CHAN**
- ii) Leave all Label and Description boxes with the autofilled text.
- iii) Calculation Type = **Share**
- iv) Click the dimension hyperlink (TIME) and select **CHANNEL** from the list.
- v) Select the **Multiply result by 100** option.

The calculation should look like this:

Name:	SALES_SHARE_TOT_CHAN
Short Label:	Sales Share Tot Chan
Long Label:	Sales Share Tot Chan
Description:	Sales Share Tot Chan
Calculation Type:	<input checked="" type="checkbox"/> Share
Calculation: Share of measure SALES (...) in CHANNEL.SALES_CHANNEL hierarchy of the CHANNEL dimension as a ratio of top of hierarchy .	
<input checked="" type="checkbox"/> Multiply result by 100.	
Expression: <code>(SHARE(SALES_CUBE.SALES OF CHANNEL.SALES_CHANNEL TOP)) * 100</code>	

Note: This calculation returns the share of sales for the currently selected channel of distribution to the total for all Channels (top of hierarchy).

- i) Click **Create**.
8. Repeat steps 1-3 to create three more calculations using the following XML files: **SALES_SHARE_PRNT_CHAN**, **SALES_SHARE_TOT_PROD**, and **SALES_SHARE_PRNT_PROD**.
9. Create a Rank measure for the Product dimension. You may either use the Calculation Builder or an XML template.

Note: If using the Calculation Builder, follow steps 9a – 9b. If using an XML template, select the **SALES_RANK_PROD_LVL.XML** file.

 - a. Right-click Calculated Measures and select **Create Calculated Measure** from the menu.
 - b. In the Create Calculated Measure dialog box, enter or select the following:
 - i) Name = **SALES_RANK_PROD_LVL**
 - ii) All Label and Description boxes = **Sales Rank in Prod Lvl**
 - iii) Calculation Type = **Rank**
 - iv) In the Calculation inputs section, click the dimension hyperlink (TIME) and select **PRODUCT** from the list.

The calculation should look like this:

Name:	SALES_RANK_PROD_LVL
Short Label:	Sales Rank In Prod Lvl
Long Label:	Sales Rank In Prod Lvl
Description:	Sales Rank In Prod Lvl
Calculation Type:	<input checked="" type="checkbox"/> Rank
<p>Calculation:</p> <p>Rank members of the PRODUCT dimension and PRODUCT.STANDARD hierarchy based on measure SALES (...) . Calculate rank using RANK method with member's level in order highest to lowest . Rank NA(null) values nulls last .</p>	
<p>Expression:</p> <pre>RANK() OVER HIERARCHY (PRODUCT.STANDARD ORDER BY SALES_CUBE.SALES DESC NULLS LAST WITHIN LEVEL)</pre>	

iv) Click **Create**.

10. Finally, repeat steps 1-3 to create two more calculations using the following XML files: **SALES_RANK_PROD_PRNT.XML** and **SALES_3P_MOV_AVG.XML**.

When you are done, the following calculations should be part of the Sales Cube:

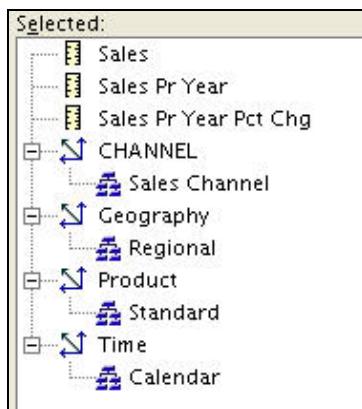
Calculated Measures:	
Name	Long Description
SALES_YTD	Sales Ytd
SALES_YTD_PY	Sales YTD Pr Year
SALES_YTD_PY_PCT_CHG	Sales YTD Pr Yr Pct Chg
SALES_PY	Sales Pr Year
SALES_PY_PCT_CHG	Sales Pr Year Pct Chg
SALES_PP	Sales Pr Period
SALES_PP_PCT_CHG	Sales Pr Period Pct Chg
SALES_SHARE_TOT_CHAN	Sales Share Tot Chan
SALES_SHARE_PRNT_CHAN	Sales Share Prnt Chan
SALES_SHARE_TOT_PROD	Sales Share Tot Prod
SALES_SHARE_PRNT_PROD	Sales Share Prnt Prod
SALES_RANK_PROD_LVL	Sales Rank in Prod Lvl
SALES_RANK_PROD_PRNT	Sales Rank in Prod Prnt
SALES_3P_MOV_AVG	Sales 3 Per Mov Avg

11. Feel free to examine any of the definitions for the calculations that you created from XML template files.

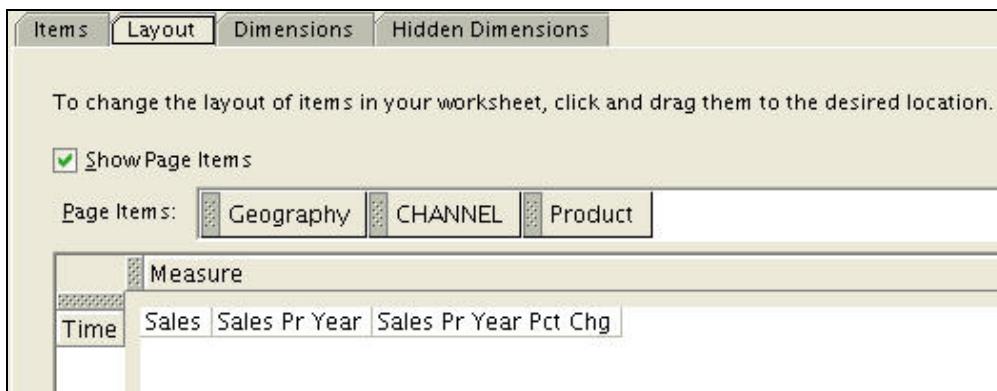
Practice 5-3: Viewing Calculated Measures

In this practice section, you use the AWM Data Viewer to see calculated measures.

- First, you will create a comparative time-series report on Product categories, using the Prior Year, and Prior Year Percent Change measures.
 - Second, you will create a Product analysis report using Rank and Share measures.
1. In the Navigator, right-click the SALES measure and select **View Data Sales** from the menu. Sales data is displayed for a default set of dimension members.
 2. Remove the Graph by clicking the down arrow of the Hide/Show tool. (▲▼)
 3. Click the **Query Builder** tool  , under the File menu.
 4. In the Items tab:
 - a. Select **Sales Pr Year** and **Sales Pr Year Pct Chg**.
 - b. Click the **Add Selected Items** tool  to move those two measures to the Selected list, like this:

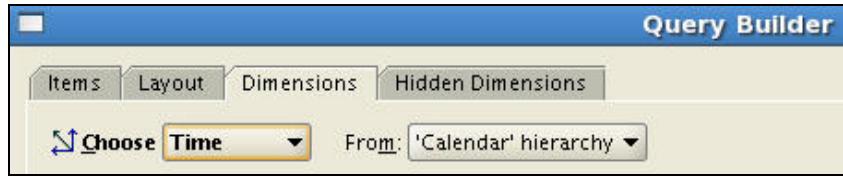


5. Click the **Layout** tab.
6. In the Layout tab, drag the appropriate dimension tiles to the correct axis so that the layout looks like this:

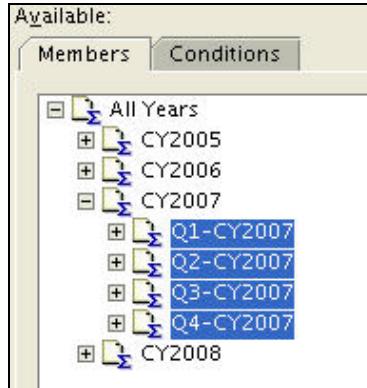


7. Click the **Dimensions** tab.
8. In the Dimensions tab, perform the following:

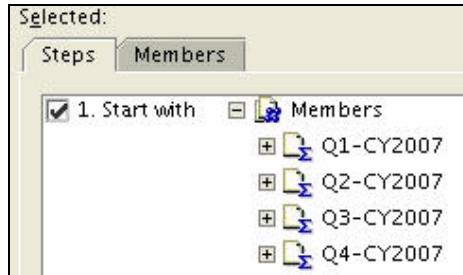
- a. Select the **Time** dimension from the Choose list.



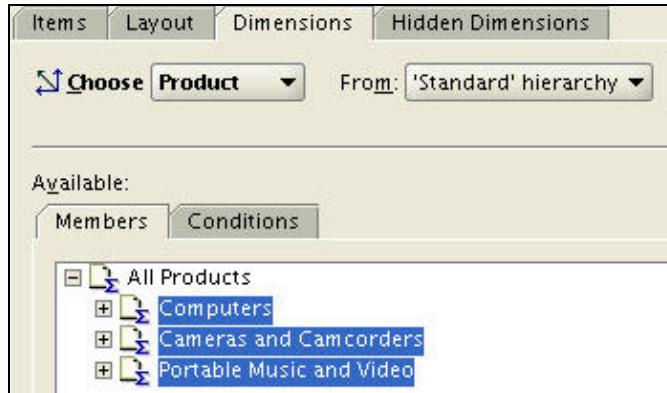
- b. Click the **Remove All Items** tool to clear the Selected list.
 c. In the Members tab of the Available list, select **All Years > CY2007**.
 d. Select all four quarters in CY2007, like this:



- e. Click the **Add Items** tool to move the 2007 quarters to the Selected list, like this:



- f. Select **Product** from the Choose list.
 g. Click the **Remove All Items** tool and then select **All Products** in the Available list. Select the three Product department members, like this:



- h. Click the **Add Items** tool to move the department members to the Selected list.

- i. Click **OK** to view the data.
9. Select any of the members from the Product dimension header. The calculations are instantaneously updated.

	Sales	Sales Pr Year	Sales Pr Peri	Product
► Q1-CY2007	33,777,199.32	28,073,254.88		Computers
► Q2-CY2007	28,581,026.01	24,191,092.73	-15.38	Cameras and Camcorders
► Q3-CY2007	30,982,913.01	28,279,705.99	8.40	Portable Music and Video
► Q4-CY2007	34,565,476.88	31,579,754.49	11.56	

10. Select any of the 2007 Quarter values to view the data at the month level.

Next, you modify the Data Viewer for a Product ranking and share report.

11. Click the **Query Builder** tool.

12. In the Items tab, perform the following:

- a. In the Selected list, select **Sales Pr Year** and **Sales Pr Year Pet Chg** measures, and click the **Remove Selected Items** tool .
- b. In the Available list, select **Sales Rank In Prod Prnt** and **Sales Share Prnt Prod**, and click the Add Selected Items tool.

Result: The rank and share measures are added to the Selected list.

13. In the Layout tab, swap the Product and Time dimensions, so that Product is in the Row axis, and Time is in the Page Items axis.
14. Click **OK** to view the data.

	Sales	Sales Rank By Prod Prnt	Sales Share Prnt Prod
► Computers	33,777,199.32	1.00	81.52
► Cameras and Camcorders	2,961,770.79	3.00	7.15
► Portable Music and Video	4,692,772.30	2.00	11.33

The calculated measures show the rank of each Product dimension member within its hierarchy parent. The share measure shows the percentage of sales returned by the product member to its parent in the hierarchy.

15. Select Computers.

Page Items	Geography All Regions	CHANNEL All Channels	Time Q1-CY2007	
	Sales	Sales Rank By Prod Prnt	Sales Share Prnt Prod	
▼ Computers	33,777,199.32	1.00	81.52	
▶ Total Personal Computers	27,723,678.35	1.00	82.08	
▶ PDAs	26,675.15	4.00	0.08	
▶ All Computer Furniture	21,313.40	5.00	0.06	
▶ Computer Printers and Supplies	5,434,375.02	2.00	16.09	
▶ Total Server Computers	571,157.40	3.00	1.69	
▶ Cameras and Camcorders	2,961,770.79	3.00	7.15	
▶ Portable Music and Video	4,692,772.30	2.00	11.33	

The rank and share measures show the relative ranking and share contribution for each of the Product division members in the Computers department.

16. Select any Time member from the Page Items axis, and the calculations update instantaneously.
17. Feel free to modify the report by selecting other dimension members. When you are done, close the Measure Data Viewer.

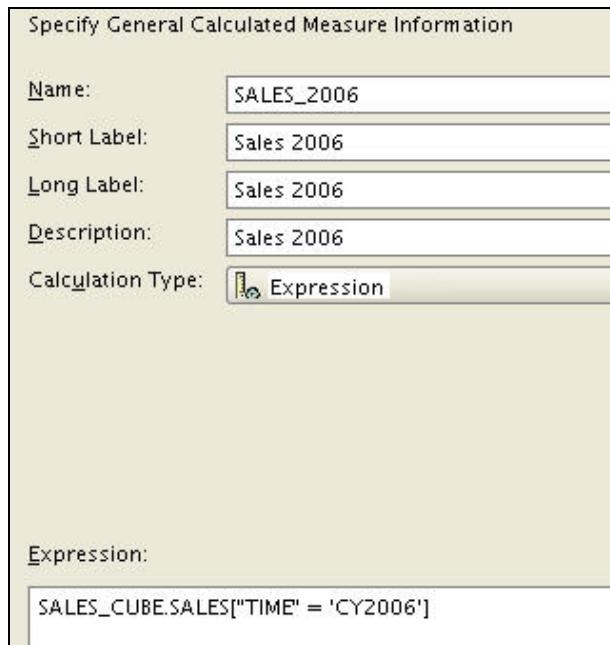
Practice 5-4: Creating and Using Custom Calculations

In this practice section, you create and use a number of custom calculations that demonstrate the following:

- Expression syntax using QDRs
 - **Sales 2006**
 - **To Go** (Sales for 2006 – Sales YTD)
 - **PCT of 2006**
- Conditional expression syntax
 - **How is Sales YTD** (a CASE Statement example)
- OLAP DML code example
 - **Product Alert**

Practice 5-4a: Create three custom calculations that illustrate the use of a QDR (qualified data reference).

1. In the navigator, right-click Calculated Measures and select **Create Calculated Measure** from the menu.
2. In the Create Calculated Measure dialog box, enter or select the following:
 - a. Name = **SALES_2006**
 - d. Leave all Label and Description boxes as autofilled
 - e. Calculation Type = **Expression**
 - f. In the Expression box, enter **SALES_CUBE.SALES["TIME" = 'CY2006']**

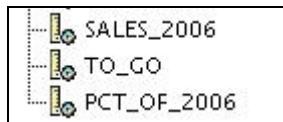


Note: The braces for the QDR are square “[“ and “].”

