Oracle® Retail Accelerators

Developing BI EE Reports for RPAS
Release 13.2

June 2011

ORACLE
RETAIL

ORACLE

Note: The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Contents

1
Introduction

The Oracle Retail Predictive Application Server (RPAS) platform supports multidimensional databases and hierarchical data. Users usually manipulate the data (in the RPAS domains) using the workbooks (graphical user interface) or by calling the RPAS API. Since the RPAS domains are not SQL compliant, the data cannot be accessed directly using Oracle Business Intelligence (BI) Enterprise Edition (EE). To solve this issue, Oracle Retail has developed the RPAS ODBC package that contains the following components:

- RPAS ODBC Driver – This component resides on the system where the BI EE is installed.
- RPAS ODBC Server – This component runs on the system where the RPAS applications are installed.

This document illustrates how you can set up this integration between BI EE and RPAS domains to generate reports and dashboards. Using this ODBC interface, you can then use BI EE to access the RPAS domains. This interface enables you to create reports on measures across multiple RPAS domains. Since BI EE is able to export reports as URLs, RPAS data can now be presented in a greater variety of ways.

RPAS ODBC enables the integration between BI EE and RPAS domains. It provides an additional way to present the intelligence from the planning and optimization applications.

2

RPAS ODBC Data Query

The RPAS ODBC server calls low-level RPAS API to access the RPAS domains. It presents the RPAS domain data (dimensions, measures) into tables, which are SQL compliant. This chapter describes the database objects that can be used for the RPAS ODBC data query.

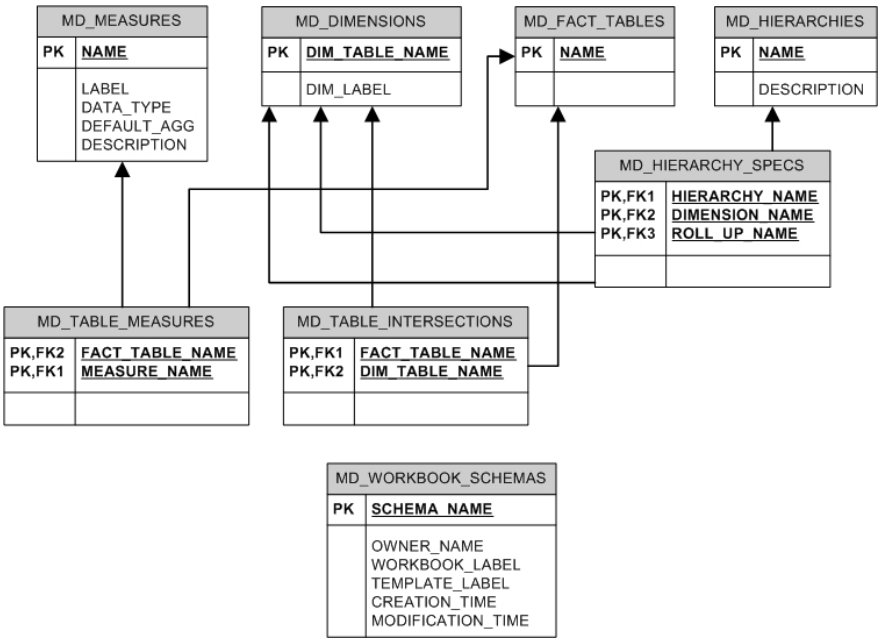
It includes the following topics:

Metadata

The following figure shows the metadata tables available in a domain or workbook. These tables can be used to examine the structure of the domain, such as:

- Measures and dimensions that exist within the database.
- Hierarchies that exist and their rollup structure.
- Fact tables that are available.
- Measures that exist at the intersections they represent.

When connected to a domain, an additional table (MD_WORKBOOK_SCHEMAS) is available to list all accessible workbooks within the domain with their schema names.



Database Diagram for All Metadata Tables in a Domain or in Each Workbook

> **Note:** The MD_WORKBOOK_SCHEMAS table is not included in workbooks.

