Techno Level 1: Basic Level

1. Overview of OBIEE Real Time Project

1.1. Why OBIEE?

- ♣ As an excellent source to get a job quickly.
- ♣ To get excellent and handsome salary.
- ♣ The Future is secured and steady in OBIEE Stream.
- ♣ Help in Settle quickly.
  Refer From Fig 1-5

1.2. What is OBIEE?

Oracle Definition:

Oracle Business Intelligence Enterprise Edition 11g (OBIEE) is an unmatched and comprehensive business intelligence and analytics platform that delivers a full range of capabilities - including interactive dashboards, ad hoc queries, mobile analytics, notifications and alerts, enterprise and financial reporting, scorecard and strategy management, business process invocation, unstructured search and collaboration, integrated systems management and more. OBIEE 11g is built on a proven and modern technological foundation that supports the highest workloads and most complex deployments, while providing timely insights to users across an enterprise at a low overall total cost of ownership.

Our Definition:

OBIEE or OBI is a reporting tool or application used by business users to query or requests the organization data stored over a period of time or in data warehouse or OLAP to obtain answers or results or reports for the analysis of business and make better decisions.

Fig6

1.3. OBIEE Real Time Project Types

1.3.1. OBIEE without ETL (Directly from OLTP or any Database)

Fig7

1.3.2. OBIEE with ETL (OLAP using ETL Tools like Informatica, DAC, and ODI)

Fig8

1.3.3. OBIEE with OBIA or Oracle BI Apps (Using OBIA 7.9.6.4, OBIA 11.1.1.10.1)

Fig9

1.4. OLTP vs OLAP

OLTP (OnLine Transaction Processing)		OLAP (OnLine Analytical Processing)	
1.	Source of data is transactions.	1.	Source of data is from various OLTP Databases.
2.	It is useful to store transactional or operational data.	2.	It is useful to store consolidated data.
3.	The nature of data is current and Detail.	3.	The nature of data is historical and summarized.
4.	Useful to RUN the Business.	4.	Useful to ANALYZE the Business.
5.	OLTP supports CRUD(create,Read,update,delete)	5.	OLAP supports Read only.
6.	OLTP is volatile.	6.	OLAP is non-volatile
7.	Querying or processing is very fast.	7.	Querying or processing depends on amount of data. Improved by creating indexes.
8.	OLTP uses normalized schemes.	8.	OLAP mostly uses De-normalized schemes.
9.	OLTP is isolation depends on Application.	9.	OLAP is integrated as per subject area.
10.	OLTP is accessed by large number of users. Ex: Customers, Employees...	10.	OLAP is accessed by less number of users. Ex: CEO, CFO, COO, GM, Managers…

I1. What is multi-dimensional data source?-----Brillio (Bangalore)

1.5. OBIEE 11g Installation

1.5.1. Certification Matrix

Fig 10

1.5.2. Downloads (OTN & edelivery)

i. Go to **Oracle.com** or **edelivery.oracle.com**

ii. Get from Front Office in Institute.

1.5.3. Installation Types

Fig 11 and Fig12

1.5.4. Installation

1.5.4.1. On Windows Server

Refer to 1.OBIEE Installation Folder provide to you.

1.5.4.2. On Linux Server

Refer to 1.OBIEE Installation Folder provide to you.

1.5.4. Bugs and Patches (Oracle Support)

For Bugs and Patches login to Oracle Support. Just Type Oracle Support in any web browser and login.

I2. Did you work any Sev-1 Issue?-----Dell (Bangalore)

Try Interview Level – Expected Questions

Section: I. Introduction

2. OBIEE Introduction

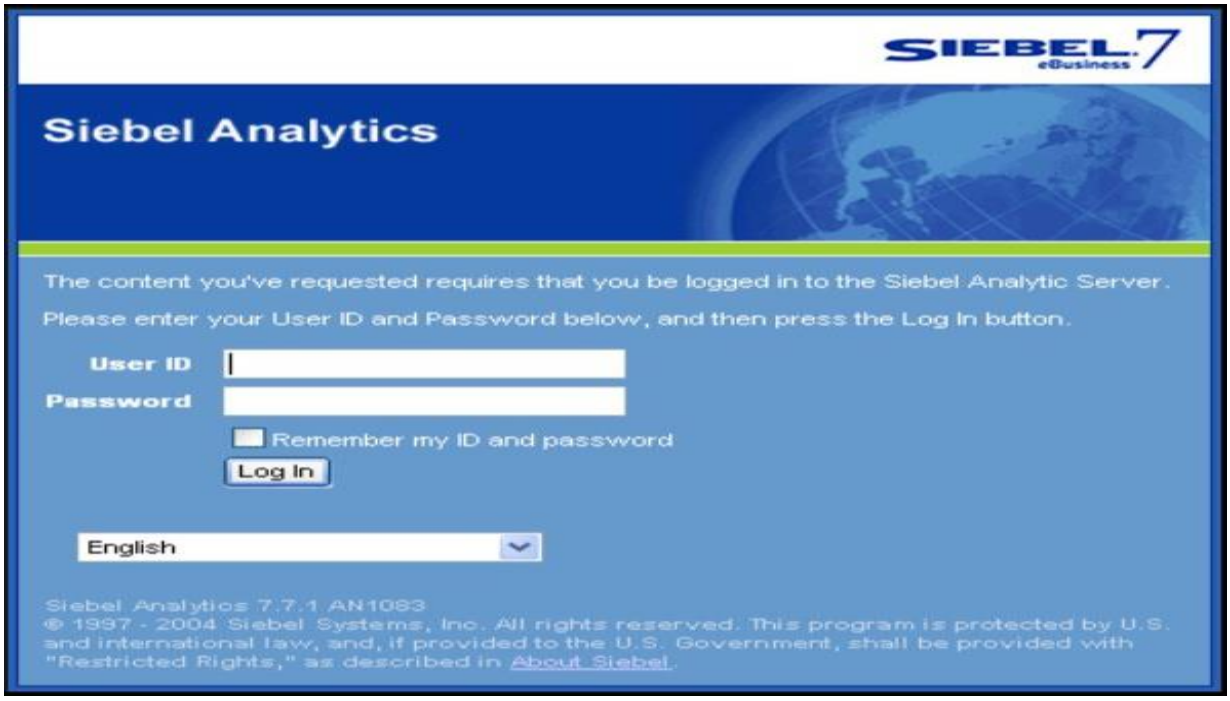
2.1. History of OBIEE

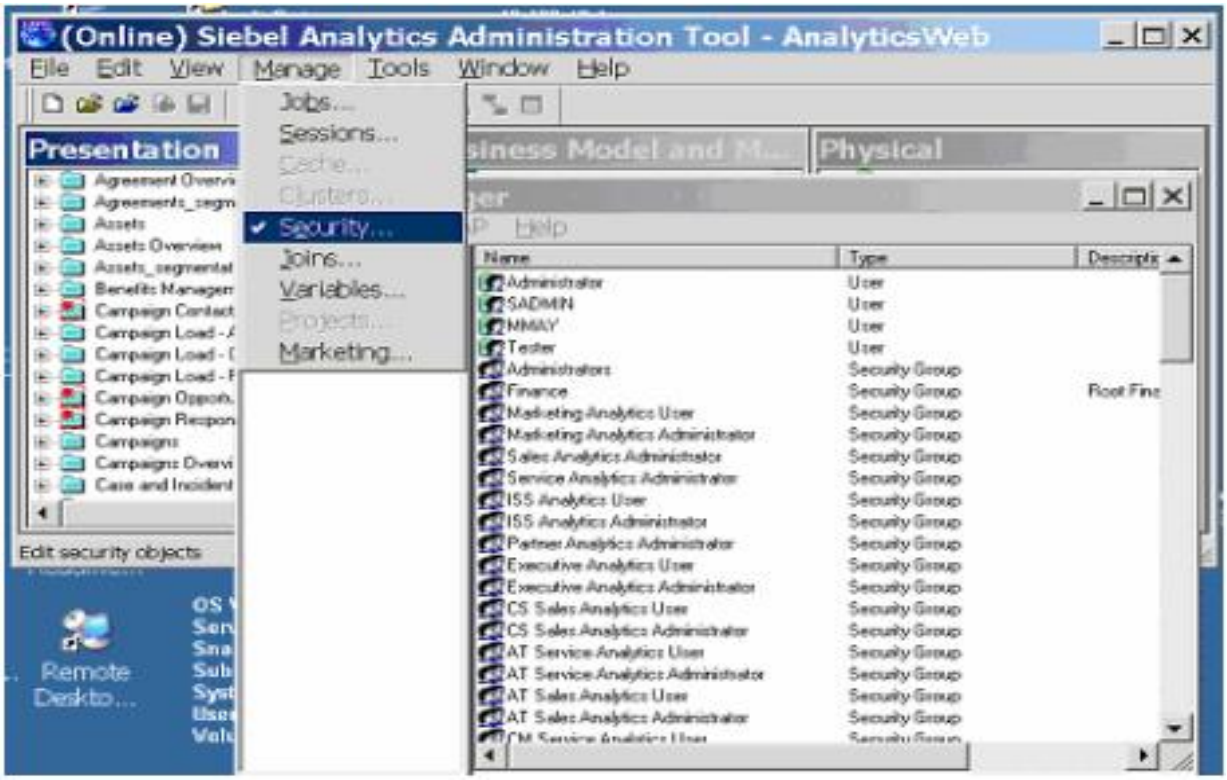
Siebel Systems Inc. acquired privately-held **nQuire Software Inc**. and integrated and shipped nQuire's scalable analytic server and intelligent Web products with Siebel 7, the e-business application suite.

San Mateo, California and Minnesota-based nQuire launched its nQuire suite of products in late 1999 to monitor and deliver real-time intelligence based on any fact-based problem or opportunity by initiating an e-mail or page to any device, such as a laptop, pager, PDA, RIM, U.S. and European mobile phones.

Siebel 7 is an integrated suite of applications for customer relationship management (CRM -- including applications for partner relationship management (PRM) and employee relationship management (ERM) -- all based on a common Web architecture.

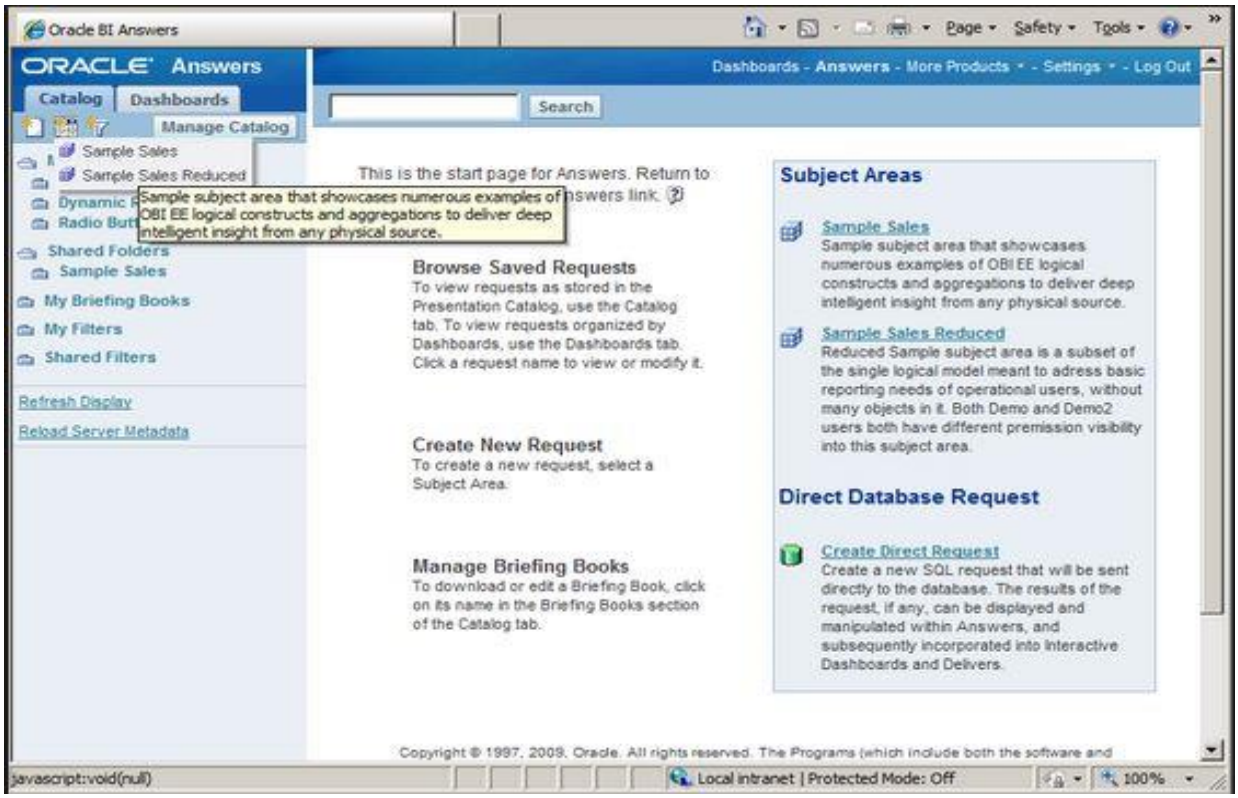
Siebel Enterprise Analytic Platform 7.7 includes Siebel Analytics Server for calculation and integration, logical business model and metadata management, and caching. Other components include the Intelligent Interaction Management module for data mining; Interactive Dashboard, which provides a portal interface to the data.


SIEBEL 7
eBusiness
Siebel Analytics
The content you've requested requires that you be logged in to the Siebel Analytic Server.
Please enter your User ID and Password below, and then press the Log In button.
User ID
Password
Remember my ID and password
Log In
English
Siebel Analytics 7.7.1 AN1083
© 1997 - 2004 Siebel Systems, Inc. All rights reserved. This program is protected by U.S. and international law, and, if provided to the U.S. Government, shall be provided with "Restricted Rights," as described in About Siebel.