- g. Click **Create**.
3. Open the Create Calculated Measure dialog box. Enter or select the following:

- a. Name = **TO_GO**
 - b. Leave all Label and Description boxes as autofilled
 - c. Calculation Type = **Expression**
 - d. In the Expression box, enter:
SALES_CUBE.SALES_2006 - SALES_CUBE.SALES_YTD
 - e. Click **Create**.
4. Open the Create Calculated Measure dialog box and enter or select the following:
 - a. Name = **PCT_OF_2006** (leave all Label and Description boxes as autofilled.)
 - b. Calculation Type = **Expression**
 - c. In the Expression box, enter:
100 * (SALES_CUBE.SALES_YTD / SALES_CUBE.SALES_2006)
 - d. Click **Create**.

The three Expression calculations are now added to the list of available calculated measures:



Now, use these three calculations in the Data Viewer by performing the following:

5. In the Navigator, right-click the SALES measure and select **View Data Sales** from the menu.
6. Remove the Graph by clicking the down arrow of the Hide/Show tool. ()
7. Click the **Query Builder** tool.
8. In the Items tab, add the following measures (in the order specified) from the Available list to the Selected list: **Sales YTD**, **Sales 2006**, **To Go**, and **Pct of 2006**.
9. Using the Layout tab, place Geography, Channel, and Product on the Page Items axis, and Time on the Row axis.
10. In the Dimension tab, select the four quarters of CY2007, using the same techniques described previously in this practice.
11. Click **OK** to view the data.

The screenshot shows a report interface with a toolbar at the top and three dropdown menus: 'Geography All Regions', 'CHANNEL All Channels', and 'Product All Products'. Below these are three columns of data:

	Sales	Sales Ytd	Sales 2006	To Go	Pct Of 2006
► Q1-CY2007	41,431,742.41	41,431,742.41	138,960,159.40	97,528,416.99	29.82
► Q2-CY2007	35,270,329.75	76,702,072.16	138,960,159.40	62,258,087.24	55.20
► Q3-CY2007	38,488,015.18	115,190,087.34	138,960,159.40	23,770,072.06	82.89
► Q4-CY2007	43,029,012.11	158,219,099.45	138,960,159.40	-19,258,940.05	113.86

The report dynamically compares several sales measures for a selected time period to a different “benchmark” sales value for the year 2006.

Notice how the:

- “To Go” calculation shows the Sales revenue required to meet CY2006 Sales
- “Pct of 2006” measure shows the percent of YTD sales against the benchmark Sales_2006 figures

Practice 5-4b: Now, create a custom calculation that includes the use of conditional expression syntax.

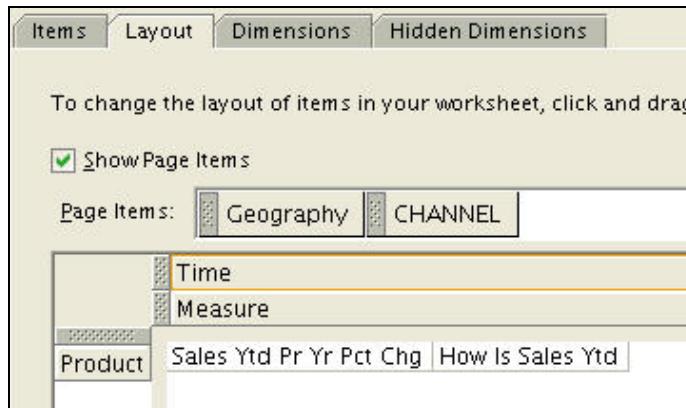
1. Open the Create Calculated Measure dialog box. Enter or select the following:

- a. Name = **HOW_IS_SALES_YTD**
- b. Leave all Label and Description boxes as autofilled
- c. Calculation Type = **Expression**
- d. In the Expression box, enter the following CASE statement:

```
CASE
WHEN SALES_CUBE.SALES_YTD_PY_PCT_CHG < 0 THEN 'Needs Improvement'
WHEN SALES_CUBE.SALES_YTD_PY_PCT_CHG > 15 THEN 'Outstanding'
ELSE 'On track'
END
```

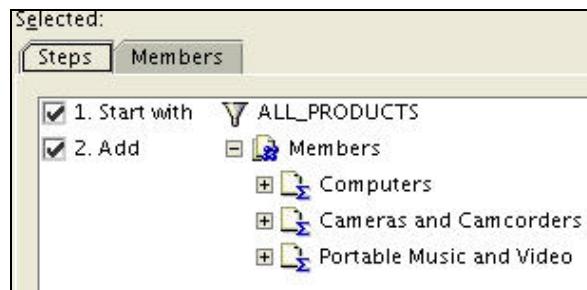
- e. Click **Create**.
2. In the Navigator, right-click the SALES_YTD_PY_PCT_CHG measure and select **View Data SALES_YTD_PY_PCT_CHG** from the menu.
3. Remove the Graph by clicking the down arrow of the Hide/Show tool.
4. Click the **Query Builder** tool.
5. In the Items tab, add **How Is Sales Ytd** to the Selected list.

6. In the Layout tab, place Time above the measures in the Column axis, like this:



7. In the Dimensions tab, select the **Time** dimension from the Choose list, and then select the four quarters of CY2007, using the same techniques described previously in this practice.
8. In the Dimensions tab, select the **Product** dimension from the Choose list. Then do the following:
- In the Available list, select **All Products**.
 - Select the three Product departments and click the **Add Selected Items** button.

The Selected list should look like this:



9. Click **OK** to view the data.

10. Select the Measure headings and select the **Wrap Text** tool . Then, resize the column edges, like this:

	Q1-CY2007		Q2-CY2007		Q3-CY2007	
	Sales Ytd Pr Yr Pct Chg	How Is Sales Ytd	Sales Ytd Pr Yr Pct Chg	How Is Sales Ytd	Sales Ytd Pr Yr Pct Chg	How Is Sales Ytd
All Products	17.39	Outstanding	17.14	Outstanding	14.98	
Computers	20.32	Outstanding	19.31	Outstanding	15.89	
Cameras and Camcorders	1.09	On Track	6.91	On Track	14.11	
Portable Music and Video	9.38	On Track	9.62	On Track	9.37	

11. Finally, drill down on Cameras and Camcorders.

The report should look something like this: (**Hint:** Resize the Measure Data Viewer.)

	Q1-CY2007		Q2-CY2007		Q3-CY2007		Q4-CY2007	
	Sales Ytd Pr Yr Pct Chg	How Is Sales Ytd	Sales Ytd Pr Yr Pct Chg	How Is Sales Ytd	Sales Ytd Pr Yr Pct Chg	How Is Sales Ytd	Sales Ytd Pr Yr Pct Chg	How Is Sales Ytd
All Products	17.39	Outstanding	17.14	Outstanding	14.98	On Track	13.86	On Track
Computers	20.32	Outstanding	19.31	Outstanding	15.89	Outstanding	14.08	On Track
Cameras and Camcorders	1.09	On Track	6.91	On Track	14.11	On Track	17.23	Outstanding
Cameras and Accessories	-8.36	Needs Improvement	-0.63	Needs Improvement	9.93	On Track	16.53	Outstanding
Camcorders and Accessories	14.66	On Track	17.56	Outstanding	19.97	Outstanding	18.16	Outstanding
Portable Music and Video	9.38	On Track	9.62	On Track	9.37	On Track	10.18	On Track

12. Close the Data Viewer.

Practice 5-4c: Finally, create a custom calculation that calls an OLAP DML program.

In order to create the final custom calculation, you must first have access to the program that it calls. Therefore, use the following instructions to import an OLAP DML program named product_alert.dml:

1. Select **Tools > OLAP Worksheet**.
2. In the command window, enter the following command (and then press **F5**):

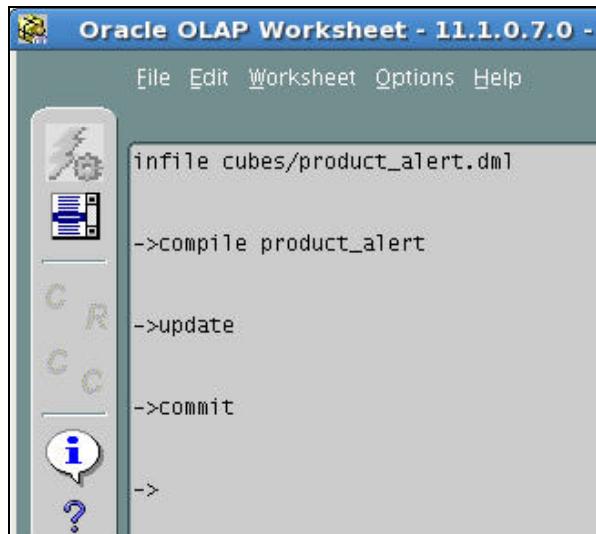
```
infile cubes/product_alert.dml
```

Result: The predefined OLAP DML program is imported into the AW.

3. Execute the following three commands in the command window:

```
compile product_alert  
update  
commit
```

When done, the OLAP Worksheet should look like this:



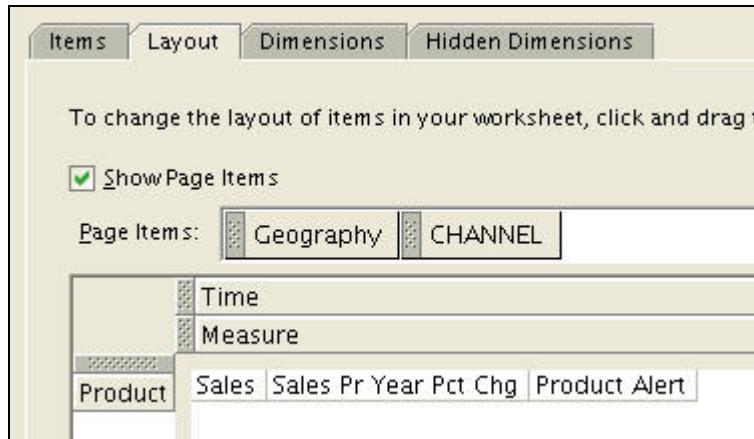
4. Select **File > Close** to close the OLAP Worksheet window.

Note: This is exactly the same DML program that was covered in the lesson.

Now, create and use a calculation that calls the OLAP DML program.

5. Open the Create Calculated Measure dialog box. Enter or select the following
 - a. Name = **PRODUCT_ALERT**
 - b. Leave all Label and Description boxes as auto-filled.
 - c. Calculation Type = **Expression**
 - d. In the Expression box, enter the following syntax:
OLAP_DML_EXPRESSION('PRODUCT_ALERT', VARCHAR2)
 - e. Click **Create**.

6. In the navigator, right-click the SALES measure and select **View Data SALES** from the menu.
7. Remove the Graph by clicking the down arrow of the Hide/Show tool.
8. Click the **Query Builder** tool.
9. In the Items tab, add **Sales Pr Year Pct Chg** and **Product Alert** to the Selected list.
10. In the Layout tab, create the following layout:



11. In the Dimension tab, select the Time dimension, and then select **CY2007** as the only time dimension member. The Selected pane should look like this:



12. Click **OK** to view the data.

13. Select **All Products**.

Page Items	Geography All Regions	CHANNEL All Channels	
CY2007			
	Sales	Sales Pr Year Pct Chg	Product Alert
▼ All Products	158,219,099.45	13.86	OKAY
► Computers	127,906,615.22	14.08	ALERT
► Cameras and Camcorders	12,393,248.54	17.23	OKAY
► Portable Music and Video	17,919,235.69	10.18	OKAY

Note:

An Alert is raised on Computers, even though it is by far the largest selling product department, and also shows good growth (a good percent change from last year).

This means that there is a hidden sales problem somewhere in the Computers department.

14. Select Computers.

Page Items	Geography All Regions	CHANNEL All Channels	CY2007		
			Sales	Sales Pr Year Pct Chg	Product Alert
▼ All Products			158,219,099.45	13.86	OKAY
▼ Computers			127,906,615.22	14.08	ALERT
▶ Total Personal Computers			104,944,431.56	14.47	ALERT
▶ PDAs			101,464.67	8.82	OKAY
▶ All Computer Furniture			66,635.00	-8.19	ALERT
▶ Computer Printers and Supplies			20,693,698.99	12.84	OKAY
▶ Total Server Computers			2,100,385.00	8.37	OKAY
▶ Cameras and Camcorders			12,393,248.54	17.23	OKAY
▶ Portable Music and Video			17,919,235.69	10.18	OKAY

Note:

As indicated by the Alert, at least one of the computer categories had a sales drop-off. The computer furniture category has had an 8.19 percent drop-off as compared to the previous year.

In addition, an Alert is raised on the Total Personal Computers category, indicating further investigation is required.

- Of all of the Personal Computer product types that were alerted, how many seemed obvious by only viewing the Sales Prior Year Percent Change measure? Which one(s)?
- On which product members in the Personal Computer category has an alert been raised?

15. After you have finished your investigations, close the Data Viewer.

16. Exit AWM.

Practices for Lesson 6

In the practices for this lesson, you use SQL Developer to query OLAP cubes directly using the OLAP cube views. You apply techniques that leverage the unique properties of OLAP cubes, including:

- Level and Member filters
- Embedded Total features
- Hierarchical (“Family”) relationships
- Parent columns in order to “drill down”

Practice 6-1: Creating a Simple OLAP Cube Query

Write a query that returns total **SALES** for products at the **DEPARTMENT** level.

Hints

- Use the Views node in SQL Developer’s Connection navigator to examine the OLAP cube views, column names, and column values for your query. (You can view the data values for any column by double-clicking the desired view in the Connection navigator, and then selecting the Data tab.)
- Ensure that all dimensions include a filter, even if they are not in the select statement. Therefore, Channel, Geography, and Time should be filtered at the “ALL” level.
- Use the *LEVELNAME_LONG_DESCRIPTION* column for the product dimension. (Notice that some column names are truncated by SQL Developer.)
- In SQL Developer, drag column names from the navigator to the SQL Statement window and drop them into the query.

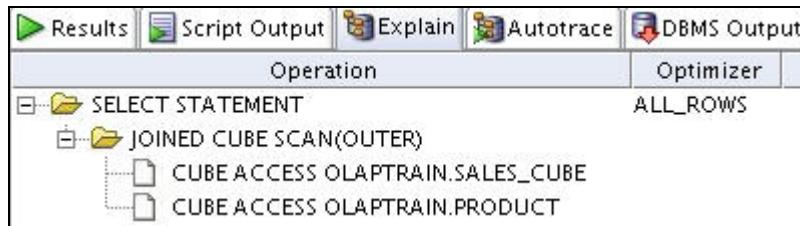
Note: Solutions for each of the following lesson 6 practices are in *Prac6_solutions.sql*. Try to complete the query on your own before examining the solution.