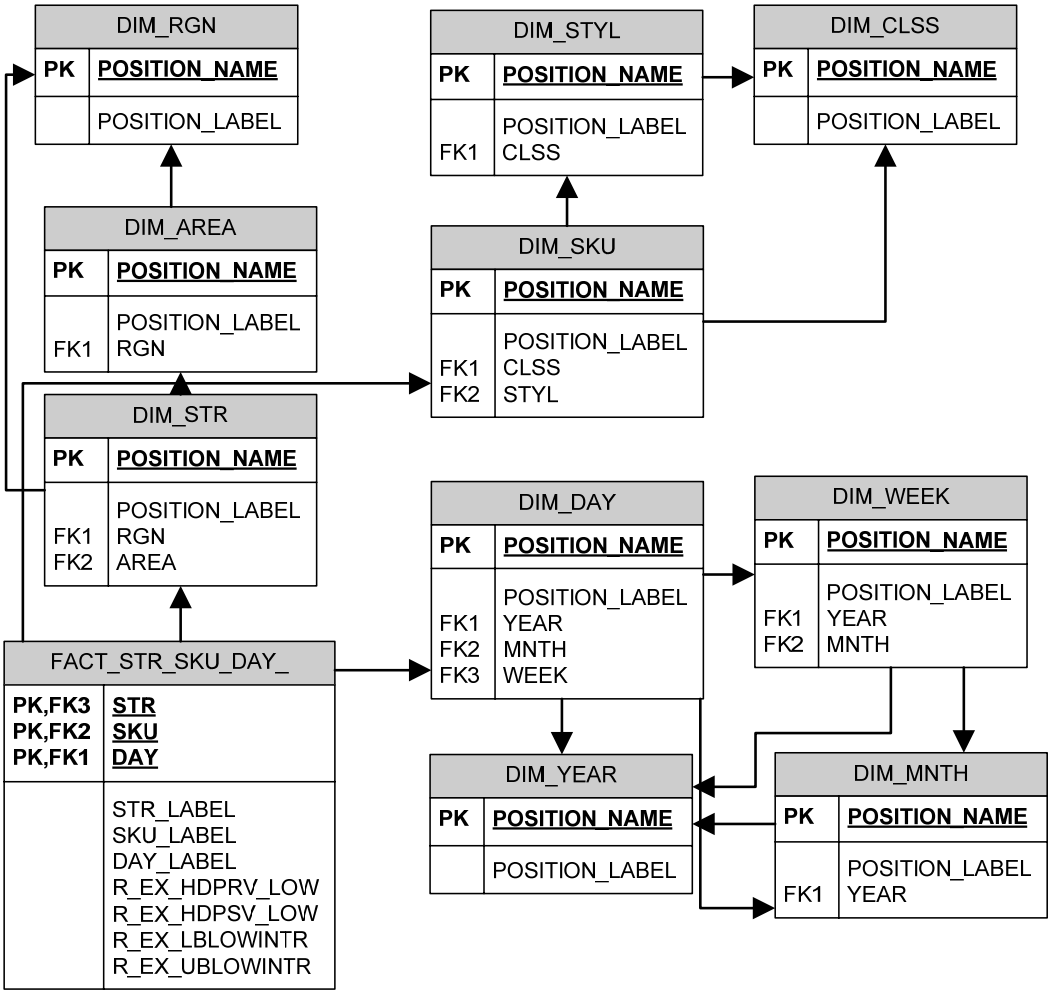
Fact and Dimension Tables

The following figure shows an example of the structure of fact and dimension tables and the relationships between them. A fact table represents an intersection where one or more measures data is stored. Each measure is represented by a column in the table.

Additionally, each dimension on the intersection is represented by a column. A record in the fact table is uniquely identified by a unique combination of position names for the intersecting dimensions.

A dimension table represents a dimension. It includes a column to list all position names, their labels, and their rollup mapping to each dimension at higher levels in the hierarchy. The fact and dimension tables have foreign key relationships between them to represent the intersection and maintain data integrity between the dimensions and the facts.

Dimension tables have foreign key relationships with other dimension tables to represent the hierarchical relationships between them.



Example of Star – De-normalized Schema to Represent Facts and Dimensions in RPAS

At connection time, all intersections at which any measure is stored at its base level are available as fact tables within the database. Additional aggregate level intersections may be made available in the database by specifying them in a custom connection property.

These fact tables are a part of the set of database entities that will be visible to reporting tools at connection time. However, the RPAS ODBC/JDBC driver supports dynamic aggregate level fact tables that can be queried even though they are not available at connection time. These tables include all intersections that are logically above the base intersection fact tables and have at least one measure in them when manifested. If the measure existence condition is not met, the driver returns an error that the fact table could not be found.

These dynamic fact tables are queried in the same fashion as the tables that are available at connection time. The name of the fact table can be constructed by piecing together dimension names (not labels) that make up the intersection in the order in which they would exist within the domain. For example, if someone wants to query facts at the store/class/day level but the fact table is not available at connection time, they can construct the fact table name as: FACT_STR_CLSSDAY_. Note that dimension names have been concatenated in the same order as the intersection and have been prefixed with 'FACT_'. Also, note that a dimension name is assumed to be four characters long and if the dimension name is less than four characters, it is padded with '_' characters to make it four characters long.

By issuing ODBC SQLs against these dimension tables and fact tables, OBIEE is able to access RPAS data.

3

Setting Up a Database Connection

This chapter describes how you can set up the ODBC data source for the MFP Retail domain. It includes the following topics:

Prerequisites

Before you proceed, ensure that you have the following installed:

- **Oracle Retail Predictive Application Server (RPAS) Release 13.2 or later and ODBC Components**
  This includes the RPAS ODBC Server and RPAS ODBC client. For more information on installing RPAS and setting up the ODBC components, refer to the *Oracle Retail Predictive Application Server Installation Guide Release 13.1.1 and Release 13.2*.
  In case you install the RPAS ODBC Server on a Linux-based system, ensure that you set up the Time Zone (TZ) environment variable.
- **MFP Retail domain**
  This includes the Merchandise Financial Planning Retail Release 13.2 solution and domain. For more information on installing Merchandise Financial Planning Retail, refer to the *Oracle Retail Merchandising Financial Planning Retail Installation Guide Release 13.2*.
- **Oracle Business Intelligence (BI) Enterprise Edition (EE) Release 10.1.3.4 or later**
  For more information on installing Oracle BI EE, refer to the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

Setting Up the ODBC Data Source for the MFP Retail Domain

Once you have the required software installed, you must set up the RPAS ODBC data source for the MFP Retail domain. You must use the Management Console to set up data sources on the RPAS ODBC server. This enables the RPAS ODBC data service to accept connections from the RPAS ODBC client.