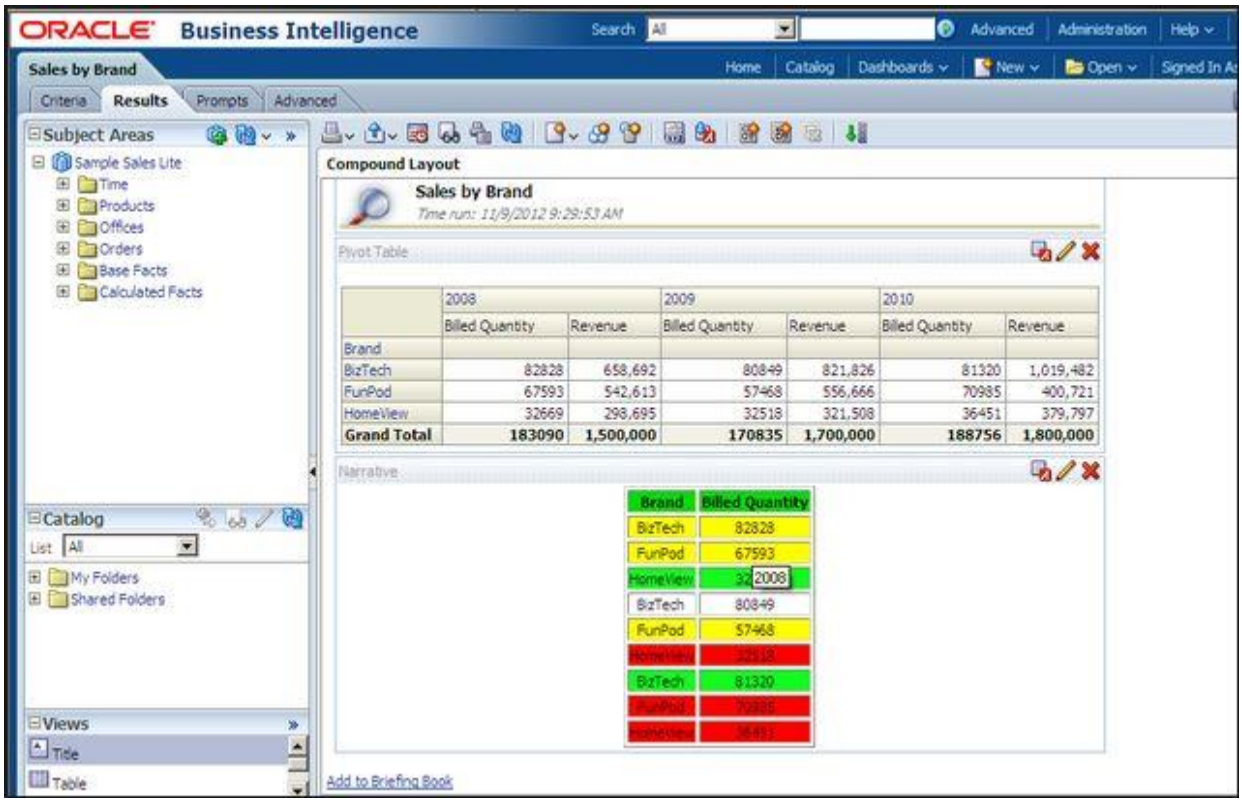

(Online) Siebel Analytics Administration Tool - AnalyticsWeb
File Edit View Manage Tools Window Help
Jobs...
Sessions...
Cache...
Clusters...
Security...
Joins...
Variables...
Projects...
Marketing...
Presentation
Physical
Name
Type
Administrator
SADMIN
NMAY
Tester
Administrators
Finance
Marketing Analytics User
Marketing Analytics Administrator
Sales Analytics Administrator
Service Analytics Administrator
ISS Analytics User
ISS Analytics Administrator
Partner Analytics Administrator
Executive Analytics User
Executive Analytics Administrator
CS Sales Analytics User
CS Sales Analytics Administrator
AT Service Analytics User
AT Service Analytics Administrator
AT Sales Analytics User
AT Sales Analytics Administrator
Edit security objects
Remote Deskto...

In 2006, Oracle purchased Seibel analytics and renamed as **Oracle Business Intelligence 10g** (OBIEE 10g)

OBIEE 10g:



OBIEE 11g:



Summary:

- 2000: nQuire formed
- 2001: nQuire courted by Siebel.
- 2002: Siebel purchases nQuire
- 2005: Oracle purchases Siebel 7.8.3.1
- 2006: OBIEE 10G or Siebel 7.8.4
- 2010: OBIEE11g Version Released
- 2015: OBIEE 12c Version Released

Following are major releases:

Siebel Analytics 7.0 – 2002
Siebel Analytics 7.5 – 2003
Siebel Analytics 7.7 – 2004
Siebel Analytics 7.8.2 and 7.8.3 – 2005
Siebel Analytics / Oracle Business Intelligence 7.8.4 and 7.8.5 – 2006
OBIEE 10gR3, 10.1.3.2 – Jan 2007 (MAUI)
OBIEE 10gR3, 10.1.3.2.1 – Apr 2007
OBIEE 10gR3, 10.1.3.3.0 – Aug 2007
OBIEE 10gR3, 10.1.3.3.1 – Oct 2007
OBIEE 10gR3, 10.1.3.3.2 – Dec 2007
OBIEE 10gR3, 10.1.3.3.3 – May 2007

OBIEE 10gR3, 10.1.3.4 – Aug 2008
OBIEE 10gR3, 10.1.3.4.1 – Apr 2009
OBIEE 11g-July 2010
OBIEE 12c-Oct 2015

OBIEE 11g versions
1. OBIEE 11.1.1.3.0
2. OBIEE 11.1.1.5.0
3. OBIEE 11.1.1.6.0
4. OBIEE 11.1.1.7.0
5. OBIEE 11.1.1.9.0

OBIEE 12c versions
1. OBIEE 12.2.1.0.0
2. OBIEE 12.2.1.1.0 (latest)

2.2. Components and Types

OBIEE components are mainly divided into two types of components −

- Server Components
- Client Components

Server Components:

Will be discussed in Section Titled: 7. OBIEE Architecture

Client Components:

There are two types, Windows based or Non Web based (Non Browser based) clients.

Non Web based (Non Browser based) clients

1. BI Administration tool:

The Oracle BI Administration Tool is a Windows application that you can use to create and edit repositories (RPD).The Administration Tool can connect directly to the repository in offline mode, or it can connect to the repository through the Oracle BI Server(online mode).

Fig14

2. Catalog Manager:

Catalog Manager is a tool that lets you perform online and offline management of Presentation Catalogs and Catalog items like reports, dashboards, permissions, Renaming reports, Migration etc.

3. Job Manager:

The Job Manager is a Windows tool that is the interface with the Oracle BI Scheduler. Through Job Manager, you can connect to, start and stop the Oracle BI Scheduler, add and manage jobs, and manage job instances.

4. ODBC Client:

Open Database Connectivity (ODBC) is an industry standard interface for connecting to databases. A Data Source Name (DSN) is used to store the information about connecting to a given database as a given database user over ODBC.

Ex: nqcmd, BI analytics etc.

Fig13

Web based or Browser based clients

1. BI Analytics (11g)/BI Answers (10g):

Oracle BI Analysis Editor is a set of graphical tools that are used to build, view, and modify Oracle BI analyses. The analysis are queries against an organization's data. It allows us to include the views that you create in an analysis for display in dashboards.

2. BI Interactive Dashboards:

Dashboards provide personalized views of corporate and external information.A dashboard consists of one or more pages. Pages can display anything that you can access or open with a web browser, including, the results of analyses, Images,Text etc.,

3. BI Delivers:

BI Delivers is the interface that is used to create alerts based on Oracle Business Intelligence Analyses. You can use Delivers to detect specific results and notify appropriate persons or groups using the web, wireless, mobile, and other communication channels.

4. BI Publisher:

Oracle BI Publisher is the reporting solution to author, manage, and deliver all pixel-perfect standardized reports and documents easier and faster than traditional reporting tools.

5. OFMW Enterprise Manager:

Fusion Middleware Control is a web based,graphical user interface that used to centrally manage, monitor, and configure Oracle Business Intelligence system

components. For example, the Oracle BI Server, Oracle BI Presentation Services, and Oracle BI Scheduler.

6. OWLS Administration Console:

The Administration Console is a Web browser-based, graphical user interface that you use to manage a WebLogic Server domain. WebLogic Server Administration Console enables you to monitor status, configure security and manage the Administration Server and Managed Servers and more

2.3 Data Modelling Elements

2.3.1. Columns

String Columns:

i) Dimensional Columns

Dimensional Columns or Dimensions are categories of attributes by which the business is defined.

EX: Time periods, Products, Customers

Within a given dimension, there may be many attributes. For example, the time period dimension can contain the attributes day, week, month, quarter, and year. Exactly what attributes a dimension contains depends on the way the business is analyzed.

I3. What is a dimension?------ Oracle India (Bangalore),Brillio (Bangalore),TCS (Bangalore)

I4. What is a Conformed dimension? ------Genpact (Hyderabad),TCS (Bangalore)

I5. What is a Slowly Changing Dimensions (SCD)? ---capegemini (Mumbai)

I6. What is a degenerate dimension? ------capegemini (Mumbai)

I7. What is a junk dimension?------ Sapient (Bangalore)

ii) Dimensional Hierarchical Columns

Dimensional Hierarchical Columns typically contain hierarchies, which are sets of top-down relationships between members within a dimension.

Ex: Year→quarter→month→week→day

Measure or Numeric Columns:

i) Fact Columns or Measure:

Facts are the metrics that business users would use for making business decisions by joining them with their associated dimensions. Generally, facts are mere numbers.

Ex: Revenue, shipped quantity, billed quantity

Note:

The data in the dimension tables are less compared to the data in the fact tables. The data in the dimension table is static and descriptive in nature whereas the fact table contains numeric and will change regularly. Facts tables are the key performance indicators of the business.

I8. What is a fact or measure?----Brillio (Bangalore), Oracle India (Bangalore),Dell (Bangalore)

I9. What are fact types?

I10. What is an Implicit fact column?--- Genpact (Hyderabad),Oracle India(Bangalore),Accenture (Kolkata)

ii) Aggregated Columns:

Aggregated columns stores precomputed (typically summed) or calculated results of facts.

Ex: Total Revenue, Total Shipped quantity

Key Columns:

A Key columns are used to identify each row/record in a database table. Or if after summing a numeric column is given a valid value then it is a Fact column otherwise it is a key column.

Ex: Primary key, foreign key, Surrogate key, Chronological key

I11. What is primary key and foreign key?

I12. What is Surrogate key or ROW_WID, Candidate Key and Super Key?— Genpact (Hyderabad)

I13. What is Chronological Key?------ TCS (Bangalore),HCL (Chennai)

Note:

Cubes:

In Mutli dimensional Source are made up of Cubes. Cubes are made up of measures and organized by dimensions. Because they are already dimensional, each cube maps easily to the logical fact and dimension tables in the business model.

I14. What is a Cube? Do you have used Cubes?---- Brillio (Bangalore)

2.3.2. Tables

i) Dimensional Tables:

Dimension tables contain attributes that describe business entities (such as Customer Name, Region, Address, Country and so on). Dimension tables also contain primary keys that identify each member.

EX: W_CUSTOMER_D, W_ORGANIZATION_D, W_PO_VENDORS_D

ii) Dimensional Hierarchical Tables:

Dimensional Hierarchical Tables has Columns typically contain hierarchies, which are sets of top- down relationships between members within a dimension.

Ex: W_CALENDER_DH, W_PRODUCTS_DH

I15. What are the tables or Dimensions or fact tables used in your projects or in your Data warehouse?------ Accion Labs (Malaysia),Deyaar (Dubai)

iii) Fact Tables:

Fact Tables contains facts or measures which are the key performance indicators of the business when joined with related dimensions.

Ex: W_AP_INVOICE_F, W_GL_COGS_F, W_ORDERS_F

I16. Did you created any dummy tables in rpd? Why? ---- TCS (Bangalore)

I17. What is Factless fact table? ------- TCS (Bangalore)

iv) Aggregated Tables:

Aggregate tables store precomputed results that are aggregated measures (typically summed) over a set of dimensional attributes. Using aggregate tables is a typical technique used to improve query response times in decision support systems.
Ex: W_AP_INVOICE_A, W_GL_COGS_A, W_ORDERS_A

2.3.3. Schemes

A Scheme is a collection of database objects (tables, views, sequences, indexes etc.,) in a database. There may be one or more schemes in a database.

Fig14

OBIEE Schemes

1. Star Scheme:

A star schema is a set of dimensional schemas (stars) that each have a single fact table with foreign key join relationships to several dimension tables

Fig15

2. Snow flake Scheme:

A snow flake schema is a set of dimensional schemas (stars) that each have a single fact table with foreign key join relationships to several dimension tables and primary key join with other dimensional tables.

Fig16

3. Normalized Schemas:

Normalized schemas (OLTP) distribute data entities into multiple tables to minimize data storage redundancy and optimize data updates

Fig17

4. Fully Denormalized Schemas:

This type of dimensional schema combines the facts and dimensions as columns in one table.

Ex: Flat files (excel), prebuilt summarization or OLAP cubes, star scheme, snow flake schemes.

5. Mixed schemes:

It is a combination of star scheme and snow flake schemes joined by means of conformed dimensions. It is combination of some stars or snowflakes or both.

I18.What are Schemes used in OBIEE? Difference between Star and Snowflake?-

------ Oracle India (Hyderabad), Brillio (Bangalore), Genpact (Hyderabad)

Try Interview Level – Expected Questions

Section: I. Introduction

3. BI ADMIN TOOL

I.19. On a scale of 5, how you rate yourself in BI Admin Tool? [OR]

On a scale of 10, how you rate yourself in OBIEE? --- Oracle India (Hyderabad),PWC (Bangalore), Capegemini (Mumbai),Accenture (Kolkata),Dubai Islamic Bank (UAE).

3.1. Physical Layer

1) The Physical layer defines the data sources to which Oracle BI Server submits queries and the relationships between physical databases and other data sources that are used to process multiple data source queries.
2) The recommended way to populate the Physical layer is by importing metadata from databases and other data sources. The data sources can be of the same or different varieties.
3) You can import schemas or portions of schemas from existing data sources. Additionally, you can create objects in the Physical layer manually.
4) When you import metadata, many of the properties of the data sources are configured automatically based on the information gathered during the import process.
5) After import, you can also define other attributes of the physical data sources, such as join relationships, that might not exist in the data source metadata.
6) There can be one or more data sources in the Physical layer, including databases, flat files, XML documents, and so forth. Here we will import and configure tables from the BISAMPLE schema from Oracle Database.

3.1.1. Hands on Main Objectives

To build the Physical layer of a repository, you perform the following steps:

1. Creating a New Repository
2. Importing Metadata
3. Verifying Connection
4. Creating Aliases
5. Creating Physical Keys and Joins

Creating a New Repository

1. Select Start > Programs > Oracle Business Intelligence > BI Administration to open the Administration Tool.
2. Select File > New Repository
3. Select the Binary method
4. Enter the Name for the Repository i.e., BISAMPLE.
5. Leave the default location as is. It points to the default repository directory.

C:\OBIEE11g_HOME\instances\instance1\bifoundation\OracleBIServerComponent\coreapplication_obis1\repository

6. Leave Import Metadata set to Yes.
7. Enter and retype a password for the repository. BISAMPLE123 is the repository password.
8. Click Next.

Importing Metadata

1. Change the Connection Type to OCI 10g/11g. The screen displays connection fields based on the connection type you selected.
2. Enter a data source name. In this example the data source name is orcl. This name is the same as the tnsnames.ora entry for this Oracle database instance.
3. Enter user name and password for the data source. In this example the username and password are both BISAMPLE.
4. Click Next.
5. Accept the default metadata types and click Next.
6. In the Data source view, expand the BISAMPLE schema.
7. Use Ctrl+Click to select the following tables from BISAMPLE schema:

SAMP_ADDRESSES_D
SAMP_CUSTOMERS_D
SAMP_PRODUCTS_D
SAMP_REVENUE_F
SAMP_TIME_DAY_D

8. Click the Import Selected button to add the tables to the Repository View.
9. The Connection Pool dialog box appears. Accept the defaults and click OK.

10. The Importing message appears.

11. When import is complete, expand BISAMPLE in the Repository View and verify that the five tables are visible.

12. Click Finish to open the repository

13. Expand orcl > BISAMPLE and confirm that the five tables are imported into the physical layer of the repository.

I20. What is a Connection Pool? How many are there in your project?---- Oracle India (Hyderabad), HCL (Chennai)

I21. How can we know the features supported by database in OBIEE?---Oracle India (Hyderabad)

I22. Do you Isolation Levels? What are they?----- Capegemini (Mumbai)

Verifying Connection

1. Select Tools > Update All Row Counts.

2. When update row counts completes, move the cursor over the tables and observe that row count information is now visible, including when the row count was last updated.

3. Expand tables and observe that row count information is also visible for individual columns.

4. Right-click a table and select View Data to view the data for the table.

5. Close the View Data dialog box when you are done. It is a good idea to update row counts or view data after an import to verify connectivity. Viewing data or updating row count, if successful, tells you that your connection is configured correctly.

Creating Aliases

1. It is recommended that you use table aliases frequently in the Physical layer to eliminate unrelated joins and to include best practice naming conventions for physical table names. Right-click SAMP_TIME_DAY_D and select New Object > Alias to open the Physical Table dialog box.

2. Enter D1 Time in the Name field

3. In the Description field, enter Time Dimension Alias at day grain. Stores one record

for each day.