1. In SQL Developer, open **/home.oracle/common_schema/olaptrain/labs/Prac6.sql**, and fill in the body of the query.
2. Execute the query. The query should return three rows, and results should look like this:

The screenshot shows the SQL Developer interface with the 'Results' tab selected. The results pane displays a table with two columns: 'DEPT' and 'SALES'. The data is as follows:

DEPT	SALES
1 Portable Music and Video	49286079.48
2 Cameras and Camcorders	31820248.4
3 Computers	336408689.39

The correct Explain plan should look like this:



Practice 6-2: Using Level Filters in an OLAP Cube Query

Write a query that returns **SALES** for:

- Channel members at the **CLASS** level
- Product members at the **DEPARTMENT** level
- **Quarters** in calendar year 2007
- **All Products**

Hints

- Use the *LEVELNAME_LONG_DESCRIPTION* columns for each dimension (notice that some of the column names are truncated by SQL Developer).
- Use the *DIMENSION_HIERARCHY_VIEW* for each dimension.
- Use the Views node in the Connection navigator to identify view names, column names, and column values.
- Drag column names from the navigator into your query.
- View the data values for any column by double-clicking the desired view in the Connection navigator, and then selecting the Data tab.

1. Scroll down in Prac6.sql to **6-2**. Examine the query, and fill in the remaining elements.

2. Execute the query. The query should return 24 rows, and results should look like this:

The screenshot shows the Oracle SQL Developer interface with the 'Results' tab selected. The results pane displays a table with four columns: CLASS, DEPT, QTR, and SALES. The data consists of 16 rows, grouped into two main categories: Direct (rows 1-8) and Indirect (rows 9-16). The DEPT column contains three distinct values: 'Cameras and Camcorders', 'Computers', and 'Portable Music and Video'. The QTR column shows the quarters from Q1-CY2007 to Q4-CY2007. The SALES column lists various monetary values.

	CLASS	DEPT	QTR	SALES
1	Direct	Cameras and Camcorders	Q1-CY2007	1242385
2	Direct	Cameras and Camcorders	Q2-CY2007	1125521
3	Direct	Cameras and Camcorders	Q3-CY2007	1354490
4	Direct	Cameras and Camcorders	Q4-CY2007	1443028
5	Direct	Computers	Q1-CY2007	13917490
6	Direct	Computers	Q2-CY2007	11756607
7	Direct	Computers	Q3-CY2007	12865030
8	Direct	Computers	Q4-CY2007	14308176
9	Direct	Portable Music and Video	Q1-CY2007	1945639
10	Direct	Portable Music and Video	Q2-CY2007	1666430
11	Direct	Portable Music and Video	Q3-CY2007	1812649
12	Direct	Portable Music and Video	Q4-CY2007	2045273
13	Indirect	Cameras and Camcorders	Q1-CY2007	1719385
14	Indirect	Cameras and Camcorders	Q2-CY2007	1573766
15	Indirect	Cameras and Camcorders	Q3-CY2007	1837557
16	Indirect	Cameras and Camcorders	Q4-CY2007	2097116

Practice 6-3: Adding Calculated Measures to the Query

Start with the completed query from practice 6-2. Then, modify it by adding the following measures:

- **SALES_YTD**
- **SALES_2006**
- **TO_GO**
- **PCT_OF_2006**
- **HOW_IS_SALES**

1. Copy the completed query from the previous practice and paste it below the commented lines for 6-3.

```
-- ****
-- 6-3:
--
```

2. Modify the query using the preceding instructions (use the **round** function with all measures except for HOW_IS_SALES).
3. Execute the query.

The results should look like this:

	CLASS	DEPT	QTR	SALES	YTD	SALES_2006	TO_GO	PCT_OF_2006	HOW_IS_SALES
1	Direct	Cameras and Camcorders	Q1-CY2007	1242385	1242385	4372207	3129821	28	On track
2	Direct	Cameras and Camcorders	Q2-CY2007	1125521	2367906	4372207	2004301	54	On track
3	Direct	Cameras and Camcorders	Q3-CY2007	1354490	3722396	4372207	649811	85	Outstanding
4	Direct	Cameras and Camcorders	Q4-CY2007	1443028	5165424	4372207	-793217	118	Outstanding
5	Direct	Computers	Q1-CY2007	13917490	13917490	46459972	32542482	30	Outstanding
6	Direct	Computers	Q2-CY2007	11756607	25674097	46459972	20785875	55	Outstanding
7	Direct	Computers	Q3-CY2007	12865030	38539127	46459972	7920845	83	Outstanding
8	Direct	Computers	Q4-CY2007	14308176	52847303	46459972	-6387331	114	On track
9	Direct	Portable Music and Video	Q1-CY2007	1945639	1945639	6777315	4831676	29	On track
0	Direct	Portable Music and Video	Q2-CY2007	1666430	3612069	6777315	3165246	53	On track
1	Direct	Portable Music and Video	Q3-CY2007	1812649	5424718	6777315	1352597	80	On track
2	Direct	Portable Music and Video	Q4-CY2007	2045273	7469990	6777315	-692676	110	On track
3	Indirect	Cameras and Camcorders	Q1-CY2007	1719385	1719385	6199891	4480506	28	On track
4	Indirect	Cameras and Camcorders	Q2-CY2007	1573766	3293152	6199891	2906740	53	On track
5	Indirect	Cameras and Camcorders	Q3-CY2007	1837557	5130709	6199891	1069183	83	On track
6	Indirect	Cameras and Camcorders	Q4-CY2007	2097116	7227825	6199891	-1027933	117	Outstanding
7	Indirect	Computers	Q1-CY2007	19859709	19859709	65663836	45804127	30	Outstanding
8	Indirect	Computers	Q2-CY2007	16824419	36684128	65663836	28979708	56	Outstanding

Practice 6-4: Leveraging Embedded Total Features of Cubes

The cube and dimensions are represented as embedded totals in the views. This makes it easy to show multiple levels of aggregation at the same time using a simple query.

Using the completed query from practice 6-3, modify the query as follows:

- For Time, select the following three levels for 2007 using a “member” filter:
 - 'CY2007'
 - 'Q3-CY2007'
 - 'Nov-2007'
 - To simplify the query, select from the LONG_DESCRIPTION column for each of the dimensions (instead of the “level” description columns).
1. Copy the completed query from the previous practice and paste it below the commented lines for 6-4.
 2. Modify the query using the preceding instructions.
 3. Execute the query.

The results should look like this:

	CHANNEL	PRODUCT	TIME	SALES	YTD	SALES_2006	TO_GO	PCT_OF_2006	HOW_IS_SALE
1	Direct	Cameras and Camcorders	Q3-CY2007	1354490	3722396	4372207	649811	85	Outstanding
2	Direct	Cameras and Camcorders	Nov-2007	441683	4586706	4372207	-214499	105	Outstanding
3	Direct	Cameras and Camcorders	CY2007	5165424	5165424	4372207	-793217	118	Outstanding
4	Direct	Computers	Q3-CY2007	12865030	38539127	46459972	7920845	83	Outstanding
5	Direct	Computers	Nov-2007	4280462	46797775	46459972	-337802	101	On track
6	Direct	Computers	CY2007	52847303	52847303	46459972	-6387331	114	On track
7	Direct	Portable Music and Video	Q3-CY2007	1812649	5424718	6777315	1352597	80	On track
8	Direct	Portable Music and Video	Nov-2007	608237	6587010	6777315	190304	97	On track
9	Direct	Portable Music and Video	CY2007	7469990	7469990	6777315	-692676	110	On track
10	Indirect	Cameras and Camcorders	Q3-CY2007	1837557	5130709	6199891	1069183	83	On track
11	Indirect	Cameras and Camcorders	Nov-2007	662219	6376675	6199891	-176783	103	Outstanding
12	Indirect	Cameras and Camcorders	CY2007	7227825	7227825	6199891	-1027933	117	Outstanding
13	Indirect	Computers	Q3-CY2007	18117883	54802011	65663836	10861825	83	Outstanding
14	Indirect	Computers	Nov-2007	6105937	66545865	65663836	-882029	101	On track
15	Indirect	Computers	CY2007	75059312	75059312	65663836	-9395476	114	On track
16	Indirect	Portable Music and Video	Q3-CY2007	2500406	7571126	9486938	1915812	80	On track

Practice 6-5: Applying Hierarchical Relationships in Queries

The dimension views also include special hierarchy columns that make it easy to navigate the dimensions (for example, drill down from ‘All Years’ to the ‘Years’). In this practice, you use “Parent” columns in order to perform drilling.

Modify the completed query from practice 6-4 by applying a condition on the PARENT columns for the Product and Geography dimensions. The query should:

- Select data for the children of “ALL_PRODUCTS” and “ALL_REGIONS.”
- Not display the Channel dimension
- Order by geography, product, and t.end_date (time)

1. Copy the completed query from the previous practice and paste it below the commented lines for **6-5**.
2. Modify the query using the preceding instructions.
3. Execute the query.

Query results should look like this:

Results:

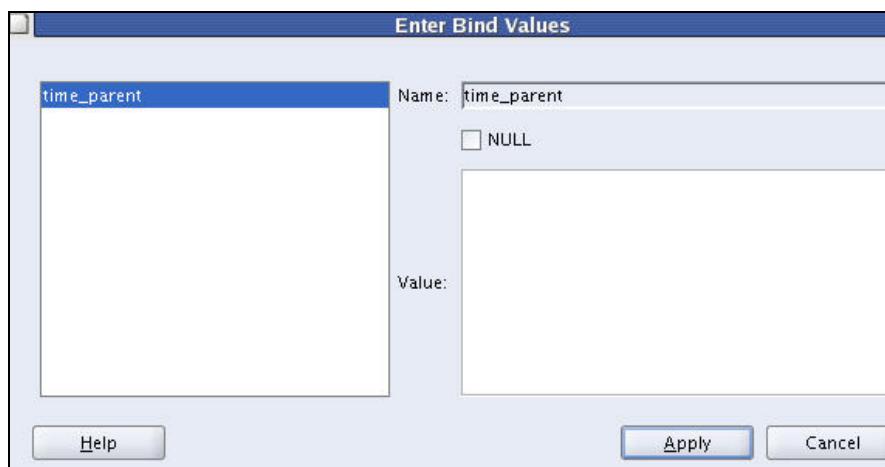
GEOGRAPHY	PRODUCT	TIME	SALES	YTD	SALES_2006	TO_GO	PCT_OF_2006	HOW_IS_SALES_
1 Africa	Cameras and Camcorders	Q3-CY2007	131006	391502	474460	82958	83	On track
2 Africa	Cameras and Camcorders	Nov-2007	51821	488912	474460	-14452	103	Outstanding
3 Africa	Cameras and Camcorders	CY2007	542473	542473	474460	-68012	114	On track
4 Africa	Computers	Q3-CY2007	1393278	4088334	5143958	1055624	79	On track
5 Africa	Computers	Nov-2007	454352	4976090	5143958	167868	97	On track
6 Africa	Computers	CY2007	5620477	5620477	5143958	-476520	109	On track
7 Africa	Portable Music and Video	Q3-CY2007	195573	566383	728667	162283	78	On track
8 Africa	Portable Music and Video	Nov-2007	65432	691098	728667	37568	95	On track
9 Africa	Portable Music and Video	CY2007	775816	775816	728667	-47149	106	On track
10 Asia	Cameras and Camcorders	Q3-CY2007	1387059	3856158	4661169	805011	83	On track
11 Asia	Cameras and Camcorders	Nov-2007	474724	4777146	4661169	-115978	102	Outstanding
12 Asia	Cameras and Camcorders	CY2007	5407279	5407279	4661169	-746110	116	Outstanding
13 Asia	Computers	Q3-CY2007	13781877	41183310	50031368	8848058	82	On track
14 Asia	Computers	Nov-2007	4570693	50032719	50031368	-1351	100	On track

Practice 6-6: Using Parameterized Drilling

In this practice, add a substitution parameter to the query for Time. This enables the user to enter a valid Time member value. Perform the following

- For the Time dimension:
 - Use the PARENT column for drilling.
 - Use the `nvl` function so that if no value is provided, the ALL_YEARS member is automatically used as the parent value.
 - For the Channel dimension:
 - Add Channel back into the results.
 - Use the PARENT column for drilling, to return the children of ‘ALL_CHANNELS.’
 - Remove Geography from the results.
1. Copy the completed query from the previous practice and paste it below the commented lines for 6-6.
 2. Modify the query using the preceding instructions and execute the query.