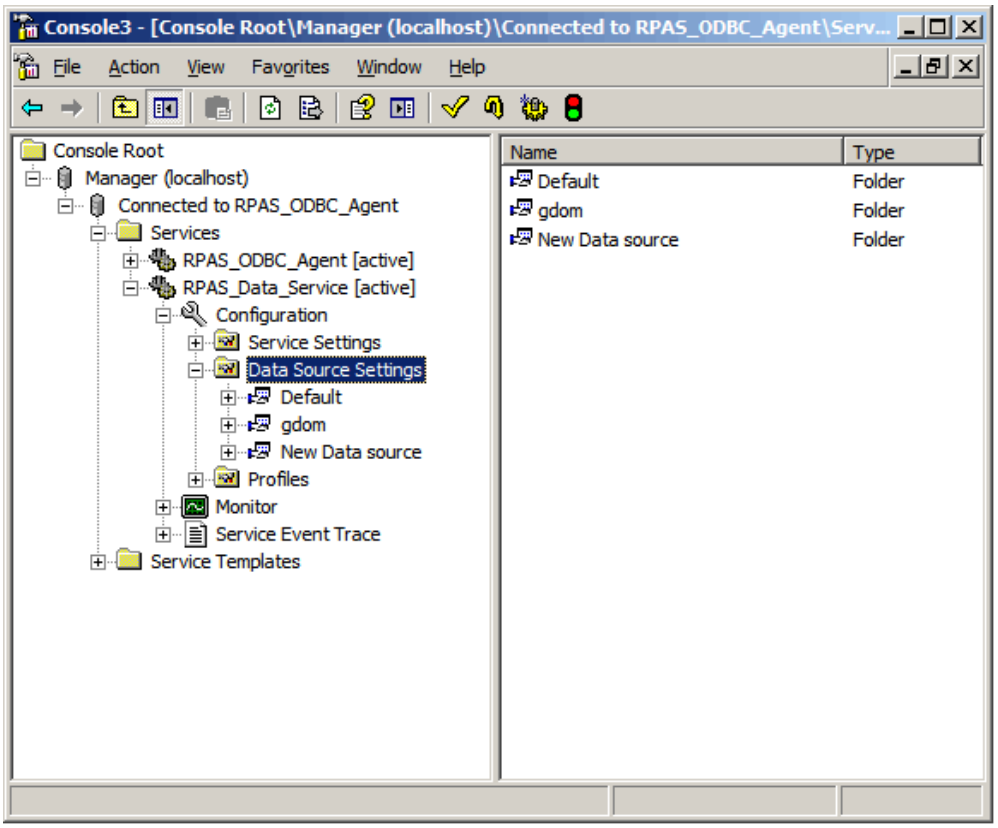
This section describes how you can set up the ODBC data source for the MFP Retail domain. It includes the following topic:

Setting Up Data Sources on the RPAS ODBC Server

To set up data sources on the RPAS ODBC server:

1. On the Windows-based system where the RPAS ODBC server is installed, click the Start menu.
2. From the **Start** menu, point to **Programs**, and then point to **Oracle RPAS ODBC Server**.

3. From the **Oracle RPAS ODBC Server** menu, click **Management Console**. The **Management Console** appears, as shown in the following figure:



Management Console Window

4. Under **Console Root**, expand the **Manager** node for the **RPAS ODBC Server Manager**.
5. Right-click on **Not connected to RPAS_ODBC_Agent** and select **Connection** to connect to the RPAS ODBC server.
6. Once connected, expand **Services**, and then expand **RPAS_Data_Service**.
7. Under **Configuration**, expand **Data Source Settings**.
8. From the right-click menu for **Data Source Settings**, point to **New**, and then click **Data Source**. A new data source is added with the name **New Data source**.
9. Rename the data source name to the relevant RPAS solution domain. For example, use mfprtl_domain for the MFP Retail domain.
10. Expand **mfprtl_domain** and set up the data source attributes with the values listed in the following table:

Attribute	Type	Value
IP Parameters		
DataSourceIPType	String	DAMIP
DataSourceIPProperties	String	DOMAIN_PATH=<*Location where the mfprtl_domain is installed*>

DataSourceIPCustomProperties	String	WORKBOOK_SCHEMA=;LANGUAGE=; SHORT_DATE_FORMAT=;QUALIFIER=; DEFAULT_SCHEMA=DOMAIN;RETURN _NA_AS_NULL=;LOG_FILE=;AGG_TABLE _NAMES=;SCHEMA_IN_CACH=;NORMALIZE _DIM_TABLES=;RPAS_LOG_LEVEL=
DataSourceIPSchemaPath	String	*<Location_where_the_ODBC_Server_is_installed>* /ip/schema/template_dynamic
User Security		
DataSourceLogonMethod	String	DBMSLogon(UID,PWD)
Workarounds		
DataSourceWorkArounds2	Integer	8192

Note: The value for the DataSourceIPSchemaPath attribute includes a placeholder text <Location_where_the_ODBC_Server_is_installed>. Ensure that you replace the placeholder text with the location where the ODBC server is installed.