4. Click the Columns tab. Note that alias tables inherit all column definitions from the source table

5. Click OK to close the Physical Table dialog box.

6. Repeat the steps and create the following aliases for the remaining physical tables.

SAMP_ADDRESSES_D = **D4 Address**
SAMP_CUSTOMERS_D = **D3 Customer**
SAMP_PRODUCTS_D = **D2 Product**
SAMP_REVENUE_F = **F1 Revenue**

I23. What is an Alias? Difference between Alias and Duplicate?---- Genpact (Hyderabad)

Creating Keys and Joins

1. Select the five alias tables in the Physical layer
2. Right-click one of the highlighted alias tables and select **Physical Diagram > Selected Object(s) Only** to open the Physical Diagram. Alternatively, you can click the Physical Diagram button on the toolbar.
3. Rearrange the alias table objects so they are all visible
4. You may want to adjust the objects in the Physical Diagram. If so, use the toolbar buttons to zoom in, zoom out, fit the diagram, collapse or expand objects, select objects, and so forth:
5. Click the **New Join** button on the toolbar.
6. Click the **F1 Revenue** table and then the **D1 Time** table. The **Physical Foreign Key** dialog box opens. It matters which table you click first. The join creates a **one-to-many (1:N)** relationship that joins the key column in the first table to a foreign key column in the second table.
7. Select the **D1 Time. CALENDAR_DATE** column, and then select **F1 Revenue.BILL_DAY_DT** to join the tables. Ensure that the Expression edit box (at the bottom) contains the following expression:
   "orcl"."""."BISAMPLE"."D1 Time"."CALENDAR_DATE" =
   "orcl"."""."BISAMPLE"."F1 Revenue"."BILL_DAY_DT"

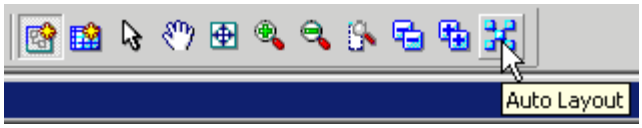
8. Click **OK** to close the Physical Foreign Key dialog box. The join is visible in the Physical Diagram.
9. Repeat the steps to create joins for the remaining tables. Use the following expressions as a guide. Please notice that **D4 Address** joins to **D3 Customer**.

"orcl"."."."BISAMPLE"."D2 Product"."PROD_KEY" = "orcl"."."."BISAMPLE"."F1 Revenue"."PROD_KEY"

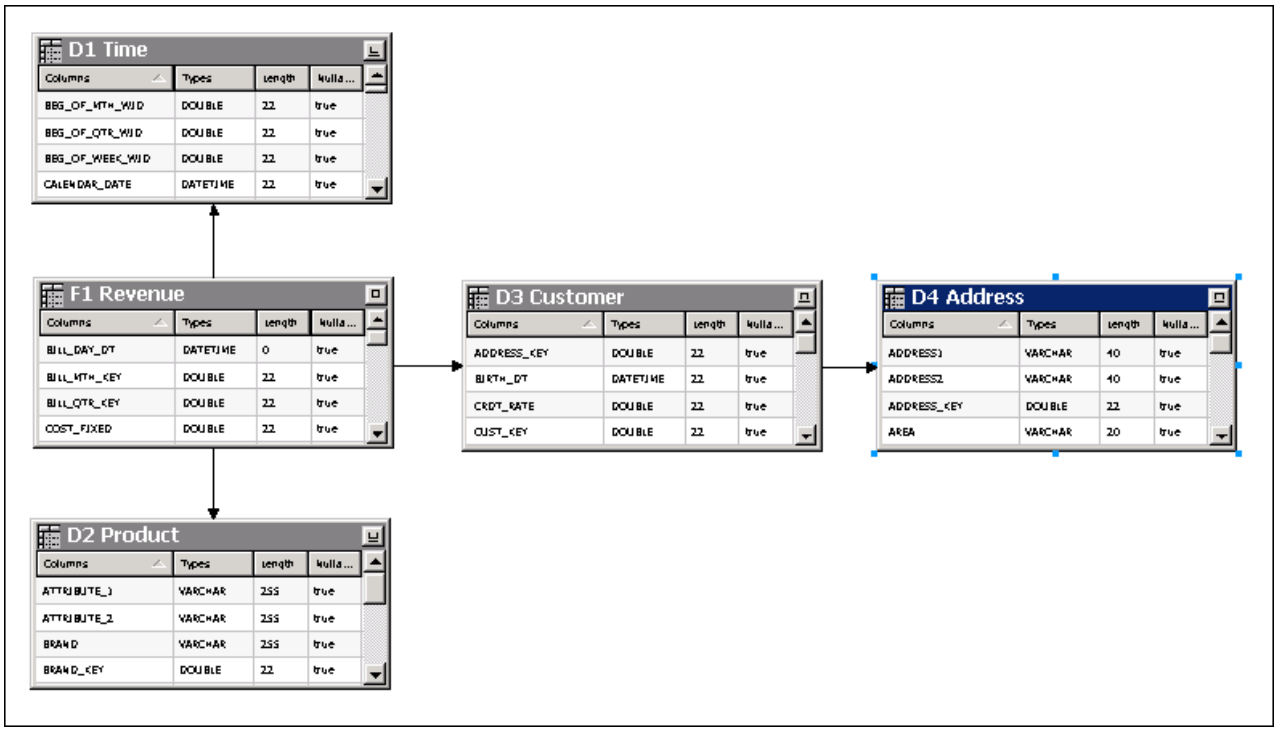
"orcl"."."."BISAMPLE"."D3 Customer"."CUST_KEY" = "orcl"."."."BISAMPLE"."F1 Revenue"."CUST_KEY"

"orcl"."."."BISAMPLE"."D4 Address"."ADDRESS_KEY" = "orcl"."."."BISAMPLE"."D3 Customer"."ADDRESS_KEY"

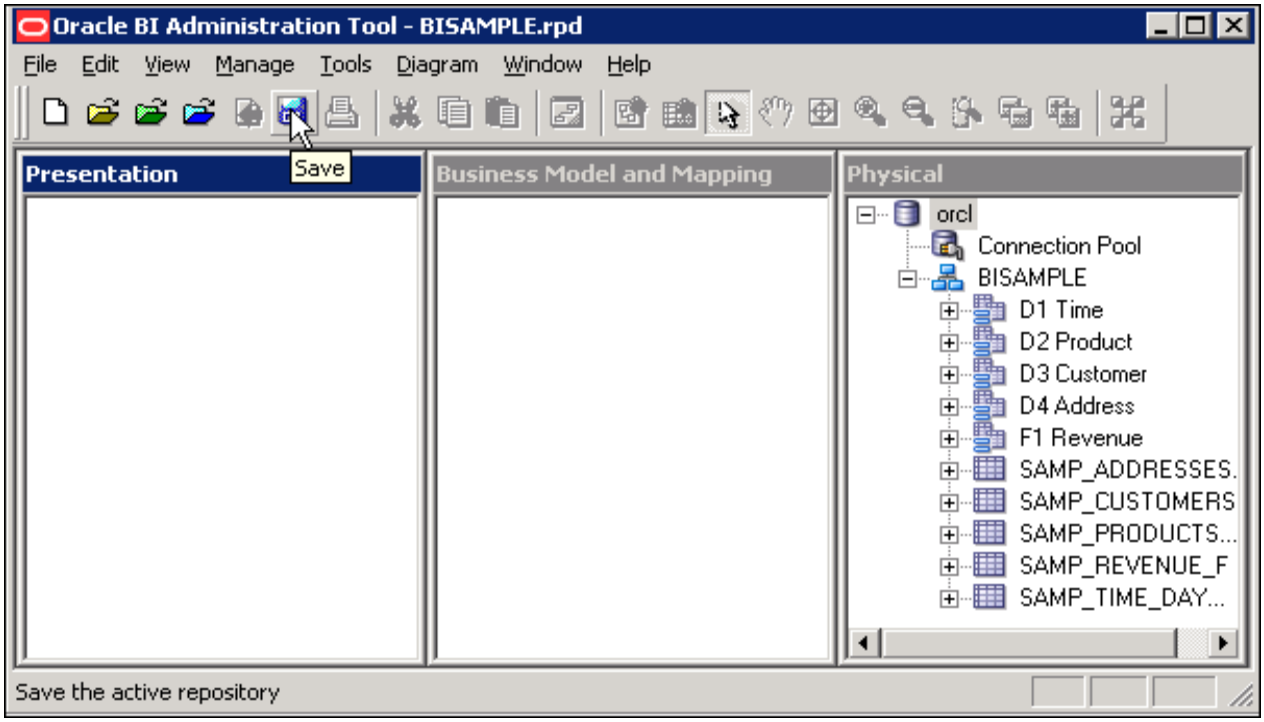
10. Click the **Auto Layout** button on the toolbar.



11. Your diagram should look similar to the screenshot:



12. Click the X in the upper right corner to close the Physical Diagram.
13. Select File > Save or click the Save button on the toolbar to save the repository.



13. Click No when prompted to check global consistency.

Note : Some of the more common checks are done in the Business Model and Mapping layer and Presentation layer. Since these layers are not defined yet, bypass this check until the other layers in the repository are built.

I24. What are joins in OBIEE?---- Genpact (Hyderabad),HCL (Chennai),Oracle India (Hyderabad), Sapient (Banglore),KPMG (Bangalore)

I25. Can we create tables in physical Layer or Did you created any views in OBIEE?------ Genpact (Hyderabad),Capegemini (Bangalore)

3.2. BMM Layer

1. The Business Model and Mapping layer (BMM) of the Administration Tool defines the business, or logical model of the data and specifies the mappings between the business model and the Physical layer schemas.
2. This layer is where the physical schemas are simplified to form the basis for the users' view of the data.
3. The Business Model and Mapping layer of the Administration Tool can contain one or more business model objects.
4. A business model object contains the business model definitions and the mappings from logical to physical tables for the business model.
5. The main purpose of the business model is to capture how users think about their business using their own vocabulary.
6. The business model simplifies the physical schema and maps the users' business vocabulary to physical sources. Most of the vocabulary translates into logical columns in the business model.

3.2.1. Hands on Main Objects

To build the Business Model and Mapping layer of a repository, you perform the following steps:

1. Creating a Business Model
2. Examining Logical Joins
3. Examining Logical Columns
4. Examining Logical Table Sources
5. Renaming Logical Objects Manually
6. Renaming Logical Objects Using the Rename Wizard
7. Deleting Unnecessary Logical Objects
8. Creating Simple Measures

Creating a Business Model

1. Right-click the white space in the Business Model and Mapping layer and select New Business Model to open the Business Model dialog box.

2. Enter Sample Sales in the Name field. Leave Disabled checked.

3. Click OK. The Sample Sales business model is added to the Business Model and Mapping layer.

4. In the Physical layer, select the following four alias tables:

D1 Time
D2 Product
D3 Customer
F1 Revenue
Do not select D4 Address at this time.

5. Drag the four alias table from the Physical layer to the Sample Sales business model in the Business Model and Mapping layer. The tables are added to the Sample Sales business model.

Note:

Notice that the three dimension tables have the same icon, whereas the F1 Revenue table has an icon with a # sign, indicating it is a fact table.

Examining Logical Joins

1. Right-click the Sample Sales business model and select Business Model Diagram > Whole Diagram to open the Business Model Diagram.

2. If necessary, rearrange the objects so that the join relationships are visible.

Note: Because you dragged all tables simultaneously from the Physical layer onto the business model, the logical keys and joins are created automatically in the business model. This is because the keys and join relationships were already created in the Physical layer. However, you typically do not drag all physical tables simultaneously, except in very simple models. Later in this tutorial, you learn how to manually build logical keys and joins in the Business Model and Mapping layer. The process is very similar to building joins in the Physical layer.

3. Double-click any one of the joins in the diagram to open the Logical Join dialog box. In this example the join between D1 Time and F1 Revenue is selected.

4. Click OK to close the Logical Join dialog box.

5. Click the X to close the Business Model Diagram.

I26. We already have joins in Physical layer then why we need to create logical joins in BMM layer or question comes in this way that why we need to

create logical joins?------ Tek Systems (Hyderabad)

I27. What is cardinality? ----- Accenture (Kolkata)

I28. What the use of driving table?----- Accenture (Kolkata)

Examining Logical Columns

Expand the D1 Time logical table. Notice that logical columns were created automatically for each table when you dragged the alias tables from the Physical layer to the BMM layer.

Examining Logical Table Sources (LTS)

1. Expand the Sources folder for the D1 Time logical table. Notice there is a logical table source, D1 Time. This logical table source maps to the D1 Time alias table in the Physical layer.
2. Double-click the D1 Time logical table source (not the logical table) to open the Logical Table Source dialog box.
3. On the General tab, rename the D1 Time logical table source to LTS1 Time. Notice that the logical table to physical table mapping is defined in the "Map to these tables" section.
4. On the Column Mapping tab, notice that logical column to physical column mappings are defined. If mappings are not visible, select Show mapped columns
5. You learn more about the Content and Parent-Child Settings tabs later in this tutorial when you build logical dimension hierarchies. Click OK to close the Logical Table Source dialog box. If desired, explore logical table sources for the remaining logical tables.

Renaming Logical Objects Manually

1. Expand the D1 Time logical table.
2. Click on the first logical column, BEG_OF_MONTH_WID, to highlight it.
3. Click on BEG_OF_MONTH_WID again to make it editable.
4. Rename BEG_OF_MONTH_WID to Beg of Mth Wid. This is the manual method for renaming objects. You can also rename an object and select Rename to manually rename an object.

Renaming Objects Using the Rename Wizard

1. Select Tools > Utilities > Rename Wizard > Execute to open the Rename Wizard.
2. In the Select Objects screen, click Business Model and Mapping in the middle pane.
3. Expand the Sample Sales business model.
4. Expand the D1 Time logical table.
5. Use Shift+Click to select all of the logical columns except for the column you already renamed, Beg of Mth Wid.
6. Click Add to add the columns to the right pane.
7. Repeat the steps for the three remaining logical tables so that all logical columns from the Sample Sales business model are added to the right pane.
8. Click Next to move to the Select Types screen.Notice that Logical Column is selected. If you had selected other object types, such as logical tables, the type would have appeared here.
9. Click Next to open the Select Rules screen.
10. In the Select Rules screen, select All text lowercase and click Add to add the rule to the lower pane.
11. Add the rule Change each occurrence of '_' into a space.
12. Add the rule First letter of each word capital.
13. Click Next to open the Finish screen. Verify that all logical columns will be named according to the rename rules you selected.
14. Click Finish.
15. In the Business Model and Mapping layer, expand the logical tables and confirm that all logical columns have been renamed as expected..
16. In the Physical layer, expand the alias tables and confirm that all physical columns have not been renamed. The point here is you can change object names in the BMM layer without impacting object names in the Physical layer. When logical objects are renamed, the relationships between logical objects and physical objects are maintained by the logical column to physical column mappings.

Deleting Unnecessary Logical Objects

1. In the BMM layer, expand Sample Sales > F1 Revenue.
2. Use Ctrl+Click to select all F1 Revenue logical columns except for Revenue and Units.
3. Right-click any one of the highlighted logical columns and select Delete.

   Alternatively you can select Edit > Delete or press the Delete key on your keyboard.
4. Click Yes to confirm the delete
5. Confirm that F1 Revenue contains only the Revenue and Units columns.

Creating Simple Measures

Double-click the Revenue logical column to open the Logical Column dialog box.

1. Click the Aggregation tab.
2. Change the default aggregation rule to Sum.
3. Click OK to close the Logical Column dialog box. Notice that the icon has changed for the Revenue logical column indicating that an aggregation rule has been applied.

Repeat the steps to define the SUM aggregation rule for the Units logical column.

Note: Measures are typically data that is additive, such as total dollars or total quantities. The F1 Revenue logical fact table contains the measures in your business model. You aggregated two logical columns by summing the column data.

4. Save the repository without checking global consistency.

Congratulations! You have successfully built a business model in the Business Model and Mapping layer of a repository and created business measures.