Result: The Enter Bind Values dialog box appears:



3. Leave the Value box blank, and click **Apply** to execute the query.

Query results include a drilldown on ‘ALL_YEARS’ for the TIME dimension:

CHANNEL	PRODUCT	TIME	SALES	YTD	SALES_2006	TO_GO	PCT_OF_2006	HOW_IS_SALES
1 Direct	Cameras and Camcorders	CY2005	3709301	3709301	4372207	662906	85 (null)	
2 Direct	Cameras and Camcorders	CY2006	4372207	4372207	4372207	0	100 Outstanding	
3 Direct	Cameras and Camcorders	CY2007	5165424	5165424	4372207	-793217	118 Outstanding	
4 Direct	Cameras and Camcorders	CY2008	(null)	(null)	4372207	(null)	(null) (null)	
5 Direct	Computers	CY2005	39562142	39562142	46459972	6897830	85 (null)	
6 Direct	Computers	CY2006	46459972	46459972	46459972	0	100 Outstanding	
7 Direct	Computers	CY2007	52847303	52847303	46459972	-6387331	114 On track	
8 Direct	Computers	CY2008	(null)	(null)	46459972	(null)	(null) (null)	
9 Direct	Portable Music and Video	CY2005	6209045	6209045	6777315	568270	92 (null)	
10 Direct	Portable Music and Video	CY2006	6777315	6777315	6777315	0	100 On track	
11 Direct	Portable Music and Video	CY2007	7469990	7469990	6777315	-692676	110 On track	
12 Direct	Portable Music and Video	CY2008	(null)	(null)	6777315	(null)	(null) (null)	
13 Indirect	Cameras and Camcorders	CY2005	5145600	5145600	6199891	1054291	83 (null)	

4. Re-execute the query.
5. This time, enter **CY2007** as the bind variable value, and click **Apply**.



Query results include a drilldown on 'CY2007' for the TIME dimension:

	CHANNEL	PRODUCT	TIME	SALES	YTD	SALES_2006	TO_GO	PCT_OF_2006	HOW_IS_SALES
1	Direct	Cameras and Camcorders	Q1-CY2007	1242385	1242385	4372207	3129821	28	On track
2	Direct	Cameras and Camcorders	Q2-CY2007	1125521	2367906	4372207	2004301	54	On track
3	Direct	Cameras and Camcorders	Q3-CY2007	1354490	3722396	4372207	649811	85	Outstanding
4	Direct	Cameras and Camcorders	Q4-CY2007	1443028	5165424	4372207	-793217	118	Outstanding
5	Direct	Computers	Q1-CY2007	13917490	13917490	46459972	32542482	30	Outstanding
6	Direct	Computers	Q2-CY2007	11756607	25674097	46459972	20785875	55	Outstanding
7	Direct	Computers	Q3-CY2007	12865030	38539127	46459972	7920845	83	Outstanding
8	Direct	Computers	Q4-CY2007	14308176	52847303	46459972	-6387331	114	On track
9	Direct	Portable Music and Video	Q1-CY2007	1945639	1945639	6777315	4831676	29	On track
10	Direct	Portable Music and Video	Q2-CY2007	1666430	3612069	6777315	3165246	53	On track
11	Direct	Portable Music and Video	Q3-CY2007	1812649	5424718	6777315	1352597	80	On track
12	Direct	Portable Music and Video	Q4-CY2007	2045273	7469990	6777315	-692676	110	On track
13	Indirect	Cameras and Camcorders	Q1-CY2007	1719385	1719385	6199891	4480506	28	On track
14	Indirect	Cameras and Camcorders	Q2-CY2007	1573766	3293152	6199891	2906740	53	On track
15	Indirect	Cameras and Camcorders	Q3-CY2007	1837557	5130709	6199891	1069183	83	On track
16	Indirect	Cameras and Camcorders	Q4-CY2007	2097116	7227825	6199891	-1027933	117	Outstanding

6. Save the Prac6.sql file.
7. Minimize SQL Developer.

Practice 6-7: Aggregating over Attributes

In this practice, you run a query that combines OLAP cube views with SQL aggregation to make use of the dimension attribute “Channel Type.” Recall that you created this attribute in Lesson 3, when you first created your dimensional model.

This example illustrates how to use attributes in an OLAP query. In this type of query:

- The Cube delivers summary data at the specified levels in the WHERE clause.
- Then, the relational engine summarizes cube data using a GROUP BY clause in combination with the SUM aggregation functions in the SELECT statement.

Perform the following:

1. Open the `/home.oracle/common_schema/olaptrain/labs/Prac6_solutions.sql` file.
2. Scroll to the bottom of the file and examine the query under **6-7**.
3. Execute the query. The query should return the following:

The screenshot shows the Oracle SQL Developer interface. The top part is a code editor with the following SQL query:

```
Enter SQL Statement:
SELECT c.channel_type AS chan_type,
       p.category_long_descriptio AS category,
       t.calendar_quarter_long_de AS qtr,
       SUM(s.sales) AS sales,      -- (B) Relational engine summarizes cube data
       SUM(s.sales_ytd) AS ytd    -- (B) Relational engine
  FROM channel_sales_channel_view c,
       product_standard_view p,
       time_calendar_view t,
       sales_cube_view s
 WHERE(c.dim_key = s.channel
   AND p.dim_key = s.product
   AND t.dim_key = s.time)      -- (A) Cube delivers summary data at Levels:
   AND c.level_name = 'CHANNEL' -- Channel
   AND s.geography = 'ALL_REGIONS' -- All Regions
   AND p.level_name = 'CATEGORY' -- Category
   AND t.level_name = 'CALENDAR_QUARTER' -- Quarter
)
 GROUP BY                      -- (B) Relational engine
       t.calendar_quarter_long_de,
       p.category_long_descriptio,
       c.channel_type;
```

The bottom part shows the results of the query execution:

	CHAN_TYPE	CATEGORY	QTR	SALES	YTD
1	Mail Order Catalog	Total Server Computers	Q1-CY2006	101717	101717
2	Mail Order Catalog	Total Server Computers	Q4-CY2006	99733.8	381878.8
3	Boutique	Total Server Computers	Q3-CY2005	98446	314513.2
4	Mall Store	Total Server Computers	Q4-CY2006	79671.8	269492.6
5	Web Store	Total Server Computers	Q2-CY2006	172787.8	365601.4
6	Web Store	Total Server Computers	Q4-CY2005	182899.6	728415
7	Web Store	Total Server Computers	Q4-CY2007	222728.4	838172.8
8	Mail Order Catalog	Computer Printers and Supplies	Q2-CY2007	923217.33	1958447.57

Practices for Lesson 7

In the practices for this lesson, you enhance the analytic content of your OLAP data model by adding the following elements:

- Cubes with varying dimensionality
- Integrated measures from data with different dimensionality
- Forecast measures of sales data

In practice sections, you will create statistical forecasts based on historic sales data.

Practice 7-1: Creating and Populating Forecast Measures

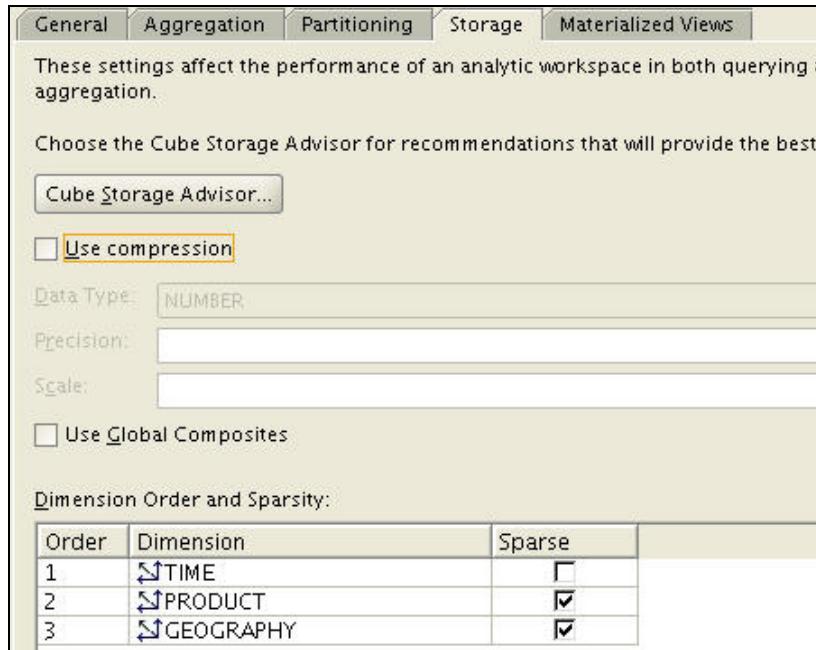
In this practice section, you create statistical forecast measures based on historic sales data. These forecast measures are designed to deliver “crossover” calculations that show actual sales for historical time periods and forecast sales for future time periods.

You will:

- Create a three-dimensional forecast cube that contains two forecast measures.
- Import and run a program that contains the logic for the forecast.
- Create and then execute a new cube script that will call the forecast program.

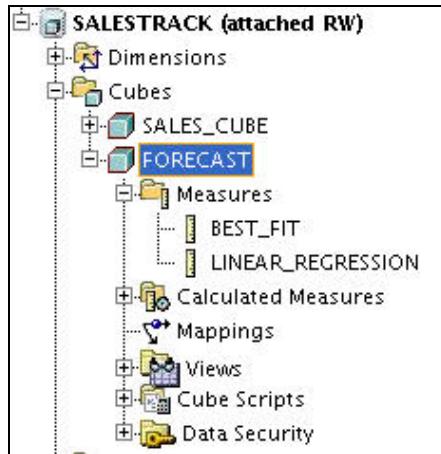
Follow these steps:

1. In AWM, right-click the Cubes node and select **Create Cube** from the menu.
2. In the General tab of the Create Cube dialog box, specify the following:
 - a. Name: **FORECAST**
 - b. Selected dimension order:
 - **TIME**
 - **GEOGRAPHY**
 - **PRODUCT**
3. In the Storage tab, deselect the Compression option, and leave the Sparse option unchecked for Time (Forecast is dense over Time).



4. Click **Create**.
5. Create two measures in the Forecast cube, with the following names:
 - BEST_FIT
 - LINEAR_REGRESSION

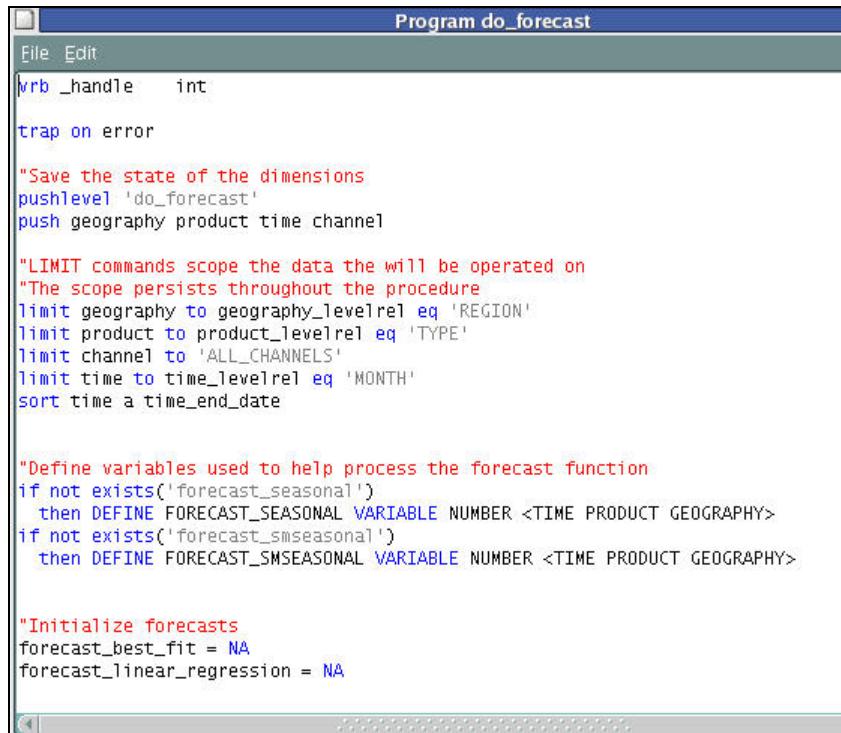
Result: The Forecast cube now contains two forecast measures.



6. Import and view a DML program that contains the forecasting logic:
 - a. Open the OLAP Worksheet, and execute the following commands:
 - **infile cubes/do_forecast.dml**
 - **update**
 - **commit**
 - b. Then, open the program by executing the following command:
 - **edit do_forecast**

The program contains four primary sections, which perform the following tasks:

- Define the historical basis for the forecast (LIMIT commands)
- Identify forecast method (definition of forecast function variables)



```

Program do_forecast
File Edit
vrb _handle int
trap on error
"Save the state of the dimensions
pushlevel 'do_forecast'
push geography product time channel

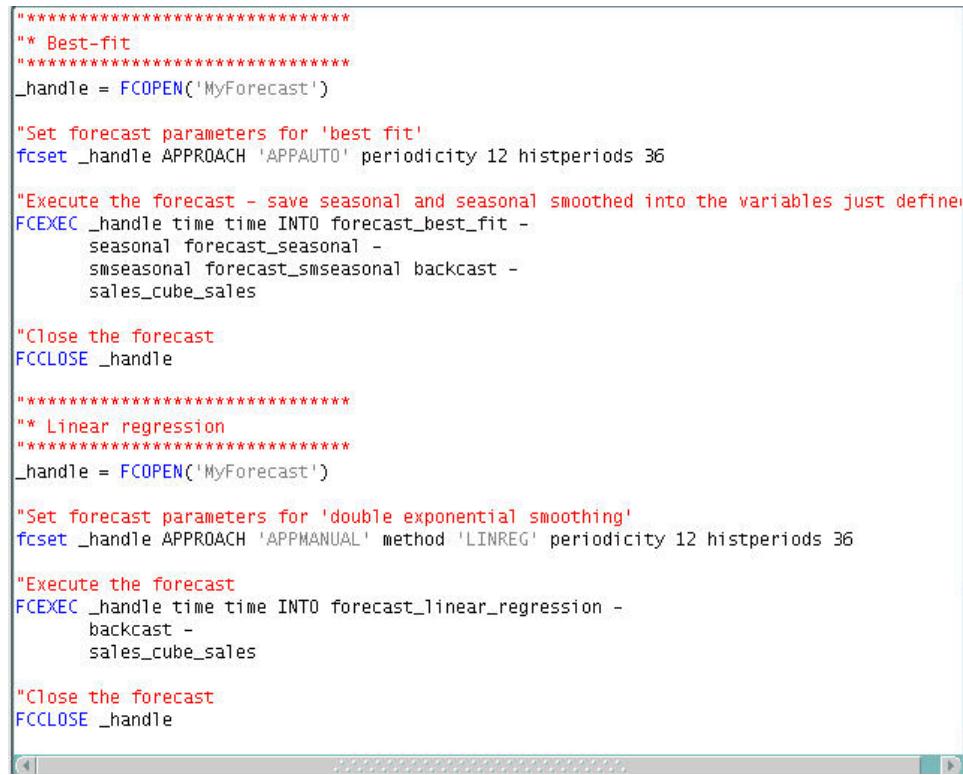
"LIMIT commands scope the data the will be operated on
"The scope persists throughout the procedure
limit geography to geography_levelrel eq 'REGION'
limit product to product_levelrel eq 'TYPE'
limit channel to 'ALL_CHANNELS'
limit time to time_levelrel eq 'MONTH'
sort time a time_end_date

"Define variables used to help process the forecast function
if not exists('forecast_seasonal')
  then DEFINE FORECAST_SEASONAL VARIABLE NUMBER <TIME PRODUCT GEOGRAPHY>
if not exists('forecast_smseasonal')
  then DEFINE FORECAST_SMSEASONAL VARIABLE NUMBER <TIME PRODUCT GEOGRAPHY>

"Initialize forecasts
forecast_best_fit = NA
forecast_linear_regression = NA