Setting Up Data Sources on the RPAS ODBC Driver

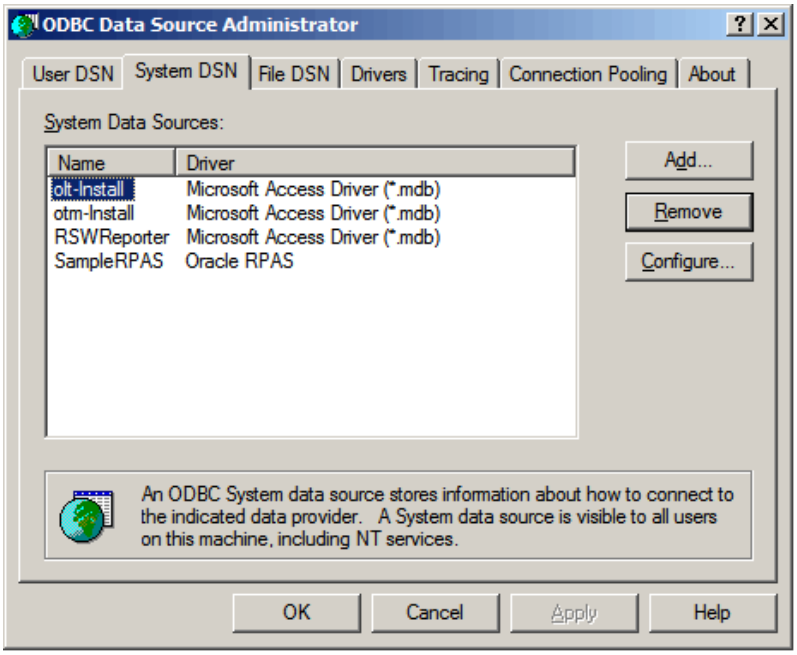
Once the data sources are created on the RPAS ODBC server, you must create the relevant data sources on the RPAS ODBC driver. Make sure the data source names are the same as on the RPAS server. This section provides information on setting up the data sources on the RPAS ODBC client running on either a Windows or a UNIX-based system.

Note that for the POC environment, OBIEE 10.1.3.4.1 is installed on a Windows Desktop. The MFP Retail domain is hosted by RPAS 13.2 on HPUX 11.31.

Windows-Based RPAS ODBC Client Configuration

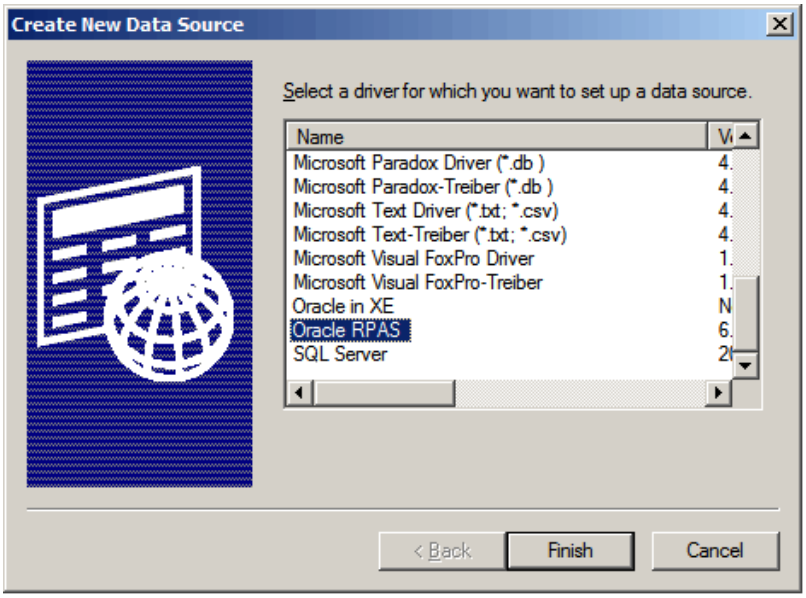
To set up data sources on the RPAS ODBC client installed on a Windows-based system:

1. On the Windows-based system where the RPAS ODBC client was installed, click the **Start** menu.
2. From the **Start** menu, point to **Programs**, and then point to **Oracle RPAS ODBC Driver**.
3. From the **Oracle RPAS ODBC Driver** menu, click **ODBC Administrator**. The **ODBC Data Source Administrator** window appears, as shown in the following figure.