3.3. Presentation Layer

1. The Presentation layer exposes the business model objects in Oracle BI user interfaces so that users can build analyses and dashboards to analyze their data.
2. You have created the initial Sample Sales business model in the repository.
3. You now create the Presentation layer of the repository.

3.3.1. Hands on Main Objects

To build the Presentation layer you perform the following steps:

1. Creating a Subject Area
2. Creating Presentation Tables
3. Creating Presentation Columns
4. Renaming Presentation Columns
5. Reordering Presentation Columns

Creating a Subject Area

1. Right-click the white space in the Presentation layer and select New Subject Area to open the Subject Area dialog box.
2. On the General tab, enter Sample Sales as the name of the subject area.
3. Click OK to close the Subject Area dialog box. The Sample Sales subject area is added to the Presentation layer.

Creating Presentation Tables

1. Right-click the Sample Sales subject area and select New Presentation Table to open the Presentation Table dialog box.
2. On the General tab, enter Time as the name of the presentation table.
3. Click OK to close the Presentation Table dialog box. The Time presentation table is added to the Sample Sales subject area
4. Repeat the process and add three more presentation tables: Products, Customers and Base Facts.

Note:

Please note that you are using the manual method for creating Presentation layer objects. For simple models it is also possible to drag objects from the BMM layer to

the Presentation layer to create the Presentation layer objects. When you create presentation objects by dragging from the BMM layer, the business model becomes a subject area, the logical tables become presentation tables, and the logical columns become presentation columns. **Note that all objects within a subject area must derive from a single business model.**

Creating Presentation Columns

1. In the BMM layer, expand the D1 Time logical table
2. Use Ctrl+ Click to select the following logical columns:

   Calendar Date
   Per Name Half
   Per Name Month
   Per Name Qtr
   Per Name Week
   Per Name Year.

3. Drag the selected logical columns to the Time presentation table in the Presentation layer.
4. Repeat the process and add the following logical columns to the remaining presentation tables:

   Products: Drag **Brand**, **Lob**, **Prod Dsc**, **Type** from **D2 Product**.
   Customers: Drag **Cust Key**, **Name** from **D3 Customer**.
   Base Facts: Drag **Revenue**, **Units** from **F1 Revenue**.

Renaming Presentation Columns

1. In the Presentation layer, expand the Products presentation table.
2. Double-click the Lob presentation column to open the presentation column dialog box. On the General tab notice that "Use Logical Column Name" is selected. When you drag a logical column to a presentation table, the resulting presentation column inherits the logical column name by default. In this example the Lob presentation column inherits the name of the logical column "Sample Sales"."D2 Product"."Lob".
3. Deselect Use Logical Column Name. The Name field is now editable.
4. Enter Line of Business in the Name field.
5. Click OK to close the Presentation Column dialog box. Notice that the presentation column name is now changed to Line of Business in the Presentation layer.

6. In the BMM layer, expand D2 Product. Notice that the Lob logical column name is not changed. The point here is you can change object names in the Presentation layer without impacting object names in the BMM or Physical layers.
7. In the BMM layer, rename the Prod Dsc logical column to Product. Notice that the name change is inherited by the corresponding presentation column
8. Make the following name changes to logical objects in the BMM layer so that the names of the corresponding presentation columns are also changed:

For the **D3 Customer** logical table:

Change **Cust Key** to **Customer Number**.
Change **Name** to **Customer Name**

9. Confirm that the corresponding presentation column names are changed.

Reordering Presentation Columns

1. In the Presentation layer, double-click the Time presentation table to open the Presentation Table dialog box.
2. Click the Columns tab.
3. Select columns and use the up and down arrows, or drag the columns. to rearrange the presentation columns into the following order from top to bottom:

Per Name Year
Per Name Half
Per Name Qtr
Per Name Month
Per Name Week
Calendar Date

4. Click **OK** to close the **Presentation Table** dialog box and confirm that the presentation column order is changed in the **Presentation** layer.
5. Repeat the steps to reorder the columns in the Products presentation table:

Brand
Line of Business
Type
Product

6. Save the repository without checking global consistency.

Congratulations! You have successfully built the Presentation layer of a repository.

Note:

Create another subject area ‘sample sales DM’ and copy Time , Product and Revenue from samples sales subject area and paste in sample sales DM. save RPD.

I. Can a Subject area refer two Business models at time? How?--- Cognizant (Bangalore)

I. Can two Subject area refer Business models at time? ----Cognizant (Bangalore)

I. Can we disable a particular Subject Area or any object in Presentation layer? How?------PWC (Bangalore)

3.4. Testing and Validating a RPD

1. You have finished building an initial business model and now need to test and validate the repository before continuing.
2. You begin by checking the repository for errors using the consistency checking option.
3. Next you load the repository into Oracle BI Server memory.
4. You then test the repository by running an Oracle BI analysis and verifying the results.
5. Finally, you examine the query log file to observe the SQL generated by Oracle BI Server.

To test and validate a repository you perform the following steps:

1. Checking Consistency
2. Loading the Repository
3. Disabling Cache
4. Setting Up Query Log
5. Creating and Running Analysis
6. Checking the Query Log

Checking Consistency

1. Select File > Check Global Consistency
2. You should receive the message Business model "Sample Sales" is consistent. Do you want to mark it as available for queries?
3. Click Yes. You should receive the message: Consistency check fixed certain object(s); there are no errors, warnings or best practice violations left to report.

Note: If you do not receive this message, you must fix any consistency check errors or warnings before proceeding.

4. Click OK. Notice that the Sample Sales business model icon in the BMM layer is now green, indicating it is available for queries.
5. Save the repository without checking global consistency again.
6. Select File > Close to close the repository. Leave the Administration Tool open.

4. OFMW Enterprise Manager

4.1. Overview

Fusion Middleware Control is used to centrally manage, monitor, and configure Oracle Business Intelligence system components. For example, the Oracle BI Server, Oracle BI Presentation Services, and Oracle BI Scheduler.

Fusion Middleware Control enables you to manage system components by performing tasks such as monitoring status, starting and stopping processes, scaling out, resolving issues, and configuring components. You can also manage some aspects of Java components. For example, you can monitor their status and start and stop them.

Locking mechanism

With large deployments, you might have multiple administrators accessing the system concurrently to view the state of the system while other administrators might want to make configuration changes. Fusion Middleware Control and Oracle WebLogic Server prevent concurrent updates of the same configuration settings by multiple administrators by using a locking mechanism that allows only one administrator to make changes at any one time.

4.2 Loading the Repository

1. Open a browser and enter the following URL to navigate to Oracle Enterprise

   Manager:

http://<machine name>:7001/em

Ex: http:\\msit:7001\em

Log in as an administrative user. Typically you use the administrative user name and password provided during the Oracle BI installation. In this example the user name is weblogic and password is weblogic123.

1. In the left navigation pane, expand Business Intelligence and click coreapplication.

2. In the right pane, click the Deployment tab.

3. Click the Repository subtab.

4. Click Lock and Edit Configuration.

5. Click Close when you receive the confirmation message "Lock and Edit

   Configuration - Completed Successfully."

6. In the "Upload BI Server Repository" section, click Browse to open the Choose file dialog box.
7. By default, the Choose file dialog box should open to the repository directory. If not, navigate to the repository directory with the BISAMPLE repository. If not, browse to EX:D:\OBIEE11g_Home\instances\instance1\bifoundation\ OracleBIServerComponent\coreapplication_obis1\repository.
8. Select the BISAMPLE.rpd file and click Open.
9. Enter BISAMPLE123 as the repository password and confirm the password.

I. Can we change the path or location of RPD? How?--- Raqmiyat (Dubai)

I. What is the size of your RPD in your Project? -------British Telecom(BT) (Bangalore)

Loading the Catalog

10. In the “BI Presentation Catalog” section, click on the catalog location and move the cursor to the end and then remove SampleApplite and give BISAMPLE name.
11. Click Apply
12. In the BI Server Repository section, confirm that the Default RPD is now BISAMPLE with an extension. In this example the file name is **BISAMPLE_BIXXXX.**
13. Click Activate Changes.
14. Allow Active Changes processing to complete. Click Close when you receive the confirmation message Activate Changes - Completed Successfully.
15. On the Availability > Processes page, select Restart all .
16. Click Yes when you receive the message “Are you sure you want to restart all components?”
17. Allow the “Restart All – In Progress” processing to complete. This may take a few moments.

18. Click Close when you receive the confirmation message “Restart All– Completed Successfully".
19. Confirm that all components are running. Oracle BI cache is now disabled and the BISAMPLE repository is loaded into BI Server.
20. Leave Oracle Enterprise Manager open.
21. Go to the catalog folder using the below path and check catalog folder MSITCAT is created.

Path:

Ex:C:\OBIEE11g_HOME\instances\instance1\bifoundation\OracleBIPresentationServicesComponent\coreapplication_obips1\catalog

I. What happen or error if empty catalog is loaded in BI Server?---Dell (Bangalore)

4.3 Disabling Cache

1. In the right pane of Oracle Enterprise Manager. Click the Capacity Management tab.
2. Click the Performance sub tab
3. Locate the Enable BI Server Cache section. Cache is enabled by default.
4. Click Lock and Edit Configuration
5. Click Close when you receive the confirmation message "Lock and Edit Configuration - Completed Successfully."
6. Deselect Cache enabled. Caching is typically not used during development. Disabling cache improves query performance.
7. Click Apply.
8. Click Activate Changes.
9. Click Close when you receive the confirmation message Activate Changes – Completed Successfully.

4.4 Setting Up Query Log

1. Return to the Administration Tool, which should still be open.
2. Select File > Open > Online to open the repository in online mode. You use online mode to view and modify a repository while it is loaded into the Oracle BI Server. The Oracle BI Server must be running to open a repository in online mode.
3. Enter BISAMPLE123 as the repository password and enter your administrative user name and password.
4. Click Open to open the repository in online mode
5. Select Manage > Identity to open Identity Manager.
6. In the left pane, select BI Repository.
7. Select Action > Set Online User Filter.
8. Enter an asterisk and click OK to fetch users from the identity store.
9. In the right pane, double-click your administrative user to open the User dialog box. In this example the administrative user is weblogic
10. In the User dialog box, on the User tab, set Logging level to 2.
11. Click OK to open the Check Out Objects dialog box.
12. In the Check Out Objects dialog box, click Check Out. When you are working in a repository open in online mode, you are prompted to check out objects when you attempt to perform various operations.
13. Select Action > Close to close Identity Manager.
14. Select File > Check In Changes. Alternatively, you can click the Check In Changes icon on the toolbar.
15. Save the repository. There is no need to check consistency.
16. Select File > Copy As to save a copy of the online repository with the security Changes.

17. In the Save Copy As dialog box, save the file as BISAMPLE.rpd, replacing the existing BISAMPLE repository.
18. Click Yes when asked if you want to replace the existing BISAMPLE repository. This will create a new BISAMPLE repository with query logging set for the weblogic user.
19. Select File > Close to close the repository.
20. Click OK when you receive the following message:

"In order for your online changes to take effect, you will have to manually restart each non-master Oracle BI Server instance in the cluster."

21. Leave the Administration Tool open.
22. Open Enterprise Manager and restart all components.

I. How to get the SQL generated by BI Server for analysis or report? ----Oracle India (Hyderabad)

I. With want the SQL in the log file starts with?----Oracle India (Hyderabad)

5. BI Analytics

5.1. Accessing Analytics for First Time

1. Oracle BI Analysis Editor is a set of graphical tools that are used to build, view, and modify Oracle BI analyses.
2. The analysis are queries against an organization's data. It allows us to include the views that you create in an analysis for display in dashboards.

Logging In

1. Open any Mozilla or chrome or IE web browser and type below addresses

a. In a browser window, enter

http://localhost:9704/analytics for Enterprise Installation

http://localhost:7001/analytics for Simple Installation

Ex: http://msit:9704/analytics

b. The Oracle Business Intelligence Sign In page is displayed. Enter your User ID and Password and click Sign In.

2. When you sign in, the Home page is displayed.
3. Home Page contains global header, Create New section, Catalog Management section, Get Started section with links to additional help and BI tools, Recent section displaying the recently viewed or created analysis or dashboards, and Most Popular section. You can always operate these features from the global header as well.

5.2. Creating or Developing First Analysis

Creating an Analysis and Using the Analysis Editor

1. To build an analysis, do the following:

   From the home page, click New > Analysis.

2. The Select Subject Area pop-up appears.

3. In the Select Subject Area pop-up, select Sample Sales. The Analysis Editor is displayed.

Note:

In this example: The selected subject area is Sample Sales

The four tabs - Criteria, Results, Prompts and Advanced are displayed at the top of the Editor

4. Selected columns pane is empty as you are yet to choose the columns and Filters is empty as well waiting for the column selections and further criteria

5a. Select the following columns for your analysis.

Folder	Columns
Customers	**Customer Name**
Products	**Product**
Base Facts	**Revenue**

b. While selecting the columns, click the plus sign  to expand the folders and double click the required column names to get them in the Selected Columns section. In this example, expand the Customers folder, and then double click Customer name to get it in the Selected Columns section.

Note: In the Selected Columns section, you can reorder the columns in your analysis by clicking and dragging them.

6. Click the Results tab. The default Compound Layout is displayed.

Note1: The Compound Layout is a composition of many views. By default, both a Title and Table view are defined for you when using attribute and measure columns. A Pivot Table view is automatically created when using hierarchical columns in your analysis.

Note2: In the Compound Layout, you can create different views of the analysis results such as graphs, tickers, and pivot tables. These are covered in this tutorial going forward.

Filtering, Sorting, and Saving your Analysis

Filtering An Analysis:

1. Perform the following steps to filter, sort and save the previously created analysis.
2. Click the Criteria tabbed page. Select the column Customer name to create a filter.

   You can create a filter by hovering over the specific column's toolbar by selecting the more drop-down menu.
3. In the More drop-down menu, select Filter.

   The column selected for this example is Customer name.
4. The New Filter dialog box is displayed. Accept the default value for the operator, that is is equal to / is in, and enter a column value (or a range of column values) for this condition. To do this, click the drop-down list for Value, and click the desired checkboxes.
5. Click OK.

   The Filters pane displays the newly created filter.
6. Click OK.

Saving an Analysis:

7. Click the Save icon to save your analysis.
8. Navigate to My Folders and click the New Folder icon. The New Folder dialog box appears.
9. Name the folder Revenue Details and click OK.
10. Name the analysis Revenue by Customer and click OK.
11. The analysis is saved to the catalog folder Revenue Details

Sorting An Analysis:

1. Go to Home page, and in the Recent area, click the Edit link for the Revenue Details analysis. Now you will add a sort to this analysis.
2. On the Criteria tabbed page, click the More Options icon for Revenue column.
3. Select Sort > Sort Descending.

Note:

Observe that a sort icon is added to Revenue. The order of the sort is indicated by an arrow; in this case, the arrows points down, indicating that it is descending. Additionally, if multiple sorts are added, a subscript number will also appear, indicating the sequence for the sort order.

4. Save your analysis again.
5. Click the Results tabbed page to verify the filter and sort are being applied to your analysis. The Compound Layout display the filtered and sorted analysis.

I. Can we add images to analysis or report? How?----ITC Infotech (Bangalore)

5.3. Enhancing an Analysis by Adding Views

1. Create a new analysis by using new columns. Click New > Analysis on the global header. Use Sample Sales Subject Area.
2. Add Per Name Year from Time, Product from Products, and Revenue from Base Facts to Selected Columns.
3. Save the analysis to your Revenue Detail folder, as Revenue by Year as the analysis name.
4. You will now add a graph to this analysis.
5. Click the Results tabbed page, and click the New View icon.
6. Select Graph > Bar > Default (Vertical) from the menus.

   The default Graph view appears below the Table view.

7. Click the Remove View from Compound Layout icon for both Title and Table views.

8. Both views are removed from the Compound Layout. Note however, that they are still available for use from the Views pane.
9. Save the analysis.