```

- Execute the appropriate forecast
- Aggregate the forecast measures



```

*****
** Best-fit
*****
_handle = FCOPEN('MyForecast')

"Set forecast parameters for 'best fit'
fcset _handle APPROACH 'APPAUTO' periodicity 12 histperiods 36

"Execute the forecast - save seasonal and seasonal smoothed into the variables just defined
FCEEXEC _handle time time INTO forecast_best_fit -
  seasonal forecast_seasonal -
  smseasonal forecast_smseasonal backcast -
  sales_cube_sales

"Close the forecast
FCCLOSE _handle

*****
** Linear regression
*****
_handle = FCOPEN('MyForecast')

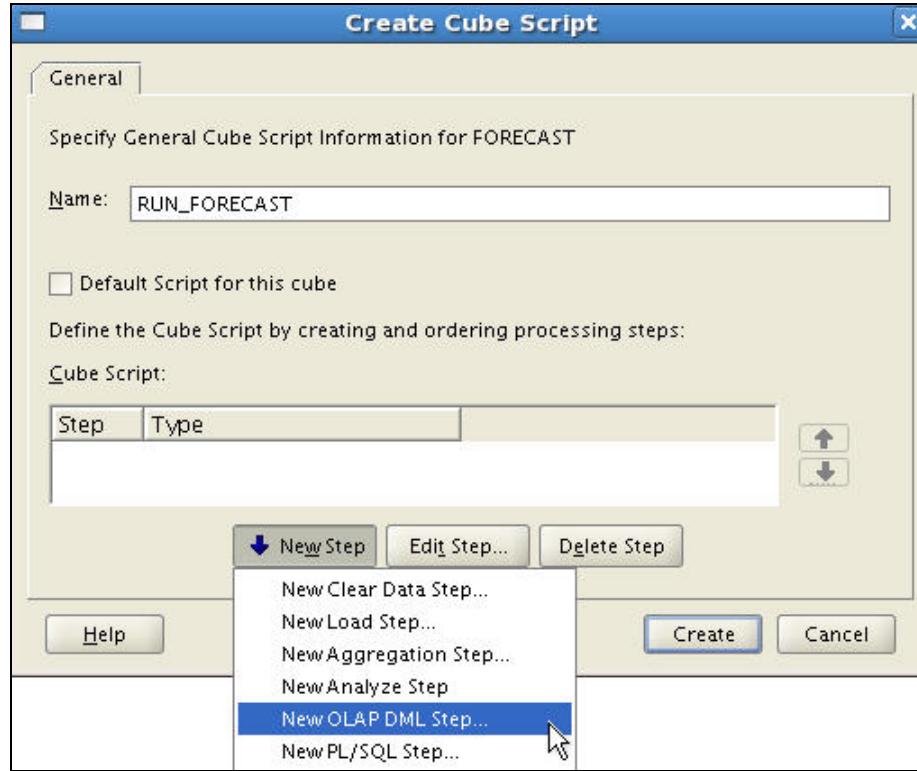
"Set forecast parameters for 'double exponential smoothing'
fcset _handle APPROACH 'APPMANUAL' method 'LINREG' periodicity 12 histperiods 36

"Execute the forecast
FCEEXEC _handle time time INTO forecast_linear_regression -
  backcast -
  sales_cube_sales

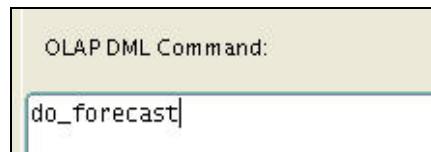
"Close the forecast
FCCLOSE _handle

```

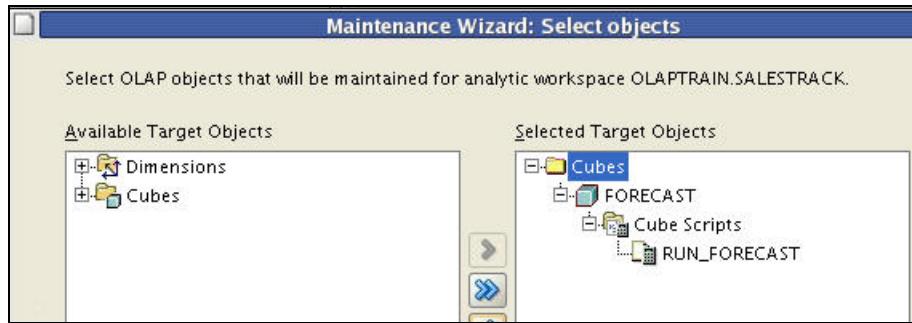
- c. Close the program window when you are done viewing the DML code.
- 7. Close the OLAP Worksheet.
- 8. Under the FORECAST cube node, right-click Cube Scripts and select **Create Cube Script** from the menu.
- 9. In the Create Cube Script dialog box, perform the following:
 - a. Enter **RUN_FORECAST** as the name.
 - b. Click the New Step button, and select **New OLAP DML Step** from the menu.



- c. In the General tab of the New OLAP DML Step dialog box, enter the DML program name **do_forecast**.



- d. Click **OK**.
- e. In the Create Cube Script dialog box, click **Create**.
- 10. To run the forecast:
 - a. Right-click the RUN_FORECAST cube script and select **Run Cube Script RUN_FORECAST** from the menu.
 - b. In the Maintenance Wizard, select all three dimensions and click the **Remove** button. The dimensions are removed from the Selected list, as the following screenshot shows (the dimensions do not need to be loaded again).



- c. Click **Finish**.
11. When the Build Log appears, close the window.

Practice 7-2: Creating and Viewing Forecast “Crossover” Calculations

In this practice section, you create two “cross-over” calculations in the Sales cube. These calculations return actual sales for historical data (available through CY2007) and forecasted sales for time periods in CY2008.

1. In the Sales cube, create a new Calculated Measure by using the techniques you learned previously. Use the following input:
 - a. Name = **CROSS_OVER_BEST_FIT**
 - b. Calculation Type = **Expression**
 - c. In the Expression box, enter the following CASE statement:

```
Expression:
CASE
WHEN "TIME".END_DATE < TO_DATE ('01-JAN-2008') THEN SALES_CUBE.SALES
WHEN CHANNEL.DIM_KEY = 'ALL_CHANNELS' THEN FORECAST.BEST_FIT
ELSE NULL
END
```

- Note:** In our example, forecasts are not broken out by channel. Therefore, the crossover measures only show forecast data for the top level in the Channel dimension ('ALL_CHANNELS').
2. Click **Create**.
 3. Create another Calculated Measure, with the following input:
 - a. Name = **CROSS_OVER_LINEAR_REGRESSION**
 - b. Calculation Type = **Expression**

- c. In the Expression box, enter the following CASE statement:

```
Expression:
CASE
WHEN "TIME".END_DATE < TO_DATE ('01-JAN-2008') THEN SALES_CUBE.SALES
WHEN CHANNEL.DIM_KEY = 'ALL_CHANNELS' THEN FORECAST.LINEAR_REGRESSION
ELSE NULL
END
```

4. Click **Create**.
5. Open the CROSS_OVER_BEST_FIT calculated measure in the Data Viewer.
6. Remove the Graph by clicking the down arrow of the Hide/Show tool.
7. Click the **Query Builder** tool.
8. In the Items tab, add **SALES** to the Selected list.
9. In the Layout tab, place Geography and Channel on the Page Items axis, place Product on the Row axis, and place Time beneath Measures in the Column axis.
10. In the Dimensions tab, remove the Time current selection and replace it with **CY2006**, **CY2007**, and **CY2008**.
11. Click **OK** to view the data.

12. Select **All Products**.

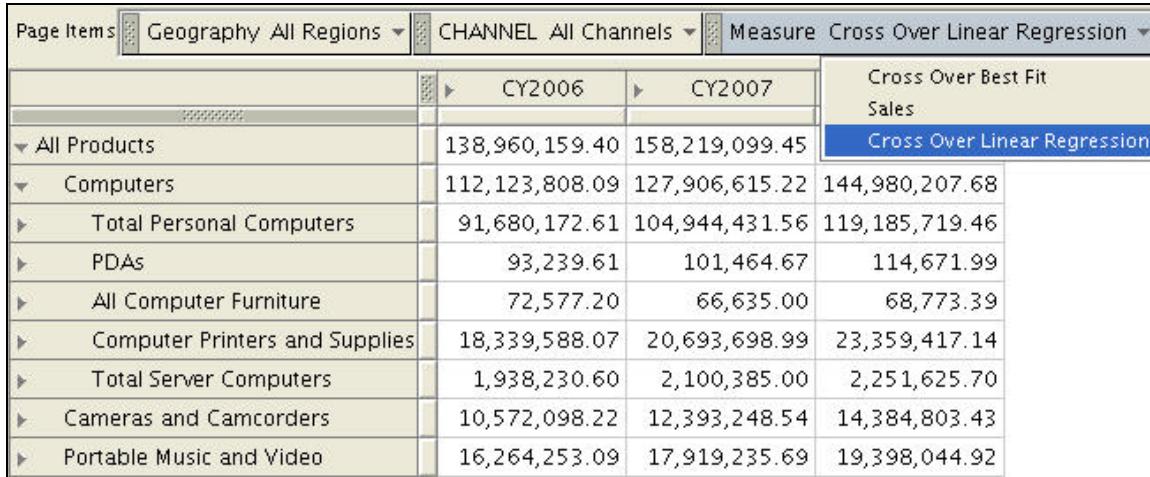
The screenshot shows the Measure Data Viewer window with the title bar 'Measure Data Viewer'. The interface includes a toolbar with icons for Dialog, Bold, Italic, Underline, and various layout options. Below the toolbar, there are dropdown menus for 'File' and other settings. The main area is divided into sections: 'Page Items' (Geography All Regions, CHANNEL All Channels), 'Cross Over Best Fit' (Time periods CY2006, CY2007, CY2008), and 'Sales' (Time periods CY2006, CY2007, CY2008). The data grid displays sales figures for different product categories over the specified time periods.

	Cross Over Best Fit			Sales		
	CY2006	CY2007	CY2008	CY2006	CY2007	CY2008
▼ All Products	138,960,159.40	158,219,099.45	172,243,622.11	138,960,159.40	158,219,099.45	
▶ Computers	112,123,808.09	127,906,615.22	138,963,361.66	112,123,808.09	127,906,615.22	
▶ Cameras and Camcorders	10,572,098.22	12,393,248.54	13,876,040.40	10,572,098.22	12,393,248.54	
▶ Portable Music and Video	16,264,253.09	17,919,235.69	19,404,220.04	16,264,253.09	17,919,235.69	

Note: The Cross Over calculation uses Sales data for historical time periods, but applies the specified forecast method for future periods

13. Experiment with the report, for example:

- Select the Cross Over Linear Regression calculation from the Query Builder and move the measures to the Page Items axis, like this:



The screenshot shows a Data Viewer interface with the following configuration:

- Page Items:** Geography All Regions, CHANNEL All Channels, Measure: Cross Over Linear Regression.
- Time Periods:** CY2006, CY2007.
- Measures:** Cross Over Best Fit Sales, Cross Over Linear Regression.

The report displays sales data for various product categories across different channels for the years CY2006 and CY2007. The 'Cross Over Linear Regression' measure is highlighted in blue.

	CY2006	CY2007	Cross Over Best Fit Sales	Cross Over Linear Regression
All Products	138,960,159.40	158,219,099.45		
Computers	112,123,808.09	127,906,615.22	144,980,207.68	
Total Personal Computers	91,680,172.61	104,944,431.56	119,185,719.46	
PDAs	93,239.61	101,464.67	114,671.99	
All Computer Furniture	72,577.20	66,635.00	68,773.39	
Computer Printers and Supplies	18,339,588.07	20,693,698.99	23,359,417.14	
Total Server Computers	1,938,230.60	2,100,385.00	2,251,625.70	
Cameras and Camcorders	10,572,098.22	12,393,248.54	14,384,803.43	
Portable Music and Video	16,264,253.09	17,919,235.69	19,398,044.92	

- Select other time periods (for example, the quarters CY2007 and CY2008).
 - Select other Product members, Geography members, or Channel members (remember, if any Channel dimension member is selected other than ALL_CHANNELS, the forecast returns a Null value).
14. When you are finished experimenting with the report, close the Data Viewer.
15. Exit AWM.

Practice 7-3: Querying Forecast Measures Using SQL

In this practice section, you use SQL Developer to query the Forecast data that you just created.

Follow these steps:

- Open SQL Developer.
- Open the .../olaptrain/labs/Prac7-3.sql file and view the query.

```
4  -- A forecast has been created using OLAP's statistical forecast capabilities.
5  -- The forecast is computed at a higher level than the actual sales data -
6  -- and it is not dimensioned by CHANNEL. Therefore, FORECAST is in a separate cube.
7  --
8  -- The CROSS_OVER measures in the SALES cube display actual sales for
9  -- historical data and forecast data for future time periods.
10 --
11 -- Note: no join is required to display SALES and FORECAST data together
12 -- using the CROSS_OVER calculation. The cube merges the data w/o
13 -- requiring joins through the calculation.
14
15 SELECT c.long_description as channel,
16     p.long_description as product,
17     t.long_description as time,
18     round(s.sales) as sales,
19     round(s.cross_over_best_fit, 0) as best_fit,
20     round(s.cross_over_linear_regres, 0) as lin_reg
21 FROM channel_sales_channel_view c,
22     product_standard_view p,
23     geographyRegional_view g,
24     time_calendar_view t,
25     sales_cube_view s
26 WHERE (c.dim_key = s.channel
27     AND g.dim_key = s.geography
28     AND p.dim_key = s.product
29     AND t.dim_key = s.TIME
30     AND c.dim_key = 'ALL_CHANNELS'          -- The children of "ALL_CHANNELS"
31     AND g.level_name = 'ALL_REGIONS'
32     AND p.parent = 'ALL_PRODUCTS'         -- The children of "ALL_PRODUCTS"
33     AND t.level_name = 'CALENDAR_QUARTER'
34     AND t.calendar_year_long_descr in ('CY2007', 'CY2008'))
```

Notice that the query is similar in every way to the other OLAP cube view queries that you created and ran in the practices for lesson 6. As you have learned, you simply select calculated measures as columns from the cube view.