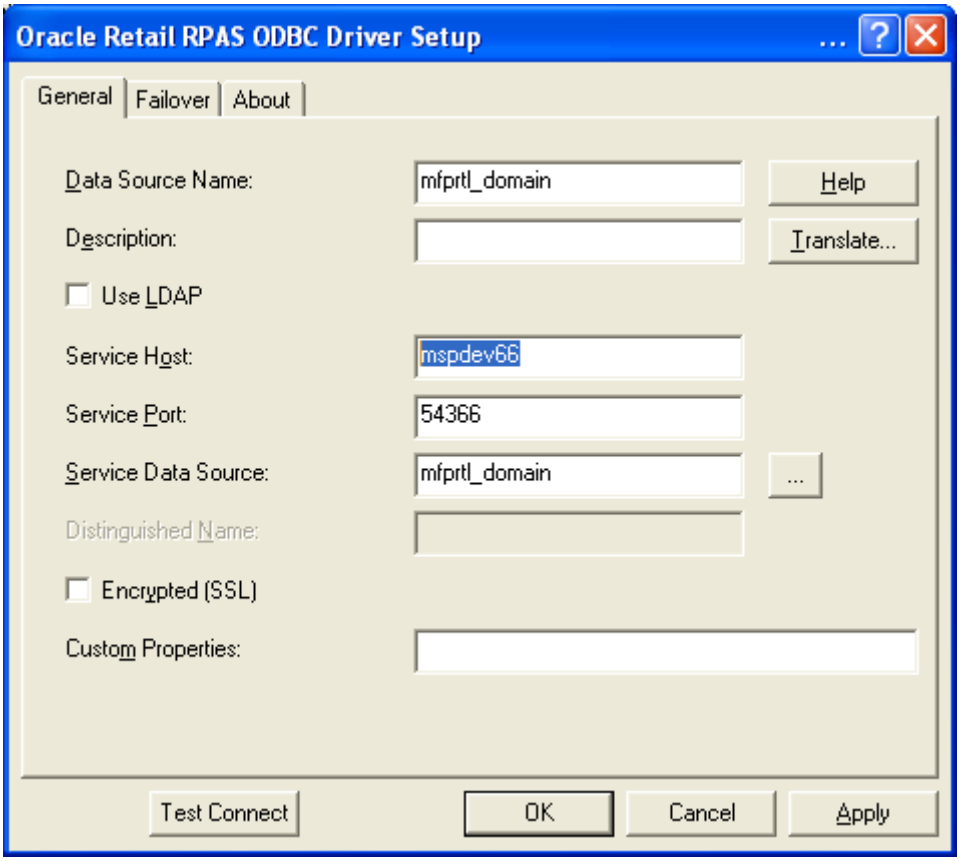
ODBC Data Source Administrator Window

4. On the **ODBC Data Source Administrator** window, click the **System DSN** tab.
5. On the **System DSN** tab, click **Add**. The **Create New Data Source** window appears, as shown in the following figure:



Create New Data Source Window

6. On the **Create New Data Source** window, select **Oracle RPAS**. The **Oracle Retail RPAS ODBC Driver Setup** window appears, as shown in the following figure.



Oracle Retail RPAS ODBC Driver Setup Window

7. On the **Oracle Retail RPAS ODBC Driver Setup** window, in the **General** tab, enter the relevant RPAS solution domain name in the **Data Source Name** field. For example, enter mfprtl_domain for the MFP Retail domain.
8. Enter relevant information in the following fields:
   - **Service Host** – enter the name of the RPAS ODBC server where the RPAS data service is set up.
   - **Service Port** – enter the port number set up for the RPAS data service.
   - **Service Data Source** – enter the relevant data source name on the RPAS ODBC server.
9. Click **Test Connect** to test the connection.
10. Once the test is successful, click **OK** to close the window and return to the **System DSN** tab.

UNIX-Based RPAS ODBC Client Configuration

To set up data sources on the RPAS ODBC client installed on a UNIX-based system:

1. On the UNIX-based system, navigate to the location where the RPAS ODBC client is installed.
2. Open the **odbc.ini** file for editing.
3. Replace **PATH_TO_ODBC_CLIENT** with the absolute path to the ODBC Client.
4. In the **[ODBC Data Sources]** area, add an entry for the relevant RPAS solution domain. For example, set up mfprtl_domain data source for the Assortment Planning domain in the following manner:

```
mfprtl_domain=MFP Retail Domain
```

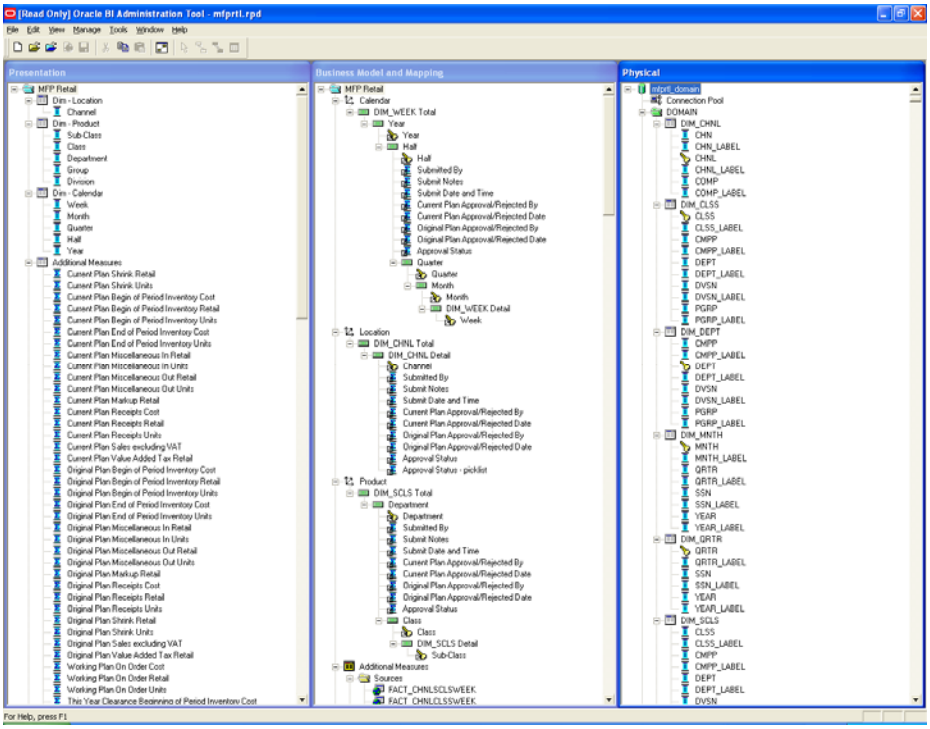
5. Create a new area with the name **[mfprtl_domain]**.
6. Copy the contents (parameters) under the **[SampleRPAS]** area, and paste the contents under **[mfprtl_domain]**.
7. Under **[mfprtl_domain]**, set the following parameters with relevant values:
   - **Host** – enter the name of the RPAS ODBC server where the RPAS data service is set up.
   - **Port** – enter the port number set up for the RPAS data service.
   - **ServerDataSource** – enter the relevant data source name on the RPAS ODBC server.
8. Copy the contents of the odbc.ini you set up in steps 4 through 7 to the odbc.ini file located in **<OracleBI_Home>/setup/**.