I. Can we add a logo to a report or analysis?-----PWC (Bangalore)

I. If copy and paste options are disabled how will you do changes to a report? ---- PWC (Bangalore)

6. BI Interactive Dashboards

6.1. Exploring and Editing My Dashboard

To open My Dashboard, perform the following steps:

1. Open any Mozilla or chrome or IE web browser and type below addresses

   In a browser window, enter http://localhost:9704/analytics.

2. Click the Dashboards link on the global header and then click My Dashboard.

   An empty My Dashboard page appears.

Note:

When you open a dashboard, including My Dashboard, the content appears in one or more dashboard tabbed pages. Pages contain the columns and sections that hold the content of a dashboard, and every dashboard has at least one page. Multiple pages are used to organize content.

3. This example shows an empty My Dashboard page with no content. Hover over the

   Edit icon to edit the dashboard and add content.

4. Click the Edit icon () to add content to your empty dashboard page.

   The Dashboard Builder appears and automatically creates page 1 of your dashboard

5. Using the Dashboard Builder, you can add pages and objects to a dashboard and

   control the page layout. The Dashboard Builder is composed of the following:

6. Dashboard Toolbar: The toolbar allows you to perform tasks such as adding or

   deleting pages, previewing, saving, and so on.

7. In the Dashboard Toolbar, the Tools toolbar button provides options to set

   dashboard properties, set page report links, and so on.

8. The Dashboard Objects pane provides you with a list of objects to add as content to

   a dashboard page. You will have to drag the object to the Page Layout pane on the

   right.

9. Drag the Column object onto the Page Layout pane. The Column object appears on the Page Layout pane.

10. In the Catalog pane, navigate to the folder where you saved your analysis.

11. Drag the Revenue Per Year analysis to the Column 1.

Revenue Per Year appears in the column.

Note:

Observe that a Section is automatically created for you. You can also drag an analysis directly onto an empty Layout Pane without first creating a column. The Dashboard Builder automatically creates the column for you. You can then add sections automatically to that column by dragging analyses below the existing sections.

12. Click the Save icon to save the dashboard page and then click the Run icon.

My Dashboard appears with the selected analysis Revenue Per Year.

Note:

1. Before creating a dashboard, click on catalog and click on My folder and Click on Expand Option under Revenue Details Folder and then click on edit under Revenue By Customer analysis.

2. Then Click on save as and save as dialog box will open .select shared folder from left pane then create new folder and name it as Revenue analysis and click ok.

6.2. Building or Developing First Dashboard

Refer above note before developing your First Dashboard

1. Click the New > Dashboard in the global header.

The New Dashboard dialog box appears.

2. Enter Revenue Summary in the name text box. Notice that you can also enter a description of your choice.

3. Click on location and select Browse catalog and select shared folders.

4. Create a new folder Revenue Details

This Folder will appear under Dashboards Link

5. Accept the default to Add content now.

6. Click OK. The Dashboard Builder appears.

7. Navigate to the Revenue By Customer analysis and drag it from the Catalog to the

Page Layout pane.

8. Save and run the dashboard. The Revenue Detail dashboard appears.

Note:

Now you can access this dashboard from Dashboards link on global header.

I. Can we run 100 or more reports at a time? How?------ Genpact (Hyderabad)

6.3. Catalog or BI Presentation Catalog

The Oracle BI Presentation Catalog stores the content that users create in a directory structure of individual files. This content includes folders, shortcuts, Oracle BI EE objects (such as analyses, filters, prompts, and dashboards), and Oracle BI Publisher objects (such as reports and templates).

Each object is composed of two files:

- **Own file:**

  For example, an analysis called Revenue By Customer analysis would be stored in a file named Revenue By Customer analysis.

- **Attributes file:**

  Each object has a corresponding attribute file. For example, the analysis called Revenue By Customer analysis would have a corresponding attribute file named Revenue By Customer analysis.atr. The attribute file contains the object's full name, access control list (ACL), description, and so on. To access an object in the catalog, users must have appropriate ACL entries for that object. All objects in the catalog use ACL entries.

- **Lock Files :**

  To guarantee that only one user can write to a file at one time, a lock file is created when an object is being written to. On rare occasions (for example, after a power outage), temporary lock files in the catalog might not be removed completely. If Presentation Services reports of such a lock file, then you must delete it manually.

The web catalog is located:

$ORACLE_INSTANCE\bifoundation\OracleBIPresentationServicesComponent\coreapplication_obips1\catalog\XXXXX (catalog name)

EX:
C:\OBIEE11g_HOME\instances\instance1\bifoundation\OracleBIPresentationServicesComponent\coreapplication_obips1\catalog\BISAMPLE

I. What do mean ACL? ----- Capegeimi (Mumbai)

Directory Structure:

When opening a web catalog, you will always find the following folders

- **The system folder** — Contains administrative elements of Presentation Services. Some of these elements are distributed with the product, and others are set up by the administrator, such as privileges.

- **The users folder** — Contains content that catalog users with the appropriate permissions have saved to their personal folders, such as individual analyses. Includes a Subject Area Contents folder where you save objects such as calculated items and groups.

- **The shared folder**— Contains content that is shared among catalog users. Dashboards that are shared across users are saved in a Dashboards subfolder under a common subfolder under the /Shared Folders folder.

Use of BI Presentation Catalog:

BI Presentation Catalog is managed by BI Presentation Services the following things can be managed

- Permissions on objects to respective users, groups and roles
- Access to objects globally
- Scheduling Analysis
- Rename objects
- Copy, save as and deleting objects
- Creating Shortcuts

Techno Level 2: Experienced Level

7. OBIEE Architecture

7.1. Physical Architecture

Fig12

7.2. Logical Architecture

Fig12

8. Oracle Weblogic Server (OWLS) Administration Console

8.1. Overview

Oracle WebLogic Server is a scalable, enterprise-ready Java Platform, Enterprise Edition (Java EE) application server. The Administration Console is a Web browser-based, graphical user interface that you use to manage a WebLogic Server domain. A domain includes one or more WebLogic Servers and may also include WebLogic Server clusters. Clusters are groups of WebLogic Servers instances that work together to provide scalability and high-availability for applications. You deploy and manage your applications as part of a domain.

8.2. Creating a user

1. Open the web browser and enter the below url

   http://<hostname>:port/console

   Ex: http://<msittest>:7001/console

And press enter

Fig18

1. Enter the username and password and click login, then Home is displayed.

   Fig19

2. In the left pane of the Home, under Domain Structure→click Security Realms.

   Fig20

3. In the Summary of Security Realms window,Click on “myrealm”.

   Fig21

4. In Settings for myrealms window,click on users and groups tab.

   Fig22

5. Under Users and Groups tab, click on Users tab and click New button.

Fig23

Note:

Under users tab, in users table there are some users already displaying ,they are system default users like WebLogic, BISystemUser.

6. Create a New User window is displayed. Enter the name(case sensitive), description(optional) and password(8 characters mandatory).

   Click ok.

   Fig 24

7. User created message will be displayed and the user is displayed in the user table.

   Fig25

   Note:

   There is no need to restart any BI server or services after creating a new user.

8. Testing the new user, open the analytics and login using new Username and Password.

   Fig26

8.3. Creating a group

Follow steps 1 to 5 of creating a New User Topic, then

6. Under Users and Groups tab, click on Groups tab and click new button.

   Fig27

Note:

Under Groups tab, in groups table there are some groups already displaying, they are system default groups like BIAdministrator, BIAuthor, and operators.

7. Create a New Group window is displayed. Enter the name (case sensitive) and description (optional) .

   Click ok.

   Fig 28

8. Group created message will be displayed and the group is displayed in the group table.

Fig29

Note:

There is no need to restart any BI server or services after creating a new group.

8.3.1. Assigning a user or users to a group or groups

Follow steps 1 to 5 of creating a New User Topic, then

6. Under Users and Groups tab, click on Users tab.
7. In the Users table, click on the user (Ex: msit).

   Fig30
8. Settings for msit window will open, click on the groups tab in the window

   Fig31
9. Choose the groups to be added to the user and click save.

   Fig32
10. Then you will get successfully updated message.

    Fig33

Note:

You can assign same group to any number of users.

8.4. Creating an application role in OFMW EM

1. Open Oracle Fusion Middleware Enterprise Manager.
2. In the Left pane, click Business Intelligence→core application.
3. In right pane, at the top under core application click on business intelligence Instance→security→application roles.

   Or

   In right pane, click on security tab and at the bottom of the tab under Application policies and roles title, click configure and manage applcation roles.

   Fig 34
4. Application roles window will open, click on create button.

   Fig35
5. Enter role name and display name and click Add buton to add groups or users or other roles.

Fig36

6. A window is prompted, select type as group and princopal name as msit and click on search button.

   Fig37

7. The group name or any name starting with ‘m’ will appear.select your group and click ok.

   Fig 38

   Click ok at the top right cornor

8. A new role created meassge is displayed and your application roles is displayed in application roles list.

   Fig39

8.4.1. Assigning a group or groups to an application role or roles

Follow all steps from above. Except at step 4, select any existing role and click on edit button.

Note:

Using Create like option we can create an application role same as desired role.

8.5. Deleting users, groups and roles

Deleting a role:

Open EM, click on core application→security tab→configure and manage application roles(bottom)→select a role and click delete button and click ok.

Fig40

Deleting a group:

Open WLS Console →security realm (left pane)→myrealm→Users and Groups tab→Groups tab→Under Groups table→ select group check box→Delete→ok.

Fig41

Deleting a User:

Open WLS Console →security realm (left pane)→myrealm→Users and Groups tab→Users tab→Under Users table→ select user check box→Delete→ok.

Fig42

9. BI ADMIN TOOL

9.1. Adding Multiple Sources

1. Open Oracle BI Administration tool,

   Start > Programs > Oracle Business Intelligence > BI Administration.
2. File →open→online→RPD Password and username and password to open BISAMPLE RPD which is loading into BI server memory in basic level.
3. In the BMM layer, expand Sample Sales > D3 Customer > Sources. Notice that the D3 Customer logical table has one logical table source named D3 Customer.
4. Rename the D3 Customer logical table source (not the logical table) to LTS1 Customer.
5. Double-click LTS1 Customer to open the Logical Table Source dialog box.
6. Click the Column Mapping tab and notice that all logical columns map to physical columns in the same physical table: D3 Customer. It may be necessary to scroll to the right to see the Physical Table column. Make sure "Show mapped columns" is selected.
7. Click OK to close the Logical Table Source dialog box.
8. In the Physical layer, expand orcl > BISAMPLE.
9. Drag D4 Address from the Physical layer to the D3 Customer logical table in the BMM layer. Notice this creates a new logical table source named D4 Address for the D3 Customer logical table. It also creates new logical columns that map to the D4 Address physical table.

Note:

1. If data is not duplicated then that physical table can add to LTS or LT (As a best practice add to LTS).
2. If data is duplicated then we need to add physical table to LT.
3. To add Physical table to LTS, the adding physical table and existing physical table of LTS must have direct join.

10. In the BMM layer, double-click the new D4 Address logical table source to open the Logical Table Source dialog box.
11. On the General tab, enter LTS2 Customer Address in the Name field.

12. Click the Column Mapping tab and notice that all logical columns map to physical columns in the same physical table: D4 Address. If necessary, select show mapped columns and deselect Show unmapped columns.
13. Click OK to close the Logical Table Source dialog box.
14. Confirm that the D3 Customer logical table now has two logical table sources: LTS1 Customer and LTS2 Customer Address. A single logical table now maps to two physical sources.
15. Right-click the new ADDRESS_KEY column and select Delete. This is a duplicate column and is not needed.
16. Click Yes to confirm the delete.
17. Use the Rename Wizard or a manual renaming technique to rename the new address logical columns (with uppercase letters) in D3 Customer.
18. Rename the remaining logical table sources according to the following table. Recall that logical table sources are located in the Sources folder for a logical table. For example: D2 Product > Sources.
19. In the Presentation layer, right-click the Sample Sales subject area and select New Presentation Table to open the Presentation Table dialog box.
20. On the General tab, enter Customer Regions in the Name field.
21. Click OK to close the Presentation Table dialog box. Confirm that the Customer Regions presentation table is added to the Sample Sales subject area in the Presentation layer.
22. In the BMM layer, expand Sample Sales > D3 Customer.
23. Drag the following logical columns from D3 Customer to Customer Regions in the Presentation layer:

Address 1
Address 2
Area
City
Country Name
Estab Name
Postal Code

Region
State Province
State Province Abbrv

Your column names may be slightly different depending on how you renamed them.

24. Reorder the Customer Regions presentation columns in the following order, from top to bottom:

Region
Area
Country Name
State Province
State Province Abbrv
City
Postal Code
Address 1
Address 2
Estab Name

25. Double-click the Sample Sales subject area in the Presentation layer to open the Subject Area dialog box.

26. Click the Presentation Tables tab.

27. Reorder the presentation tables so that Customer Regions appears after Customers.

28. Click OK to close the Subject Area dialog box. Confirm that the presentation tables appear in the expected order.

Note: You now have two presentation tables, Customers and Customer Regions, mapped to the same logical table, D3 Customer. The D3 Customer logical table is mapped to two physical sources: D3 Customer and D4 Address.

29. Save the repository and check global consistency when prompted. You should receive a message that there are no errors, warnings, or best practice violations to report.

30. Click OK to close the consistency check message.

31. Open BI analytics, in the left navigation pane, under Create... Analysis and

Interactive Reporting, select Analysis.

32. Select the Sample Sales subject area.

33. Select “Reload Server Metadata”.

34. In the left navigation pane, expand the folders and confirm that the Customer Regions folder and corresponding columns appear.

35. Create the following analysis by double-clicking column names in the Subject Areas pane:

Customer Regions.Region
Customers.Customer Name
Products.Type
Base Facts.Revenue

36. Click Results to view the analysis results. Use the Get more rows button at the bottom of the results screen to see more rows.

37. Click on Administration (At the top right corner.)

38. Click Leave Page when prompted with the message: “Are you sure?”

39. On the Administration page, under Session Management, select Manage Sessions.

40. In the Cursor Cache section, locate your query and select View Log.

41. Locate the SQL Request section. This section contains the logical SQL issued by the query.

42. Click the browser back button to return to the Manage Session page.

43. Click the browser back button to return to the Administration screen.

44. Click Home to return to the Home page.

I. There are two physical columns,one with 50 char and not nullable and other with 20 char and nullable what happen if we create a logical coumn with these physical columns.

9.2. Calculations

1. In general, if the calculations are complex and large, then such calculations are done during ETL Stage.
2. If the calculations are reusable and reliable, then such calculations are done in RPD BMM Layer.
3. If the calculations are specific, personnel and temporary, then such calculations are done at Report level.
4. In OBIEE BMM Layer we can do calculations in 3 ways.

- ★ Creating calculation based on logical columns
- ★ Creating calculation based on physical columns.
- ★ Creating calculations by using calculation wizard.

9.2.1. Calculations based on logical columns

1. In the BMM layer, expand Sample Sales > F1 Revenue.
2. Right-click F1 Revenue and select New Object > Logical Column to open the Logical Column dialog box.
3. On the General tab, enter Actual Unit Price in the Name field.
4. Click the Column Source tab.
5. Select Derived from existing columns using an expression.
6. Click the Edit Expression button to open Expression Builder.
7. In the left pane select Logical Tables > F1 Revenue > Revenue.
8. Click the Insert selected item button to move the Revenue column to the right pane.
9. Click the division operator to add it to the expression.
10. In the left pane select Logical Tables > F1 Revenue and then double-click Units to add it to the expression.
11. Click OK to close Expression Builder. Notice that the formula is added to the

Logical Column dialog box.

12. Click OK to close the Logical Column dialog box. The Actual Unit Price calculated measure is added to the business model.
13. Drag Actual Unit Price from the BMM layer to the Base Facts presentation table in the Presentation layer.
14. Save the repository and check consistency. Fix any errors or warnings before proceeding.