3. Click inside the query and press **F9** to execute.

The results should look like this:

Results:						
	CHANNEL	PRODUCT	TIME	SALES	BEST_FIT	LIN_REG
1	All Channels	Cameras and Camcorders	Q1-CY2007	2961771	2961771	2961771
2	All Channels	Cameras and Camcorders	Q2-CY2007	2699287	2699287	2699287
3	All Channels	Cameras and Camcorders	Q3-CY2007	3192047	3192047	3192047
4	All Channels	Cameras and Camcorders	Q4-CY2007	3540144	3540144	3540144
5	All Channels	Cameras and Camcorders	Q1-CY2008	(null)	3526328	3419082
6	All Channels	Cameras and Camcorders	Q2-CY2008	(null)	3068517	3537161
7	All Channels	Cameras and Camcorders	Q3-CY2008	(null)	3437529	3655241
8	All Channels	Cameras and Camcorders	Q4-CY2008	(null)	3843667	3773320
9	All Channels	Computers	Q1-CY2007	33777199	33777199	33777199
10	All Channels	Computers	Q2-CY2007	28581026	28581026	28581026
11	All Channels	Computers	Q3-CY2007	30982913	30982913	30982913
12	All Channels	Computers	Q4-CY2007	34565477	34565477	34565477
13	All Channels	Computers	Q1-CY2008	(null)	36313991	34705567
14	All Channels	Computers	Q2-CY2008	(null)	31450588	35731890
15	All Channels	Computers	Q3-CY2008	(null)	34078273	36758213
16	All Channels	Computers	Q4-CY2008	(null)	37120510	37784537
17	All Channels	Portable Music and Video	Q1-CY2007	4692772	4692772	4692772
18	All Channels	Portable Music and Video	Q2-CY2007	3990017	3990017	3990017

4. Close SQL Developer.

Practices for Lesson 8

In the practices for this lesson, you access OLAP data using APEX. You perform three tasks:

1. Create an interactive report that allows you to view highly stylized output. This report will use parameters; however the parameters will not be used until the second practice.
2. Add formatting to the report.
3. Leverage the parameters to enable data exploration using “drilling” across the product, time, and geography dimensions.

You will find that this style of drilldown will work with any type of hierarchy: level-based, parent-child, skip-level, ragged, and so on.

When you are done, you will have a formatted, interactive sales analysis report that enables users to explore data by drilling down on products, times, and geographies.

The resulting report will look like the following:

Geography : Africa								
Product	Time	Sales	% Chg Pr Period	% Chg Pr Year	Product Alert	Ytd	Ytd % Chg Pr Year	
Cameras and Camcorders	CY2005	392434	-	-	OKAY	392434	-	
Portable Music and Video	CY2005	647626	-	-	OKAY	647626	-	
Computers	CY2005	4199753	-	-	OKAY	4199753	-	
Portable Music and Video	CY2006	728666	12.5	12.5	OKAY	728666	12.5	
Cameras and Camcorders	CY2006	474460	20.9	20.9	OKAY	474460	20.9	
Computers	CY2006	5143957	22.4	22.4	ALERT	5143957	22.4	
Portable Music and Video	CY2007	775815	6.4	6.4	OKAY	775815	6.4	
Cameras and Camcorders	CY2007	542472	14.3	14.3	OKAY	542472	14.3	
Computers	CY2007	5620477	9.2	9.2	ALERT	5620477	9.2	
Cameras and Camcorders	CY2008	-	-	-	OKAY	-	-	
Portable Music and Video	CY2008	-	-	-	OKAY	-	-	
Computers	CY2008	-	-	-	OKAY	-	-	

Geography : Asia								
Product	Time	Sales	% Chg Pr Period	% Chg Pr Year	Product Alert	Ytd	Ytd % Chg Pr Year	
Cameras and Camcorders	CY2005	3888650	-	-	OKAY	3888650	-	
Portable Music and Video	CY2005	6704484	-	-	OKAY	6704484	-	
Computers	CY2005	42606313	-	-	OKAY	42606313	-	

1 - 15

Using this report, users:

- Can drill down on Geography, Product, and Time to any level of detail
- May want to follow the “Product Alert” column when drilling down, since this column indicates that Sales have dropped compared to last year for at least one of the children of the current product

For example, in the preceding report, Computers’ Sales in 2006 have improved by 22.4% compared to last year, but there is an ALERT. Drilling down on Computers will show that there is a problem with sales within the department.

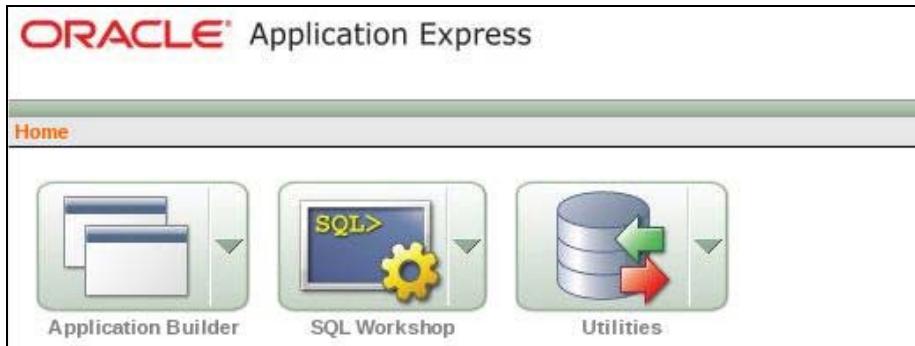
Practice 8-1: Creating an Interactive Sales Analysis Report

In this practice section, you log into APEX as a developer and create an interactive report in the predefined **OLAP ILT – Hands On** application. This application currently contains a login page and an empty Sales Analysis page.

Follow these steps to create the interactive report:

1. Start your web browser and navigate to the Application Express login page:
<http://localhost:8080/apex>
2. Log in using the following details:
 - a. Workspace: **SALESTRACK**
 - b. Username: **olaptrain**
 - c. Password: **oracle**
3. If you are prompted for a new password (if not, move to step 4):
 - a. Enter **oracle** as both the old and the new password, and then click **Apply**.
 - b. Click **Return**.
 - c. Once again, enter **oracle** as the password, and click **Login**.

Result: You should now be at the APEX developer Home page, as shown here:

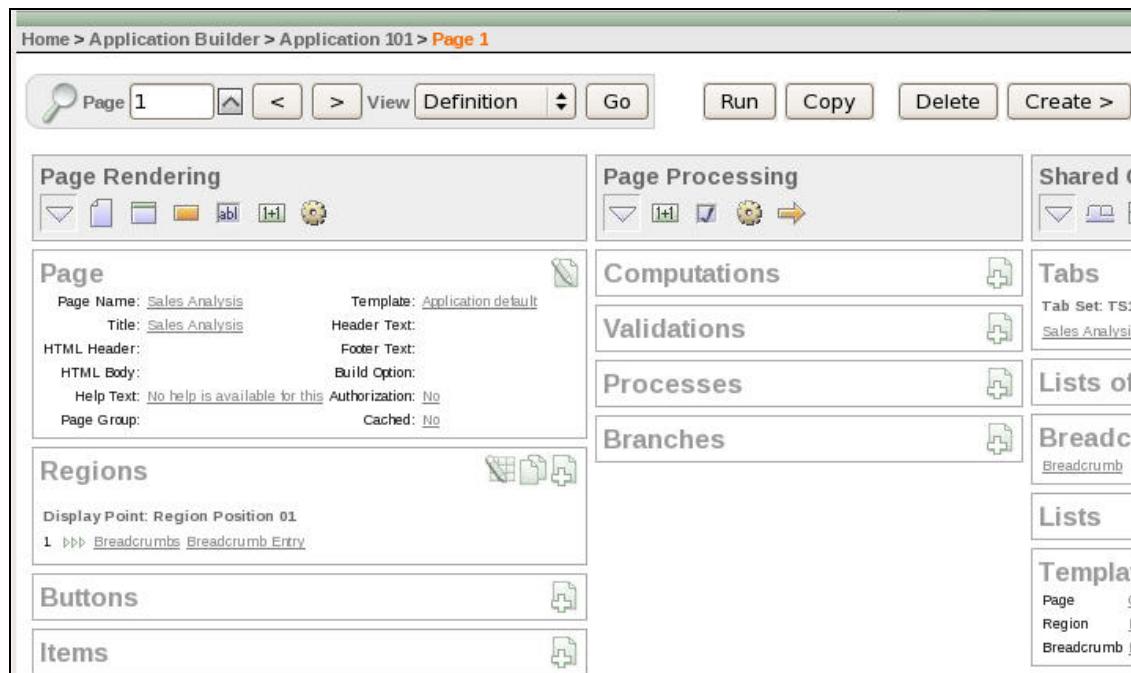


4. To access the predefined application, click **Application Builder > OLAP ILT – Hands On**.



- Click the Sales Analysis icon.

Result: You are now in the Page Editor.



Next, you will perform the following tasks:

- Create three Hidden Items that will be used for your parameterized query. These parameters will be used for setting up dimension “drilling” in the next practice.
- For example, a hidden item named P1_PRODUCT will be used as a parameter for drilling down on the Product dimension in the sales query as follows:

```
p.parent = nvl (:P1_PRODUCT, 'ALL_PRODUCTS')
```

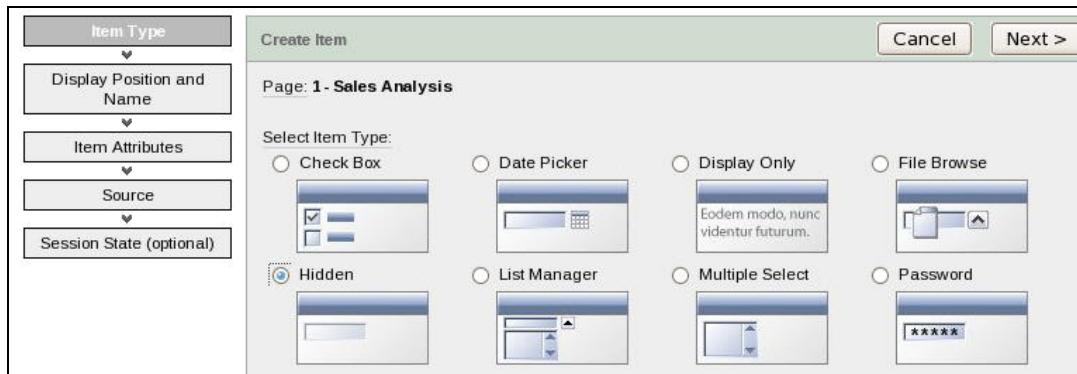
This condition selects the children of the product represented by the parameter (and selects “ALL_PRODUCTS” if P1_PRODUCT is null).

- Create an interactive report region that shows sales performance by region, product, and time.

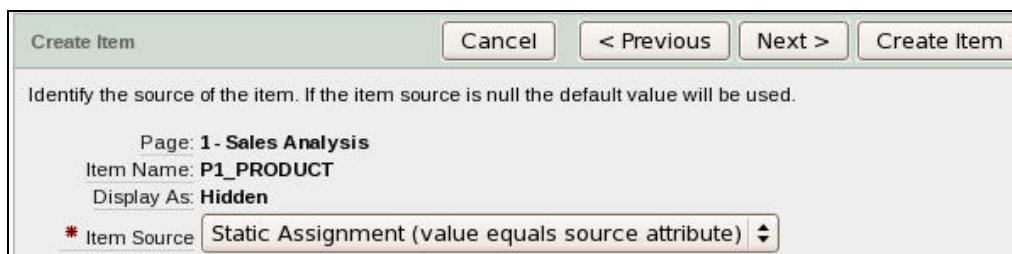
First, create the Hidden Items.

6. Create a hidden item P1_PRODUCT:

- Click the **Create**  button in the Items pane.
- On the Create Item page, select **Hidden** and click **Next**.



- Select **Hidden** for the Hidden Item Type and click **Next**.
- Name the item **P1_PRODUCT** and click **Next**.
- Accept the defaults for the item source and click **Create Item**.



7. Create a second hidden item **P1_GEOGRAPHY** using the preceding steps.

8. Create a final hidden item **P1_TIME** using the preceding steps.

When you are done, the Items pane should look like this:



10	<u>P1_PRODUCT</u>	Hidden
20	<u>P1_GEOGRAPHY</u>	Hidden
30	<u>P1_TIME</u>	Hidden

Next, create the Interactive report.

9. Create an interactive report region:

- On the Page Definition page, click the **Create**  button in the Regions pane.
- On the Create Region page, select **Report** and click **Next**.
- On the Report Implementation page, select **Interactive Report** and click **Next**.
- On the Display Attributes page, enter **Explore** as the title and click **Next**.

- e. On the Source page, you will use a predefined query. Perform the following:
- Select **File > Open File** from the browser's main menu.
 - Open the **home/oracle/common_schema/olaptrain/labs/Prac8-1.sql** file.
 - Select the query (that is explained in the following) and copy it to the clipboard.

```

SELECT
    g.long_description "Geography",
    p.long_description "Product",
    t.long_description "Time",
    trunc(sales) "Sales",
    trunc(sales_pp_pct_chg, 1)
        "% Chg Pr Period",
    trunc(sales_py_pct_chg, 1)
        "% Chg Pr Year",
    product_alert "Product Alert",
    trunc(sales_ytd) "YTD",
    trunc(sales_ytd_py_pct_chg, 1)
        "YTD % Chg Pr Year",
    t.dim_key as time_dim_key,
    t.parent as time_parent,
    t.end_date as end_date,
    g.parent as geography_parent,
    g.dim_key as geography_dim_key,
    p.parent as product_parent,
    p.dim_key as product_dim_key
FROM sales_cube_view s,
     time_calendar_view t,
     product_standard_view p,
     geographyRegional_view g
WHERE s.time = t.dim_key
    AND s.product = p.dim_key
    AND s.geography = g.dim_key
    AND s.channel = 'ALL_CHANNELS'
    AND t.parent =
        nvl(:P1_TIME, 'ALL_YEARS')
    AND g.parent =
        nvl(:P1_GEOGRAPHY, 'ALL_REGIONS')
    AND p.parent =
        nvl(:P1_PRODUCT, 'ALL_PRODUCTS')
}

```

The diagram illustrates the structure of the SQL query with five labeled sections:

- Dimension Descriptions:** Groups the first three columns (Geography, Product, Time).
- Measures:** Groups the Sales column and its associated calculations (truncation, percentage changes).
- Dimension Key and Parent Columns (used for drilling):** Groups the time_dim_key, time_parent, end_date, geography_parent, geography_dim_key, product_parent, and product_dim_key columns.
- Cube and dimension views:** Groups the FROM clause, listing the four views: sales_cube_view, time_calendar_view, product_standard_view, and geographyRegional_view.
- Parent conditions that select children of parameter value:** Groups the WHERE clause, which includes conditions on time, product, geography, channel, and parent dimensions, along with their corresponding NVL clauses for parameter values.

Note: This single select statement is used to navigate vast regions of the cube. Notice that the parameter names match the hidden items you created previously.

- Paste the query into the SQL SELECT statement box.

v) Select **No** for the Link to Single Row View option.