4

Setting Up the RPD File

This chapter describes how you can create the RPD file. The created rpd file, including the physical model, the business model, and the presentation model, is showing in the following figure.

Note that for the POC environment, the presentation layer and the business model are the same.



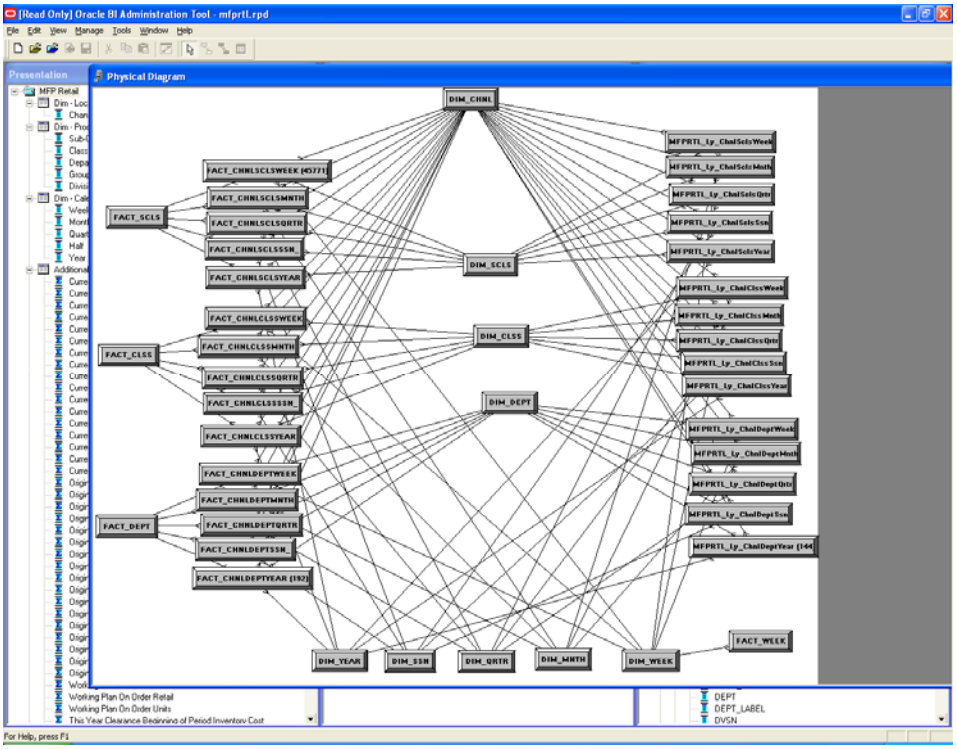
The rpd File

Complete the following steps to create the rpd file:

1. Import from the database in order to create the physical layer.
   The sample rpd uses the following measures:
   - Approval Status - picklist
   - Approval Status
   - Clearance Sales Retail
   - Current Plan Approval/Rejected By
   - Current Plan Approval/Rejected Date
   - Current Plan Approved/Rejected Notes
   - EOP Inventory Retail
   - Gross Margin %
   - Gross Margin
   - Ly Clearance Sales Retail
   - Ly EOP Inventory Retail

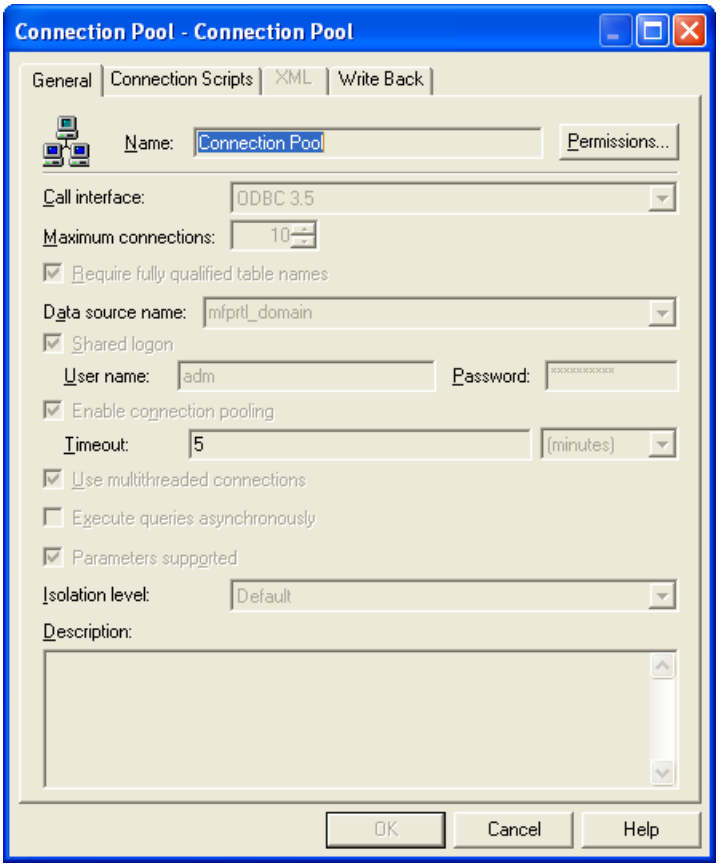
- Ly Gross Margin %
- Ly Gross Margin
- Ly Promotional Sales Retail
- Ly Regular Sales Retail
- Ly Retail Sell Thru %
- Ly Sales Retail
- Original Plan Approval/Rejected By
- Original Plan Approval/Rejected Date
- Original Plan Approved/Rejected Notes
- Plan Clearance Sales Retail
- Plan EOP Inventory Retail
- Plan Gross Margin %
- Plan Gross Margin
- Plan Promotional Sales Retail
- Plan Regular Sales Retail
- Plan Retail Sell Thru %
- Plan Sales Retail
- Promotional Sales Retail
- Regular Sales Retail
- Retail Sell Thru %
- Sales Retail % Variance to Ly
- Sales Retail % Variance to Plan
- Sales Retail
- Submit Date and Time
- Submit Notes
- Submitted By
- User Defined Performance Threshold
- What Plan Version has been submitted for approval - picklist
- What Plan Version has been submitted for approval