9.2.2. Calculations based on Physical columns

1. In the BMM layer, expand Sample Sales > F1 Revenue>Sources and Double Click on LTS1 Revenue.

   Fig43
2. Go to Column mapping Tab, then click on Add New Column Icon to open a logical column dialog box.

   Fig44
3. In logical column dialog box, enter name as Actual units price and click ok.
4. Check show unmapped columns and the new column is displayed below, select it and click on Expression builder button .

   Fig45
5. In Expression Builder left pane, select physical tables in the top and at bottom select revenue column.
6. Then click on division operator and select units column as denominator in the expression.

   Fig46
7. Click ok to close expression builder and click ok to close logical column dialog box.

8. Set aggregation for the new column by double click the new column→aggregation tab→select SUM aggregation rule and click ok.
9. Drag Actual Unit Price from the BMM layer to the Base Facts presentation table in the Presentation layer.

   Fig48
10. Save the repository and check consistency. Fix any errors or warnings before proceeding.

9.2.3. Calculations using Calculations wizard

1. In Physical Layer, Expand BISAMPLE>F1 revenue and select columns – COST_FIXED and COST_VARIABLE.

   Fig49
2. Drag these columns to F1 Revenue in BMM layer and rename as Fixed Cost and variable Cost.

   Fig50
3. Double click fixed cost column and set SUM aggregation rule in aggregation tab, do same for variable cost column.
4. Right click on variable cost column, select calculation wizard>click next>check fixed cost column> click next.

   Fig51 and 52
5. Maximize the calculation wizard box, in the Generate calculation section check only change option and uncheck other options .

   Fig53
6. Under calculation name give name as change in costs and click next and again click next and click finish.

7. Drag Change In Cost, variable cost, Fixed Cost from the BMM layer to the Base Facts presentation table in the Presentation layer.

   Fig54

8. Save the repository and check consistency. Fix any errors or warnings before proceeding.

9. Open BI Analytics and test your work.

Develop a report using Per name year, Type, revenue, units, Actual unit price columns.

Develop a report using Per name year,Type,variable cost,fixed cost,change in cost columns.

Develop a report using Product, Revenue, Revenue Rank ,Units, Actual unit Price columns.

9.3. Dimensional Hierarchies

In OBIEE, there are two types of logical dimensions

1. Dimensions with level-based hierarchies (structure hierarchies)

   ★ Ragged and Skipped hierarchies
   ★ Time Dimension Hierarchy

2. Dimensions with parent-child hierarchies (value hierarchies)

9.3.1. Level based Hierarchy

In level-based hierarchies, members can be of different types and members of the same type come only at single level. It has two mandatory levels that is, Grand Total and detail levels.

Creating a Logical Dimension for Product

1. In the BMM layer, right-click the Sample Sales business model and select New Object > Logical Dimension > Dimension with Level-Based Hierarchy to open the Logical Dimension dialog box.

2. Name the logical dimension H2 Product.

3. Click OK. The logical dimension is added to the Sample Sales business model.

Creating Logical Levels

1. Right-click H2 Product and select New Object > Logical Level.

2. Name the logical level as Product Total.

3. Because this level represents the grand total for products, select the Grand total level check box. Note that when you do this, the Supports rollup to higher level of aggregation field is grayed out and protected.

4. Click OK to close the Logical Level dialog box. The Product Total level is added to the H2 Product logical dimension.

5. Right-click Product Total and select New Object > Child Level to open the Logical

Level dialog box.

6. Name the logical level Product Brand.

7. Click OK to close the Logical Level dialog box. The Product Brand level is added to the logical dimension.

8. Repeat the steps to add the following child levels:

Product LOB as a child of Product Brand
Product Type as a child of Product LOB
Product Detail as a child of Product Type

Associating Logical Columns with Logical Levels

1. Expand the D2 Product logical table.

2. Drag the Brand column from D2 Product to the Product Brand level in H2 Product.

3. Continue dragging logical columns from the D2 Product logical table to their corresponding levels in the H2 Product logical dimension:

Logical Column	Logical Level
Lob	Product LOB
Type	Product Type
Product	Product Detail
Prod Key	Product Detail

Setting Logical Level Keys

1. Double-click the Product Brand logical level to open the Logical Level dialog box. On the General tab, notice that the Product LOB child level is displayed.

2. Click the Keys tab.

3. Enter Brand for Key Name.

4. In the Columns field, use the drop down list to select D2 Product.Brand.

5. Check Use for Display. When this is selected, users can drill down to this column from a higher level.

6. Set Brand as the Primary key.

7. Click OK to close the Logical Level dialog box. The icon changes for Brand to show that it is the key for the Product Brand level.

8. Use a different technique to create a logical level key: Right-click Lob for the Product LOB level and select New Logical Level Key to open the Logical Level Key dialog box.

9. In the Logical Level Key dialog box, accept the defaults and click OK.

10. The icon changes for Lob to show that it is the key for the Product LOB level.

11. Use either method to set the remaining keys for the H2 Product logical dimension:

Logical Level	**Logical Level Key**	**Use for Display**
Product Type	Type	Yes
Product Detail	Product	Yes
Product Detail	Prod Key	No

Note: Please note that the Detail level (lowest level of the hierarchy) must have the column that is the logical key of the dimension table associated with it and it must be the key for that level: Prod Key in this example.

12. Set Prod Key as the primary key for the Product Detail level. Hint: Double-click the level and select the Keys tab.

Creating a Logical Dimension for Time (Time Dimension Hierarchy)

1. Use a different technique to create a logical dimension for Time. Right-click the D1 Time logical table and select Create Logical Dimension > Dimension with Level-Based Hierarchy.

2. A new logical dimension, D1 Time Dim in this example, is automatically added to the business model.

3. Rename D1 TimeDim to H1 Time.
4. Expand H1 Time . Notice that two levels were created automatically: D1 Time Total and D1 Time Detail. D1 Time Detail is populated with all of the columns from the D1 Time logical table.
5. Rename D1 Time Total to Time Total, and rename D1 Time Detail to Time Detail.
6. Right-click Time Detail and select New Object > Parent Level to open the Logical Level dialog box.
7. On the General tab, name the logical level Week, and check Supports rollup to higher level of aggregation.
8. Click OK to close the Logical Level dialog box. The Week level is added to the H1 Time logical dimension.
9. Repeat the steps to add the remaining logical levels:

Month as a parent of Week
Quarter as a parent of Month
Half as a parent of Quarter
Year as a parent of Half

Associating Time Logical Columns with Logical Levels

1. Use a different technique to associate logical columns with logical levels. Drag the logical columns from the Time Detail logical level (not from the D1 Time logical table) to their corresponding levels in the H1 Time logical dimension. This is a convenient technique when logical columns are buried deep in the business model.

Logical Column	**Logical Level**
Per Name Year	Year
Per Name Half	Half
Per Name Qtr	Quarter
Per Name Month	Month
Per Name Week	Week

2. Delete all remaining columns from the Time Detail level except for Calendar Date so that only Calendar Date is associated with the Time Detail level. Notice that deleting objects from the hierarchy does not delete them from the logical table in the business model.
3. Set the logical keys for the H1 Time logical dimension according to the following table:

Logical Level	Level Key	Use for Display
Year	Per Name Year	Yes
Half	Per Name Half	Yes
Quarter	Per Name Qtr	Yes
Month	Per Name Month	Yes
Week	Per Name Week	Yes
Time Detail	Calendar Date	Yes

Creating a Logical Dimension for Customer

1. Use either technique to create a logical dimension with a level-based hierarchy named H3 Customer for the D3 Customer logical table with the following levels, columns, and keys.

Level	Column	Key	Use for Display
Customer Total	<none>	<none>	<none>
Customer Region	Region	Region	Yes
Customer Area	Area	Area	Yes
Customer Country	Country Name	Country Name	Yes
Customer State	State Province	State Province	Yes
Customer City	City	City	Yes
Customer Postal Code	Postal Code	Postal Code	Yes
Customer Detail	Customer Name Customer Number	Customer Name Customer Number	Yes No

2. Set Customer Total as the grand total level.

3. Set Customer Number as the primary key for the Customer Detail level.

Setting Aggregation Content for Logical Table Sources

1. Expand D1 Time > Sources.

2. Double-click the LTS1 Time logical table source to open the Logical Table Source dialog box.

3. Click the Content tab.

4. Confirm that Aggregation content, group by is set to Logical Level and the logical level is set to Time Detail for the H1 Time logical dimension.

5. Click OK to close the Logical Table Source dialog box.

6. Repeat to verify or set content settings for the remaining logical table sources using the table :

Logical Table Source	Logical Dimension	Logical Level
LTS1 Product	H2 Product	Product Detail
LTS1 Customer	H3 Customer	Customer Detail
LTS2 Customer Address	H3 Customer	Customer Detail
LTS1 Revenue	H1 Time H2 Product H3 Customer	Time Detail Product Detail Customer Detail

7. Save the repository and check global consistency. Fix any errors or warnings before

proceeding. Notice that you did not have to make any changes to the Presentation layer.

Testing Your Work

1. Return to Oracle BI, which should still be open, and sign in if necessary.

2. Create the following analysis to test the Product hierarchy.

Products.Brand
Base Facts.Revenue

3. Click Results.

4. Click on the BizTech brand and verify that you can drill down through the hierarchy to see revenue data at each level.

5. Select New > Analysis > Sample Sales.

6. Click Leave Page when prompted with the message: “Are you sure?”

7. Create the following analysis:

Time.Per Name Year
Base Facts.Revenue

8. Click Results and verify that you can drill down through the Time hierarchy.

9. Repeat the steps and create the following analysis to test the Customers hierarchy:

Customer Regions.Region
Base Facts.Revenue

10. Click Results and verify that you can drill down through the Customers hierarchy.

11. Sign out of Oracle BI. Click Leave Page when prompted about navigating away from this page. Leave the Oracle BI browser page open.

9.3.2. Time Dimension Hierarchy

A Hierarchy which contains at least one level of a time dimension must having a chronological key is called Time Dimension hierarchy.

Refer sections titled: Creating a Logical Dimension for Time and Modelling Time Series Data.

9.3.3. Unbalanced Hierarchies (Ragged and Skipped)

1. **Ragged Hierarchy:** A hierarchy in which all levels end at different depth or end at different levels.
2. **Skipped Hierarchy:** A hierarchy in which immediate parent level of a child level is skipped or not present.

Importing Metadata and Define Physical Layer Objects

1. Open the BISAMPLE repository in online mode.
2. In the Physical layer, expand orcl.
3. Right-click Connection Pool and select Import Metadata to open the Import Wizard.
4. In the Select Metadata Types screen, accept the defaults and click Next.
5. In the Select Metadata Objects screen, in the data source view, expand BISAMPLE.
6. In the data source view, select the SAMP_PRODUCTS_DR table for import
7. Click the Import Selected button to move the table to the Repository View.
8. Expand BISAMPLE in the Repository View and confirm that the SAMP_PRODUCT_DR table is visible.
9. Click Finish to close the Import Wizard.
10. Confirm that the SAMP_PRODUCT_DR table is visible in the Physical layer of the repository.
11. Create the following alias for the table: D20 Product
12. Use the Physical Diagram to create the following physical join for the alias table:

"orcl"."""BISAMPLE"."D20 Product"."PROD_KEY" =
"orcl"."""BISAMPLE"."F1 Revenue"."PROD_KEY"

13. Right-click D20 Product and select View Data.

Note: There are skipped levels in the hierarchy. For example, brand A - Brand2 has a NULL value for LOB for the product D - Product 8.

14. Close View Data.

Creating Logical Table and Logical Columns

1. Drag D20 Product from the Physical layer to the Sample Sales business model in the BMM layer to create a D20 Product logical table. The logical join to F1 Revenue is created automatically based on the join in the Physical layer.

2. Rename the D20 Product logical columns:

Old Name	New Name
BRAND	Brand
LOB	LOB
PROD_DSC	Product
PROD_KEY	Product Number
Type	Product Type

3. Rename the D20 Product logical table source to LTS1 Product (Ragged).

Creating a Ragged/Skipped Levels Logical Dimension

1. Right-click the D20 Product logical table and select Create Logical Dimension > Dimension with Level-Based Hierarchy to automatically create a logical dimension named D20 ProductDim.

2. Rename D20 ProductDim to H20 Product.

3. Double-click the H20 Product logical dimension to open the Logical Dimension dialog box.

4. On the General tab, select both Ragged and Skipped Levels.

5. Click OK to close the Logical Dimension dialog box.

6. Expand H20 Product.

7. Create the following hierarchy:

Level	**Column**	**Key**	**Use for Display**
Product Total	<none>	<none>	<none>
Product Brand	Brand	Brand	Yes
Product LOB	LOB	LOB	Yes
Product Type	Product Type	Product Type	Yes
Product	Product	Product	Yes
Product Detail	Product Number	Product Number	Yes

Creating Presentation Layer Objects

1. Drag the D20 Product logical table to the Sample Sales subject area in the Presentation layer.
2. In the Presentation layer, rename D20 Product to Products (Ragged) and move Products (Ragged) to appear after Products.
3. Expand Products (Ragged) and notice that the H20 Product logical dimension is automatically added to the Presentation layer.
4. Save the repository and check consistency. Fix any errors or warnings before proceeding.
5. Close the repository. Leave the Administration Tool open.

Testing Your Work

1. Return to Oracle BI, which should still be open, and sign in.

2. Create the following analysis to test the ragged / skipped level hierarchy:

Products (Ragged).Brand
Products (Ragged).LOB
Products (Ragged).Product Type
Products (Ragged).Product
Base Facts.Revenue

3. Click Results.

4. The results display correctly even though there are skipped levels (levels with NULL values) and ragged levels (leaves with varying depth).

5. Sign out of Oracle BI. Click Leave Page when prompted about navigating away from this page. Leave the Oracle BI browser page open.

9.3.4. Parent-Child Hierarchy

A parent-child hierarchy is a hierarchy of members that all have the same type. This contrasts with level-based hierarchies, where members of the same type occur only at a single level of the hierarchy

Importing Metadata and Define Physical Layer Objects

1. Open the BISAMPLE repository in online mode.

2. In the Physical layer, expand orcl.

3. Right-click Connection Pool and select Import Metadata to open the Import Wizard.

4. In the Select Metadata Types screen, accept the defaults and click Next.

5. In the Select Metadata Objects screen, in the data source view, expand BISAMPLE and select the following tables for import:

SAMP_EMPL_D_VH
SAMP_EMPL_PARENT_CHILD_MAP
SAMP_EMPL_POSTN_D

6. Click the Import Selected button to move the tables to the Repository View.

7. Click Finish to close the Import Wizard.

8. Confirm that the three tables are visible in the Physical layer of the repository.

9. Right-click SAMP_EMPL_PARENT_CHILD_MAP and select View Data.

Note:

This is an example of a parent-child relationship table with rows that define the inter-member relationships of an employee hierarchy. It includes a Member Key column, which identifies the member (employee); an Ancestor Key, which identifies the ancestor (manager) of the member; a Distance column, which specifies the number of parent-child hierarchy levels from the member to the ancestor; and a Leaf column, which indicates if the member is a leaf member.

10. Create the following aliases for the tables:

Table	**Alias**
SAMP_EMPL_D_VH	D50 Sales Rep
SAMP_EMPL_PARENT_CHILD_MAP	D51 Sales Rep Parent Child
SAMP_EMPL_POSTN_D	D52 Sales Rep Position

11. Use the Physical Diagram to create the following physical joins for the alias tables:

"orcl"."."."BISAMPLE"."D52 Sales Rep Position"."POSTN_KEY" = "orcl"."."."BISAMPLE"."D50 Sales Rep"."POSTN_KEY"

"orcl"."."."BISAMPLE"."D50 Sales Rep"."EMPLOYEE_KEY" = "orcl"."."."BISAMPLE"."D51 Sales Rep Parent Child"."ANCESTOR_KEY"

"orcl"."."."BISAMPLE"."D51 Sales Rep Parent Child"."MEMBER_KEY" = "orcl"."."."BISAMPLE"."F1 Revenue"."EMPL_KEY"

Creating Logical Table and Logical Columns

1. In the BMM layer, right-click the Sample Sales business model and select New Object > Logical Table to open the Logical Table dialog box.