The Source page should look like this:

The screenshot shows the 'Create Region' dialog box. At the top, there are buttons for 'Cancel', '< Previous', 'Next >', and 'Create Region'. The title bar says 'Create Region' and the sub-title is 'Page: 1 - Sales Analysis'. Below that, 'Region Title' is set to 'Explore'. A large text area contains a SQL SELECT statement:

```
FROM sales_cube_view s, time_calendar_view t,
product_standard_view p, geographyRegional_view g
WHERE s.time = t.dim_key
AND s.product = p.dim_key
AND s.geography = g.dim_key
AND s.channel = 'ALL_CHANNELS'
AND t.parent = nvl(:P1_TIME, 'ALL_YEARS')
AND g.parent = nvl(:P1_GEOGRAPHY, 'ALL_REGIONS')
AND p.parent = nvl(:P1_PRODUCT, 'ALL_PRODUCTS')
```

Below the SQL area is a 'Query Builder' section with three fields:

- 'Link to Single Row View' is set to 'No'.
- 'Uniquely Identify Rows by' is set to 'ROWID'.
- 'Unique Column' is empty.

vi) Click **Create Region**.

The Regions pane should look like this:



Practice 8-2: Formatting the Interactive Sales Analysis Report

Now that your report has been created, you will apply formatting to it. You will apply the following type of formatting:

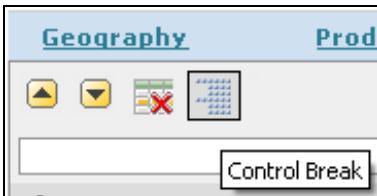
- Select the columns to display.
- Sort the report by time.
- Add a control break on Geography to the report.
- Add color coding to the report.

Follow these steps to format the interactive report:

1. Click the **Run** button in the middle of the page to view and then format the report.
2. In the Login screen, enter **olaptrain/oracle** and click **Login**.
3. Select the report columns to display:
 - a. Click the Gear icon  and then click **Select Columns**.
 - b. Select the following dimension and measures columns in the **Display in Report** list. Then, click **Apply**.

Do Not Display	Display in Report
Time Dim Key Time Parent End Date Geography Parent Geography Dim Key Product Parent Product Dim Key	Geography Product Time Sales % Chg Pr Period % Chg Pr Year Product Alert Ytd Ytd % Chg Pr Year

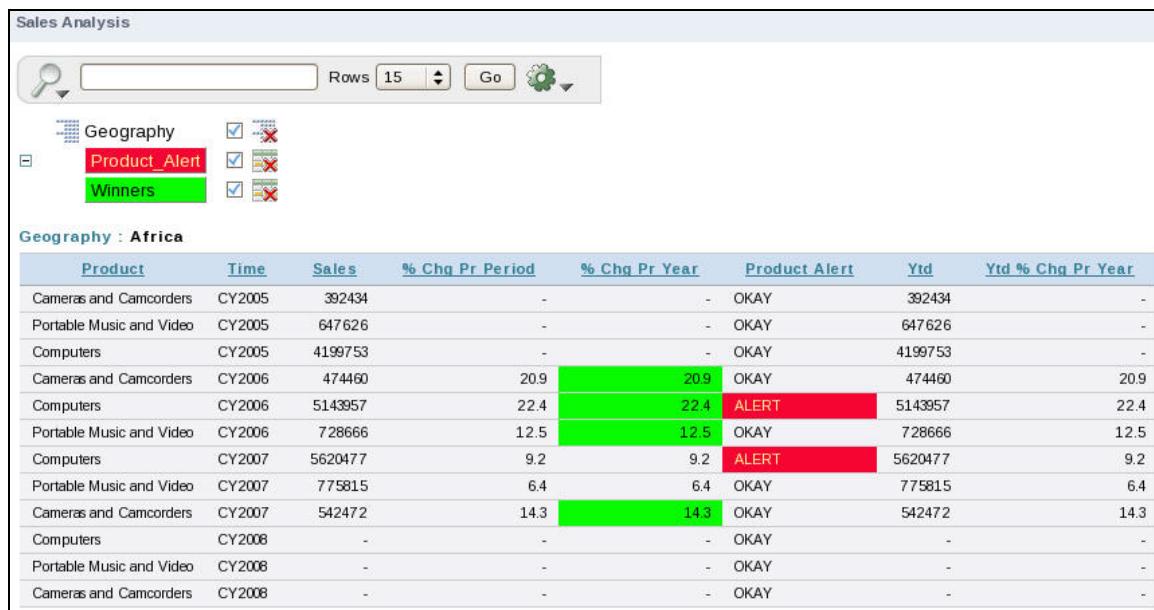
4. Sort the rows by the End Date column:
 - a. Click the Gear icon  and then select **Sort**.
 - b. For Column 1, select **End Date** and then click **Apply**.
5. To add a control break on Geography to the report, click the **Geography** report header and select **Control Break**.



6. Color the % Chg Pr Year cells green if sales growth exceeds 10%:
 - a. Click the Gear icon  and then select **Highlight**.
 - b. Name the Highlight format **Winners**.

- c. Set the Highlight type to **Cell**.
 - d. Set the Background Color to **green**.
 - e. Specify the following Highlight Condition: **% Chg Pr Year > 10**
 - f. Click **Apply**.
7. Color Product Alert red if there is an ALERT.
- a. Click the Gear icon  and then select **Highlight**.
 - b. Name the format **Product_Alert**.
 - c. Set the Highlight type to **Cell**.
 - d. Set the Background Color to **red** and the Text Color to **yellow**.
 - e. Specify the following Highlight Condition: **Product Alert = ALERT**
 - f. Click **Apply**.

Your report should now look something like the following:



The screenshot shows a report titled "Sales Analysis" with a search bar and a gear icon. Under "Geography", "Product_Alert" is selected. The main table is titled "Geography : Africa" and has columns: Product, Time, Sales, % Chg Pr Period, % Chg Pr Year, Product Alert, Ytd, and Ytd % Chg Pr Year. The data includes rows for various products and years, with some cells highlighted in green or red based on the defined conditions.

Product	Time	Sales	% Chg Pr Period	% Chg Pr Year	Product Alert	Ytd	Ytd % Chg Pr Year
Cameras and Camcorders	CY2005	392434	-	-	OKAY	392434	-
Portable Music and Video	CY2005	647626	-	-	OKAY	647626	-
Computers	CY2005	4199753	-	-	OKAY	4199753	-
Cameras and Camcorders	CY2006	474460	20.9	20.9	OKAY	474460	20.9
Computers	CY2006	5143957	22.4	22.4	ALERT	5143957	22.4
Portable Music and Video	CY2006	728666	12.5	12.5	OKAY	728666	12.5
Computers	CY2007	5620477	9.2	9.2	ALERT	5620477	9.2
Portable Music and Video	CY2007	775815	6.4	6.4	OKAY	775815	6.4
Cameras and Camcorders	CY2007	542472	14.3	14.3	OKAY	542472	14.3
Computers	CY2008	-	-	-	OKAY	-	-
Portable Music and Video	CY2008	-	-	-	OKAY	-	-
Cameras and Camcorders	CY2008	-	-	-	OKAY	-	-

8. Scroll through the report to view the formatting.
9. Save the Report as the default:
 - a. Click the Gear icon  and then select **Save Report**.
 - b. Name the report **Default Report Settings** and click **Apply**.

Practice 8-3: Adding Drilldown Capabilities to the Report

The final step is to enable drilldown capabilities to the report. When users click a Geography, Product or Time value, the details for the selected item will replace what was selected. For example, clicking CY2007 displays the calendar quarters for that year.

Application Express allows you to specify links for columns on the Report Attributes page. In our case, the link will set the relevant dimension parameter (a.k.a. Hidden Item) to the item the user selected.

Consider the flow for the time drilldown previously described:

- User clicks the CY2007 link.
- The logic behind the link specifies that the parameter P1_TIME is set to the selected value.
- The query is re-executed, whereby the condition:

```
t.parent = nvl(:P1_TIME, 'ALL_YEARS')
```

Selects all time values whose parent is ‘CY2007.’ This is the equivalent to selecting the children of ‘CY2007’.

Result: A drilldown has been executed.

Perform the following steps to add drilldown capabilities to the report:

1. Return to the page editor by clicking the **Edit Page 1** link at the bottom of the report page.
2. Go to the Report Attributes page by clicking the **Interactive Report** link in the Regions pane.
3. Add a drilldown link to Geography:
 - a. Click the **Edit** icon  for Geography.
 - b. Click the **Link** button above the Column Definition page.
 - c. For Link Text, click the **[Geography]** link.
 - d. In the Page text box, enter 1 (that is, the current page number is “1”).
 - e. Use the flashlight icons  to enter name value pairs as follows:

Name	Value
Item 1 P1_GEOGRAPHY	#GEOGRAPHY_DIM_KEY#
Item 2 P1_PRODUCT	#PRODUCT_PARENT#
Item 3 P1_TIME	#TIME_PARENT#

Note:

- You set the “link” dimension to the current member. This value, as described earlier, will cause the drilldown on Geography.

- You also set parameters for the Product and Time dimensions as well. By setting these values to the parent column, you are effectively preserving the state of the report (the children of a parent produces the current set of members). This allows browser operations like bookmarking to re-create the report in its drilldown state; these parameters appear in the URL.
- f. Click **Apply Changes**.
4. Add a drilldown link to Product:
- Click the **Edit** icon  for Product.
 - For Link Text, click **[Product]**.
 - For Page, enter **1** (that is, the current page number is “1”).
 - Use the flashlights  to enter name value pairs as follows:
- | Name | Value |
|---------------------|---|
| Item 1 P1_GEOGRAPHY | #GEOGRAPHY_PARENT#  |
| Item 2 P1_PRODUCT | #PRODUCT_DIM_KEY#  |
| Item 3 P1_TIME | #TIME_PARENT#  |
- e. Click **Apply Changes**.
5. Finally, add a drilldown link for Time:
- Click the **Edit** icon  for Time.
 - For Link Text, click **[Time]**.
 - For Page, enter **1** (that is, the current page number is “1”).
 - Use the flashlights  to enter name value pairs as follows:
- | Name | Value |
|---------------------|--|
| Item 1 P1_GEOGRAPHY | #GEOGRAPHY_PARENT#  |
| Item 2 P1_PRODUCT | #PRODUCT_PARENT#  |
| Item 3 P1_TIME | #TIME_DIM_KEY#  |
- e. Click **Apply Changes**.
- The Column Attributes window should look like this:
- | Column Attributes | | | | |
|---|-----------|--------|-------------------------------------|------------------------|
| | Heading | Type | Link | Display Text As |
|  | Geography | STRING | <input checked="" type="checkbox"/> | Standard Report Column |
|  | Product | STRING | <input checked="" type="checkbox"/> | Standard Report Column |
|  | Time | STRING | <input checked="" type="checkbox"/> | Standard Report Column |
|  | Sales | NUMBER | <input checked="" type="checkbox"/> | Standard Report Column |
6. In the Region Name pane, click **Apply Changes**.

7. Run the report from the Application Page. The report should look like this:

Product	Time	Sales	% Chg Pr Period	% Chg Pr Year	Product Alert	Ytd	Ytd % Chg Pr Year
Cameras and Camcorders	CY2005	392434	-	-	OKAY	392434	-
Portable Music and Video	CY2005	647626	-	-	OKAY	647626	-
Computers	CY2005	4199753	-	-	OKAY	4199753	-
Cameras and Camcorders	CY2006	474460	20.9	20.9	OKAY	474460	20.9
Computers	CY2006	5143957	22.4	22.4	ALERT	5143957	22.4
Portable Music and Video	CY2006	728666	12.5	12.5	OKAY	728666	12.5
Computers	CY2007	5620477	9.2	9.2	ALERT	5620477	9.2
Portable Music and Video	CY2007	775815	6.4	6.4	OKAY	775815	6.4
Cameras and Camcorders	CY2007	542472	14.3	14.3	OKAY	542472	14.3
Computers	CY2008	-	-	-	OKAY	-	-
Portable Music and Video	CY2008	-	-	-	OKAY	-	-
Cameras and Camcorders	CY2008	-	-	-	OKAY	-	-

8. For Africa, click the **Computers** link for CY2005. This is the row that had excellent percent growth in sales, but showed an Alert in the following data.

Computers	CY2005	4199753	-	-	OKAY
Cameras and Camcorders	CY2006	474460	20.9	20.9	OKAY
Computers	CY2006	5143957	22.4	22.4	ALERT
Portable Music and Video	CY2006	728666	12.5	12.5	OKAY
Computers	CY2007	5620477	9.2	9.2	ALERT
Portable Music and Video	CY2007	775815	6.4	6.4	OKAY
Cameras and Camcorders	CY2007	542472	14.3	14.3	OKAY

The resulting report shows the Computer division products that contain underlying alerts:

Geography : Africa							
Product	Time	Sales	% Chg Pr Period	% Chg Pr Year	Product Alert	Ytd	Ytd % Chg Pr Year
Total Personal Computers	CY2005	3419646	-	-	OKAY	3419646	-
All Computer Furniture	CY2005	499	-	-	OKAY	499	-
PDAs	CY2005	3969	-	-	OKAY	3969	-
Total Server Computers	CY2005	76129	-	-	OKAY	76129	-
Computer Printers and Supplies	CY2005	699508	-	-	OKAY	699508	-
Total Personal Computers	CY2006	4186144	22.4	22.4	ALERT	4186144	22.4
Total Server Computers	CY2006	102661	34.8	34.8	OKAY	102661	34.8
All Computer Furniture	CY2006	2665	434	434	OKAY	2665	434
PDAs	CY2006	3350	-15.6	-15.6	ALERT	3350	-15.6
Computer Printers and Supplies	CY2006	849136	21.3	21.3	OKAY	849136	21.3
Total Server Computers	CY2007	94985	-7.4	-7.4	OKAY	94985	-7.4
Total Personal Computers	CY2007	4578214	9.3	9.3	ALERT	4578214	9.3
All Computer Furniture	CY2007	1917	-28	-28	ALERT	1917	-28
PDAs	CY2007	3508	4.7	4.7	OKAY	3508	4.7
Computer Printers and Supplies	CY2007	941852	10.9	10.9	OKAY	941852	10.9

9. Feel free to experiment with the interactive report. If you have time, there is an optional exercise to examine on the following page.

Practice 8-4: Optional Interactive Report Steps

It is often useful to add a reset button to the report. This allows you to reset the report to its original state.

To add a reset button:

1. In the Page Editor, click **Create**  in the Buttons area.
2. Add the button to the **Explore** region and click **Next**
3. Select position **Create a button in a region** and click **Next**.
4. Specify button properties:
 - a. Button Name: **Reset**
 - b. Label: **Reset**
 - c. Select **HTML Button**, and then click **Next**
5. Add the button to the **Top and Bottom of Region** and click **Next**
6. Select branch to Page: **1** (the current page). Click **Create Button**.