The following figure shows the physical diagram:



Physical Diagram

The following figure shows the MFP Retail ODBC data source for the connection pool:



MFP Retail ODBC Data Source in Connection Pool

2. Create the business model and mappings.
   The Business Model and Mapping layer (known also as Business Model) is where you can apply your business logic according to the business requirement. An example business model can be seen in the rpd file above. It has three dimension tables and two fact tables.
3. Drag the business model to be the presentation model.
   The presentation layer is used to provide the customized view that hides all the business logic.

Creating Reports and the Dashboard

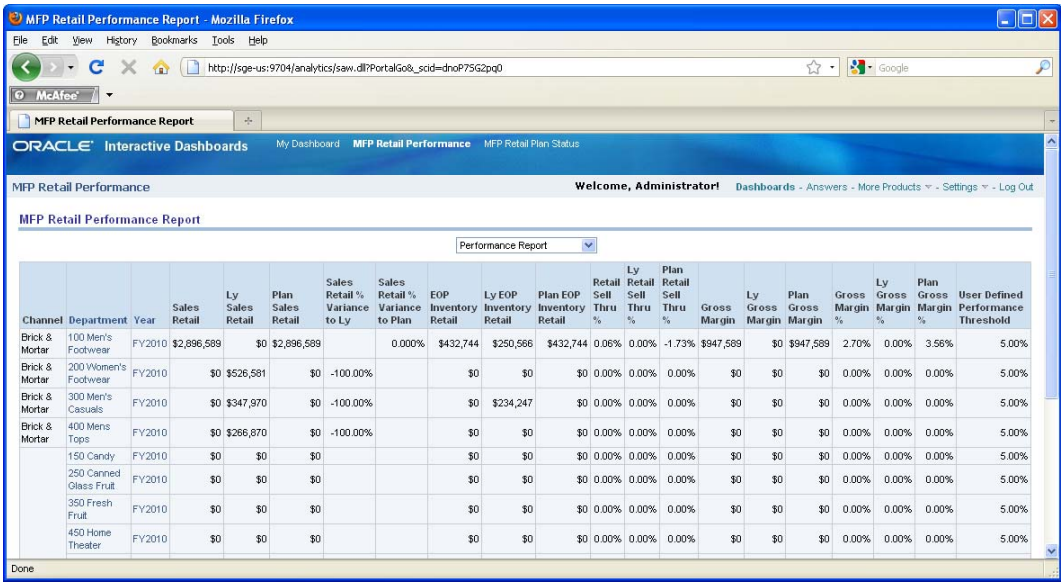
In the BI EE environment, create the following:

- Two reports: the MFP Retail Performance Report and the MFP Retail Plan Status Report.
- Two dashboards: The MFP Retail Performance Dashboard and the MFP Retail Plan Status Dashboard.

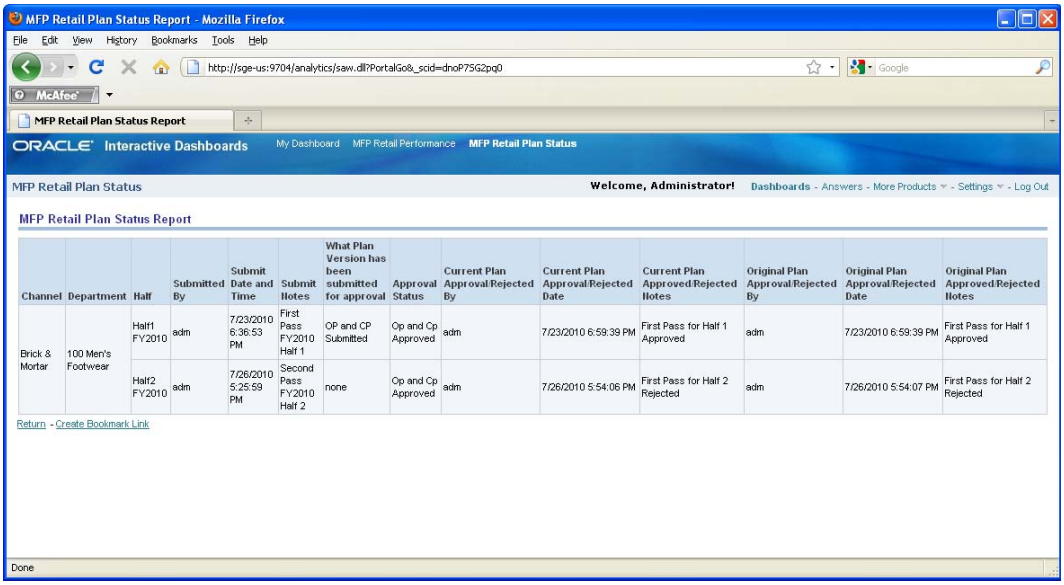
The MFP Retail Performance Report contains the MFP Retail Performance Dashboard.