2. On the General tab, name the logical table D5 Sales Rep.

3. Click OK to add the logical table to the business model.

Note: Note that the D5 Sales Rep icon has a # sign. This is because you have not yet defined the logical join relationship.

4. Drag all six columns from D50 Sales Rep in the Physical layer to D5 Sales Rep in the BMM layer. This action creates logical columns and adds a D50 Sales Rep logical table source to D5 Sales Rep.
5. Rename the D50 Sales Rep logical table source to LTS1 Sales Rep.
6. In the Physical layer, expand D52 Sales Rep Position.
7. Drag POSTN_DESC and POSTN_LEVEL from D52 Sales Rep Position to LTS1 Sales Rep. Note that you are dragging the columns to the logical table source, not the logical table. Dragging to the logical table would create a second logical table source.
8. Drag DISTANCE from D51 Sales Rep Parent Child to LTS1 Sales Rep. Again, you drag the column to the logical table source, not the logical table.
9. Rename the logical columns:

Old Name	**New Name**
POSTN_KEY	Position Key
TYPE	Sales Rep Type
EMPL_NAME	Sales Rep Name
EMPLOYEE_KEY	Sales Rep Number
HIRE_DT	Hire Date
MGR_ID	Manager Number
POSTN_DESC	Position
POSTN_LEVEL	Position Level
DISTANCE	Closure Distance

Creating a Logical Join

1. In the BMM layer, select D5 Sales Rep and F1 Revenue.

2. Right-click either highlighted table and select Business Model Diagram > Selected Tables Only to open the Business Model Diagram.
3. Create a logical join between D5 Sales Rep and F1 Revenue with F1 Revenue at the many end of the join.
4. Close the Business Model Diagram. Notice that the icon has changed for the D5 Sales Rep table.

   Notice that the D5 Sales Rep icon is not having # sign.

Creating a Parent-Child Logical Dimension

1. Right-click the D5 Sales Rep logical table and select Create Logical Dimension > Dimension with Parent-Child Hierarchy.
2. In the Logical Dimension dialog box, on the General tab, name the logical dimension H5 Sales Rep.
3. Click Browse next to Member Key. The Browse window shows the physical table and its corresponding key.
4. Click View to open the Logical Key dialog box. Confirm that the Sales Rep Number column is selected.
5. Click Cancel to close the Logical Key dialog box.
6. Click OK to close the Browse window.
7. Click Browse next to Parent Column. The Browse window shows the columns other than the member key.
8. Deselect Show Qualified Names and select Manager Number as the parent column for the parent-child hierarchy.
9. Click OK to close the Browse window, but do not close the Logical Dimension dialog box.

Defining Parent-Child Settings

1. Click Parent-Child Settings to display the Parent-Child Relationship Table Settings dialog box. Note that at this point the Parent-Child Relationship Table is not defined.

Note:

For each parent-child hierarchy defined on a relational table, you must explicitly define the inter-member relationships in a separate parent-child relationship table. In the process of creating the parent-child relationship table, you may choose one of the following options:

2. Select a previously-created parent-child relationship table.
3. Use a wizard that will generate scripts to create and populate the parent-child relationship table. In the next set of steps you select a previously created and populated parent-child relationship table.
4. Click the Select Parent-Child Relationship Table button to open the Select Physical Table dialog box.
5. In the Select Physical Table dialog box, select the D51 Sales Rep Parent Child alias you created.
6. The D51 Sales Rep Parent Child alias is now displayed in the Parent-Child Relationship Table column.
7. In the Parent-Child Table Relationship Column Details section, set the appropriate Columns :

Member Key	MEMBER_KEY
Parent Key	ANCESTOR_KEY
Relationship Distance	DISTANCE
Leaf Node Identifier	IS_LEAF

Explanation:

- Member Key identifies the member.

- Parent Key identifies an ancestor of the member, The ancestor may be the parent of the member, or a higher-level ancestor.
- Relationship Distance specifies the number of parent-child hierarchical levels from the member to the ancestor.
- Leaf Node Identifier indicates if the member is a leaf member (1=Yes, 0=No).

8. Click OK to close the Parent-Child Relationship Table Settings dialog box.
9. Click OK to close the Logical Dimension dialog box.
10. Right-click H5 Sales Rep and select Expand All. Note that a parent-child logical dimension has only two levels.
11. Delete all columns from the Detail level except for Sales Rep Name and Sales Rep Number.
12. Double-click the Detail level to open the Logical Level dialog box.
13. On the Keys tab, create a new key named Display Key that maps to the Sales Rep Name column.
14. Deselect Use for Display for the Sales Rep Number column and select Use for Display for the Sales Rep Name column.
15. Make sure that Member Key is still set to D50 Sales Rep_Key.
16. Click OK to close the Logical Level dialog box.
17. Expand F1 Revenue > Sources and double-click LTS1 Revenue to open the Logical Table Source dialog box.
18. On the Content tab, set the logical level to Detail for the H5 Sales Rep logical dimension.
19. Click OK to close the Logical Table Source dialog box.
20. Expand D5 sales rep>Sources and double click LTS1 Sales rep to open the Logical Table Source dialog box.
21. On the Content tab, set the logical level to Detail for the H5 Sales Rep logical

dimension.

22. On parent-child settings tab, click browse and select D51 Sales Rep Parent Child .
23. In the Parent-Child Table Relationship Column Details section, set the appropriate columns:

Member Key	MEMBER_KEY
Parent Key	ANCESTOR_KEY
Relationship Distance	DISTANCE
Leaf Node Identifier	IS_LEAF

Creating Presentation Layer Objects

1. Drag the D5 Sales Rep logical table from the BMM layer to the Sample Sales subject area in the Presentation layer.
2. Rename the D5 Sales Rep presentation table to Sales Reps.
3. Move the Sales Reps presentation table above the Base Facts table.
4. Expand the Sales Reps presentation table and notice that the H5 Sales Rep parent-child logical dimension is automatically included as a presentation hierarchy.
5. Double-click the H5 Sales Rep presentation hierarchy to open the Presentation Hierarchy dialog box.
6. On the Display Columns tab, confirm that Sales Rep Name is set as the display column.
7. Click OK to close the Presentation Hierarchy dialog box.
8. Save the repository and check consistency. Fix any errors or warnings before proceeding.
9. Close the repository. Leave the Administration Tool open.

Testing Your Work

1. Return to Oracle BI, which should still be open, and sign in.

2. Create the following analysis to test the parent-child logical dimension.

   Sales Reps.H5 Sales Reps
   Sales Reps.Position
   Base Facts.Revenue

3. Click Results.

4. Expand the pivot table to view data at different levels of the hierarchy. Notice that the Revenue measure rolls up through each level.

5. Sign out of Oracle BI. Click Leave Page when prompted about navigating away from this page. Leave the Oracle BI browser page open.

9.4. Level Based Measures (LBM)

1. A level-based measure is a column whose values are always calculated to a specific level of aggregations present in a logical dimension hierarchy.
2. To create LBM, a logical dimension hierarchy must be created or present.
3. Level based measures are useful to calculate Grand Totals, shares or percentages.
4. Nested aggregate functions are not supported by RPD. Hence we need learn LBM Ex: SUM (SUM(Revenue))

Creating Level-Based Measures

1. Open RPD in Online Mode.
2. In the BMM layer, right-click the F1 Revenue table and select New Object > Logical Column to open the Logical Column dialog box.
3. On the General tab, enter Product Total Revenue in the Name field.
4. Click the Column Source tab.
5. Select Derived from existing columns using an expression.
6. Open the Expression Builder.
7. In the Expression Builder, add Logical Tables > F1 Revenue > Revenue to the expression. Recall that the Revenue column already has a default aggregation rule of Sum.
8. Click OK to close Expression Builder.
9. Click the Levels tab.
10. For the H2 Product logical dimension, select Product Total from the Logical Level drop-down list to specify that this measure should be calculated at the grand total level in the product hierarchy.
11. Click OK to close the Logical Column dialog box. The Product Total Revenue

measure appears in the Product Total level of the H2 Product logical dimension and the F1 Revenue logical fact table.

12. Repeat the steps to create a second level-based measure:

Name	Logical Dimension	Logical Level
Product Type Revenue	H2 Product	Product Type

13. Expose the new columns to users by dragging Product Total Revenue and Product Type Revenue to the Base Facts presentation table in the Sample Sales subject area in the Presentation layer. You can drag the columns from either the H2 Product logical dimension or the F1 Revenue logical table.

Creating a Share Measure

1. In the BMM layer, right-click the F1 Revenue table and select New Object > Logical Column to open the Logical Column dialog box.
2. On the General tab, name the logical column Product Share.
3. On the Column Source tab, select "Derived from existing columns using an expression".
4. Open the Expression Builder.
5. In the Expression Builder, Select Functions > Mathematic Functions > Round.
6. Click Insert selected item. The function appears in the edit box.
7. Click Source Number in the formula.
8. Enter 100* followed by a space.
9. Insert Logical Tables > F1 Revenue > Revenue.
10. Using the toolbar, click the Division button. Another set of angle brackets appears, <<expr>>.
11. Click <<expression>>.

12. Insert Logical Tables > F1 Revenue > Product Total Revenue. Recall that this is the total measure for the hierarchy.
13. Click between the last set of angle brackets, <<Digits>>, and enter 1. This represents the number of digits of precision with which to round the integer.
14. Check your work:

    Round(100* "Sample Sales"."F1 Revenue"."Revenue" / "Sample Sales"."F1 Revenue"."Product Total Revenue", 1)
15. This share measure will allow you to run an analysis that shows how revenue of a specific product compares to total revenue for all products.
16. Click OK to close the Expression Builder. The formula is visible in the Logical Column dialog box.
17. Click OK to close the Logical Column dialog box. The Product Share logical column is added to the business model.
18. Add the Product Share measure to the Base Facts presentation table.
19. Save the repository. Check consistency. You should receive the following message.
20. If there are consistency errors or warnings, correct them before you proceed.
21. Close the repository.

Testing Your Work

1. Return to Oracle BI, which should still be open, and sign in.

2. Create the following analysis to test the level-based and share measures.

    Products.Product
    Base Facts.Revenue
    Base Facts.Product Type Revenue
    Base Facts.Product Share

1. For the Product Share column, select Column Properties.

2. On the Data Format tab, select Override Default Data Format.

3. Change Treat Numbers As to Percentage and set Decimal Places to 2. Deselect Use 1000's separator.

4. Click OK to close the Column Properties dialog box.

5. Sort Product Share in descending order.

6. Click Results.

Notice that Product Type Revenue returns dollars grouped by Type even though the query is at a different level than Type; Product in this example. Product Share shows the percent of total revenue for each product sorted in descending order.

7. Sign out of Oracle BI. Click Leave Page when prompted about navigating away from this page. Leave the Oracle BI browser page open.

9.5. Modelling Time Series Data

1. Time series functions provide the ability to compare business performance with previous time periods,allowing you to analyze data that spans multiple time periods. For example, time series functions enable comparisons between current sales and sales a year ago, a month ago, and so on

2. Time series functions are 3 types.

Ago function:

i) The AGO function offsets the time dimension to display data from a past period.

ii) Example: 1 year ago Revenue, 3 month Ago Revenue, 6 months ago Revenue

iii) Syntax: Ago (<<Measure>>, <<Level>>, <<Number of Periods>>)

EX: Ago ("Sample Sales"."F1 Revenue"."Revenue","Sample Sales"."H1 Time "."Per name month", 3)

iv)Ago function will use dense_rank analytics function to achieve the calculations

ToDate function:

i) Aggregates a measure attribute from the beginning of a specified time period to the currently displayed time

ii) Example: YTD (Year To Date) , QTD (Quarter To Date) ……etc

iii) Syntax: ToDate (<<Measure>>, <<Level>>)

Ex: ToDate ("Sample Sales"."F1 Revenue"."Revenue " , "Sales"."H1 Time"." Per name Year")

iv) Todate function will use row_number ,min and Rank analytics function to achieve the calculations.

Period Rolling function:

i) This function computes the aggregate of a measure over the period starting x units

of time and ending y units of time from the current time.

ii) Example 3 Months rollup (Current Month + two previous Months)

iii) Syntax PeriodRolling (<<Measure>>, <<Starting Period Offset>>, <<Ending Period Offset>>)

EX: PeriodRolling ("Sample Sales"."F1 Revenue"."Revenue", -2, 0)

iv) Period rolling function will use dense_rank analytics function to achieve the calculations.

3. Time series functions operate on time-oriented dimensions. To use these functions on a particular dimension, you must designate the dimension as a Time dimension and set one or more keys at one or more levels as chronological keys.

Identifying a Logical Dimension as a Time Dimension

1. Return to the Administration Tool and open the BISAMPLE repository in online mode.
2. In the BMM layer, double-click the H1 Time logical dimension to open the Logical Dimension dialog box.
3. In the Structure section, select Time.
4. Click OK to close the Logical Dimension dialog box.

Identifying Level Keys as Chronological Keys

1. Expand the H1 Time logical dimension and double-click the Time Detail level to open the Logical Level dialog box.
2. Click the Keys tab.
3. Select the Chronological Key check box for Calendar Date.
4. Click OK to close the Logical Level dialog box.
5. Repeat and set chronological keys for the following levels:

Logical Level	Chronological Key
Year	Per Name Year
Half	Per Name Half
Quarter	Per Name Qtr
Month	Per Name Month
Week	Per Name Week

6. It is best practice to designate a chronological key for every level of a time logical dimension**.**

Creating a Measure Using the AGO Function

1. Right-click the F1 Revenue logical table and select New Object > Logical Column.
2. On the General tab, name the column Month Ago Revenue.
3. On the Column Source tab, select "Derived from existing columns using an expression".
4. Open the Expression Builder.
5. Select Functions > Time Series Functions > Ago.
6. Double-click Ago or click Insert selected item to add the Ago function to the Expression Builder.
7. Click <<Measure>>in the expression.
8. Select Logical Tables > F1 Revenue and then double-click Revenue to add it to the expression.
9. Click <<Level>> in the expression.
10. Select Time Dimensions > H1 Time and then double-click Month to add it to the expression.
11. Click <<Number of Periods>> and enter 1. The Ago function will calculate the Revenue value one month before the current month.
12. Click OK to close the Expression Builder. Check your work in the Logical

Column dialog box:

13. Click OK to close the Logical Column dialog box. The Month Ago Revenue time series measure is added to the F1 Revenue logical table.
14. Drag the Month Ago Revenue logical column to the Base Facts presentation folder.

Creating a Measure Using the TODATE Function

1. Right-click the F1 Revenue logical table and select New Object > Logical Column.
2. On the General tab, name the new logical column Year To Date Revenue.
3. On the Column Source tab, select "Derived from existing columns using an expression".
4. Open the Expression Builder.
5. Select Functions > Time Series Functions and double-click ToDate to insert the expression.
6. Click <<Measure>> in the expression.
7. Select Logical Tables > F1 Revenue and then double-click Revenue to add it to the expression.
8. Click <<Level>> in the expression.

9 Select Time Dimensions > H1 Time and then double-click Year to add it to the expression.

10. Click OK to close the Expression Builder.
11. Check your work in the Logical Column dialog box.
12. Click OK to close the Logical Column dialog box.
13. Drag the Year To Date Revenue logical column to the Base Facts presentation folder.