Now that the button has been created, you need to specify an action for the button:

7. In the **Page Processing > Branches** section of the page editor, click **Go to Page**.
8. In the **Action** section of the page, enter “1” for **Clear Cache**. This will clear the parameter values for P1_PRODUCT, P1_TIME, and P1_GEOGRAPHY.
9. Click **Apply Changes**.
10. Run the page.

Result: You will see two reset buttons. Clicking these buttons will reset the report.

Practices for Lesson 9

In this practice, you will enable secured access to cube data. Specifically, you will perform two exercises:

1. Allow the DM user to see all of the data in the OLAPTRAIN sales cube.
2. Restrict DM's view of the data to a) indirect distribution channels and b) sales across the entire company.

You will use SQL Developer to perform the first exercise. The second exercise will be performed using AWM.

Practice 9-1: Grant DM User Access to Sales Cube

In this practice section, you connect to both the DM and OLAPTRAIN schemas using SQL Developer. The OLAPTRAIN user will grant the DM user access to all the objects required to access data in the SALES_CUBE plus the associated dimension views.

Follow these steps to grant the DM user access to the SALES_CUBE and the related dimension views:

1. In SQL Developer, open a connection to the database using the OLAPTRAIN user by clicking the **local (OLAPTRAIN)** node in the **Connections** pane. You will use the SQL Worksheet for the following steps.

2. Grant select privileges on the SALESTRACK analytic workspace:

```
grant select on AW$SALESTRACK to dm;
```

3. Grant select privileges on the dimensions and the dimension views:

```
grant select on product to dm;
grant select on product_view to dm;
```

```
grant select on time to dm;
grant select on time_view to dm;
```

```
grant select on geography to dm;
grant select on geography_view to dm;
```

```
grant select on channel to dm;
grant select on channel_view to dm;
```

4. Grant select privileges on the sales cube and its view:

```
grant select on sales_cube to dm;
grant select on sales_cube_view to dm;
```

5. Remember, some of the calculations in the SALES_CUBE refer to measures in the FORECAST cube. If you plan to query the “crossover” calculations, make sure you grant access to the FORECAST cube (and optionally, the FORECAST_VIEW) as well:

```
grant select on forecast to dm;
grant select on forecast_view to dm;
```

6. Open a connection to the database using the DM user by clicking the **local (DM)** node in the **Connections** pane.
7. In the SQL Worksheet for the following steps, select sales data from the SALES_CUBE:

```
select c.long_description,
       s.sales
  from olaptrain.sales_cube_view s,
       olaptrain.channel_view c
 where s.product = 'ALL_PRODUCTS'
   and s.geography = 'ALL_REGIONS'
   and s.time = 'ALL_YEARS'
   and s.channel = c.dim_key;
```

The results should look like the following:

The screenshot shows two SQL Workbench windows side-by-side. The left window, titled 'local (olaptrain)', contains SQL grants for the 'AW' user:

```
1 -- Grant access to the AW
2 grant select on AW$SALESTRACK to dm;
3
4 -- Grant access to the dimensions and their views
5 grant select on product to dm;
6 grant select on product_view to dm;
7 grant select on time to dm;
8 grant select on time_view to dm;
9 grant select on geography to dm;
10 grant select on geography_view to dm;
11 grant select on channel to dm;
12 grant select on channel_view to dm;
13
14 -- Grant access to cube and cube view
15 grant select on sales_cube to dm;
16 grant select on sales_cube_view to dm;
17
18 grant select on forecast to dm;
19 grant select on forecast_view to dm;
20
21
```

The right window, titled 'local (dm)', shows the execution of the provided query:

```
1 select c.long_description,
       s.sales
  from olaptrain.sales_cube_view s,
       olaptrain.channel_view c
 where s.product = 'ALL_PRODUCTS'
   and s.geography = 'ALL_REGIONS'
   and s.time = 'ALL_YEARS'
   and s.channel = c.dim_key;
```

The results pane displays the output:

LONG_DESCRIPTION	SALES
All Channels	417515017.27
Indirect	244942316.79
Direct	172572700.48
Catalog	80443485.61
New York Retail	15659829.02
Lisbon Retail	9150317.77
San Francisco Retail	23141158.51
Internet	164498831.18
London Retail	28676863.28
Shanghai Retail	8243332.07
Los Angeles Retail	16359393.03
Paris Retail	17852437.69

Practice 9-2: Limit the Channels That the DM User May View

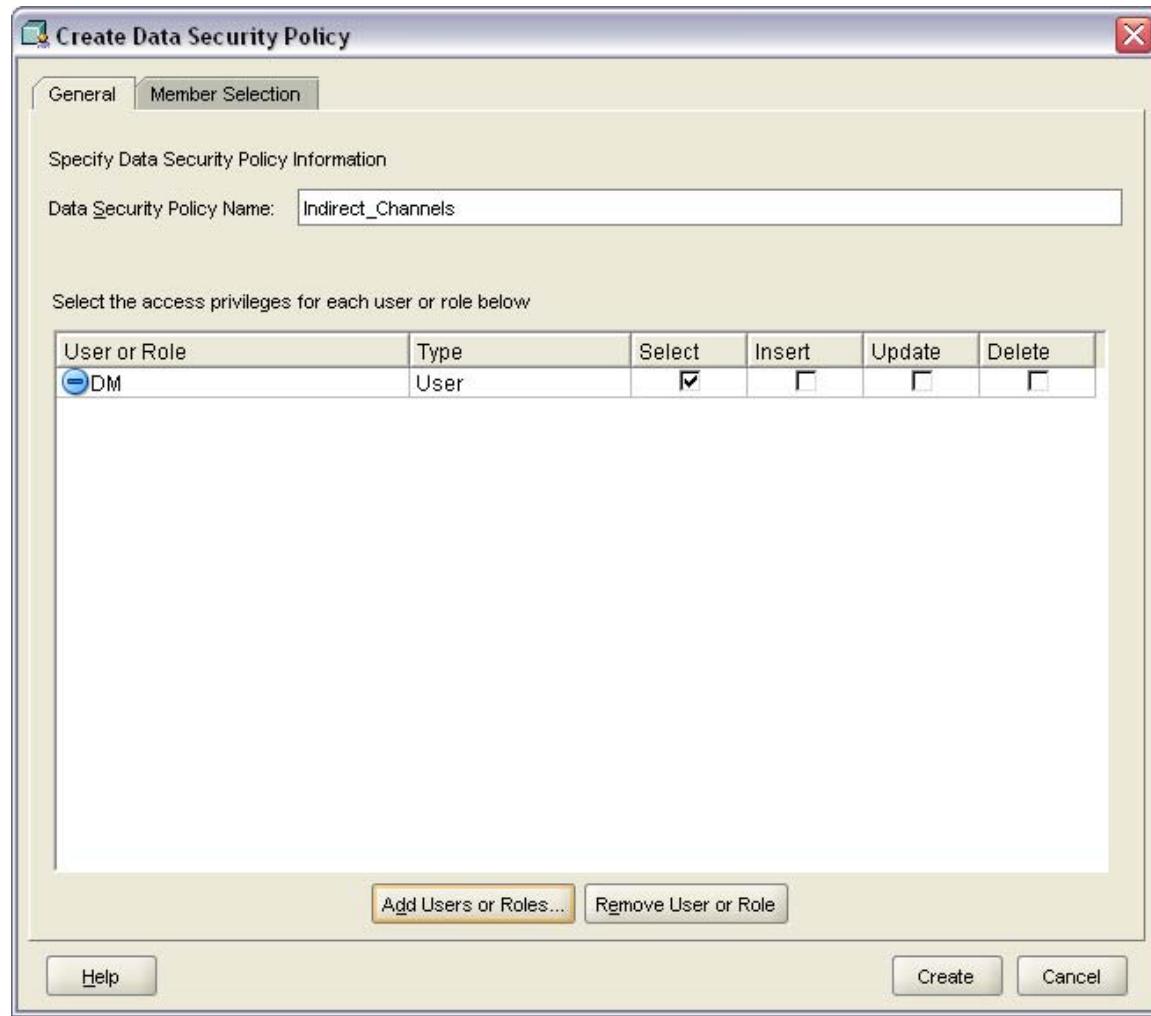
In this practice section, you will use AWM to define a data security policy over the Channel dimension.

The DM user will only be allowed to see the following Channel members:

- All Channels
- Indirect, Web
- Catalog
- Indirect Warehouse.

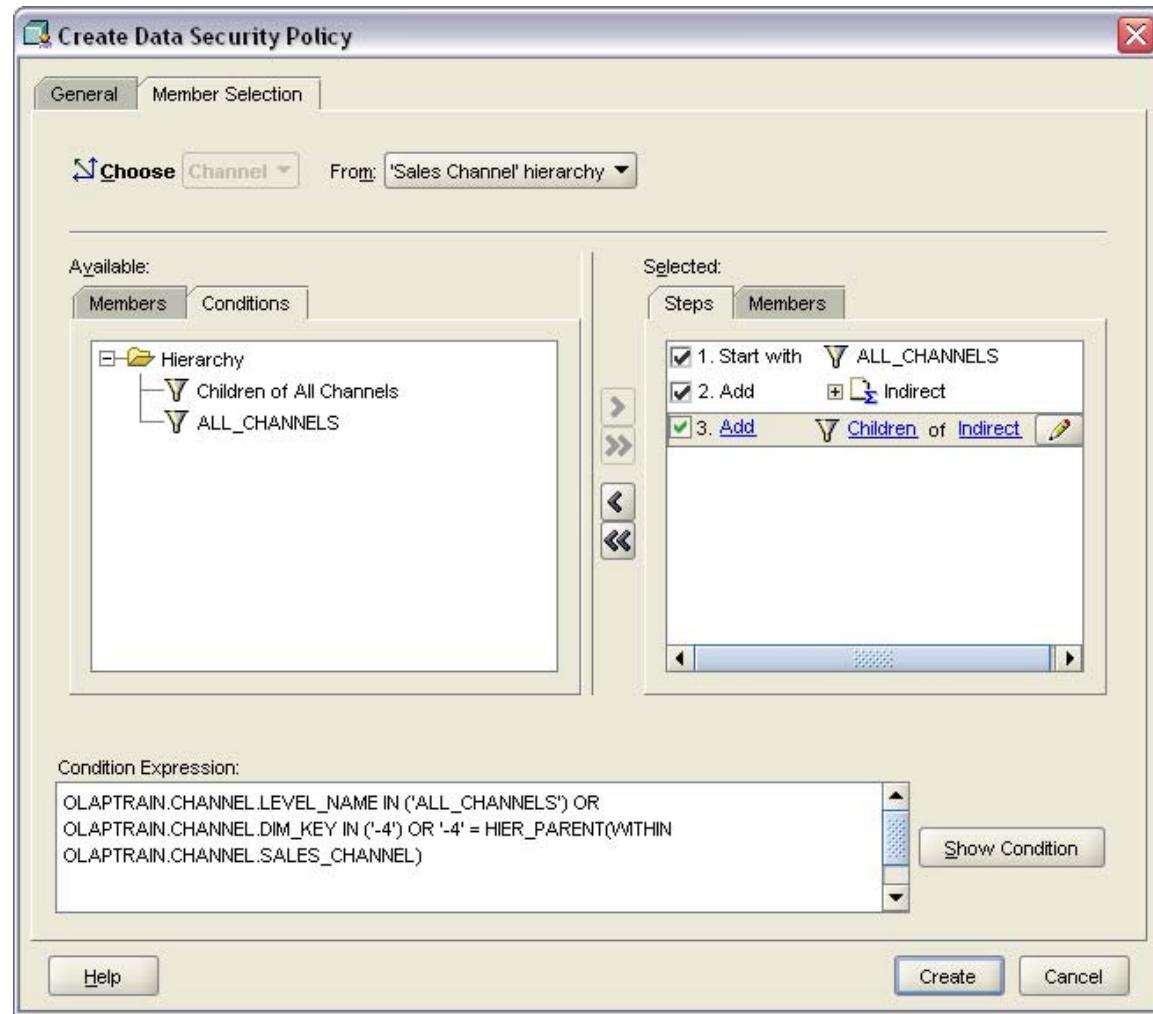
Follow these steps in AWM to limit the Channels that the DM user may view:

1. Right-click the **Data Security** node for the Channel dimension and select **Create Data Security Policy**.
2. In the General tab of the Create Data Security Policy dialog box, specify the following:
 - a. Name: **Indirect_Channels**
 - b. User or Role: **DM**
 - c. Ensure that the user has **Select** privileges, like this:



3. Click the **Member Selection** tab to identify the Channels that DM may select.
4. Define a three-step query using the Query Builder. Use the hypertext links to edit the conditions as follows:
 - a. In the Selected list, change the first step to **Start with ALL_CHANNELS**.
 - b. In the Members tab of the Available list, select **Indirect** and shuttle it to the Selected list.
 - c. In the Conditions tab of the Available list, select **Children of All Channels** and shuttle it to the Selected list.
 - d. In the third step of the Selected list, change **Children of All Channels** to **Children of Indirect**. (Hint: To do this, click the “All Channels” hyperlink, and select “Indirect” from the list.)
 - e. Click the **Show Condition** button to display the condition expression that this security policy will create.

The completed dialog box should look like this:



f. Click **Create**.

5. Go back to SQL Developer, and log off of DM. Then log back on as DM.
6. Re-execute the following query using the DM connection:

```
select c.long_description,
       s.sales
  from olaptrain.sales_cube_view s,
       olaptrain.channel_view c
 where s.product = 'ALL_PRODUCTS'
   and s.geography = 'ALL_REGIONS'
   and s.time = 'ALL_YEARS'
   and s.channel = c.dim_key;
```

The screenshot shows the Oracle SQL Developer interface. The top bar indicates the connection is 'local (dm)'. Below it is a toolbar with various icons. The main area has a status bar showing '0.35445037 seconds' and a progress bar. The 'Enter SQL Statement:' pane contains the SQL query shown above. The 'Results' tab is selected, displaying a table with four rows. The table has two columns: 'LONG_DESCRIPTION' and 'SALES'. The data is as follows:

LONG_DESCRIPTION	SALES
1 All Channels	417515017.27
2 Indirect	244942316.79
3 Catalog	80443485.61
4 Internet	164498831.18

Note:

- The results have changed; DM only has access to the “All Channels” and the indirect channels.
- Every view in the database that uses the Channel dimension will now be automatically scoped.