The MFP Retail Plan Status Report contains the MFO Retail Plan Status Dashboard.

These two dashboards are shown in the following two figures:



MFP Retail Performance Dashboard



MFP Retail Plan Status Report Dashboard

Appendix A

Best Practices and Findings

This appendix contains recommendations about best practices as well as a discussion of relevant findings.

Fact Tables Recommendations

An RPAS measure has a base intersection. For example, the "Regular Sales Retail" measure in the MFP Retail domain has a base intersection of CHNL, SCLS and WEEK.

RPAS ODBC presents this measure in different tables such as fact_chnlsclsweek, fact_chnlclssweek, and fact_chnldeptweek. High-level fact tables are those having dimensions higher in the hierarchy. For example, fact_chnldeptweek is a higher- level table than fact_chnlclssweek since dept dimension is a higher level than clss dimension.

When possible, query the high-level fact tables rather than lower-level fact tables, since the high- level fact tables have fewer records.

Calculation Recommendations

Calculations can be done either in OBIEE or RPAS.

Performing calculations in OBIEE provides the following two benefits:

- Reporting on real time data. If calculating in RPAS, users have to run a batch before OBIEE can read the calculated measure. There will be time delay between the batch and the read.
- No need to store the calculated measure anywhere.

Performing calculations in RPAS provides the following two benefits:

- Faster. Since OBIEE issues fewer SQLs, the RPAS engine is closer to the data, and the batch run has already completed the calculation before OBIEE read it.
- Some calculations are not supported in OBIEE. For example, RPAS has a function called cover, which calculates how many days the inventory can supports the sales for a period of days. This function is not supported by OBIEE.

Using RPAS is preferable for the following reasons:

- Speed. When having a reasonable amount of data, calculating in RPAS is much faster.
- Nature of the applications. Since analytical applications generally use a data set that is changed infrequently, running a batch periodically to populated calculated measure is acceptable.
- For some applications, the batch can be incorporated into the workbook actions: like commit or custom menu. Whenever these actions happen, the calculated measures are populated.

BI EE Reports

If the BI EE report does not need "order by", do not use it. In this way the generated SQL will not have "order by" clause and will be faster.

BI EE-Generated SQL

BI EE generates SQLs automatically. These SQLs may not always be optimized. Users can manually modify them.

Caching

BI EE uses caching at three levels: repository (rpd), BI server, and presentation service.

In order to report the real time domain data, caching can be disabled.

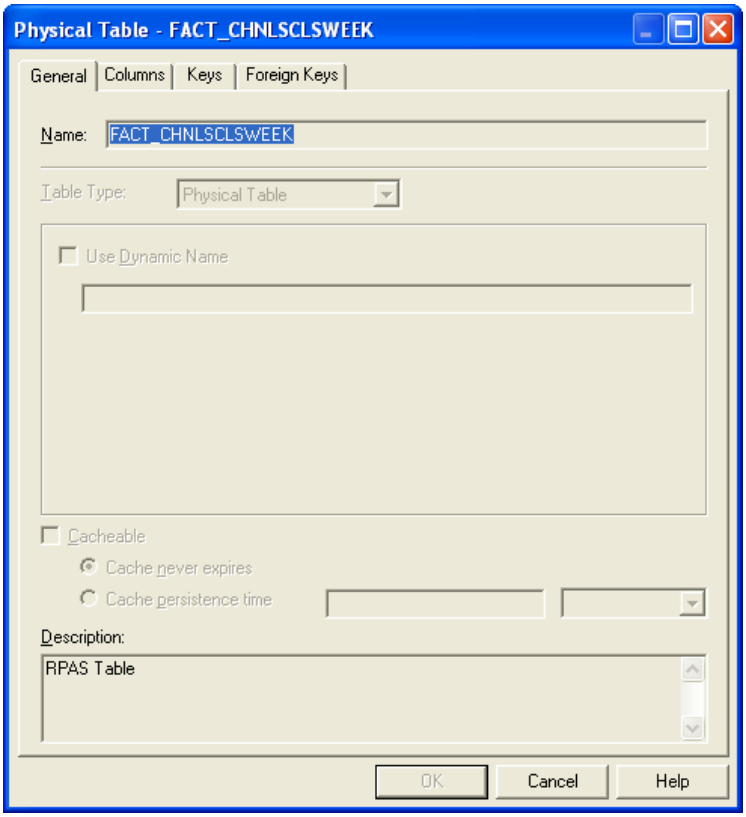
Whether caching is enabled or disabled is a retailer's call. Here are some limitations when disabling caching to have real time access to the RPAS domains:

- Only one user at a time can write to the RPAS domain.
- When one user is writing, all reads and writes are pending.
- When no one is writing, multiple reads are allowed.

For more information, refer *Oracle Retail Predictive Application Server Administration Guide* Release 13.1.1 and Release 13.2.

Repository Level Caching

In the following figure for repository level caching, the cachable button is not checked.



Repository Level Caching

BI Server Level

In the following example, caching is disabled at the BI server level:

```
Change <OracleBI_HOME>\server\Config\NQSConfig.INI
Change ENABLE to NO:
[ CACHE ]
```

```
ENABLE =   NO;
```

Presentation Service Level

In the following example, caching is disabled at the presentation service level:

```
Change < OracleBIData_HOME>\web\config\instanceconfig.xml
Add the follows:
<WebConfig>
   <ServerInstance>
     <ForceRefresh>TRUE</ForceRefresh>
  </ServerInstance>
</WebConfig>
```

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

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com