Creating a Measure Using the PERIODROLLING Function

1. Right-click the F1 Revenue logical table and select New Object > Logical Column.
2. On the General tab, name the new logical column Revenue 3-Period Rolling Sum.
3. On the Column Source tab, select "Derived from existing columns using an expression".
4. Open the Expression Builder.
5. Select Functions > Time Series Functions and double-click PeriodRolling to insert t he expression.
6. Click <<Measure>> in the expression.
7. Select Logical Tables > F1 Revenue and then double-click Revenue to add it to the expression.
8. Click <<Starting Period Offset>> in the expression.
9. Enter -2. This identifies the first period in the rolling aggregation.
10. Click <<Ending Period Offset>>.
11. Enter 0. This identifies the last period in the rolling aggregation.
12. These integers are the relative number of periods from a displayed period. In this example, if the query grain is month, the 3 month rolling sum starts two months in the past (-2) and includes the current month (0).
13. Click OK to close the Expression Builder.
14. Check your work in the Logical Column dialog box.
15. Click OK to close the Logical Column dialog box.
16. Drag the Revenue 3-Period Rolling Sum logical column to the Base Facts presentation folder.
17. Save the repository and check consistency. Fix any errors or warnings before you

proceed.

18. Close the repository. Leave the Administration Tool open.

Testing Your Work

1. Return to Oracle Enterprise Manager and load the BISAMPLE repository.

2. Return to Oracle BI and sign in.

3. Create the following analysis to test AGO and TODATE functions:

    Time.Per Name Month
    Time.Per Name Year
    Base Facts.Revenue
    Base Facts.Month Ago Revenue
    Base Facts.Year to Date Revenue

4. Set the following filter for the analysis:

5. Per Name Year is equal to / is in 2008.

6. For the Per Name Year column, select Column Properties > Column Format > Hide. This will prevent Per Name Year from displaying in the analysis results.

7. Sort Per Name Month in ascending order.

8. Click Results.

9. Month Ago Revenue displays revenue from the previous month. Year To Date Revenue calculates a running sum of revenue for the year on a monthly basis.

10. Create the following new analysis and filter to test the PERIODROLLING function at the month grain:

    Time.Per Name Month
    Time.Per Name Year
    Base Facts.Revenue
    Base Facts.Revenue 3-Period Rolling Sum

    Per Name Year is equal to / is in 2008

11. For the Per Name Year column, select Column Properties > Column Format > Hide. This will prevent Per Name Year from displaying in the analysis results.
12. Sort Per Name Month in ascending order.
13. Click Results.
14. Revenue 3-Period Rolling Sum is calculated based on the month grain.
15. Create the following new analysis and filter to test the PERIODROLLING function at the year grain:

    Time.Per Name Year
    Base Facts.Revenue
    Base Facts.Revenue 3-Period Rolling Sum

16. Sort Per Name Year in ascending order.
17. Click Results.

9.6. Variables

1. Variables in the Oracle BI are used to streamline administrative tasks and dynamically modify metadata content to adjust to a changing data environment.
2. There are five types of variables that you can use:
    1. Repository Variables
    2. Session Variables
    3. Presentation Variable
    4. Request Variable
    5. Global Variable
3. Repository variables and session variables will be defined in RPD & can be used in
        1. RPD Calculations
        2. RPD Filters
        3. Report Calculations
        4. Report Filters
        5. Dashboard prompts.
4. Presentation variables, request variables and global variable will be defined in dashboard prompt & can be utilized only in
    1. Report Calculations
    2. Report Filters

9.6.1. Repository Variables

1. A repository variables or Oracle BI Server variables is a variable that has a single Value at any point in time.
2. There are two types of repository variables:

    Static - Repository variables whose value persist and do not change until the administrator decides to change them.

    Dynamic - Repository variables whose values are refreshed by data returned

from queries

3. The administrator creates repository variables using the Oracle BI Administration Tool.

   Syntax: VALUEOF (<VARIABLE_NAME>)

   Ex: VALUEOF (DBUSER)

4. Repository variable values will be updated whenever oracle BI server is restarted or started and based on schedule in Initialization blocks.

i) Static Repository variable:

1. Open RPD in online mode→click on Manage tab→ click on Variables to open variable manager dialog box.
2. Click on Action→New→Repository→variable to open Static Repository variable dialog box.
3. Enter Name as DSN, select type as static and enter default initializer as 'orcl' (quotations are mandatory).click ok.

Note:

The value of a static repository variable is initialized in the Variable dialog. This value persists, and does not change until an administrator decides to change it.

4. Similary create another variable as Name as DBUSER,select type as static and enter default initializer as 'BISAMPLE' and click ok and close the Static Repository variable dialog box.
5. Expand orcl database in Physcial layer and double click connectionpool.
6. Under data source name remove orcl and enter VALUEOF (DSN) and in username VALUEOF (DBUSER).
7. Similary create another variable as Name as CURRENT_YEAR,select type as static and enter default initializer as '2008' and click ok and close the Static Repository

variable dialog box.

8. Save the RPD and check consistency.
9. Open BI analytics and create a new analysis from sample sales with per name year,type and revenue columns.
10. Click on per name year column filter→click on add more opions→click on repository variable→provide name as CURRENT_YEAR and click ok and see results.

ii) Dynamic Repository variable:

1. These variables will be associated with initialization block.
2. An initialization block contains the SQL statement that will be executed to initialize Or refresh the variables associated with that block.
3. These initialization blocks will be executed with oracle BI server refresh and based on the schedule of initialization block.
4. Schedule option is available only for dynamic repository variable initialization blocks.

Creating an Initialization Block

1. Select Manage > Variables to open the Variable Manager.
2. Select Action > New > Repository > Initialization Block.
3. Name the initialization block Current Periods.
4. Click the Edit Data Source... button to open the Repository Variable Initialization Block Data Source dialog box.
5. Click the Browse button to open the Select Connection Pool dialog box.
6. Double-click the Connection Pool object to select it.

The connection pool is added.

7. Enter the following SQL to determine the value of the current day, month, and year by finding the maximum value of the period key (BILL_DAY_DT) in the fact table:

SELECT CALENDAR_DATE, PER_NAME_MONTH, PER_NAME_YEAR FROM BISAMPLE.SAMP_TIME_DAY_D WHERE CALENDAR_DATE = (SELECT MAX (BILL_DAY_DT) FROM BISAMPLE.SAMP_REVENUE_F)

8. Click Test and confirm the expected results are returned. In this example, the results are determined by the data in the sample database used for this tutorial, which holds data through December 2010.
9. Close the Results window.
10. Click OK to close the Repository Variable Initialization Block Data Source dialog box.

Creating Variables

1. Click Edit Data Target to open the Repository Variable Initialization Block Variable Target dialog box.
2. Use the New button to create three new variables: CurrentDay, CurrentMonth, CurrentYear. The order is important. The value returned from the first column in the initialization block SQL, CALENDAR_DATE, is assigned to the CurrentDay variable. The value of the second column, PER_NAME_MONTH, is assigned to CurrentMonth (the second variable), and the value of the third column, PER_NAME_YEAR, is assigned to CurrentYear (the third variable). If necessary, use the Up and Down buttons to arrange the variables.
3. Click OK to close the Repository Variable Initialization Block Variable Target dialog box.
4. Leave the default Refresh Interval set to every hour. This means that the variables will be reinitialized every hour.

5. Click the Test button and check the results:
6. In this example, the results are determined by the data in the sample database used for this course, which holds data through December 2010.
7. Close the Results window.
8. Click OK to close the Repository Variable Initialization Block dialog box.
9. Check your work in the Variable Manager
10. Close the Variable Manager.
11. Save the repository and check consistency. Fix any errors or warnings before proceeding.
12. Close the repository. Leave the Administration Tool open.

Testing Your Work

1. Return to Oracle Enterprise Manager and load the BISAMPLE repository.
2. Return to Oracle BI and sign in.
3. Create the following analysis to test the variables.

    Time.Per Name Year
    Time.Per Name Month
    Time.Calendar Date
    Base Facts.Revenue

4. Click Filter for the Per Name Year column. The New Filter dialog box opens.
5. Select Add More Options > Repository Variable.
6. In the Repository Variable field, enter CurrentYear to create a filter for the Per Name Year column using the CurrentYear repository variable.
7. Click OK to close the New Filter dialog box. The filter is added to the Filters pane.
8. Repeat the steps to add the CurrentMonth and CurrentDay repository variables as filters for Per Name Month and Calendar Date columns, respectively.
9. Click Results and confirm that data only for the current year, month, and day is

returned (based on the sample data set).

10. Sign out of Oracle BI. Click Leave Page when prompted about navigating away from this page. Leave the Oracle BI browser page open.

9.6.2. Session Variables

1. A session variables is a variable that is initialized at login time for each user. When a user begins a session, the Oracle BI Server creates a new instance of a session variable and initializes it.
2. There are as many instances of a session variable as there are active sessions on the Oracle BI Server.
3. Each instance of a session variable could be initialized to a different value.
4. There are two types of session variables:

System - A session variable that the Oracle BI Server and Oracle BI Presentation Services use for specific purposes.

System session variables have reserved names that cannot be used for other kinds of variables (such as static or dynamic repository variables and non-system session variables).

Non-system - A system variable that the administrator creates and names.

5. The administrator creates non-system session variables using the Oracle BI Administration Tool.
6. Syntax VALUEOF (NQ_SESSION.Variablename)

   EX: VALUEOF (NQ_SESSION.USER)

System session variables:

1. These are pre-defined session variables used by oracle BI server for specific purpose such as authenticating users

2. We have below pre-defined system variables.[case-sensitive must be in CAPITAL]

1. USER.
2. PASSWORD
3. DISPLAYNAME.
4. GROUP.
5. WEBGROUP.
6. LOGLEVEL.
7. ROLES.
8. PERMISSIONS.
9. USERLOCALE.
10. TIMEZONE.
11. PORTALPATH
12. DISABLE_CACHE_HIT
13. DISABLE_CACHE_SEED

3. These variables are useful in special cases such as authenticating user. These variables should not use for any other purpose. (as a static variable name or dynamic variable,non system session variable etc)

Using a system variable

1. Open BI analytics and create a new analysis →sample sales→per name year, revenue columns
2. Click on per name year edit formula→remove per name year and click on variable option.
3. Variable→click on session→type USER→click on ok→again ok→click on Results.
4. Observe the results displaying the current session user name that is weblogic
5. Now sign out as weblogic and sign in as msit user and create same report and Observe the results displaying msit in the results.

Non system session variables:

1. These are application specific customized variables.
2. These variables required Session initialization blocks
3. Session initialization blocks will be executed whenever user login into analytics application.

Using system & non system variables enabling data level security or Row Level Security

1. It is useful to hide some of the data based on user.

2. Here we will enable data security to see customer corresponding region data.

3. Data Filters/security can be applied on BMM layer /Presentation layer objects.

Creating users, group and role:

Refer section 8. Weblogic Server (WLS) Admin Console for Creating users, group and role

Users	Group	Application role
Robin Fisher	Customer	Customers
Fay Mills	Customer	Customers
Linda Larson	Customer	Customers

Synchronize roles in BI Admin tool:

1. Open above RPD in online mode.

2. Go to Manage menu →click on identity→go to action menu→click on Synchronize application roles.

3. New roles will be displayed in Identity Manager.

Create Dedicated connection pool

1. In Physical Layer,right click on orcl database object→clcik on new object→click on connectionpool.

2. Enter connection pool name as IB_SESSION→datasource name:orcl→ username:BISAMPLE→Password:BISAMPLE→click on ok→again ok.

Create initialization block(IB) and assign values to non-system variables

1. Creating non-system variable & initialization block.

2. Go to manage menu→click on variables→click on action menu→click on new session variable→name it as V_USER_REGION→type default initializer as

'A'→click on new →name it as IB_USER_REGION→Edit datasource→type below Sql:

Select region from samp_customers_d c ,samp_addresses_d a

where a.address_key=c. address_key and name =':USER'

3. Clcik on connection pool browse→select IB_SESSION connection pool→click on select →click on ok→again ok.

Create a data filter on role

1. Go to Manage menu→click on Identity→select BI Repsoitory→click on application roles tab.
2. Application roles tab→Double click on customers→ click on premissions→clickon data filters.
3. Data filters→click on add→expand sample sales→select customer regions→click on select →click ok.
4. Under data filter→click your mouse→click on edit expression→select D3 customer table→double click on region →type =→select session variable→double click on V_USER_REGION→click on ok→again ok→click close.

"Sample.sales"."D3 customer"."Region"=

VALUEOF (NQ_SESSION.V_USER_REGION)

Test your work

1. Login BI analytics as WebLogic and create a new analysis with columns region, revenue and Observe the results.
2. Login BI analytics as Robin Fisher and create a new analysis with columns region, revenue and Observe the results.
3. Repeat for Fay Mills and Linda Larson and Observe the results.

Row Wise Initialization Blocks

1. Are used to retrieve and initialization a list of values to a non system session variables.
2. Returns list of values separated by colon .
   Ex: AMERICAS: EMEA
3. Used for implementing data security when one user normally belongs to more than One group.

Enabling Row Wise Initialization

1. In above processes, Modify Session Initialization Block query as shown below

Select 'V_USER-REGION',region from samp_customers_d c ,samp_addresses_d a where a.address_key=c. address_key and name =':USER'

2. Click on edit target and select V_USER_REGION and remove it.
3. Below enable Row –wise initialization and click ok.
4. Some reason the moment you are deleting v_user_region session variable, one static repository variable created (it is a bug)→Delete this static repository variable V_USER_REGION (in variable manager,click on static in left pane)→check in→click yes→save RPD.
5. As we deleted v_user_region session variable, data security deleted
6. So go to manage menu→identity→select role of customers→clcik on permissions→data filters→check the filter expression→replace entire filter expression or replace MISSING with V_USER_REGION to make it right.
7. Close these windows→check in →click yes→ save RPD and check consistency.

Test Your Work

1. Open sql developer or Toad and insert the record into samp_customers_d and samp_addresses_d tables.

Insert into samp_customers_d(name,addres_key) values('Fay Mills','1234');

Insert into samp_ addresses_d (region,addres_key) values('APAC','1234');

Delete from samp_ addresses_d where region='EMEA' and address_key='1234';

Commit;

Verify

Select region,name,address_key from samp_customers_d c ,samp_addresses_d a where a.address_key=c. address_key and name in ('Robin Fisher','Fay Mils', 'Linda Larson');

2. Now login BI analytics as Fay Mills and observer the results,now he is showing both EMEA and APAC region data.
3. Logout from BI Analytics.

9.6.3. Presentation Variable

Refer section tiltled: 11.2. Dashboard Prompts

9.6.4. Request Variable

Refer section tiltled: 11.2. Dashboard

9.6.5. Global Variable

1. A global variable is a column created by combining a specific data type with a value. The value can be a string, number, date, time, expression, formula, and so on.
2. The global variable is then saved in the catalog and made available to all other analyses.
3. Global variables can be of the following types:

- Date
- Date and Time
- Number
- Text
- Time

Note:

Global variables apply to Oracle BI EE 11.1.1.7.10 and later versions, and might not be available in earlier versions.

Creating a global variable

1. Open BI analytics an create a new analysis with per name year,revenue,revenue columns.
2. Click on 1st revenue columns→select edit formula→remove revenue and click on variable button and select global .
3. The Insert Global Variable dialog is displayed.
4. Click the Add New Global Variable button. The "New Global Variable dialog" displays.
5. Enter a unique name as gv_txt_multiply_rev
6. Select a data type as text.
7. Enter a value as “Base facts”.”revenue”*3.1415
8. Click OK. The new global variable is added to the Insert Global Variable dialog.
9. Select the new global variable that you just created, and then click OK. The Edit Column Formula dialog is displayed with the global variable inserted in the Column Formula pane.

Note:

The Custom Headings check box is automatically selected.

10. Enter a new name for the column to which you have assigned a global variable to more accurately reflect the variable.

11. Click OK. The global variable is evaluated at the time the analysis is executed, and the value of the global variable is substituted appropriately. Only users with appropriate privileges can manage (add, edit, and delete) global variables

----------lot more